



**ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ**

**ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧ/ΚΩΝ & ΜΗΧ/ΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ**

**ΤΟΜΕΑΣ ΕΠΙΚΟΙΝΩΝΙΩΝ ΗΛΕΚΤΡΟΝΙΚΗΣ & ΣΥΣΤΗΜΑΤΩΝ  
ΠΛΗΡΟΦΟΡΙΚΗΣ**

**ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ**

**ΚΑΤΑΣΚΕΥΗ ΚΑΙ ΠΛΟΗΓΗΣΗ ΡΟΜΠΟΤ**

**ΑΚΟΥΣΤΙΚΩΝ ΜΕΤΡΗΣΕΩΝ**

**ΔΗΜΟΣΘΕΝΗΣ ΡΩΣΣΙΔΗΣ**

**ΕΠΙΒΛΕΠΩΝ ΚΑΘΗΓΗΤΗΣ : ΓΕΩΡΓΙΟΣ ΚΑΜΠΟΥΡΑΚΗΣ**

**ΑΘΗΝΑ**

**ΣΕΠΤΕΜΒΡΙΟΣ 2012**



ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ  
ΚΑΤΑΣΚΕΥΗ ΚΑΙ ΠΛΟΗΓΗΣΗ ΡΟΜΠΟΤ ΑΚΟΥΣΤΙΚΩΝ ΜΕΤΡΗΣΕΩΝ

ΔΗΜΟΣΘΕΝΗΣ ΡΩΣΣΙΔΗΣ

ΕΠΙΒΛΕΠΩΝ: ΓΕΩΡΓΙΟΣ ΚΑΜΠΟΥΡΑΚΗΣ  
ΕΠΙΚΟΥΡΟΣ ΚΑΘΗΓΗΤΗΣ Ε.Μ.Π.

ΤΡΙΜΕΛΗΣ ΕΞΕΤΑΣΤΙΚΗ ΕΠΙΤΡΟΠΗ

ΓΕΩΡΓΙΟΣ ΚΑΜΠΟΥΡΑΚΗΣ

Επ. Καθηγητής Ε.Μ.Π.

ΕΛΕΥΘΕΡΙΟΣ ΚΑΓΙΑΦΑΣ

Καθηγητής Ε.Μ.Π.

ΒΑΣΙΛΕΙΟΣ ΛΟΥΜΟΣ

Καθηγητής Ε.Μ.Π.

ΑΘΗΝΑ

ΣΕΠΤΕΜΒΡΙΟΣ 2012

Copyright © Δημοσθένης Ρωσσίδης, 2012

Με επιφύλαξη παντός δικαιώματος. All rights reserved.

Απαγορεύεται η αντιγραφή, αποθήκευση και διανομή της παρούσας εργασίας, εξ ολοκλήρου ή τμήματος αυτής, για εμπορικό σκοπό. Επιτρέπεται η ανατύπωση, αποθήκευση και διανομή για σκοπό μη κερδοσκοπικό, εκπαιδευτικής ή ερευνητικής φύσης, υπό την προϋπόθεση να αναφέρεται η πηγή προέλευσης και να διατηρείται το παρόν μήνυμα. Ερωτήματα που αφορούν τη χρήση της εργασίας για κερδοσκοπικό σκοπό πρέπει να απευθύνονται προς τον συγγραφέα.

Οι απόψεις και τα συμπεράσματα που περιέχονται σε αυτό το έγγραφο εκφράζουν τον συγγραφέα και δεν πρέπει να ερμηνευθεί ότι αντιπροσωπεύουν τις επίσημες θέσεις του Εθνικού Μετσόβιου Πολυτεχνείου.

## ΠΡΟΛΟΓΟΣ

Η παρούσα διπλωματική εργασία έχει ως αντικείμενο την κατασκευή και την πλοήγηση ενός ρομπότ το οποίο κινείται αυτόνομα στον χώρο, με σκοπό την προσέγγιση ενός σημείου και τη λήψη ακουστικών μετρήσεων. Μέσα από αυτήν την εργασία αναδεικνύεται η χρησιμότητα που μπορούν να έχουν οι εφαρμογές της ρομποτικής στην καθημερινότητα, προκειμένου να αντικαταστήσουν από απλές έως πιο σύνθετες διαδικασίες. Η εφαρμογή της συγκεκριμένης κατασκευής μπορεί να αποτελέσει ένα αντιπροσωπευτικό παράδειγμα, αφού με τον κατάλληλο εξοπλισμό θα μπορούσε να μετατραπεί σε ρομπότ συλλογής διαφορετικού τύπου μετρήσεων, ή γενικά να επιτελέσει μια διαφορετική εργασία.

Μέσα στην διπλωματική εργασία, θα περιγράψουμε τους τρόπους με τους οποίους σχεδιάστηκε, κατασκευάστηκε και προγραμματίστηκε το ρομποτικό σύστημα, θα παρουσιάζουμε την πρακτική εφαρμογή του, θα αναλύσουμε τα προβλήματα που συναντήθηκαν καθώς επίσης και τους τρόπους με τους οποίους τα αντιμετωπίσαμε, ενώ τέλος, θα κάνουμε και αρκετές προτάσεις με σκοπό την περαιτέρω βελτίωση του όλου project.

Η διπλωματική εργασία έχει χωριστεί σε 2 μέρη. Το πρώτο μέρος αποτελεί το θεωρητικό κομμάτι της διπλωματικής, και ξεκινάει στο πρώτο κεφάλαιο με μία γενικότερη εισαγωγή στον τομέα της ρομποτικής, ενώ συνεχίζει με μία ιστορική αναδρομή στα κινούμενα ρομπότ (στα οποία ανήκει και το ρομπότ που κατασκευάσαμε), καθώς επίσης και στο οδόμετρο του Αρχιμήδη, η λογική του οποίου αποτελεί τη βάση της εργασίας. Στο δεύτερο κεφάλαιο αναλύεται το θεωρητικό υπόβαθρο στο οποίο βασιστήκαμε, κάνοντας κυρίως μία εκτεταμένη αναφορά στα συστήματα διαφορικής οδήγησης και στον τρόπο κίνησής τους, ενώ ένα επίσης σημαντικό μέρος του δευτέρου κεφαλαίου είναι η περιγραφή της τεχνολογίας των αισθητήρων υπέρυθρης ακτινοβολίας.

Το δεύτερο μέρος αποτελεί και το κυρίως μέρος της διπλωματικής εργασίας, καθώς γίνεται η αναλυτική περιγραφή του ρομπότ ακουστικών μετρήσεων που κατασκευάσαμε. Το τρίτο κεφάλαιο λοιπόν, πραγματεύεται την κατασκευή ολόκληρου του ρομπότ και των κυκλωματικών διατάξεων που χρησιμοποιεί, περιγράφεται το σύστημα αισθητήρων, οι τρόποι με τους οποίους πραγματοποιείται η τροφοδοσία του ρομπότ, καθώς επίσης και οι κινητήρες που χρησιμοποιήθηκαν για την μετακίνησή του. Στο τέταρτο κεφάλαιο, περιγράφεται η όλη λειτουργία του συστήματος και ο τρόπος με τον οποίο προγραμματίστηκε, αναλύεται η διαδικασία αποφυγής εμποδίων και ο τρόπος υλοποίησης του οδομέτρου, παρουσιάζονται τα μηνύματα που εξάγονται κατά την διάρκεια της λειτουργίας, ενώ αναλύεται και η χρησιμότητά τους. Τέλος, όπως αναφέραμε, στο πέμπτο κεφάλαιο γίνεται μία αναφορά στα σφάλματα τα οποία εισάγονται κατά την κίνηση του ρομποτικού οχήματος, προτείνονται τρόποι αντιμετώπισής τους, ενώ στην συνέχεια παρουσιάζονται αρκετές επιλογές που μπορούν να γίνουν μελλοντικά, με σκοπό την τελειοποίηση της συνολικής λειτουργίας του συστήματος.

Η συγκεκριμένη διπλωματική εργασία έχει ιδιαίτερο ενδιαφέρον, καθώς συνδυάζει την εφαρμογή γνώσεων από διαφορετικές επιστημονικές περιοχές. Εκτός από τους τομείς της Ηλεκτρονικής και της Ρομποτικής, εφαρμόζονται γνώσεις από το πεδίο των Μικροϋπολογιστών, του Αυτομάτου Ελέγχου, καθώς επίσης και της Μηχανικής. Το ρομπότ ακουστικών μετρήσεων

σχεδιάστηκε και κατασκευάστηκε εξολοκλήρου από την αρχή, και το γεγονός αυτό αποτέλεσε την αιτία να αποκτηθούν σημαντικές γνώσεις σε όλους τους προαναφερθέντες τομείς.

Το ρομπότ που κατασκευάσαμε δεν είναι ένα παιχνίδι. Αν και μπορεί να είναι διασκεδαστικό να το παρακολουθούμε να ακολουθεί και να αποφεύγει μόνο του εμπόδια, όπως και το να κινείται μόνο του στο χώρο ενώ βρισκόμαστε μακριά, εν τούτης πρόκειται για μια κατασκευή υψηλής τεχνολογίας με πολλές δυνατότητες ανάπτυξης και χρήσης. Στόχος της όλης κατασκευής ήταν η έρευνα και η τεχνογνωσία που προκύπτει από την χρήση τεχνικών οπτικής αναγνώρισης και εφαρμογής τους σε πραγματικές συνθήκες.

Στο σημείο αυτό θα ήθελα να ευχαριστήσω τον κ. Γιώργο Καμπουράκη, καθηγητή του Εθνικού Μετσόβιου Πολυτεχνείου (ΕΜΠ), ο οποίος με τις συμβουλές και την καθοδήγησή του με βοήθησε σημαντικά στην εκπόνηση της διπλωματικής εργασίας. Θα ήθελα επίσης να ευχαριστήσω ιδιαίτερα, και τον υποψήφιο διδάκτορα Δημήτρη Πιπερίδη για την πολύτιμη βοήθειά του.

# ΠΕΡΙΕΧΟΜΕΝΑ

## ΜΕΡΟΣ Α

### ΚΕΦΑΛΑΙΟ 1

#### ΙΣΤΟΡΙΚΗ ΑΝΑΔΡΟΜΗ

1.1	Ρομπότ & Ρομποτική	11
1.1.1	Εισαγωγή	11
1.1.2	Ορισμός του ρομπότ	11
1.1.3	Οι κατηγορίες των ρομπότ	12
1.1.4	Το μέλλον των ρομπότ	14
1.2	Ιστορία των κινούμενων ρομπότ	15
1.3	Οδόμετρο του Αρχιμήδη	20

### ΚΕΦΑΛΑΙΟ 2

#### ΕΙΣΑΓΩΓΗ ΣΤΗ ΡΟΜΠΟΤΙΚΗ ΚΙΝΗΣΗΣ

2.1	Εισαγωγή	23
2.2	Βαθμοί Ελευθερίας	24
2.3	Διαφορική Οδήγηση	25
2.3.1	Εισαγωγή στα Ρομπότ Διαφορικής Οδήγησης	25
2.3.2	Κινηματική Διαφορικής Οδήγησης	26
2.3.3	Ορθή Κινηματική των Ρομπότ Διαφορικής Οδήγησης	27
2.3.4	Ανάστροφη Κινηματική Κινούμενου Ρομπότ	30
2.3.5	Αντιστοίχιση Γωνιακής Ταχύτητας Τροχού με Γραμμική Ταχύτητα	31
2.4	Πλοήγηση	33
2.4.1	Εισαγωγή	33
2.4.2	Τεχνικές Εντοπισμού Θέσης	34
2.4.3	Ταυτόχρονος Εντοπισμός και Χαρτογράφηση (SLAM)	35
2.5	Αισθητήριες Διατάξεις	36
2.5.1	Εισαγωγή	36
2.5.2	Infrared LED – Πομπός Υπερύθρων	37
2.5.3	Φωτοδίοδος και δίοδος PIN – Ο δέκτης υπερύθρων	39
2.5.4	Οι ολοκληρωμένοι δέκτες υπερύθρων	41

## ΜΕΡΟΣ Β

### ΤΟ ΡΟΜΠΟΤ ΑΚΟΥΣΤΙΚΩΝ ΜΕΤΡΗΣΕΩΝ

44

### ΚΕΦΑΛΑΙΟ 3

#### ΚΑΤΑΣΚΕΥΗ

3.1	Εισαγωγή	48
3.2	Το μηχανολογικό μέρος – σασί του ρομπότ	49
3.3	Ο Επεξεργαστής	54
3.4	Αισθητήρες Υπερύθρων	58
3.5	Κωδικοποιητές	63
	3.5.1 Εισαγωγή	63
	3.5.2 Encoder discs	64
	3.5.3 Ο αισθητήρας CNY70	66
3.6	Κινητήρες	71
	3.6.1 Εισαγωγή	71
	3.6.2 Ο servo κινητήρας SM-S4315R	72
	3.6.3 Ροπή κινητήρα	73
	3.6.4 Οδήγηση κινητήρων	75
3.7	Τροφοδοσία	77
	3.7.1 Μπαταρίες	77
	3.7.2 Κατανάλωση Ισχύος	80

### ΚΕΦΑΛΑΙΟ 4

#### Η ΛΕΙΤΟΥΡΓΙΑ ΤΟΥ ΡΟΜΠΟΤ

4.1	Εισαγωγή	82
4.2	Αποφυγή Εμποδίων	82
	4.2.1 Εισαγωγή	82
	4.2.2 Εντοπισμός εμποδίου	83
	4.2.3 Ευθυγράμμιση με εμπόδιο	88
4.3	Εφαρμογές Κωδικοποιητών	96
	4.3.1 Εισαγωγή – Κύλιση χωρίς ολίσθηση	96
	4.3.2 Υλοποίηση οδομέτρου	98
	4.3.3 Συντονισμός τροχών	101
4.4	Περιγραφή της λειτουργίας	105
4.5	Περιορισμοί	108
4.6	Ο κώδικας	110
4.7	Μηνύματα Εξόδου	122



4.8	Είσοδος Δεδομένων	128
-----	-------------------	-----

## ΚΕΦΑΛΑΙΟ 5

### ΣΦΑΛΜΑΤΑ ΚΑΙ ΒΕΛΤΙΩΣΕΙΣ

5.1	Σφάλματα	130
5.1.1	Εισαγωγή	130
5.1.2	Είδη Σφαλμάτων	130
5.2	Αντιμετώπιση Σφαλμάτων	133
5.2.1	Εισαγωγή	133
5.2.2	Αντικατάσταση Περιστρεφόμενου Τροχού	133
5.2.3	Αναβάθμιση Κωδικοποιητών	138
5.3	Κατασκευαστικές Μετατροπές	139
5.3.1	Εισαγωγή	139
5.3.2	Αισθητήρες Υπερήχων	140
5.3.3	Ο αισθητήρας υπερύθρων Sharp GP2D12	143
5.3.4	Άλλες κατασκευαστικές τροποποιήσεις	146

ΠΑΡΑΡΤΗΜΑ Α	Προγραμματισμός Μικροελεγκτή ΑΤmega168	148
-------------	--	-----

ΠΑΡΑΡΤΗΜΑ Β	Περιεχόμενα συνοδευτικού δίσκου (CD)	153
-------------	--------------------------------------	-----

ΒΙΒΛΙΟΓΡΑΦΙΑ		154
--------------	--	-----

# **ΜΕΡΟΣ Α**

# ΚΕΦΑΛΑΙΟ 1

## ΙΣΤΟΡΙΚΗ ΑΝΑΔΡΟΜΗ

### 1.1 Ρομπότ & Ρομποτική

#### 1.1.1 Εισαγωγή

Η ρομποτική είναι μια «διεπιστημονική περιοχή» που συνδυάζει τη φυσική, την ηλεκτρολογία, τη μηχανολογία, την πληροφορική, τις τηλεπικοινωνίες, τη θεωρία συστημάτων, τον αυτόματο έλεγχο, την τεχνητή νοημοσύνη, την τεχνολογία των αισθητήριων διατάξεων (δύναμης, αφής, όρασης, κλπ), την εικονική πραγματικότητα, την επεξεργασία σήματος, την υπολογιστική όραση και την τεχνητή ζωή. Η ρομποτική ανήκει στον ευέλικτο αυτοματισμό και είναι μια τεχνολογία με μέλλον και για το μέλλον. Η ιστορική αρχή της ανιχνεύεται στο «αυτόματο ρολόι νερού» του Έλληνα μηχανικού Κτησίβιου (~300 π.Χ.) και το μηχανισμό «αυτομάτου ανοίγματος – κλεισίματος θυρών» του Ήρωνα της Αλεξάνδρειας (~50μ.Χ). Ο όρος *ρομπότ* (ο οποίος προέρχεται από τη σλαβική λέξη *robota*), οφείλεται στον Τσέχο δραματικό συγγραφέα Karel Capek (1921) και σημαίνει «αναγκαστική εργασία».

Τα σημερινά ρομπότ αναπτύχθηκαν ταυτόχρονα με τον υπολογιστικό έλεγχο των εργαλειομηχανών. Το πρώτο βιομηχανικό ρομπότ κατασκευάστηκε στις ΗΠΑ από τους Devol και Engelberger και τέθηκε σε λειτουργία το 1960 σε ένα χυτήριο μετάλλων. Από τότε, στα πενήντα χρόνια που ακολούθησαν, η ανάπτυξη της ρομποτικής ήταν ραγδαία και σήμερα βρίσκονται σε δράση παγκοσμίως πάνω από 120.000 ρομπότ (με πρωταγωνιστές τις ΗΠΑ, την Ιαπωνία και την Ευρώπη) που καλύπτουν βιομηχανικές, ιατρικές, επιστημονικές και κοινωνικές εφαρμογές.

#### 1.1.2 Ορισμός του ρομπότ

Σύμφωνα με το Robot Institute of America, ως ρομπότ μπορούμε να ορίσουμε έναν μηχανισμό σχεδιασμένο ώστε, μέσω προγραμματιζόμενων κινήσεων, να μεταφέρει υλικά, τεμάχια, εργαλεία ή ειδικευμένες συσκευές, με σκοπό την επιτέλεση ποικιλίας εργασιών.

Ένας τέτοιος μηχανισμός περιλαμβάνει συνήθως τις ακόλουθες συνιστώσες :

- Ένα *μηχανολογικό υποσύστημα*, το οποίο ενσωματώνει τη δυνατότητα του ρομπότ για εκτέλεση έργου. Το υποσύστημα αυτό αποτελείται από μηχανισμούς που επιτρέπουν στο ρομπότ να κινείται, όπως αρθρώσεις, συστήματα μετάδοσης κίνησης, επενεργητές – κινητήρες, οδηγούς, κτλ.

- Ένα *υποσύστημα αίσθησης*, μέσω του οποίου το ρομπότ συγκεντρώνει πληροφορίες για την κατάσταση στην οποία βρίσκεται τόσο το ίδιο, όσο και το περιβάλλον του. Το υποσύστημα αυτό, εκτός των άλλων, είναι υπεύθυνο για την αποδοχή των εξωτερικών εντολών, την επεξεργασία τους, την μετάφρασή τους σε ηλεκτρική ισχύ που δοθεί στους κινητήρες του ρομπότ, καθώς επίσης και για την παραγωγή σημάτων εξόδου που θα πληροφορούν για την κατάσταση του συστήματος. Στο υποσύστημα αίσθησης περιλαμβάνονται όργανα μέτρησης, αισθητήρες, ηλεκτρονικά στοιχεία κτλ.
- Ένα *σύστημα ελέγχου*, το οποίο συνδυάζει κατάλληλα την αίσθηση με τη δράση, έτσι ώστε το ρομπότ να λειτουργεί αποτελεσματικά και με τον επιθυμητό τρόπο. Ο ελεγκτής του ρομπότ επιβλέπει και συντονίζει ολόκληρο το σύστημα, για τη σχεδίαση και την υλοποίησή του δε, απαιτείται ο συνδυασμός γνώσεων από πολλές γνωστικές περιοχές, όπως είναι ο αυτόματος έλεγχος, η τεχνητή νοημοσύνη, η επιστήμη των υπολογιστών κτλ.

### 1.1.3 Οι κατηγορίες των ρομπότ

Ως ρομπότ χαρακτηρίζεται κάθε ελεγχόμενη από υπολογιστή μηχανή που μπορεί να εκτελέσει εργασίες τις οποίες κάνει ο άνθρωπος. Τα ρομπότ της πρώτης γενιάς δεν είχαν ικανότητα υπολογισμού και αίσθησης, ενώ τα ρομπότ της 2<sup>ης</sup> γενιάς διαθέτουν περιορισμένη υπολογιστική ικανότητα. Τα ρομπότ της 3<sup>ης</sup> γενιάς διαθέτουν «*νοημοσύνη*» (είναι όπως λέμε «έξυπνα ρομπότ») και μπορούν να λύνουν προβλήματα και να παίρνουν αποφάσεις κατά τη διάρκεια της εργασίας τους. Τις ικανότητες αυτές αποκτούν με τεχνικές «*τεχνητής νοημοσύνης*» και «*αίσθησης*».

Το ρεπερτόριο των εφαρμογών διευρύνεται συνεχώς σε νέα πεδία της ανθρώπινης δραστηριότητας και οι επιστήμονες συνεχίζουν αδιάκοπα την προσπάθεια ανάπτυξης και κατασκευής αληθινά «*έξυπνων ρομπότ*» τα οποία να μπορούν να συμπεριφέρονται, όσο γίνεται πιο πολύ, όπως και ο άνθρωπος.

Στη λαϊκή αντίληψη τα ρομπότ συνδέθηκαν από τη δεκαετία του 1960 με ταινίες όπως «*Η τελευταία ημέρα του κόσμου*» και «*Ο πόλεμος των άστρων*». Χαρακτηριστικά παραδείγματα ρομπότ με συγκεκριμένους ρόλους ανθρώπινης μορφής είναι τα ρομπότ R2D2 και C3PO.

Σήμερα υπάρχει ένας θαυμαστός κόσμος ρομπότ που μπορούν να μετακινούνται, να βαδίζουν, να βλέπουν, να ομιλούν και να εκτελούν λεπτούς χειρισμούς που απαιτούν έξυπνάδα και επιδεξιότητα. Ο κόσμος αυτός διαρκώς εξαπλώνεται και ικανοποιεί τις ανάγκες εργασίας, παραγωγής, υγείας, ευημερίας και ψυχαγωγίας του ανθρώπου. Οι πέντε βασικές κατηγορίες ρομπότ είναι:

- *Βιομηχανικά ρομπότ*
- *Κινητά ρομπότ*
- *Ιατρικά ρομπότ*
- *Τηλερομπότ*
- *Κοινωνικά ρομπότ*

## **Βιομηχανικά ρομπότ**

Τα βιομηχανικά ρομπότ (ή ρομποτικοί βραχίονες) έχουν τη μορφή ενός ανθρώπινου βραχίονα με αρθρώσεις (ώμο, αγκώνα, καρπό) και παλάμη (αρπάγη/δαγκάνα, δάκτυλα). Η επιλογή του τύπου της κίνησής τους (γραμμική, κυλινδρική, σφαιρική, αρθρωτή) εξαρτάται από το είδος της εργασίας που πρέπει να εκτελέσουν.

Τα βιομηχανικά ρομπότ είναι κατάλληλα για επαναλαμβανόμενες εργασίες σε πλήρως δομημένα και σταθερά περιβάλλοντα. Τέτοιες εργασίες είναι: φόρτωμα/ξεφόρτωμα μηχανών, συναρμολόγηση, συγκόλληση, πρεσάρισμα, βαφή, γυάλισμα, κοκ. Τα πλεονεκτήματα που παρέχουν τα βιομηχανικά ρομπότ είναι: απαλλαγή των εργαζομένων από κουραστικές, ανιαρές και επικίνδυνες εργασίες ευελιξία, υψηλή παραγωγικότητα, καλύτερη ποιότητα προϊόντος και βελτιωμένη ποιότητα ζωής.

## **Κινούμενα ρομπότ**

Τα ρομπότ αυτά αποτελούνται από μια πλατφόρμα (όχημα) με ρόδες (3 ή 4) η οποία κινείται με κατάλληλο πρόγραμμα ελέγχου και είναι εφοδιασμένη με αισθητήρες όρασης (κάμερες), υπερήχων, απόστασης κ.α. Πάνω στην πλατφόρμα μπορεί να είναι προσαρμοσμένοι ρομποτικοί βραχίονες (ένας ή περισσότεροι) για την εκτέλεση εργασιών.

Τα ρομπότ του είδους αυτού, που καλούνται «κινούμενοι ρομποτικοί χειριστές», χρησιμοποιούνται για προσφορά υπηρεσιών, όπως μεταφορά υγειονομικού και λοιπού υλικού στα νοσοκομεία, μεταφορά φαρμάκων σε μεγάλες φαρμακαποθήκες, συλλογή φρούτων από δέντρα, κούρεμα προβάτων, κ.α. Χρησιμοποιούνται επίσης σε υποθαλάσσιες έρευνες για τη συλλογή οργανισμών, καθιζημάτων και άλλων αντικειμένων σε βάθη ωκεανών που είναι απαγορευτικά για τον άνθρωπο, αλλά και σε έρευνες στο εσωτερικό ηφαιστειών.

## **Ιατρικά ρομπότ**

Τα ιατρικά ρομπότ διακρίνονται σε «μακρο-ρομπότ» (χειρουργικά ρομπότ, ρομπότ αποκατάστασης ΑΜΕΑ, αυτόνομες ρομποτικές καρέκλες) και «μικρο-ρομπότ» (για καθοδηγούμενη από εικόνες χειρουργική, ελάχιστης επέμβασης/ενδοσκοπική χειρουργική, αγγειοπλαστική, εμβολισμός (γέμισμα) εγκεφαλικών ανευρυσμάτων κ.α.). Τα ιατρικά ρομπότ ενισχύονται σημαντικά από τηλεχειριστές και εικονική πραγματικότητα, ιδιαίτερα όταν ο ασθενής δεν μπορεί να μεταφερθεί στον τόπο του ειδικευμένου χειρουργού (τραυματίες πολέμου, ασθενείς απομακρυσμένων νησιών κ.λπ.). Ένα ιατρικό ρομπότ ευρείας χρήσης είναι το χειρουργικό ρομπότ Da Vinci.

## **Τηλερομπότ**

Τα τηλερομπότ συνδυάζουν τηλεχειρισμό από τον άνθρωπο και αυτονομία και μπορούν να λειτουργήσουν τόσο σε ημιδομημένα όσο και σε πλήρως αδόμητα περιβάλλοντα. Μπορούν να εκτελούν μη επαναλαμβανόμενες εργασίες χωρίς να έχουν τέλεια γνώση του χώρου εργασίας τους. Το μεγαλύτερο πρόβλημά τους είναι οι μεταβαλλόμενες χρονικές καθυστερήσεις ανάμεσα στο ρομπότ και το χειριστή, που οφείλονται κυρίως στα συστήματα επικοινωνίας. Οι κυριότερες εφαρμογές τους είναι οι ιατρικές, οι υποθαλάσσιες και οι διαστημικές εφαρμογές.

## Κοινωνικά ρομπότ

Κοινωνικό ρομπότ είναι ένα αυτόνομο ρομπότ που επικοινωνεί και αλληλεπιδρά με τον άνθρωπο ακολουθώντας κανόνες κοινωνικής συμπεριφοράς τους οποίους έχει διδαχθεί και μάθει. Οι τρεις βασικοί κανόνες τους οποίους πρέπει να ακολουθεί ένα κοινωνικό ρομπότ (πέρα από τους ειδικούς κανόνες ανθρώπινης συμπεριφοράς) είναι οι τρεις ρομποτικοί νόμοι του Ρώσου συγγραφέα Isaac Asimov που δημοσίευσε το 1941 στο μυθιστόρημα επιστημονικής φαντασίας (I, Robot).

Οι νόμοι αυτοί είναι: (1) Ένα ρομπότ δεν πρέπει να βλάψει τον άνθρωπο ενεργά ή παθητικά, (2) Ένα ρομπότ πρέπει να υπακούει στον άνθρωπο εκτός εάν αυτό αντίκειται στο νόμο 1, (3) Ένα ρομπότ πρέπει να προστατεύει την ύπαρξή του εφόσον τούτο δεν αντιβαίνει στους δύο προηγούμενους νόμους.

Στα κοινωνικά ρομπότ ανήκουν και τα ανθρωποειδή ρομπότ που μπορούν να βαδίζουν και πολλά από αυτά έχουν ανθρώπινη μορφή (πρόσωπο, χέρια, κ.λπ.). Οι ικανότητές τους εξαρτώνται από τις εργασίες που πρέπει να εκτελέσουν. Για παράδειγμα, ένα ρομπότ σερβιτόρος πρέπει να ακολουθεί τους κανόνες καλής εξυπηρέτησης.

Τρία γνωστά κοινωνικά ρομπότ είναι το ρομπότ «Kismet» (μοίρα/ειμαρμένη στην Τουρκική), το ρομπότ «μουσικός» και το ρομπότ «Asimo» της Honda. Το Kismet, είναι ένα ρομποτικό κεφάλι με στόμα, μάτια και αυτιά που μπορεί να αποκρίνεται με συναισθηματικούς μορφασμούς (χαράς, θαυμασμού, έκπληξης, θυμού) ανάλογα με την περίπτωση που αντιμετωπίζει.

### 1.1.4 Το μέλλον των ρομπότ

Τα σύγχρονα ρομπότ έχουν μηχανικές και νοητικές ικανότητες που προηγουμένως ανήκαν στη σφαίρα επιστημονικής φαντασίας. Η ανάπτυξή τους θα συνεχίσει να επεκτείνεται με ολοένα νέα είδη ρομπότ κατάλληλα για τη βιομηχανία, την επιστήμη και την καθημερινή ζωή του ανθρώπου. Ήδη σήμερα υπάρχουν ιπτάμενα μη επανδρωμένα οχήματα-ρομπότ, ρομποτικοί οδηγοί (ρομποτικά μπαστούνια) τυφλών, ρομπότ ποδοσφαιριστές, πολύποδα ρομπότ ανίχνευσης ηφαιστειών, σμήνη συνεργαζόμενων ρομπότ, ρομποτικά έντομα, κοκ. Η έρευνα και ανάπτυξη προς την κατεύθυνση αυτή συνεχίζεται αδιάκοπα έχοντας ως βασική προτεραιότητα την ασφάλεια και άνεση του ανθρώπου και την απαλλαγή του από δύσκολες, επίπονες και επικίνδυνες εργασίες.

Στο μέλλον ο άνθρωπος πιθανώς θα ζει ανάμεσα στα ρομπότ στο σπίτι, το δρόμο, την εργασία, το νοσοκομείο, κλπ και θα συμβιώνει με αυτά για την επιτυχία του κοινού στόχου υγείας, υψηλής ποιότητας ζωής και μακροζωίας.

Ο άνθρωπος φανταζόταν πάντοτε μηχανές που να του μοιάζουν και να τον υπηρετούν στην καθημερινή του ζωή. Στην ελληνική μυθολογία είναι γνωστός ο *Τάλως*, το πρώτο ρομπότ που κατασκευάστηκε από τον Δαίδαλο (ή Ηφαιστο) με εντολή του Δία και δόθηκε στον Μίνωα για την προστασία της Κρήτης από τους εχθρούς της.

Σήμερα σχεδιάζονται και κατασκευάζονται «έξυπνα ρομπότ» για τη βιομηχανία, τις υπηρεσίες, τις οικιακές εργασίες, αλλά δυστυχώς και για πολεμικούς σκοπούς. Όπως και άλλα επιτεύγματα της επιστήμης (λ.χ., της πυρηνικής φυσικής και τεχνολογίας) έχουν χρησιμοποιηθεί εναντίον του ανθρώπου, έτσι και τα ρομπότ μπορούν να χρησιμοποιηθούν (και έχουν χρησιμοποιηθεί) για καταστροφικούς σκοπούς. Οι αυτοκαθοδηγούμενες βόμβες λέιζερ δεν είναι παρά «έξυπνα ρομπότ».

Ο αρχικός φόβος ότι ο αυτοματισμός θα αυξήσει την ανεργία αποδείχθηκε στην πράξη ότι δεν ισχύει, γιατί ο αυτοματισμός (μηχανοποίηση, ρομποτική, πληροφορική) δημιούργησαν πολύ περισσότερα νέα επαγγέλματα από όσα εξαφάνισαν. Έτσι συνολικά ο αυτοματισμός οδήγησε σε μείωση της ανεργίας, η οποία όμως εξαρτάται από την οικονομική κατάσταση τόσο στο επίπεδο κάθε χώρας όσο και διεθνώς.

Η πορεία της ρομποτικής δείχνει ότι αυτή θα συνεχίσει να αναπτύσσεται για πολύ μεγάλο χρονικό διάστημα. Στην ανάπτυξη αυτή μπορούν και πρέπει να συμβάλλουν φυσικοί, μαθηματικοί, μηχανικοί, πληροφορικοί, αλλά και γιατροί, χημικοί, βιολόγοι, και οικονομοτεχνικοί επιστήμονες.

## **1.2 Ιστορία των κινούμενων ρομπότ**

Στην παράγραφο αυτήν, θα κάνουμε μία εκτεταμένη αναφορά στην ιστορία των κινούμενων ρομπότ, η οποία και έχει στιγματίσει την πορεία της τεχνολογίας και της επιστήμης, από τα μέσα του 20<sup>ου</sup> αιώνα μέχρι σήμερα. Η εξέλιξη των κινούμενων ρομπότ προέκυψε με σκοπό την επιτέλεση διάφορων λειτουργιών, από την εκτέλεση απλών καθημερινών εργασιών, μέχρι τη διενέργεια πολύπλοκων ερευνών, ενώ σήμερα χρησιμοποιούνται και στις πλέον μεγάλης σημασίας διαστημικές αποστολές.

1939 – 1945 : Κατά τη διάρκεια του Β' Παγκοσμίου Πολέμου, εμφανίστηκαν τα πρώτα κινούμενα ρομπότ, σαν αποτέλεσμα της τεχνολογικής προόδου ενός αριθμού από νέα σχετικά ερευνητικά πεδία, όπως η Επιστήμη Υπολογιστών ή η Κυβερνητική.

1948–1949 : Ο William Grey Walter κατασκεύασε τα ρομπότ Elmer και Elsie, δυο αυτόνομα ρομπότ, τα οποία ονομάστηκαν Machina Speculatrix, επειδή τους άρεσε να εξερευνούν το περιβάλλον τους. Το κάθε ένα από τα ρομπότ Elmer και Elsie ήταν εξοπλισμένο με έναν αισθητήρα φωτός. Αν έβρισκαν κάποια πηγή φωτός κινούνταν προς αυτήν, αποφεύγοντας παράλληλα τα εμπόδια που βρίσκονταν στον δρόμο τους. Αυτά τα ρομπότ αποδείκνυαν ότι μία πολύπλοκη συμπεριφορά μπορούσε να προκύψει από έναν απλό σχεδιασμό.

1961–1963 : Το Johns Hopkins University ανέπτυξε το ρομπότ 'Beast'. Το ρομπότ Beast για να κινηθεί χρησιμοποιούσε ένα sonar. Όταν οι μπαταρίες του αποφορτιζόνταν, τότε το ρομπότ θα εντόπιζε μία βάση φόρτισης με την οποία και θα συνδεόταν.

1969 : Το Mowbot ήταν το πρώτο ρομπότ το οποίο κούρευε αυτόματα το γρασίδι.

1970 : Το Cart line follower του Πανεπιστημίου Stanford, ήταν ένα κινούμενο ρομπότ το οποίο ήταν ικανό να ακολουθήσει μία άσπρη γραμμή, χρησιμοποιώντας μία κάμερα. Ήταν ασύρματα συνδεδεμένο με έναν μεγάλο mainframe, ο οποίος και πραγματοποιούσε τους υπολογισμούς.

Την ίδια περίπου περίοδο (1966–1972), το Ερευνητικό Κέντρο του Stanford κατασκευάζει και πραγματοποιεί έρευνα πάνω σε ένα ρομπότ που ονομάστηκε Shakey, εξαιτίας της σπασμωδικής του κίνησης. Το Shakey ήταν εξοπλισμένο με μία κάμερα, αισθητήρα μέτρησης της απόστασης, αισθητήρες επαφής καθώς επίσης και κεραία για ασύρματη επικοινωνία. Το Shakey ήταν το πρώτο ρομπότ που επιχειρηματολογούσε για τις πράξεις του. Αυτό σημαίνει ότι δίνονταν στο ρομπότ πολύ γενικές εντολές, και εκείνο θα αντιλαμβανόταν τα απαραίτητα βήματα που έπρεπε να κάνει, ώστε να φέρει εις πέρας την ζητούμενη εργασία.

Τέλος, η Σοβιετική Ένωση εξερευνάει την επιφάνεια της Σελήνης, με το Lunokhod 1, ένα σεληνιακό ρόβερ.

1976 : Με το πρόγραμμα Viking, η NASA στέλνει δύο μη επανδρωμένα διαστημικά σκάφη στον Άρη.

1980 : Το ενδιαφέρον του κοινού γύρω από τα ρομπότ αυξάνεται, και αυτό έχει ως αποτέλεσμα την σταδιακή παραγωγή ρομπότ τα οποία μπορούν να αγοραστούν για οικιακή χρήση. Τα ρομπότ αυτά εξυπηρετούσαν ψυχαγωγικούς ή εκπαιδευτικούς σκοπούς. Παραδείγματα αποτελούν το ρομπότ RB5X (το οποίο υπάρχει και σήμερα) και η σειρά HERO (Heathkit Educational RObot).

Επίσης, το Cart του Stanford είναι πλέον ικανό να πλοηγείται μεταξύ εμποδίων και να χαρτογραφεί το περιβάλλον του.

Αρχές 1980 : Η ομάδα του Ernst Dickmanns κατασκευάζει στο Bundeswehr University του Μονάχου, τα πρώτα ρομπότ – αυτοκίνητα, τα οποία είχαν τη δυνατότητα να κινηθούν έως και 55mph σε άδειους δρόμους.

1987 : Το HRL (Hughes Research Laboratories) κατασκευάζει την πρώτη, χαρτογραφημένη και βασιζόμενη σε αισθητήρες, αυτόνομη λειτουργία ενός ρομποτικού οχήματος.

1989 : Ο Mark Tilden επινοεί την ρομποτική BEAM (Biology, Electronics, Aesthetics, Mechanics).

Αρχές 1990 : Ο Joseph Engelberger, πατέρας του βιομηχανικού ρομποτικού βραχίονα, εργάζεται με συναδέλφους προκειμένου να σχεδιάσει τα πρώτα εμπορικά διαθέσιμα αυτόνομα κινούμενα ρομπότ για νοσοκομειακή χρήση (τα οποία επρόκειτο να βγουν στην αγορά από την HelpMate).

Την ίδια περίοδο, το Υπουργείο Άμυνας των ΗΠΑ επενδύει πάνω στο project MDARS-I, το οποίο βασίζεται στα ρομπότ ασφάλειας εσωτερικών χώρων.

1991 : Ο Edo. Franzi, ο André Guignard και ο Francesco Mondada ανέπτυξαν το Khepera, ένα μικρό αυτόνομο κινούμενο ρομπότ, που έχει ως σκοπό ερευνητικές δραστηριότητες. Το συγκεκριμένο project υποστηρίχτηκε από το εργαστήριο του Πολυτεχνείου της Λωζάνης, LAMI-EPFL lab.



1993–1994 : Τα Dante I και Dante II αναπτύχθηκαν από το Carnegie Mellon University. Και τα δύο ήταν βαδίζοντα ρομπότ που είχαν σκοπό την εξερεύνηση ενεργών ηφαιστείων.

1994 : Τα 2 δίδυμα ρομπότ – οχήματα VaMP της Daimler-Benz και VITA-2 της UniBwM, ταξίδεψαν (με επιβαίνοντες σε αυτά) περισσότερα από 1000km σε αυτοκινητόδρομο του Παρισιού 3 λωρίδων και με φυσιολογικά υψηλή κίνηση, με ταχύτητες κοντά στα 130 km/h. Τα ρομπότ αυτά επιδείκνυαν αυτόνομη οδήγηση σε ελεύθερες λωρίδες, οδήγηση σε φάλαγγα αυτοκινήτων, καθώς και αλλαγές λωρίδας (αριστερά και δεξιά), με αυτόνομη προσπέραση άλλων αυτοκινήτων.

1995 : Το ημιαυτόνομο αυτοκίνητο ALVINN ήταν κατευθυνόμενο υπό τον έλεγχο ηλεκτρονικού υπολογιστή, για μία πολύ μεγάλη διαδρομή. Παρόλα αυτά, γκάζι και φρένο ήταν ελεγχόμενα από άνθρωπο οδηγό.

1995 : Την ίδια χρονιά, ένα από τα ρομπότ – αυτοκίνητα του Ernst Dickmanns (με αυτοελεγχόμενα γκάζι και φρένα) ταξίδεψε περισσότερα από 1000 μίλια, από το Μόναχο στην Κοπεγχάγη, και επέστρεψε, εν μέσω κίνησης, με ταχύτητες κοντά στα 120 mph, εκτελώντας περιστασιακά ελιγμούς για να προσπεράσει άλλα αυτοκίνητα (μόνο σε κάποιες κρίσιμες περιπτώσεις, ένας οδηγός ασφαλείας ανέλαβε τον χειρισμό). Ενεργής όραση χρησιμοποιήθηκε για να αντιμετωπίσει τις ξαφνικές αλλαγές σε ορισμένα σημεία του δρόμου.

1995 : Το προγραμματιζόμενο κινούμενο ρομπότ της Pioneer γίνεται εμπορικά διαθέσιμο σε μία προσιτή τιμή, επιτρέποντας την εκτεταμένη ανάπτυξη στην έρευνα και την πανεπιστημιακή μελέτη της ρομποτικής μέσα στην επόμενη δεκαετία, καθώς η ρομποτική κίνηση γίνεται βασικό μέρος του κύκλου σπουδών πολλών πανεπιστημίων.

1996–1997 : Η NASA στέλνει στον Άρη το Mars Pathfinder μαζί με το ρόβερ του, το Sojourner, το οποίο είχε ως σκοπό την εξερεύνηση της επιφάνειας, δεχόμενο εντολές από τη Γη. Το Sojourner ήταν εξοπλισμένο με σύστημα αποφυγής κινδύνων, το οποίο επέτρεπε στο ρόβερ να βρίσκει αυτόνομα το δρόμο του μέσα στο άγνωστο έδαφος του Άρη.

1999 : Η Sony παρουσιάζει τον Aibo, έναν ρομποτικό σκύλο με δυνατότητα να βλέπει, να βαδίζει και να αλληλεπιδρά με το περιβάλλον του. Επίσης, παρουσιάζεται το τηλεκατευθυνόμενο στρατιωτικό ρομπότ PackBot.

2001 : Ξεκινάει το Swarm-bots project. Τα Swarm bots μοιάζουν με αποικίες εντόμων, και αποτελούνται από έναν μεγάλο αριθμό μικρών ανεξάρτητων ρομπότ, τα οποία αλληλεπιδρούν μεταξύ τους και όλα μαζί εκτελούν πολύπλοκες εργασίες.

2002 : Εμφανίζεται το Roomba, ένα οικιακό αυτόνομο κινούμενο ρομπότ που καθαρίζει το πάτωμα.

2003 : Η Axxon Robotics αγοράζει την Intellibot, κατασκευάστρια εταιρία μίας σειράς εμπορικών ρομπότ τα οποία τρίβουν και σκουπίζουν τα πατώματα νοσοκομείων, γραφείων κτλ. Τα ρομπότ περιποίησης δαπέδου από την Intellibot Robotics, λειτουργούν εντελώς αυτόνομα, χαρτογραφώντας το περιβάλλον τους και χρησιμοποιώντας ένα σύνολο αισθητήρων για την αποφυγή εμποδίων κατά την πλοήγησή τους.

2004 : Γίνεται εμπορικά διαθέσιμο το Robosapien, ένα βιομετρικό ρομπότ – παιχνίδι σχεδιασμένο από τον Mark Tilden.

Επίσης, με το 'The Centibots Project', 100 αυτόνομα ρομπότ λειτουργούν μαζί, προκειμένου να καταρτίσουν τον χάρτη ενός άγνωστου χώρου, και ψάχνουν για αντικείμενα μέσα στον χώρο αυτόν.

Τέλος, στον πρώτο διαγωνισμό DARPA Grand Challenge, πλήρως αυτόνομα οχήματα συναγωνίζονται, σε έναν κενό χώρο.

2005 : Η Boston Dynamics δημιουργεί ένα τετράποδο ρομπότ, προκειμένου να κουβαλάει βαριά φορτία, σε έδαφος που είναι πολύ ανώμαλο για οχήματα.

2006 : Η Sony σταματάει την κατασκευή του Aibo, και η HelpMate διακόπτει την παραγωγή της, αλλά το χαμηλότερου κόστους, αυτόνομο ρομπότ – βοηθός PatrolBot γίνεται διαθέσιμο, στην συνεχόμενη προσπάθεια να γίνουν τα κινούμενα ρομπότ εμπορικώς βιώσιμα.

Το Υπουργείο Άμυνας των ΗΠΑ διακόπτει το πρόγραμμα MDARS-I, αλλά επενδύει στο MDARS-E, ένα αυτόνομο ρομπότ ειδικά κατασκευασμένο για την εκτέλεση γεωργικών εργασιών.

Επίσης, παρουσιάζεται το TALON-Sword, το πρώτο εμπορικά διαθέσιμο ρομπότ με εκτοξευτή χειροβομβίδων και άλλες ενσωματωμένες οπλικές επιλογές.

Τέλος, το Asimo της Honda εκπαιδεύεται να τρέχει και να ανεβαίνει σκάλες.

2007 : Γράφεται ιστορία με το DARPA Urban Grand Challenge, όπου έξι οχήματα ολοκληρώνουν αυτόνομα μία πολύπλοκη πορεία, η οποία περιλάμβανε άλλα επανδρωμένα οχήματα και εμπόδια.

Εξαπλώνονται τα έξυπνα ρομπότ στην διανομή εργασιών, της Kiva Systems. Αυτές οι ευφυείς μονάδες ταξινόμησης, κατατάσσουν τη λειτουργία τους ανάλογα με τη δημοτικότητα του περιεχομένου της.

Το Tug γίνεται ένα δημοφιλές μέσο για την μετακίνηση νοσοκομειακών αποθεμάτων μεγάλου φορτίου, ενώ το Speci-Minder και το Motivity ξεκινούν τη μεταφορά αίματος και άλλων δειγμάτων των ασθενών, από τους σταθμούς των νοσοκόμων σε διάφορα εργαστήρια.

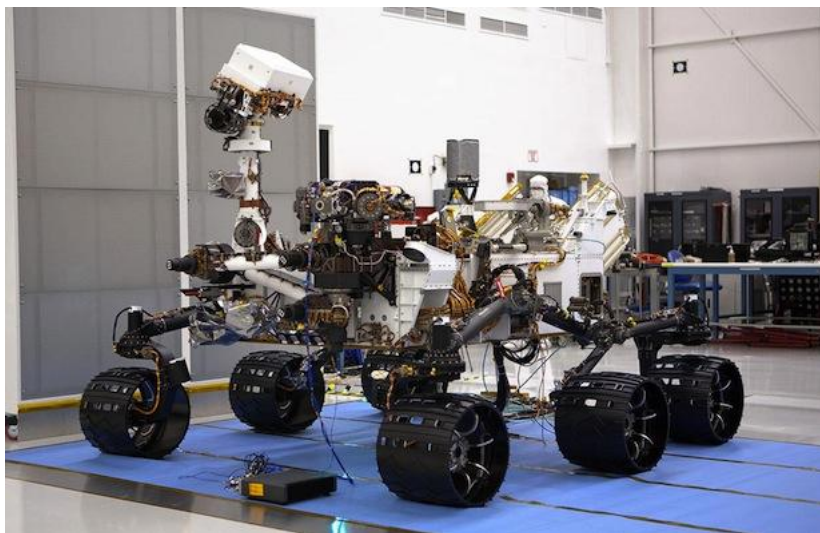
Το Seekur, το πρώτο ευρέως διαθέσιμο, μη στρατιωτικό, βοηθητικό ρομπότ εξωτερικών χώρων, έχει τη δυνατότητα να ρυμουλκήσει ένα τριών τόνων όχημα από μία θέση παρκινγκ, να το οδηγήσει αυτόνομα σε έναν εσωτερικό χώρο και εν συνεχεία, να καθοδηγηθεί και πάλι προς τα έξω.

Εν τω μεταξύ, το PatrolBot εκπαιδεύεται να ακολουθεί ανθρώπους και να εντοπίζει πόρτες που είναι μισάνοιχτες.

2008 : Η Boston Dynamics κυκλοφορεί το απόσπασμα ενός βίντεο, για μίας νέας γενιάς BigDog ικανό να βαδίζει σε παγωμένο έδαφος και να ανακτά την ισορροπία του σπρώχνεται από το πλάι.

2010 : Ο διαγωνισμός Multi Autonomous Ground-robotic International Challenge (MAGIC) παρουσιάζει ομάδες αυτόνομων οχημάτων που χαρτογραφούν ένα μεγάλο αστικό περιβάλλον, που αναγνωρίζουν και παρακολουθούν ανθρώπους, και που αποφεύγουν εχθρικά αντικείμενα.

Στο σημείο αυτό, δεν θα μπορούσαμε να μην αναφέρουμε την πολύ πρόσφατη προσεδάφιση στον Άρη, του ρομπότ Curiosity της NASA. Το πυρηνοκίνητο Curiosity (ένα εξάτροχο όχημα ενός τόνου) αποτελεί το μεγαλύτερο τροχοφόρο ρομπότ που έχει εκτοξευτεί μέχρι σήμερα στο Διάστημα. Το ρομπότ των δύομισι δισεκατομμυρίων δολαρίων, πέντε φορές μεγαλύτερο από τα δίδυμα ρομπότ Spirit και Opportunity που προσεδαφίστηκαν στον Άρη το 2004, εκτοξεύτηκε τον Νοέμβριο του 2011 με κύριο αντικειμενικό στόχο να εξετάσει αν ο γειτονικός πλανήτης ήταν ποτέ φιλόξενος για τη ζωή. Το Curiosity προσεδαφίστηκε μέσα στον κρατήρα Γκέιλ, το μεγάλο βάθος του οποίου, θα επιτρέψει στο ρομποτικό γεωλόγο να εξετάσει τι κρύβεται κάτω από την απέραντη, άνυδρη έρημο που καλύπτει σήμερα σχεδόν ολόκληρο τον πλανήτη. Τροφοδοτούμενο από μία θερμοηλεκτρική γεννήτρια πλουτωνίου, το ρομπότ θα περάσει στη συνέχεια τουλάχιστον ένα γήινο έτος εξετάζοντας τα πετρώματα με κάμερες, λέιζερ και άλλα όργανα.

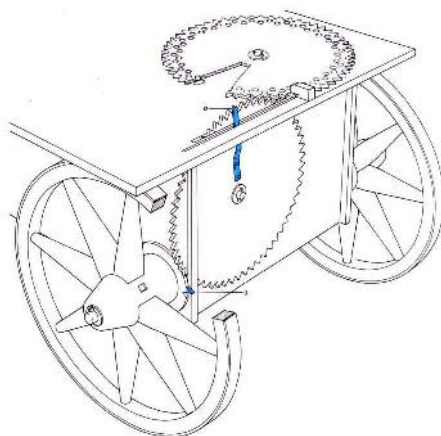


*Σχήμα 1.2.1 Το πυρηνοκίνητο τροχοφόρο ρομπότ Curiosity της NASA.*

Μετά την αποστολή του ρομπότ Curiosity, η NASA ετοιμάζει νέα αποστολή για τον πλανήτη Άρη, αυτήν του ρομπότ InSight, το οποίο και θα είναι πολύ πιο προηγμένης τεχνολογίας από το Curiosity. Σκοπός του θα είναι να εξερευνήσει σε τι κατάσταση βρίσκεται ο πυρήνας του Άρη, αλλά και η πραγματοποίηση σεισμολογικών ερευνών στον πλανήτη. Τον έλεγχο της αποστολής θα έχει η Jet Propulsion Laboratory, όπως συμβαίνει και με την αποστολή του Curiosity. Για την κατασκευή του ρομπότ θα χρειαστεί η συνεργασία του κέντρου Αεροδιαστήματος της Γερμανίας και της Γαλλικής Διαστημικής Υπηρεσίας, ενώ στο έργο θα συμμετάσχουν και πανεπιστήμια από Ευρώπη και Ηνωμένες Πολιτείες. Το κόστος κατασκευής υπολογίζεται στα 350 εκατομμύρια ευρώ, ενώ το ταξίδι του θα ξεκινήσει το 2016.

### 1.3 Οδόμετρο του Αρχιμήδη

Ένα από τα πιο βασικά ζητήματα της παρούσας διπλωματικής εργασίας, είναι η συνεχόμενη εκτίμηση της θέσης ενός κινούμενου ρομπούτ. Η πραγματοποίηση της συγκεκριμένης διαδικασίας θα γίνει, όπως θα δούμε στα επόμενα κεφάλαια, με τεχνικές *οδομετρίας*, δηλαδή με συνεχόμενη μέτρηση της απόστασης που διανύεται από το ρομποτικό όχημα. Οι τεχνικές αυτές βασίζονται στη λογική του *Οδομέτρου του Αρχιμήδη*, το οποίο και θα περιγράψουμε στην συνέχεια, μαζί με μία σύντομη ιστορική αναδρομή.



Σχήμα 1.3.1. Το οδόμετρο του Αρχιμήδη.

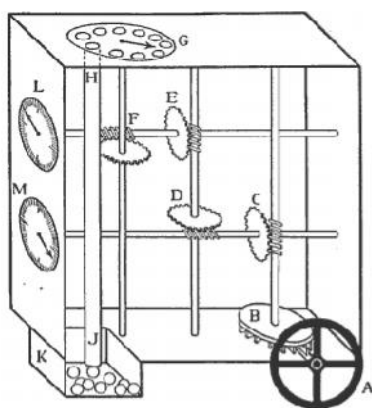
Το οδόμετρο θεωρείται ο πρόδρομος του σημερινού κοντέρ, του οργάνου που μετρά τη χιλιομετρική απόσταση, ενώ εφευρέτης του ήταν ο μεγαλύτερος μαθηματικός της αρχαιότητας, ο Αρχιμήδης. Το οδόμετρο που περιγράφει ο Ρωμαίος αρχιτέκτονας και μηχανικός Βιτρούβιος, κατασκευάστηκε τον 1ο αιώνα π.Χ. και είναι μηχανισμός που προσαρμοζόταν στους τροχούς μιας άμαξας και κατέγραφε την απόσταση που αυτή είχε διανύσει. Περιλαμβάνει ξύλινους οδοντωτούς τροχούς κι αυτή είναι η βασική του διαφορά από το οδόμετρο που περιγράφει ο Ήρωνας, το οποίο χρησιμοποιεί ατέρμονες κοχλίες.

Πρώτη αναφορά, λοιπόν, στο οδόμετρο γίνεται από τον Βιτρούβιο, που αφιερώνει και ειδικό τμήμα γι' αυτό χωρίς όμως να παρέχεται καμία απεικόνιση του οργάνου. Με την ανάγνωση αυτού του τμήματος γίνεται αμέσως φανερό ότι ο συγγραφέας περιγράφει κάποιο μηχανήμα προγενέστερο της εποχής του, του οποίου πιθανότατα είχε δει μόνο τα υπολείμματα και γι' αυτό δεν το απεικόνισε. Ίσως επρόκειτο για κάποια μορφή άμαξας που διέθετε έναν δυσανάλογα μεγάλο τροχό, αναγκαίο για το οδόμετρο και κατά πάσα πιθανότητα δεν επρόκειτο για χρηστικό μηχανήμα, γνώμη με την οποία συμφωνεί και ο D.Priice σε σχετική εργασία του. Αυτή η ασάφεια ως προς την περιγραφή και η ανυπαρξία κάποιου σχεδίου είναι μια από τις αιτίες που έχει καθυστερήσει η έρευνα γύρω από το μηχανισμό αυτό. Μολονότι ο Βιτρούβιος δεν ονομάζει κάποιον ως τον εφευρέτη και κατασκευαστή του μηχανισμού, εντούτοις η σύγχρονη έρευνα έχει αποδώσει την εφεύρεση του οργάνου στον Αρχιμήδη, δεδομένου ότι υπάρχει και μια αναφορά από τον ιστοριοδίφη Ιωάννη Τζέτζη σχετικά με την ανακάλυψή του από τον Συρακούσιο.

Το όργανο αυτό το γνωρίζουμε επακριβώς χάρη στην περιγραφή του Ήωνα του Αλεξανδρινού, μεγάλου μαθηματικού και μηχανικού της ελληνιστικής περιόδου, την οποία

ενσωμάτωσε στο έργο του (όταν είχε διατελέσει διευθυντής του περιφημου Μουσείου της Αλεξάνδρειας, ενός επιστημονικού κέντρου εφάμιλλου με τα σημερινά πολυτεχνεία).

Το μηχανικό οδόμετρο ή αλλιώς και «δρομόμετρο», αποτελείται από ένα σύμπλεγμα οδοντωτών τροχών (γρاناζιών), που με την βοήθεια ελίκων μεταφέρουν την κίνηση του οχήματος και την μετατρέπουν σε μονάδες μέτρησης. Έτσι είναι εύκολο σε οποιονδήποτε να ενημερωθεί για την απόσταση που έχει διανύσει το όχημα, συμβουλευόμενος την διαβαθμισμένη πλάκα που υπάρχει στην επάνω πλευρά του κιβωτίου το οποίο περικλείει το μηχανισμό, ένα μηχανισμό που σύγχρονοι ερευνητές ονόμασαν «ταξίμετρο», όπως φαίνεται και στο παρακάτω σχήμα.



Σχήμα 1.3.2. Περιγραφή της λειτουργίας του οδομέτρου.

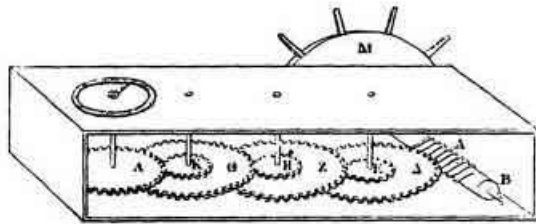
Η απόδοση της εφεύρεσης του οδομέτρου στον Αρχιμήδη, εκτός από τις φιλολογικές υπομνήσεις, στηρίζεται και σε ανάλογους συσχετισμούς. Λόγου χάρη, η άποψη του Sleeswyk, συνεπικουρούμενη από τις έρευνες του Hill τις σχετικές με το υδραυλικό ρολόι είναι αξιοπρόσεκτη. Αναφέρει ότι το οδόμετρο χρησιμοποιήθηκε κυρίως για να μετρά τις αποστάσεις στους μεγάλους ρωμαϊκούς δρόμους, και σε κάθε ρωμαϊκό μίλι που συμπληρωνόταν, ένα στρογγυλό βότσαλο έπεφτε προκαλώντας ένα ακουστικό σήμα που προειδοποιούσε ότι είχε διανυθεί η ίδια απόσταση, την οποία και επισήμαιναν με μία μαρμάρινη στήλη (μιλιάριο) που έγραφε τις αποστάσεις σε μίλια κατά αύξοντα αριθμό, μια συνήθεια που χρησιμοποιείται έως σήμερα με τις πινακίδες χιλιομετρικών αποστάσεων.

Αυτό το σύστημα ειδοποίησης είναι ακριβώς ίδιο με εκείνο που περιγράφεται σε ένα αραβικό χειρόγραφο του 1150 μ.Χ και αναφέρει το υδραυλικό ρολόι, που από τους Άραβες αποδίδεται στον Αρχιμήδη. Βάσει αυτής της λογικής, και δεδομένου ότι το σύστημα γρاناζιών που χρησιμοποιήθηκε βρίσκεται στη μέση, ανάμεσα στο κοινό παρακλάδι του οδοντωτού μηχανισμού και του ατέρμονα κοχλία, ο οποίος σύμφωνα με τον Αθήναιο (3ος π.Χ αιώνας) αποδίδεται στον Αρχιμήδη. Το υποσύστημα γρاناζιών του οδομέτρου μπορεί ως εκ τούτου να πάρει μια λογικά θέση στα τεχνολογικά επιτεύγματα του Αρχιμήδη.

Θα ήταν άδικο για την ελληνική επιστήμη να μην γραφεί ότι το 1910 ο Κ. Ν. Ράδος, ένας από τους επιστήμονες που «διασταύρωσαν τα ξίφη» τους έντονα μετά την ανακάλυψη και μελέτη του περίφημου υπολογιστή των Αντικυθήρων, ανακατασκεύασε στα πλαίσια αυτής της διαμάχης το ναυτικό δρομόμετρο στηριζόμενος στην περιγραφή του Ήρωνα.

Ο συγγραφέας διατύπωσε την άποψη ότι το ναυτικό δρομόμετρο εξελίχθηκε κυρίως στην Αλεξάνδρεια, ναυτική πρωτεύουσα της εποχής, όπου έδρασε ο Ήρων, και υποστήριξε ότι τα σημερινά δρομόμετρα έχουν μικρή διαφορά από τα αρχαία.

Με βάση την περιγραφή του Ήωνα, ο Κ. Ν. Ράδος κατασκεύασε ένα όμοιο μηχανήμα από ξύλο και χαλκό, το οποίο εξέθεσε στη Διεθνή Έκθεση του Μπορντώ το 1907, μαζί με άλλα αντικείμενα σχετικά με την ιστορία του ελληνικού ναυτικού και της ελληνικής ναυτιλίας, από τους αρχαίους χρόνους. Η συλλογή εκείνη απέσπασε μεγάλο έπαινο.



Σχήμα 1.3.3. Το ναυτικό δρομόμετρο.

Αυτά είναι όλα όσα γνωρίζουμε για το οδόμετρο («ταξίμετρο») του Ήωνα και το αντίστοιχο ναυτικό οδόμετρο (ή δρομόμετρο). Του οφείλουμε ότι διέσωσε αρχικά τα στοιχεία που υπήρχαν για το μηχανισμό αυτό, ο οποίος πιθανότατα ανακαλύφθηκε από τον Αρχιμήδη, και σε δεύτερη βάση του οφείλουμε τη σημαντική βελτίωση του οργάνου αλλά και την προσαρμογή του για χρήση στην θάλασσα. Το ναυτικό δρομόμετρο, με τις βελτιώσεις του Ήωνα, έγινε ένα εύχρηστο και αναγκαίο όργανο για την μέτρηση των ναυτικών αποστάσεων και συνέβαλε στη γνώση της ναυτικής τέχνης. Και αυτά από τον 1ο μ.Χ αιώνα, μια περίοδο κατά την οποία βρισκόταν σε εξέλιξη ένας μεγάλος αριθμός εμπορικών και εξερευνητικών αποστολών εκ μέρους των Ελλήνων σε Ανατολή, Δύση, Βορρά και Νότο. Πρόκειται για την πρώτη περίοδο των εξερευνήσεων και ανακαλύψεων, η οποία χρονολογείται ανάμεσα στα 300 π.Χ – 300μ.Χ και επέκτεινε τα όρια του γνωστού τότε κόσμου.

## ΚΕΦΑΛΑΙΟ 2

### ΕΙΣΑΓΩΓΗ ΣΤΗ ΡΟΜΠΟΤΙΚΗ ΚΙΝΗΣΗΣ

#### 2.1 Εισαγωγή

Η μετακίνηση (*locomotion*) ενός κινούμενου ρομποτικού συστήματος από κάποιο σημείο του χώρου σε κάποιο άλλο, είναι από τα πιο δύσκολα προβλήματα στη Ρομποτική. Περιλαμβάνει ζητήματα όπως :

- Κατανόηση θέσης
- Εντοπισμός στόχου
- Εύρεση μονοπατιού
- Υπολογισμός τροχιάς
- Ευστάθεια
- Αποφυγή συγκρούσεων

Στο παρόν κεφάλαιο, θα κάνουμε μία εισαγωγή στην ρομποτική κίνησης, παρουσιάζοντας το θεωρητικό υπόβαθρο που θα χρειαστούμε κατά την πλοήγηση του ρομποτικού μας συστήματος. Θα παρουσιαστούν κάποιες βασικές αρχές και έννοιες του αντικειμένου της ρομποτικής, ενώ θα αναλυθούν σε μεγαλύτερο βάθος τα στοιχεία που σχετίζονται με την παρούσα διπλωματική εργασία. Θα παρουσιάσουμε και θα αναλύσουμε, κυρίως, όσα σχετίζονται με τα κινούμενα τροχοφόρα ρομπότ (ρομπότ – οχήματα).

Κατά κύριο λόγο, θα επικεντρωθούμε στα συστήματα διαφορικής οδήγησης, ενώ θα κάνουμε και μία αναλυτική περιγραφή της κινηματικής των συστημάτων αυτών. Ο συγκεκριμένος τρόπος οδήγησης είναι αυτός που θα ακολουθήσει και το ρομπότ ακουστικών μετρήσεων, το οποίο και αποτελεί το αντικείμενο της διπλωματικής εργασίας. Στην συνέχεια, θα κάνουμε μία σύντομη περιγραφή των ζητημάτων που προκύπτουν κατά την προσπάθεια πλοήγησης ενός κινούμενου ρομποτικού συστήματος, όπως είναι για παράδειγμα η κατανόηση της θέσης του αλλά και της κατάστασης του περιβάλλοντός του, με μεθόδους χαρτογράφησης.

Τέλος, θα κάνουμε μία εκτενή ανάλυση της τεχνολογίας των αισθητήρων εκπομπής και λήψης υπέρυθρης ακτινοβολίας, οι οποίοι και θα αποτελέσουν το υποσύστημα αίσθησης του παρόντος ρομποτικού συστήματος. Το αισθητήριο σύστημα αποτελεί ένα από τα σημαντικότερα μέρη ενός κινούμενου ρομπότ, καθώς δίνει τη δυνατότητα σε αυτό να κατανοήσει καλύτερα τον χώρο εργασίας του, αποφεύγοντας τις συγκρούσεις με τα αντικείμενα που βρίσκονται μέσα σε αυτόν, ενώ χρησιμοποιείται και για την υλοποίηση των διαδικασιών εκτίμησης θέσης (τεχνικές οδομετρίας στην συγκεκριμένη περίπτωση).

## 2.2 Βαθμοί Ελευθερίας

Βασικό γνώρισμα κάθε ρομπότ αποτελούν οι *βαθμοί ελευθερίας* του. Βαθμοί ελευθερίας (*degrees of freedom – DOF*) είναι ο ελάχιστος αριθμός των ανεξάρτητων παραμέτρων (συντεταγμένων) που απαιτούνται, για να προσδιοριστούν πλήρως η θέση και η κίνηση ενός μηχανικού συστήματος στο χώρο. Οι γενικευμένες συντεταγμένες που χρησιμοποιούνται για να περιγράψουν την κατάσταση του συστήματος, μπορούν να είναι αποστάσεις, γωνίες ή ακόμα και ποσότητες που εξαρτώνται από τέτοια μεγέθη. Σε γενικές γραμμές, οι βαθμοί ελευθερίας δηλώνουν το πόσο ευκίνητο είναι ένα ρομπότ στο χώρο.

Για να περιγραφεί ακριβώς η θέση ενός μηχανικού σώματος (π.χ. ενός ρομπότ) στο χώρο, απαιτούνται 6 βαθμοί ελευθερίας. Τρεις βαθμοί ελευθερίας αφορούν τη θέση του συστήματος (*translational DOF*) σε  $x$ ,  $y$ ,  $z$ , και τρεις αφορούν τον προσανατολισμό του (*rotational DOF*) στους 3 άξονες  $x'x$ ,  $y'y$ ,  $z'z$ . Κατά συνέπεια, για να μπορεί ένα ρομπότ να κινηθεί απολύτως ελεύθερα στον χώρο, σε οποιοδήποτε σημείο και με οποιοδήποτε προσανατολισμό, πρέπει να έχει τουλάχιστον 6 βαθμούς ελευθερίας.

Στην ρομποτική, ένας άξονας κίνησης (ένας άξονας, δηλαδή, ο οποίος είναι συνδεδεμένος με έναν κινητήρα) ισοδυναμεί με έναν βαθμό ελευθερίας ως προς τον οποίο μπορεί να κινηθεί το ρομπότ. Αν ένα ρομπότ έχει μηχανισμό κίνησης για κάθε βαθμό ελευθερίας, τότε όλοι οι βαθμοί ελευθερίας ονομάζονται *ελεγχόμενοι* (*controllable*). Αν κάποιος βαθμός ελευθερίας δεν ελέγχεται από μηχανισμό, ονομάζεται *μη-ελεγχόμενος* (*uncontrollable*).

Ένα σώμα το οποίο κινείται ελεύθερα πάνω σε ένα επίπεδο (όπως για παράδειγμα ένα τροχοφόρο όχημα πάνω στην επιφάνεια της γης) διαθέτει 3 βαθμούς ελευθερίας. Οι 2 βαθμοί ελευθερίας αφορούν τη μετατόπιση ( $x$ ,  $y$ ) και ένας βαθμός την περιστροφή  $\varphi$ . Στην περίπτωση, για παράδειγμα, του αυτοκινήτου, παρότι ως σώμα που κινείται στο επίπεδο έχει 3 βαθμούς ελευθερίας, διαθέτει μηχανισμούς κίνησης μόνο για 2 από αυτούς. Οι μηχανισμοί κίνησης του αυτοκινήτου είναι το κιβώτιο ταχυτήτων/γκάζι για την κίνηση πάνω στον άξονα  $y'y$ , και το τιμόνι για την περιστροφή γύρω από τον άξονα  $z'z$ . Κατά συνέπεια, το αυτοκίνητο διαθέτει 3 βαθμούς ελευθερίας, εκ των οποίων οι 2 είναι ελεγχόμενοι και ο ένας μη-ελεγχόμενος.

Ο λόγος των ελεγχόμενων βαθμών ελευθερίας (CDOF) ως προς το σύνολο των βαθμών ελευθερίας ενός συστήματος (TDOF), καθορίζει το πόσο εύκολα ελέγχεται η κίνησή του. Διακρίνουμε τις παρακάτω περιπτώσεις :

- Αν  $CDOF = TDOF$ , τότε τα συστήματα αυτά ονομάζονται *ολονομικά* (*holonomic*). Ένα παράδειγμα ολονομικών συστημάτων αποτελούν τα σύγχρονα ελικόπτερα.
- Αν  $CDOF < TDOF$ , τότε τα συστήματα αυτά ονομάζονται *μη-ολονομικά* (*nonholonomic*), και χαρακτηριστικό παράδειγμα αποτελούν τα αυτοκίνητα.
- Αν  $CDOF > TDOF$ , τότε τα συστήματα αυτά ονομάζονται *πλεονάζοντα* (*redundant*). Για παράδειγμα, ο ανθρώπινος βραχίονας έχει συνολικά 7 βαθμούς ελευθερίας χωρίς να συμπεριληφθεί το χέρι (3 στον ώμο, 1 στον αγκώνα, 3 στον καρπό).



## 2.3 Διαφορική Οδήγηση

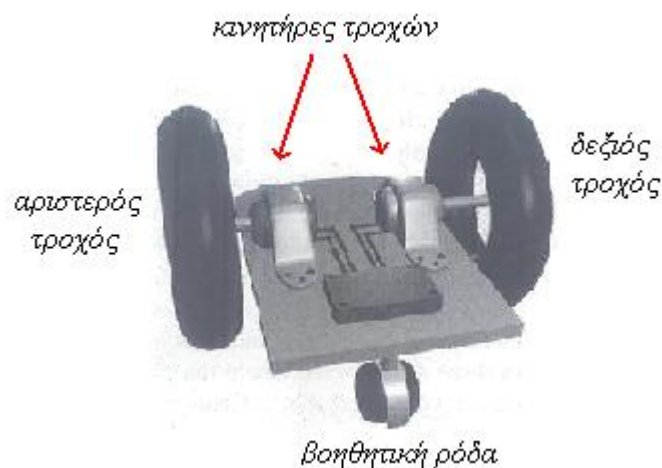
### 2.3.1 Εισαγωγή στα Ρομπότ Διαφορικής Οδήγησης

Γενικά, υπάρχουν διάφοροι τύποι ρομπότ των οποίων η κίνηση βασίζεται σε μέσα, όπως πόδια, βραχίονες, φτερά, πτερύγια, κτλ. Παρόλα αυτά, η πλειοψηφία των ρομπότ χρησιμοποιεί τροχούς για την κίνησή τους. Οι τροχοί είναι το πιο σύνηθες όργανο δράσης στα ρομπότ κυρίως καθώς συμβάλουν στην απλότητα του συστήματος, αφού ελέγχονται με μεγάλη ευκολία. Παράλληλα, δίνουν τη δυνατότητα σε ένα ρομπότ να αναπτύξει μεγαλύτερες ταχύτητες από αυτές που θα μπορούσε να αναπτύξει αν χρησιμοποιούσε ως όργανα δράσης άλλα μέσα κίνησης, όπως για παράδειγμα πόδια. Από την άλλη πλευρά όμως, τα συστήματα που βασίζονται στην κίνησή τους σε τροχούς δεν είναι ολονομικά.

Ένα ρομπότ που μετακινείται μέσω συστήματος τροχών, μπορεί να ελεγχθεί με διαφορετικούς τρόπους. Είτε με ταυτόχρονη κίνηση τροχών, όπως για παράδειγμα ένα αυτοκίνητο, είτε με ανεξάρτητη κίνηση τροχών. Στην πρώτη περίπτωση το σύστημα αποτελείται από έναν μηχανισμό κίνησης (κινητήρας) για τους τροχούς, και έναν μηχανισμό κίνησης για την στρέψη των τροχών.

Η δεύτερη περίπτωση ονομάζεται *διαφορική οδήγηση (differential drive)*, κατά την οποία, οι δύο κινητήριои τροχοί του οχήματος μπορούν να περιστρέφονται ανεξάρτητα, ο άξονας όμως της περιστροφής τους είναι κοινός και σταθερός. Αυτός ο τρόπος οδήγησης δεν χρησιμοποιεί τιμόνι, αλλά αποτελείται από έναν μηχανισμό κίνησης για τον αριστερό και έναν για τον δεξιό τροχό, με αποτέλεσμα το όχημα να στρίβει περιστρέφοντας τους δύο τροχούς του με διαφορετικές ταχύτητες ή/και φορά περιστροφής. Για να κινηθεί το όχημα ευθεία, για παράδειγμα, θα πρέπει να περιστραφούν και οι δύο τροχοί του με την ίδια ταχύτητα. Για να στρίψει δεξιά θα πρέπει να περιστραφεί ο αριστερός τροχός με μεγαλύτερη ταχύτητα από τον δεξιό. Για να στρίψει επιτόπου δεξιά θα πρέπει ο δεξιός τροχός να περιστραφεί προς τα πίσω και ο αριστερός προς τα εμπρός.

Τα συστήματα αυτού του τρόπου οδήγησης χρησιμοποιούν συνήθως και έναν βοηθητικό τροχό με δυνατότητα περιστροφής (*caster wheel*), όπως φαίνεται και στο παρακάτω σχήμα.



Σχήμα 2.3.1.1. Παράδειγμα Συστήματος Διαφορικής Οδήγησης

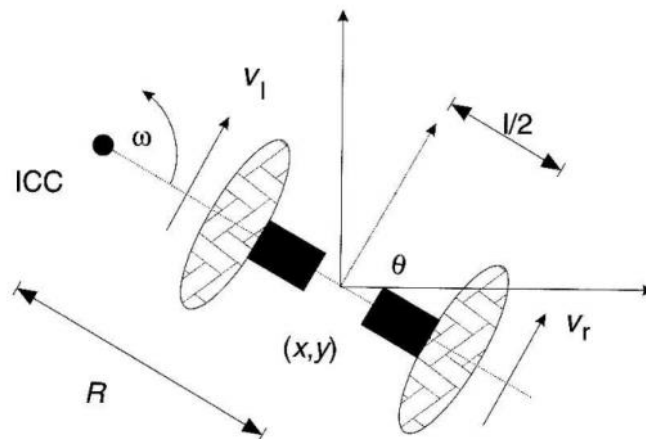
Τα ρομπότ διαφορικής οδήγησης έχουν (όπως όλα τα τροχοφόρα οχήματα) 3 βαθμούς ελευθερίας, εκ των οποίων οι 2 είναι ελεγχόμενοι (από τους 2 μηχανισμούς κίνησης – κινητήρες για κάθε έναν από τους 2 τροχούς) και ο ένας μη-ελεγχόμενος, και κατά συνέπεια τα ρομπότ αυτής της κατηγορίας είναι μη-ολονομικά, ή αλλιώς μη-ελέγξιμα.

Παρακάτω θα αναλύσουμε την κινηματική του συγκεκριμένου τρόπου οδήγησης, ο οποίος συναντάται πολύ συχνά σε εφαρμογές της ρομποτικής, κυρίως εξαιτίας της δυνατότητας που παρέχει για εύκολο χειρισμό. Βασικά πλεονεκτήματα αυτής της μεθόδου οδήγησης είναι επίσης, η απλότητα κατασκευής αλλά και των κινηματικών μοντέλων που περιγράφουν την κίνηση, καθώς και η δυνατότητα να περιστρέφεται το όχημα επιτόπου.

### 2.3.2 Κινηματική Διαφορικής Οδήγησης

Πολλά κινούμενα ρομπότ χρησιμοποιούν, λοιπόν, έναν μηχανισμό οδήγησης που είναι γνωστός, όπως αναφέραμε, ως διαφορική οδήγηση. Αποτελείται από 2 κινητήριους τροχούς τοποθετημένους σε έναν κοινό άξονα, ενώ κάθε ένας από τους τροχούς μπορεί ανεξάρτητα να κινηθεί είτε προς τα εμπρός είτε προς τα πίσω.

Αφού έχουμε τη δυνατότητα να μεταβάλλουμε την ταχύτητα του κάθε τροχού, ώστε το ρομπότ να πραγματοποιήσει περιστροφική κίνηση, το ρομπότ πρέπει να περιστραφεί ως προς ένα σημείο το οποίο βρίσκεται πάνω στον κοινό άξονα του αριστερού και του δεξιού τροχού. Το σημείο, λοιπόν, ως προς το οποίο το ρομπότ περιστρέφεται είναι γνωστό ως *ICC - Instantaneous Center of Curvature* (στιγμιαίο κέντρο καμπυλότητας), το οποίο και περιγράφεται αναλυτικά μέσω του σχήματος 2.3.2.1.



Σχήμα 2.3.2.1. Κίνηση Διαφορικής Οδήγησης

Μεταβάλλοντας τις ταχύτητες των 2 τροχών, μπορούμε να αλλάξουμε την πορεία που ακολουθεί κάθε φορά το ρομπότ. Επειδή η ταχύτητα περιστροφής  $\omega$  ως προς το *ICC* πρέπει να είναι η ίδια και για τους 2 τροχούς, μπορούμε να γράψουμε τις παρακάτω εξισώσεις :

$$\omega \left( R + \frac{l}{2} \right) = V_r \quad (1)$$

$$\omega \left( R - \frac{l}{2} \right) = V_l \quad (2)$$

όπου  $l$  είναι η απόσταση μεταξύ των κέντρων των 2 τροχών,  $V_r$ ,  $V_l$  είναι οι ταχύτητες ως προς το έδαφος δεξιού και αριστερού τροχού αντίστοιχα, και  $R$  είναι η απόσταση ανάμεσα στο ICC και στο μέσο μεταξύ των τροχών. Λύνοντας ως προς  $R$  και  $\omega$ , έχουμε :

$$R = \frac{l}{2} \frac{V_r + V_l}{V_r - V_l} \quad \omega = \frac{V_r - V_l}{l} \quad (3)$$

Παρακάτω παραθέτουμε 3 ενδιαφέρουσες περιπτώσεις αυτού του τύπου οδήγησης.

1. Αν  $V_l = V_r$ , τότε έχουμε ευθύγραμμη κίνηση προς τα εμπρός. Το  $R$  γίνεται άπειρο, και δεν υπάρχει περιστροφική κίνηση ( $\omega = 0$ ).
2. Αν  $V_l = -V_r$ , τότε  $R = 0$ , και έχουμε περιστροφή ως προς το μέσο του άξονα των 2 τροχών (επιτόπια περιστροφή).
3. Αν  $V_l = 0$ , τότε έχουμε περιστροφή ως προς τον αριστερό τροχό. Στην περίπτωση αυτή  $R = l/2$ . Αντίστοιχα ισχύει για  $V_r = 0$ .

Είναι αξιοσημείωτη η ιδιομορφία που έχουν τα ρομπότ διαφορικής οδήγησης να μην μπορούν να κινηθούν κατά μήκος του άξονα των τροχών. Τα οχήματα διαφορικής οδήγησης είναι πολύ ευαίσθητα ακόμα και σε πολύ μικρές μεταβολές της ταχύτητας καθενός από τους 2 τροχούς. Μικρά σφάλματα στη σχέση που πρέπει να έχουν οι ταχύτητες των 2 τροχών, μπορούν να επηρεάσουν την τροχιά του ρομπότ. Τα οχήματα αυτού του τύπου είναι επίσης πολύ ευαίσθητα σε μικρές διακυμάνσεις του εδάφους, και ίσως μπορεί να χρειάζονται επιπλέον βοηθητικοί τροχοί για υποστήριξη.

### 2.3.3 Ορθή Κινηματική των Ρομπότ Διαφορικής Οδήγησης

Στο παραπάνω σχήμα, θεωρούμε ότι το ρομπότ βρίσκεται σε κάποια θέση  $(x, y)$ , κινούμενο προς μια κατεύθυνση η οποία σχηματίζει μία γωνία  $\theta$  με τον άξονα  $X$ . Θεωρούμε ότι το ρομπότ έχει ως κέντρο το μέσο μεταξύ των 2 τροχών. Ρυθμίζοντας κατάλληλα τις παραμέτρους ελέγχου  $V_l$  και  $V_r$ , μπορούμε να κάνουμε το ρομπότ να κινηθεί προς συγκεκριμένες θέσεις και κατευθύνσεις.

(σημειώνουμε ότι  $V_l$ ,  $V_r$  οι ταχύτητες των τροχών ως προς το έδαφος)

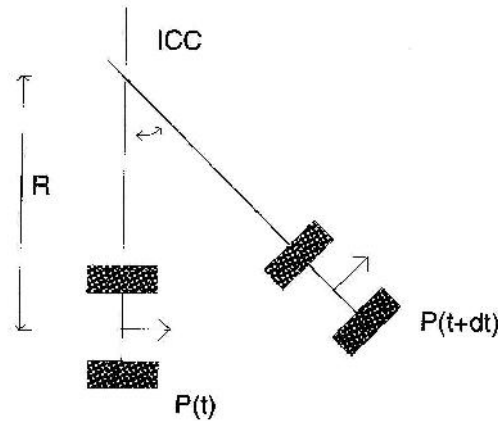
Γνωρίζοντας τις ταχύτητες  $V_l$ ,  $V_r$  και χρησιμοποιώντας την εξίσωση 3, μπορούμε να βρούμε την θέση του ICC :

$$ICC = [x - R \sin(\theta), y + R \cos(\theta)] \quad (4)$$

ενώ σε χρόνο  $t + \delta t$  η θέση του ρομπότ θα είναι :

$$\begin{bmatrix} x' \\ y' \\ \theta' \end{bmatrix} = \begin{bmatrix} \cos(\omega\delta t) & -\sin(\omega\delta t) & 0 \\ \sin(\omega\delta t) & \cos(\omega\delta t) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x - ICC_x \\ y - ICC_y \\ \theta \end{bmatrix} + \begin{bmatrix} ICC_x \\ ICC_y \\ \omega\delta t \end{bmatrix} \quad (5)$$

Αυτή η εξίσωση περιγράφει την κίνηση του ρομπότ το οποίο περιστρέφεται σε απόσταση  $R$  από το  $ICC$  με γωνιακή ταχύτητα  $\omega$ , όπως φαίνεται και στο σχήμα 2.3.3.1.

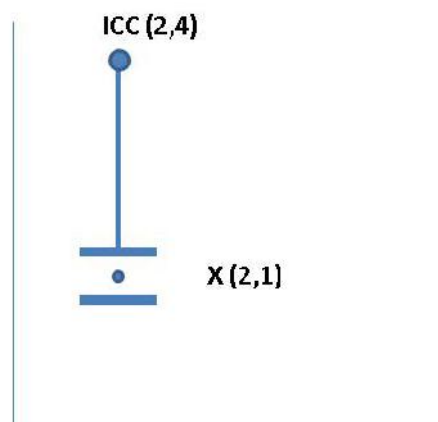


Σχήμα 2.3.3.1. Περιγραφή της κίνησης των ρομπότ διαφορικής οδήγησης.

Ένας άλλος τρόπος να κατανοήσουμε την παραπάνω εξίσωση είναι (σχήμα 2.3.3.1.) το ότι η κίνηση του ρομπότ ισοδυναμεί με :

- 1) μεταφορά του  $ICC$  στην αρχή του συστήματος συντεταγμένων,
- 2) περιστροφή του ρομπότ ως προς την αρχή των αξόνων κατά μία γωνία της τάξης  $\omega\delta t$ , και
- 3) μεταφορά πίσω στο  $ICC$ .

Τα βήματα της παραπάνω διαδικασίας περιγράφονται αναλυτικά μέσω των επόμενων σχημάτων.

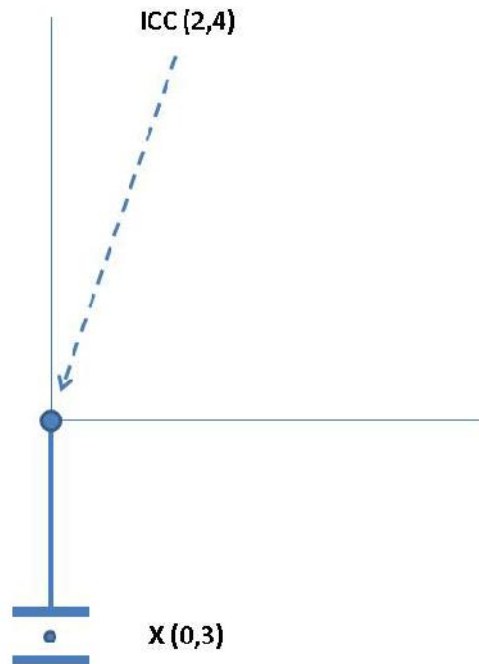


Σχήμα 2.3.3.2.a

Παράδειγμα : Ρομπότ Διαφορικής Οδήγησης

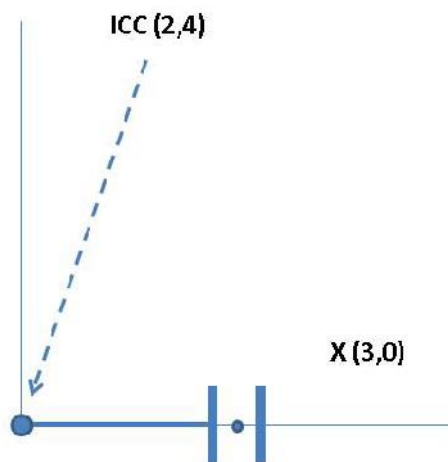
Περιστροφή  $90^\circ$  ως προς το ICC

Πώς γνωρίζουμε που καταλήγει το ρομπότ?



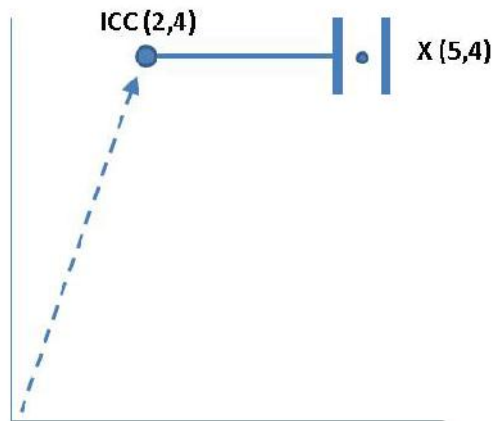
Σχήμα 2.3.3.2.b

Μεταφορά του ICC στην αρχή των αξόνων.



Σχήμα 2.3.3.2.c

Περιστροφή  $90^\circ$  ως προς τον άξονα Z.



Σχήμα 2.3.3.2.d

Μεταφορά πίσω στο πραγματικό ICC.

$$\begin{bmatrix} x' \\ y' \\ \theta' \end{bmatrix} = \begin{bmatrix} \cos(\omega\delta t) & -\sin(\omega\delta t) & 0 \\ \sin(\omega\delta t) & \cos(\omega\delta t) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x - ICC_x \\ y - ICC_y \\ \theta \end{bmatrix} + \begin{bmatrix} ICC_x \\ ICC_y \\ \omega\delta t \end{bmatrix}$$

νέο σημείο	περιστροφή κατά $\omega\delta t$	μεταφορά του ICC	μεταφορά του
$(x, y)$	ως προς τον άξονα Z	στην αρχή	ICC στην σωστή
		των αξόνων	θέση

Σχήματα 2.3.3.2.a,b,c,d. Παράδειγμα. Ορθή κινηματική των ρομπότ διαφορικής οδήγησης.

### 2.3.4 Ανάστροφη Κινηματική Κινούμενου Ρομπότ

Γενικά, μπορούμε να περιγράψουμε τη θέση ενός ρομπότ ικανού να κινηθεί σε μία συγκεκριμένη κατεύθυνση  $\theta(t)$ , με μία δεδομένη ταχύτητα  $V(t)$ , ως εξής :

$$x(t) = \int_0^t V(t) \cos[\theta(t)] dt$$

$$y(t) = \int_0^t V(t) \sin[\theta(t)] dt$$

$$\theta(t) = \int_0^t \omega(t) dt$$

Για την ειδική περίπτωση ενός ρομπότ διαφορικής οδήγησης (όπως το *iRobot Create*), οι εξισώσεις γίνονται :

$$x(t) = \frac{1}{2} \int_0^t [v_r(t) + v_l(t)] \cos[\theta(t)] dt$$

$$y(t) = \frac{1}{2} \int_0^t [v_r(t) + v_l(t)] \sin[\theta(t)] dt$$

$$\theta(t) = \frac{1}{l} \int_0^t [v_r(t) - v_l(t)] dt$$

Μία σχετική ερώτηση είναι το πώς μπορεί να γίνει ο έλεγχος ενός ρομπότ, προκειμένου να πλησιάσει έναν ζητούμενο συνδυασμό  $(x, y, \theta)$ . Το ζήτημα αυτό είναι γνωστό και ως *πρόβλημα ανάστροφης κινηματικής*.

Δυστυχώς, ένα ρομπότ διαφορικής οδήγησης επιβάλλει τους λεγόμενους *μη-ολονομικούς* περιορισμούς, όσον αφορά στο να εγκατασταθεί στη θέση του. Για παράδειγμα, ένα ρομπότ αυτού του τύπου δεν μπορεί να κινηθεί πλάγια, κατά μήκος του άξονά του. Ένας παρόμοιος μη-ολονομικός περιορισμός είναι ένα αυτοκίνητο το οποίο μπορεί να στρίψει μόνο τους μπροστινούς του τροχούς. Δεν μπορεί να κινηθεί απευθείας πλάγια, και είναι χαρακτηριστικό πως το παρκάρισμα ενός αυτοκινήτου απαιτεί ένα πιο πολύπλοκο σύνολο ελιγμών στην οδήγηση. Για τον λόγο αυτόν, δεν μπορούμε απλά να καθορίσουμε μία αυθαίρετη θέση  $(x, y, \theta)$  και να βρούμε τις ταχύτητες οι οποίες θα οδηγήσουν το ρομπότ στη θέση αυτήν.

Για την ειδική περίπτωση όπου  $v_r = v_l = v$  (το ρομπότ κινείται σε ευθεία γραμμή), οι εξισώσεις κίνησης γίνονται :

$$\begin{bmatrix} x' \\ y' \\ \theta' \end{bmatrix} = \begin{bmatrix} x + v \cos(\theta) \delta t \\ y + v \sin(\theta) \delta t \\ \theta \end{bmatrix} \quad (6)$$

Αν  $v_r = -v_l = v$  τότε το ρομπότ περιστρέφεται επιτόπου και οι εξισώσεις γίνονται :

$$\begin{bmatrix} x' \\ y' \\ \theta' \end{bmatrix} = \begin{bmatrix} x \\ y \\ \theta + 2v \delta t / l \end{bmatrix} \quad (7)$$

Η παραπάνω λογική συνιστά την τακτική πλοήγησης των ρομπότ διαφορικής οδήγησης κατά την οποία το ρομπότ κινείται αρχικά σε ευθεία γραμμή, μετά περιστρέφεται για να πραγματοποιήσει μία επιτόπια στροφή, και στην συνέχεια κινείται ξανά ευθεία.

### 2.3.5 Αντιστοίχιση Γωνιακής Ταχύτητας Τροχού με Γραμμική Ταχύτητα

Οι ταχύτητες αριστερού και δεξιού τροχού που χρησιμοποιήθηκαν παραπάνω,  $V_l$  και  $V_r$ , είναι γραμμικές ταχύτητες. Ουσιαστικά, ελέγχουμε τους τροχούς καθορίζοντας μία γωνιακή ταχύτητα

$V_{wheel}$  για έναν τροχό (ορισμένη σε  $rad/s$ ). Με δεδομένη την  $V_{wheel}$ , χρειάζεται να βρούμε με ποια γραμμική ταχύτητα κινείται ο συγκεκριμένος τροχός.

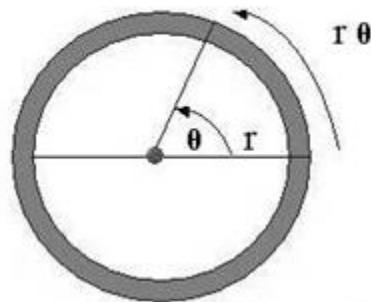
Σαν παράδειγμα, θα χρησιμοποιήσουμε το ρομπότ *Khepera*, το οποίο και είναι ένα μικρό ρομπότ διαφορικής οδήγησης, παρόμοιο με το iRobot Create. Ορίζουμε τους ακόλουθους όρους :

$r_{wheel}$  : η ακτίνα του τροχού

$D_{robot}$  : το μήκος του άξονα διαφορικής οδήγησης (η απόσταση μεταξύ των 2 τροχών)

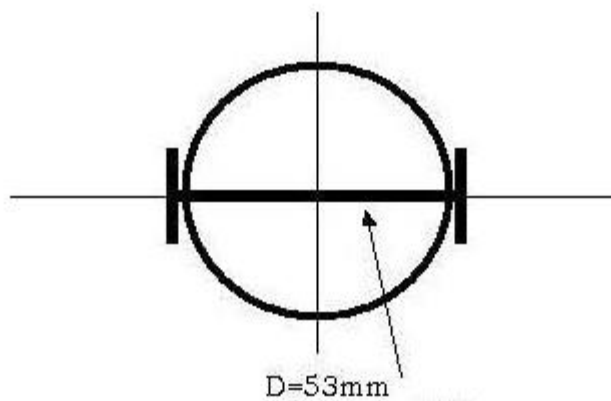
$V_{wheel}$  : το μέτρο της ταχύτητας του τροχού σε  $rad/s$ .

Για να υπολογίσουμε μία γωνία περιστροφής  $\varphi$  του ρομπότ, πρέπει να βρούμε μία εξίσωση για τον χρόνο που χρειάζεται να περιστρέψουμε τον κινητήρα του τροχού με ταχύτητα  $V_{wheel}$ , προκειμένου το ρομπότ να στρίψει κατά γωνία  $\varphi$ . Σε αυτό το παράδειγμα, ας θεωρήσουμε ότι οι 2 τροχοί περιστρέφονται με την ίδια ταχύτητα αλλά αντίθετη φορά (το ρομπότ στρίβει επιτόπου).



$$r_{wheel}=8mm$$

Σχήμα 2.3.5.1.a. Διαστάσεις τροχού.



Σχήμα 2.3.5.1.b. Διαστάσεις ρομπότ.

Ο τροχός διαγράφει μία γραμμική απόσταση  $d = r_{wheel} \cdot \theta$ , όπου  $\theta = V_{wheel} \cdot t$  (βλ. σχήματα 2.3.5.1.a,b). Κατά συνέπεια, για δεδομένο χρόνο  $t$  ο τροχός θα διαγράψει απόσταση :



$$d = r_{wheel} \cdot V_{wheel} \cdot t$$

Δηλαδή, αν θεωρήσουμε ταχύτητα τροχού  $V_{wheel} = 10rad/s$ , τότε ο τροχός θα προχωρήσει κατά ένα διάστημα  $d = 10 \cdot 8 = 80mm$  σε 1 sec.

Για να καθορίσουμε τον χρόνο που χρειάζεται το ρομπότ για να στρίψει επιτόπου κατά μία συγκεκριμένη γωνία, πρέπει πρώτα να σημειώσουμε ότι ολόκληρη η περιφέρεια  $C$  που διαγράφει το ρομπότ όταν περιστρέφεται κατά  $360^\circ$  είναι :

$$C = \pi \cdot D_{robot}$$

Μπορούμε να στρίψουμε κατά μία γωνία  $\varphi$  σε χρόνο  $t$  χρησιμοποιώντας την εξίσωση :

$$\frac{d}{C} = \frac{\varphi}{2\pi}$$

όπου η γωνία  $\varphi$  είναι σε *rad*.

Άρα έχουμε :

$$\frac{r_{wheel} \cdot V_{wheel} \cdot t}{C} = \frac{\varphi}{2\pi}$$

$$t = \frac{\varphi \cdot C}{2\pi \cdot r_{wheel} \cdot V_{wheel}}$$

## 2.4 Πλοήγηση

### 2.4.1 Εισαγωγή

Για ένα κινούμενο ρομπότ, η ικανότητα της πλοήγησης είναι από τις σημαντικότερες λειτουργίες του. Το να είναι λειτουργικό, δηλαδή να αποφεύγει επικίνδυνες καταστάσεις, όπως για παράδειγμα συγκρούσεις με αντικείμενα που βρίσκονται ενός του χώρου εκτέλεσης της λειτουργίας του, και να παραμένει σε ασφαλές συνθήκες (π.χ. θερμοκρασιακά, ακτινοβολία, έκθεση σε καιρικές συνθήκες κτλ) είναι πρωταρχικό ζήτημα (αλλά και για οποιαδήποτε εργασία που θα πρέπει να πραγματοποιηθεί και σχετίζεται με συγκεκριμένα μέρη στο περιβάλλον του ρομπότ, η πλοήγηση είναι απαραίτητη).

Πλοήγηση ενός συστήματος θεωρείται ο συνδυασμός των τριών παρακάτω υπολειτουργιών:

- Προσδιορισμός της θέσης του (ατομική πλοήγηση)
- Σχεδιασμός τροχιάς – τοπική αποφυγή εμποδίων (τοπική πλοήγηση)
- Σχεδιασμός και ερμηνεία χάρτη (γενική πλοήγηση)

Ο χάρτης με αυτήν την έννοια αποτελεί μια ένα-προς-ένα απεικόνιση του χώρου σε μια εσωτερική αναπαράσταση. Ο προσδιορισμός θέσης αποτελεί μια δεξιότητα του ρομπότ να εκτιμάει τη διάταξή του ως προς ένα γενικό πλαίσιο αναφοράς. Ο σχεδιασμός τροχιάς, είναι μια επέκταση

της διαδικασίας προσδιορισμού θέσης στο ότι απαιτεί τον προσδιορισμό της τρέχουσας θέσης και της θέσης που επιθυμούμε να φτάσουμε, ενώ και οι δύο θέσεις θα συνυπάρχουν σε ένα κοινό πλαίσιο αναφοράς. Έτσι η κατασκευή του χάρτη δεν συμπεριλαμβάνει μόνο τα στοιχεία που γνωρίζουμε ότι περιέχει κάθε χάρτης (μετρικά χαρακτηριστικά του περιβάλλοντος), αλλά και κάθε άλλο σημάδι που υποδηλώνει τοποθεσία στο πλαίσιο αναφοράς.

#### 2.4.2 Τεχνικές Εντοπισμού Θέσης

Το πρόβλημα εντοπισμού της θέσης ενός ρομποτικού συστήματος, είναι το ζήτημα που περιλαμβάνει την ερώτηση του «πού βρίσκομαι». Αυτό σημαίνει ότι το ρομπότ πρέπει να ξέρει την θέση του σε σχέση με το περιβάλλον. Όταν μιλάμε για την *θέση (location)*, εννοούμε τις συντεταγμένες  $x$  και  $y$ , καθώς επίσης και την γωνία κατεύθυνσης ενός ρομπότ ως προς ένα σύστημα αναφοράς. Η διαδικασία εντοπισμού θέσης είναι αντικείμενο εντατικής έρευνας την τελευταία δεκαετία και υπάρχουν πολλές διαφορετικές μέθοδοι οι οποίες ισχύουν υπό διαφορετικές προϋποθέσεις και σε διαφορετικά περιβάλλοντα.

Οι *σχετικές τεχνικές εντοπισμού* βασίζονται στον προσδιορισμό της θέσης και τον προσανατολισμό του ρομπότ γνωρίζοντας ένα αρχικό σημείο. Η πιο συνηθισμένη τεχνική σχετικού εντοπισμού είναι η *οδομετρία (odometry)*, η οποία και εφαρμόζεται σε ρομπότ με τροχούς, χρησιμοποιώντας αισθητήρες περιστροφών (συνήθως *κωδικοποιητές – encoders*) για την μέτρηση της απόστασης που διανύει κάθε τροχός. Η οδομετρία απασχολεί απλές γεωμετρικές εξισώσεις (κινηματική του κινητού ρομπότ), με την ενσωμάτωση των οποίων υπολογίζεται η θέση και ο προσανατολισμός του οχήματος. Η οδομετρία είναι ικανοποιητική για μικρές αποστάσεις αλλά δεν παρέχει αξιόπιστες πληροφορίες για μεγαλύτερες αποστάσεις.

Οι *τεχνικές του απόλυτου εντοπισμού* καθορίζουν τη θέση του ρομπότ σε σχέση με ένα παγκόσμιο πλαίσιο αναφοράς, για παράδειγμα, χρησιμοποιώντας φάρους ή ορόσημα. Η πιο δημοφιλής τεχνική είναι το *GPS (Global Positioning System)* η οποία χρησιμοποιώντας δορυφορικά σήματα μπορεί να προσδιορίσει την απόλυτη θέση (γεωγραφικό μήκος, πλάτος και ύψος) ενός αντικειμένου στη Γη. Στην περίπτωση του απόλυτου εντοπισμού η αύξηση του σφάλματος μειώνεται όταν οι μετρήσεις είναι διαθέσιμες. Η θέση του ρομπότ δεν εξαρτάται από το χρόνο και την αρχική θέση. Τα κύρια προβλήματα των τεχνικών που βασίζονται στα ορόσημα (landmarks) είναι:

- Απαιτεί μια δαπανηρή εγκατάσταση σχετικά με τον τομέα στον οποίο λειτουργεί το ρομπότ.
- Το κινητό ρομπότ μπορεί μόνο να περιηγηθεί πάνω από την περιοχή στην οποία βρίσκονται τα ορόσημα.
- Μεταξύ ορόσημων το ρομπότ δεν μπορεί να καθορίσει τον εντοπισμό του.

Εάν μπορούσαμε να προσαρμόσουμε έναν ακριβή αισθητήρα GPS σε ένα κινούμενο ρομπότ, το μεγαλύτερο μέρος του προβλήματος θα παρακάμπτονταν. Το GPS θα ενημέρωνε το ρομπότ για την ακριβή του θέση, εντός και εκτός κτιρίων, επομένως το ζήτημα της εκτίμησης θέσης του συστήματος θα λυνόταν άμεσα. Δυστυχώς, ένας τέτοιος αισθητήρας δεν είναι προς το παρόν πρακτικός. Το υπάρχον δίκτυο GPS παρέχει ακρίβεια κάποιων μέτρων, το οποίο είναι μη αποδεκτό

για κινούμενα ρομπότ μικρού μεγέθους, και κυρίως για ρομπότ τα οποία είναι κατασκευασμένα με σκοπό την κάλυψη μικρών αποστάσεων (στα οποία απαιτείται ακρίβεια εκατοστών). Επιπλέον, οι τεχνολογίες GPS δεν μπορούν να λειτουργήσουν σε εσωτερικούς χώρους ή σε αποκλεισμένες περιοχές και επομένως έχουν περιορισμένο χώρο εργασίας.

### 2.4.3 Ταυτόχρονος Εντοπισμός και Χαρτογράφηση (SLAM)

Η ταυτόχρονη εκτίμηση θέσης και κατάρτιση χάρτη είναι μια τεχνική που χρησιμοποιούν τα ρομπότ και τα αυτόνομα οχήματα για να κατασκευάσουν ένα χάρτη ενός άγνωστου χώρου, ενώ ταυτόχρονα προσδιορίζουν τη θέση τους σε αυτόν. Το *SLAM* (*simultaneous localization and mapping* – ταυτόχρονος εντοπισμός και χαρτογράφηση) δεν έχει τελειοποιηθεί, όμως χρησιμοποιείται ήδη σε εφαρμογές της ρομποτικής όπως οικιακά ρομπότ και αυτόνομα οχήματα μεταξύ άλλων. Η λύση του προβλήματος θεωρείται μεγάλο επίτευγμα για την επιστήμη της ρομποτικής, καθώς θα επιτρέψει πραγματική αυτονομία στα οχήματα. Παράλληλα όμως πρέπει να λυθούν και τα προβλήματα της *συσχέτισης δεδομένων* (*data association*) και της υπολογιστικής πολυπλοκότητας με τα οποία έχει άμεση σχέση.

Το πρόβλημα της ταυτόχρονης εκτίμησης θέσης και κατάρτισης χάρτη είναι πιο εύκολο να κατανοηθεί αν δούμε τα δυο επιμέρους προβλήματα που καλούμαστε να αντιμετωπίσουμε. Το ένα από τα δυο μέρη του προβλήματος του SLAM είναι η *εκτίμηση της θέσης* (*localization*) του ρομπότ με βάση κάποιο εξωτερικό σύστημα. Με τη λέξη θέση εννοούμε τόσο τη θέση, όσο και τον προσανατολισμό που έχει το ρομπότ μια δεδομένη χρονική στιγμή. Για τον υπολογισμό της χρησιμοποιούνται οι ανιχνευτές, των οποίων το θόρυβο πρέπει να γνωρίζουμε, καθώς και ο μηχανισμός κίνησης και το σφάλμα που και αυτός εισάγει. Αν αντιμετωπίζουμε μόνο αυτήν την πλευρά του SLAM, τότε θεωρούμε τις πληροφορίες για τον περιβάλλοντα χώρο, που αποτελούν το *χάρτη* (*map*), γνωστές. Εξαιτίας των σφαλμάτων η εκτίμηση είναι καθαρά πιθανολογική. Στη γενική περίπτωση μάλιστα η αβεβαιότητα για την ακριβή θέση του ρομπότ εισάγει σφάλμα και στη θέση του εμποδίου.

Το δεύτερο μέρος είναι η *κατάρτιση χάρτη* (*mapping*). Για να αλληλεπιδράσει με το περιβάλλον του το ρομπότ πρέπει να γνωρίζει τη θέση των αντικειμένων γύρω του. Σε αυτήν την περίπτωση πρέπει να αποφασίσουμε τι είδους χάρτη θα χρησιμοποιήσουμε, καθώς αυτό που ενδιαφέρει είναι η γεωμετρική θέση των αντικειμένων. Αυτό σημαίνει ότι οι *τοπολογικοί* χάρτες με πληροφορία για το πώς ενώνονται οι χώροι, επειδή δεν είναι ιδιαίτερα ακριβείς, δε χρησιμοποιούνται για την επίλυση του SLAM. Επιπλέον μπορεί να χρησιμοποιηθεί μια δυαδική μεταβλητή για την ύπαρξη ή όχι ενός αντικειμένου σε ένα σημείο, όσο και πιο λεπτομερείς πληροφορίες για τη θερμοκρασία, την κλίση ή άλλα στοιχεία που το ορίζουν (πληροφορίες οι οποίες δεν θα μας απασχολήσουν καθώς, όπως θα δούμε στην συνέχεια, θα κάνουμε κάποιες παραδοχές ως τη φύση των αντικειμένων που περιέχονται στο χώρο εργασίας). Η συγκεκριμένη εργασία περιορίζεται σε χάρτες δυο διαστάσεων, ορισμένων σε συγκεκριμένη περιοχή στο χώρο, όπου εξετάζεται αν κάποιο κελί του πλέγματος που τον καλύπτει είναι κατειλημμένο ή όχι. Αυτό το είδος του χάρτη ονομάζεται *πλέγμα κατάληψης* (*occupancy grid*) και παρότι φαίνεται απλουστευμένο χρησιμοποιείται συχνά σε κινούμενα ρομπότ. Για τα στοιχεία του χάρτη

χρησιμοποιούνται οι ονομασίες *χαρακτηριστικό (feature)*, *ορόσημο (landmark)*, ή απλά εμπόδιο. Για να πάρουμε σωστά αποτελέσματα πρέπει να έχουμε όλες τις μετρήσεις σε ένα κοινό σύστημα αναφοράς και το πετυχαίνουμε αλλάζοντας τη θέση και τον προσανατολισμό του ρομπότ, που σε αυτή την περίπτωση θεωρούμε ότι είναι γνωστά.

Όπως έχουμε αναφέρει, σκοπός της εργασίας (εκτός από την κατασκευή) είναι η πλοήγηση του ρομπότ ακουστικών μετρήσεων και η μετακίνησή του από ένα δοθέν αρχικό σημείο σε ένα σημείο προορισμού. Κατά τη διάρκεια αυτής της διαδικασίας, το ρομπότ έχει απόλυτη επίγνωση του προσανατολισμού και της θέσης στην οποία βρίσκεται, καθώς επίσης και του τρόπου (γεωμετρική θέση – γωνία) με τον οποίο είναι τοποθετημένα στον χώρο εργασίας, τα αντικείμενα τα οποία κάθε φορά συναντά. Όπως θα αναλύσουμε και αργότερα, όλες αυτές οι πληροφορίες καταγράφονται συνεχώς με τη μορφή μηνυμάτων (που θα παρουσιάσουμε στην συνέχεια), στην μνήμη του ενσωματωμένου φορητού υπολογιστή. Με τον τρόπο αυτόν, θα πληρούνται όλες οι προϋποθέσεις, ώστε να είναι όσο το δυνατόν πιο απλή η διαδικασία χαρτογράφησης του χώρου εκτέλεσης, από τον μελλοντικό χρήστη του παρόντος ρομποτικού συστήματος.

## 2.5 Αισθητήριες Διατάξεις

### 2.5.1 Εισαγωγή

Όπως αναφέραμε και στο προηγούμενο κεφάλαιο, τα κινούμενα ρομπότ είναι εξοπλισμένα με ένα σύνολο αισθητήριων διατάξεων, όπως αισθητήρες υπερύθρων, απόστασης, οπτικούς αισθητήρες κτλ. Η διαδικασία της πλοήγησης ενός κινούμενου ρομπότ περιλαμβάνει ζητήματα όπως είναι η εκτίμηση της τρέχουσας θέσης και του προσανατολισμού του συστήματος, καθώς επίσης και η αποφυγή συγκρούσεων με αντικείμενα που βρίσκονται στον χώρο εργασίας. Για την αντιμετώπιση αυτών των ζητημάτων, είναι απαραίτητη η χρήση ενός υποσυστήματος αίσθησης (σύστημα αισθητήρων), το οποίο και αποτελεί το βασικότερο μέσο για την κατανόηση του περιβάλλοντος χώρου ενός αυτόνομου ρομποτικού οχήματος.

Στην παρούσα εφαρμογή, αρμοδιότητα του συστήματος αισθητήρων είναι η επίλυση του ζητήματος της αποφυγής εμποδίων, καθώς επίσης και της υλοποίησης του οδομέτρου για την μέτρηση της διανυόμενης απόστασης. Για την αντιμετώπιση, λοιπόν, των παραπάνω προβλημάτων πλοήγησης, χρησιμοποιήθηκαν *αισθητήρες υπέρυθρης ακτινοβολίας*, οι οποίοι αποτελούν και τον πιο συνηθισμένο τύπο αισθητήρων σε ρομποτικές εφαρμογές κινούμενων συστημάτων.

Οι αισθητήρες αυτοί χωρίζονται σε *ενεργούς* και *παθητικούς*. Οι ενεργοί αισθητήρες υπερύθρων στέλνουν μια δέσμη υπέρυθρης ακτινοβολίας, και αποκρίνονται στην ανάκλαση της δέσμης από ένα αντικείμενο (στόχος). Εκτός από την ικανότητα να δείξουν την παρουσία ή την απουσία ενός αντικειμένου, μπορούν (ανάλογα με τον τύπο του αισθητήρα υπερύθρων) να δείξουν και τη θέση του αντικειμένου στον χώρο (απόστασή του από τον αισθητήρα – αισθητήρες απόστασης). Από την άλλη πλευρά, οι παθητικοί αισθητήρες υπερύθρων είναι συσκευές οι οποίες απλά ανιχνεύουν την παρουσία υπέρυθρης ακτινοβολίας στο περιβάλλον.

Στις επόμενες παραγράφους, θα περιγράψουμε και θα αναλύσουμε την τεχνολογία των αισθητήρων εκπομπής και λήψης υπέρυθρης ακτινοβολίας, οι οποίοι και θα αποτελέσουν το σύστημα αισθητήρων που θα χρησιμοποιήσει το ρομπότ ακουστικών μετρήσεων, προκειμένου να αντιλαμβάνεται το περιβάλλον του, και να μπορεί να πλοηγείται σε αυτό.

## 2.5.2 Infrared LED – Πομπός Υπερύθρων

Ένα *Infrared LED* (*IR LED – LED υπέρυθρων*) δε διαφέρει σημαντικά από ένα κοινό LED (για παράδειγμα κόκκινου χρώματος). Με την ορθή πόλωση προκαλείται ροή οπών μέσα στο υλικό n και ροή ηλεκτρονίων προς το υλικό p (τα ηλεκτρόνια ξαναμπαίνουν στο υλικό n, μέσω του εξωτερικού κυκλώματος).

Η επανασύνδεση των φορέων μειονότητας ( $e^-$ ) στην ένωση, προκαλεί την εκπομπή ακτινοβολούμενης ενέργειας. Η ενέργεια αυτή είναι ανάλογη με το ενεργειακό χάσμα του υλικού. Όσο μεγαλύτερο το ενεργειακό χάσμα, τόσο πιο υψηλή είναι η συχνότητα του εκπεμπόμενου φωτός. Τα LEDs που έχουν μικρά ενεργειακά χάσματα, φτιαγμένα από Αρσενικούχο Γάλλιο (GaAs), ακτινοβολούν στην κόκκινη ή στην υπέρυθρη περιοχή, ειδικότερα όταν εκπέμπουν στην υπέρυθρη περιοχή, ονομάζονται *IR LEDs* (*Infrared LEDs*).

Όπως αναφέραμε και προηγουμένως τα LEDs αποτελούνται από μία επαφή pn, ακριβώς όπως και οι κοινές δίοδοι. Η διαφορά είναι ότι δεν βασίζονται στο πυρίτιο Si και αυτό γιατί είναι αδύνατο να επιτευχθεί εκπομπή ακτινοβολίας, αφού πρόκειται για υλικό έμμεσου ενεργειακού διακένου. Υλικά έμμεσου ενεργειακού διακένου λέγονται τα υλικά για τα οποία ο πυθμένας της ζώνης αγωγιμότητας και η κορυφή της ζώνης σθένους δεν είναι στο ίδιο k στο διάγραμμα των ενεργειακών ζωνών  $E_n(k)$ . Το GaAs, από την άλλη πλευρά, είναι άμεσου ενεργειακού διακένου.

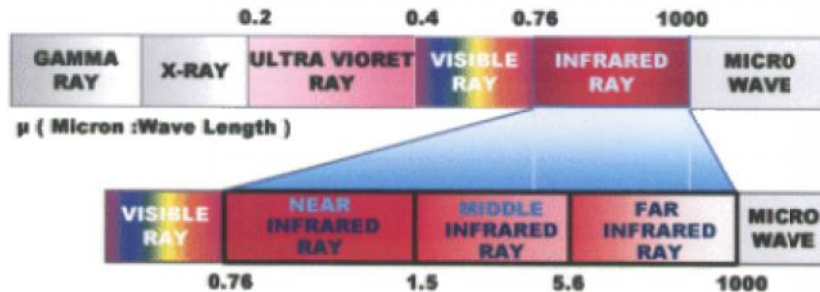
Στον παρακάτω πίνακα παρατίθεται η αντιστοιχία μερικών υλικών, που χρησιμοποιούνται στην κατασκευή LEDs (Light Emitting Diode), με το χρώμα της εκπεμπόμενης ακτινοβολίας.

Υλικό	Μήκος Κύματος (nm)	Χρώμα	Ισχύς Ακτινοβολίας ( $\mu W$ )	Βαθμός Απόδοσης (%)
GaAs : Si	950	IR (υπέρυθρο)	400	4
GaAs : Zn	900	IR (υπέρυθρο)	50	0,5
GaP	690	Κόκκινο	180	1
GaAsP	690	Κόκκινο	15	0,1
GaP	590	Κίτρινο	4	0,02
GaP	565	Πράσινο	8	0,04

Πίνακας 2.1.

Τα IR LEDs είναι περισσότερο κατάλληλα για τηλεκατεύθυνση και τηλεχειρισμό και γενικότερα για μετάδοση ψηφιακής ή αναλογικής πληροφορίας. Η μετάδοση αναλογικής πληροφορίας είναι δυνατή εξαιτίας της αναλογίας μεταξύ της έντασης του φωτός και του ρεύματος

που διαρρέει την δίοδο (διπλασιάζοντας το ρεύμα, διπλασιάζεται η ένταση του φωτός κ.ο.κ.). Η υπέρυθη ακτινοβολία στην οποία εκπέμπουν τα IR LEDs είναι το *εγγύς υπέρυθρο (near infrared)*, που είναι και αόρατη στο ανθρώπινο μάτι (βλ. εικόνα 2.5.2.1).



Εικόνα 2.5.2.1. Οι βασικές διαβαθμίσεις της υπέρυθρης ακτινοβολίας του ηλεκτρομαγνητικού φάσματος.

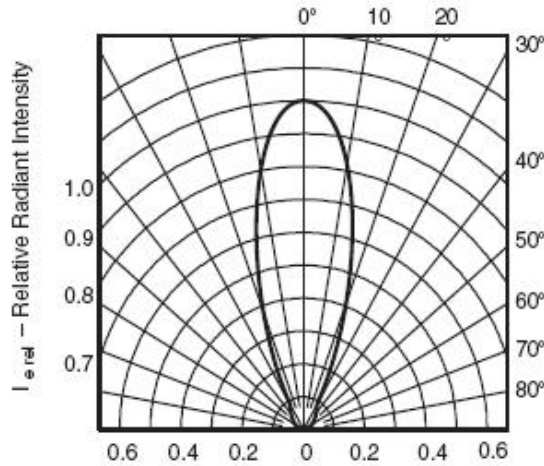
Το εγγύς υπέρυθρο (μαζί με το μέσο) χαρακτηρίζεται και ως ανακλώμενο υπέρυθρο. Γενικά, παρουσιάζει παρόμοια συμπεριφορά με το ορατό τμήμα του φάσματος.



Εικόνα 2.5.2.2. Ένα σύγχρονο IR LED.

Ένα IR LED (όπως και οποιοδήποτε LED) δεν εκπέμπει ακτινοβολία προς όλες τις κατευθύνσεις. Παρουσιάζει έντονη κατευθυντικότητα. Έτσι, η ακτινοβολία δεν εκπέμπεται σε ολόκληρο το χώρο, αλλά σε μία ορισμένη γωνία. Μία σημαντική παράμετρος είναι η γωνία ημίσειας ισχύος που δίνεται από τον κατασκευαστή. Τυπική τιμή της (για συνηθισμένα IR LEDs) είναι γύρω στις  $40^\circ$  ( $\pm 20^\circ$ ).

Ένα σημαντικό διάγραμμα που αφορά την κατευθυντικότητα ενός IR LED, είναι αυτό της *σχετικής ακτινικής έντασης (relative radiant intensity)*, όπως αυτό που ακολουθεί και το οποίο δείχνει την ένταση της ακτινοβολίας σε σχέση με τη γωνία στο χώρο. Η γωνία ημίσειας ισχύος υπολογίζεται από το διάγραμμα, βρίσκοντας τα σημεία στα οποία τέμνεται η καμπύλη από το ημικύκλιο 0,5. Οι ακτίνες δείχνουν τη γωνία (στο σχήμα 2.5.2.1, η γωνία είναι περίπου  $17^\circ$ . Άρα, συνολικά θα έχουμε  $\pm 17^\circ = 34^\circ$ ).



Σχήμα 2.5.2.1. Σχετική ακτινική ένταση.

### 2.5.3 Φωτοдиодος και διόδος PIN – Ο δέκτης υπερύθρων

Προκειμένου να ανιχνεύσουμε την υπέρυθη ακτινοβολία χρειαζόμαστε έναν δέκτη υπερύθρων. Βασικός ανιχνευτής ακτινοβολίας (φωτεινής και υπέρυθρης) είναι η *φωτοδιόδος*. Όπως γνωρίζουμε, ένας τρόπος για να διασπάσουμε δεσμούς σε έναν ημιαγωγό και να δημιουργήσουμε ελεύθερα ηλεκτρόνια και οπές είναι να τον προσβάλλουμε με φωτεινή ακτινοβολία κατάλληλου μήκους κύματος (ένας δεύτερος τρόπος διάσπασης των δεσμών είναι με την αύξηση της ανάστροφης τάσης). Προκειμένου να συμβεί κάτι τέτοιο, θα πρέπει τα φωτόνια της ακτινοβολίας να έχουν όσο ενέργεια  $hf$  χρειάζεται για τη διάσπαση του δεσμού. Σε αυτό ακριβώς το φαινόμενο στηρίζεται η λειτουργία των φωτοδιόδων.

Η φωτοδιόδος αποτελείται από μία επαφή pn που είναι τοποθετημένη μέσα σε διαφανές υλικό. Το υλικό αυτό είναι κατάλληλα χρωματισμένο ή σκεπάζεται με ειδικό μεταλλικό περίβλημα, έτσι ώστε το φως να πέφτει σε ορισμένη περιοχή της επαφής. Το σύνολο έχει πολύ μικρό όγκο με διαστάσεις μερικά δέκατα του cm.

Όταν πολώσουμε ανάστροφα την επαφή pn σε σκοτεινό χώρο, θα έχουμε ένα πολύ μικρό και σταθερό ανάστροφο ρεύμα, το οποίο λέγεται ρεύμα σκότους. Φωτίζοντας τη φωτοδιόδο με ακτινοβολία κατάλληλης συχνότητας, το ανάστροφο ρεύμα αυξάνεται, αφού τα προσπίπτοντα φωτόνια διασπών τους δεσμούς του καθαρού ημιαγωγού, με αποτέλεσμα να δημιουργούνται νέα ηλεκτρόνια και οπές μέσα στην περιοχή απογύμνωσης, που προστίθενται σε αυτά που ήδη προϋπήρχαν. Η τιμή του ρεύματος εξαρτάται τόσο από το είδος της διόδου, όσο και από την φωτεινή ροή και το μήκος κύματος της ακτινοβολίας που προσπίπτει.

Τα κυριότερα χαρακτηριστικά των φωτοδιόδων είναι :

1. Η περιοχή φάσματος της φωτεινής ακτινοβολίας στην οποία μπορεί να λειτουργεί.
2. Το ρεύμα σκότους.
3. Η μέγιστη ανάστροφη τάση λειτουργίας.
4. Η απόκριση συχνότητας. Μέχρι ποια συχνότητα λειτουργεί ικανοποιητικά.

5. Η ευαισθησία ρεύματος. Πόσο ρεύμα, δηλαδή, μπορεί να δώσει ανά μονάδα φωτισμού.

Πέρα από τις φωτοδιόδους επαφής pn, υπάρχουν και φωτοδιόδοι ακίδας που έχουν μεγάλες ταχύτητες λειτουργίας. Η χρησιμοποίησή τους είναι δυνατή σε διατάξεις που είναι εύκολη η εστίαση της ακτινοβολίας σε μία πολύ μικρή περιοχή. Υπάρχουν επίσης και φωτοδιόδοι χιονοστιβάδας. Αυτές λειτουργούν με ανάστροφες τάσεις μεγαλύτερες από την τάση Zener, κυρίως στην περιοχή των μικροκυμάτων.

Φωτοδιόδοι που μπορούν να λειτουργήσουν σε μεγάλες ταχύτητες και με μεγάλη επιλεκτικότητα στην συχνότητα, κατασκευάζονται με επαφή pn από διαφορετικά υλικά, όπως για παράδειγμα, GaAs τύπου P σε επαφή με Ge τύπου N.

Ένας άλλος φωτοανιχνευτής είναι το *φωτοτρανζίστορ*. Το φωτοτρανζίστορ έχει την ίδια κατασκευή με ένα τρανζίστορ επαφής, με τη διαφορά ότι προβλέπεται η δυνατότητα φωτισμού της επαφής βάσεως – συλλέκτη. Μπορεί να λειτουργήσει είτε με τη βάση πολωμένη με κάποια πόλωση, είτε με τη βάση ασύνδετη (στον αέρα). Γνωρίζουμε ότι το ρεύμα συλλέκτη σε ένα τρανζίστορ επαφής δίνεται από την εξίσωση :

$$I_c = \beta I_b + (\beta + 1)I_{cbo}$$

Όταν η βάση είναι ασύνδετη, η προηγούμενη σχέση γίνεται :

$$I_c = (\beta + 1)I_{cbo}$$

Δηλαδή, το ρεύμα  $I_c$  είναι  $\beta+1$  φορές το ανάστροφο ρεύμα βάσεως – συλλέκτη. Το ρεύμα αυτό είναι ουσιαστικά το ρεύμα της επαφής  $J_c$  (επαφή βάσης – συλλέκτη). Όταν φωτιστεί η επαφή αυτή, δημιουργούνται επιπλέον φορείς στην περιοχή απογυμνώσεως, με αποτέλεσμα την αύξηση του ανάστροφου ρεύματος  $I_{cbo}$  (η επαφή  $J_c$  είναι πολωμένη ανάστροφα), και συνεπώς του  $I_c$ .

Οι χαρακτηριστικές καμπύλες του φωτοτρανζίστορ είναι όμοιες με αυτές του διπολικού τρανζίστορ σε συνδεσμολογία κοινού εκπομπού, με τη διαφορά ότι αντί να έχουν ως παράμετρο το ρεύμα  $I_b$ , έχουν τη φωτεινή ισχύ.

Στις μέρες μας η φωτοανιχνευτική διάταξη που χρησιμοποιείται περισσότερο στους ολοκληρωμένους ανιχνευτές υπέρυθρης ακτινοβολίας, είναι η *δίοδος PIN*. Η δίοδος αυτή, ανάμεσα στις ισχυρά νοθευμένες περιοχές p και n, έχει μία επιπλέον ανόθευτη περιοχή i (καθαρός ημιαγωγός). Εξαιτίας αυτής της περιοχής, όταν η δίοδος πολώνεται ορθά, συμπεριφέρεται σαν μεταβλητή αντίσταση (που εξαρτάται από το ρεύμα ορθής φοράς – η αντίστασή της ελαττώνεται όταν αυξάνεται το ρεύμα  $I_F$ ). Αντίθετα, όταν πολώνεται ανάστροφα συμπεριφέρεται σχεδόν ως πυκνωτής σταθερής χωρητικότητας, ανεξάρτητης από την τάση πόλωσης.

Η δίοδος PIN χρησιμοποιείται ως ανιχνευτής ακτινοβολίας σε ανάστροφη πόλωση. Όταν ένα φωτόνιο πέφτει στην ανόθευτη περιοχή i, τότε ελευθερώνει έναν φορέα. Κάτω από την επίδραση της πόλωσης, ο φορέας επιταχύνεται και απομακρύνεται από την περιοχή αυτή. Αυτό δημιουργεί ένα ηλεκτρικό ρεύμα.

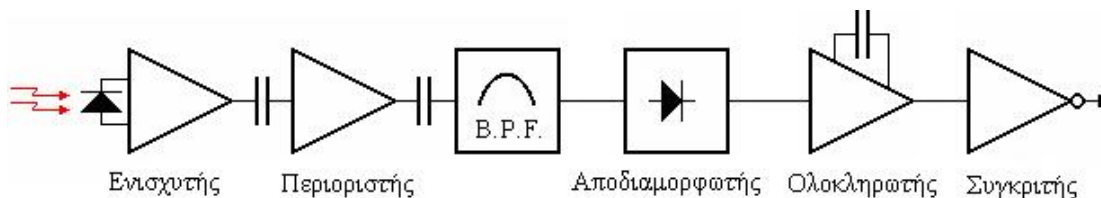


## 2.5.4 Οι ολοκληρωμένοι δέκτες υπέρυθρων

Εξαιτίας της μεγάλης τυποποίησης στον τηλεχειρισμό με υπέρυθρες, έχει αναπτυχθεί και δημιουργηθεί ένας μεγάλος αριθμός από έτοιμους ολοκληρωμένους δέκτες υπέρυθρων. Σήμερα, κατασκευάζονται έτοιμα ολοκληρωμένα κυκλώματα (συνήθως 3 ακροδεκτών) προστατευμένα μέσα σε ένα ειδικό πλαστικό περίβλημα (μαύρου χρώματος), που ταυτόχρονα λειτουργεί και ως φίλτρο ορατού φωτός (δηλαδή επιτρέπει τη διέλευση μόνο της υπέρυθρης ακτινοβολίας). Μέσα σε αυτά τα ολοκληρωμένα ενσωματώνεται ο IR δέκτης (PIN δίοδος) καθώς και το απαραίτητο ηλεκτρονικό κύκλωμα.

Το σημαντικότερο κριτήριο επιλογής ενός τέτοιου δέκτη, είναι η συχνότητα του φέροντος κύματος που χρησιμοποιείται στη διαμόρφωση, η οποία στον δέκτη μεταφράζεται ως η κεντρική συχνότητα ζωνοπερατού φίλτρου.

Στο παρακάτω σχήμα παρουσιάζεται το μπλοκ διάγραμμα ενός τυπικού δέκτη.

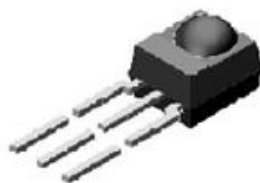


Σχήμα 2.5.4.1. Το μπλοκ διάγραμμα ενός ολοκληρωμένου δέκτη υπέρυθρων.

Το σήμα IR λαμβάνεται από τον δέκτη και ενισχύεται από τον ενισχυτή που ακολουθεί. Ο περιοριστής λειτουργεί ως ένα κύκλωμα αυτόματης ρύθμισης της απολαβής, αφού στην έξοδό του λαμβάνεται ένα σήμα σταθερού πλάτους, ανεξάρτητα της απόστασης της συσκευής. Στο ζωνοπερατό φίλτρο οδηγείται μόνο το ac σήμα (εξαιτίας του πυκνωτή σύζευξης). Η κεντρική συχνότητα του φίλτρου είναι ίση με τη συχνότητα του διαμορφωμένου φέροντος του πομπού. Οποιαδήποτε άλλη συχνότητα απορρίπτεται. Οι συνηθισμένες συχνότητες που χρησιμοποιούνται στο εμπόριο κυμαίνονται στην περιοχή από 30KHz έως 60KHz.

Οι επόμενες βαθμίδες είναι ο αποδιαμορφωτής, ο ολοκληρωτής και ο συγκριτής. Ο σκοπός τους είναι να ανιχνεύσουν την παρουσία της φέρουσας συχνότητας. Όσο αυτή είναι παρούσα, η έξοδος του συγκριτή μεταβαίνει σε χαμηλή στάθμη (0V). Στην αντίθετη περίπτωση παραμένει σε υψηλή λογική στάθμη (5V).

Όλο το προηγούμενο κύκλωμα υλοποιείται μέσα σε ένα ολοκληρωμένο, όπως φαίνεται στην παρακάτω εικόνα.



Εικόνα 2.5.4.1. Ένας σύγχρονος ολοκληρωμένος IRδέκτης.

Υπάρχουν αρκετές διαφορετικές κατασκευάστριες εταιρίες που φτιάχνουν τέτοιου είδους ολοκληρωμένα, για διαφορετικές τιμές της φέρουσας συχνότητας. Στην Ευρώπη υπάρχουν κυρίως η Siemens, η Vishay και η Telefunken. Η Siemens προωθεί την σειρά SFH506-xx, όπου το xx παίρνει τις τιμές 30, 33, 36, 38, 40 και 56 που αντιπροσωπεύουν τις αντίστοιχες τιμές της φέρουσας συχνότητας σε KHz. Η Telefunken είχε δημιουργήσει τις σειρές TFMS5xx0 και TK18xx, όπου το xx υποδεικνύει και πάλι τη συχνότητα του φέροντος κύματος για το οποίο προορίζεται το κάθε ολοκληρωμένο (ουσιαστικά είναι η κεντρική συχνότητα του ζωνοπερατού φίλτρου). Τα προηγούμενα ολοκληρωμένα έχουν καταργηθεί και έχουν αντικατασταθεί από τα αντίστοιχα ολοκληρωμένα της εταιρίας Vishay. Η τελευταία έχει δημιουργήσει τις σειρές TSOP12xx, TSOP17xx, TSOP18xx, TSOP348xx κ.α. Εκτός από τις Ευρωπαϊκές, υπάρχουν και οι 3 Ασιατικές εταιρίες Sharp, Xiamen Hualian και Japanese Electric, που κατασκευάζουν αντίστοιχα ολοκληρωμένα.

# **ΜΕΡΟΣ Β**

## **ΤΟ ΡΟΜΠΟΤ ΑΚΟΥΣΤΙΚΩΝ ΜΕΤΡΗΣΕΩΝ**

Σκοπός της παρούσας διπλωματικής εργασίας είναι η κατασκευή και η πλοήγηση ενός αυτόνομου ρομπότ – οχήματος, το οποίο θα μετακινείται από ένα δοθέν σημείο του χώρου σε κάποιο άλλο, προκειμένου να λάβει ακουστικές μετρήσεις, αποφεύγοντας παράλληλα τα εμπόδια που συναντά. Οι ακουστικές μετρήσεις πραγματοποιούνται με τη χρήση ενός προσαρμοσμένου μικροφώνου, και αποθηκεύονται στην μονάδα μνήμης ενός μικρού φορητού Η/Υ (netbook), ο οποίος είναι τοποθετημένος πάνω στην κατασκευή.

Επιδίωξή μας είναι η χρήση της εφαρμογής για την καταγραφή της ηχητικής ενός μεγάλου μέρους του χώρου. Ζητούμενο, λοιπόν, αποτελεί η «σάρωση» του χώρου, με σκοπό τη λήψη ακουστικών μετρήσεων ανά δεδομένα διαστήματα. Για αυτόν το λόγο, ο χώρος χωρίζεται σε κανονικά τετράγωνα, τα οποία και θα αποτελέσουν ενδιάμεσους και τελικούς προορισμούς του οχήματος. Συνεπώς, πριν την έναρξη της λειτουργίας, εισάγεται στον υπολογιστή του ρομπότ, ένας πίνακας δεδομένων που περιέχει τα σημεία – τετράγωνα στα οποία ζητείται η λήψη ακουστικών μετρήσεων. Γίνεται ανάγνωση του επόμενου σημείου προορισμού, και στην συνέχεια το ρομπότ ξεκινάει την λειτουργία του. Όταν το όχημα καταλήξει στο τελικό σημείο, σταματάει με σκοπό να

Ξεκινήσει η διαδικασία λήψης των μετρήσεων. Αφού ολοκληρωθεί η παραπάνω διαδικασία, το ρομπότ αναμένει την ανάλογη εντολή, ώστε να προχωρήσει στο επόμενο τετράγωνο. Παράλληλα με την κίνηση του οχήματος, πρέπει να ελέγχεται η δυνατότητα πρόσβασης στο εκάστοτε σημείο προορισμού. Κατά συνέπεια, αν το τελικό σημείο είναι μη προσβάσιμο, δηλαδή αν βρίσκεται εκτός των ορίων του χώρου, τότε το ρομπότ πρέπει να το αντιλαμβάνεται, ώστε να συνεχιστεί η πλοήγησή του προς το επόμενο τετράγωνο.

Ο πίνακας των δεδομένων εισόδου μπορεί να είναι ένα φύλλο excel, στο οποίο θα περιέχονται όλα τα σημεία έναρξης και προορισμού, με έναν αριθμοδείκτη για κάθε ζεύγος αρχικού – τελικού σημείου, όπως περιγράφεται στον παρακάτω πίνακα.

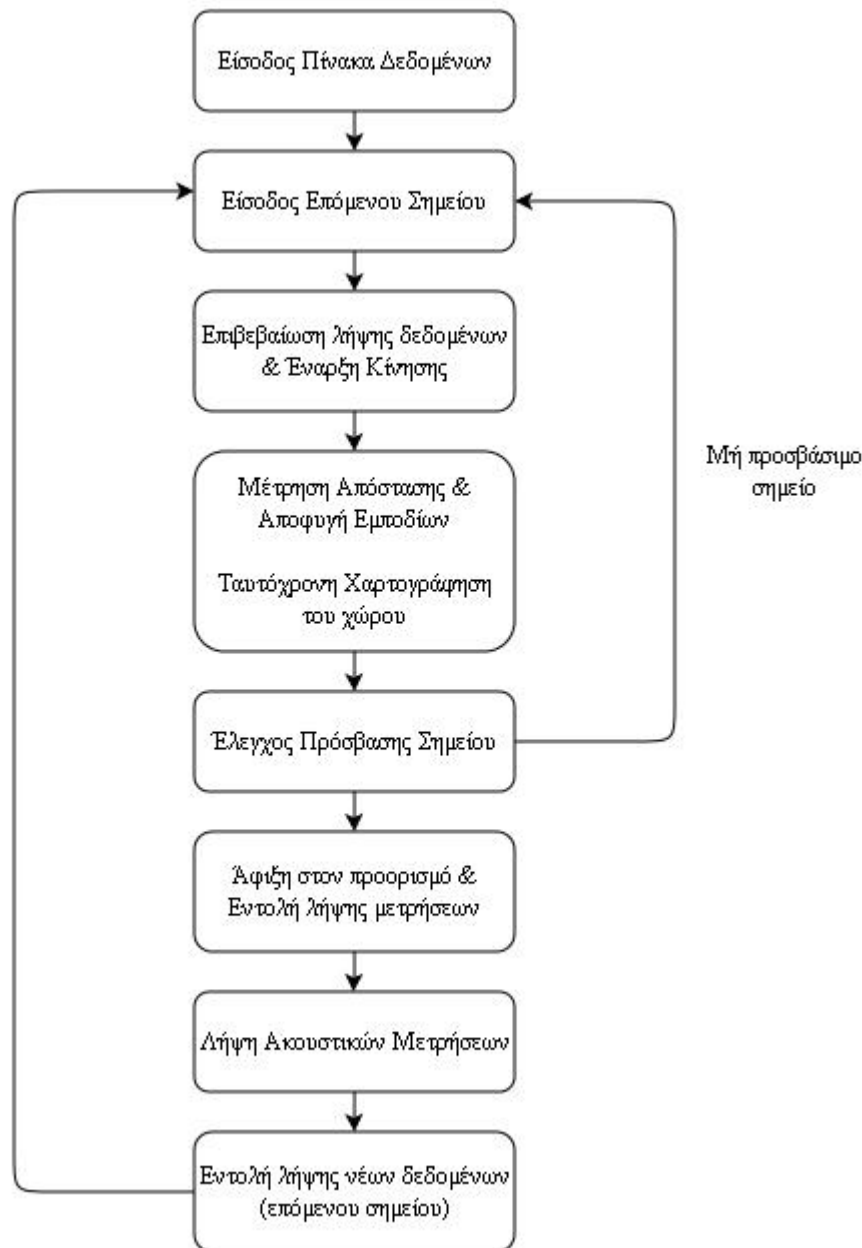
No.	Αρχικό Σημείο	Τελικό Σημείο
1	$(x_0, y_0)$	$(x_1, y_1)$
2	$(x_1, y_1)$	$(x_2, y_2)$
3	$(x_2, y_2)$	$(x_3, y_3)$
.	.	.
.	.	.
.	.	.
N	$(x_{N-1}, y_{N-1})$	$(x_N, y_N)$

*Πίνακας. Περιγραφή της μορφής του πίνακα δεδομένων εισόδου.*

Ένας πίνακας της παραπάνω μορφής λοιπόν, καταχωρείται στην μνήμη του ενσωματωμένου υπολογιστή. Αρχικά, ο δείκτης που χρησιμοποιούμε έχει την τιμή 1, δείχνοντας το πρώτο ζεύγος σημείων. Πραγματοποιείται η πλοήγηση μεταξύ του σημείου  $(x_0, y_0)$  και του σημείου  $(x_1, y_1)$ , και όταν το ρομπότ φτάσει στο σημείο προορισμού  $(x_1, y_1)$  δίνεται η ανάλογη εντολή για την έναρξη της λήψης ακουστικών μετρήσεων. Όταν ολοκληρωθεί η διαδικασία αυτή, χρησιμοποιώντας μία εντολή NEXT για παράδειγμα (ανάλογα με τον τρόπο που θα προγραμματιστεί η συγκεκριμένη λειτουργία), αυξάνουμε τον αριθμοδείκτη ώστε να δείχνει πλέον στο επόμενο ζεύγος σημείων. Το νέο σημείο προορισμού είναι το  $(x_2, y_2)$ , και με τον τρόπο αυτόν συνεχίζεται η διαδικασία «σάρωσης» του χώρου.

Μελλοντικά, μπορεί να προστεθεί και δυνατότητα ασύρματης αποστολής των δεδομένων εισόδου, προκειμένου να έχει κάθε φορά ο χρήστης την ευχέρεια να στέλνει το επιθυμητό σημείο προορισμού, όταν του δοθεί η ανάλογη εντολή. Στην περίπτωση αυτήν δεν θα ήταν απαραίτητη η χρήση ενός πίνακα δεδομένων, με ορισμένα εξαρχής, όλα τα σημεία ενδιάμεσων και τελικών προορισμών.

Στο παρακάτω μπλοκ διάγραμμα, περιγράφεται η ολοκληρωμένη εργασία που έχει ως σκοπό να επιτελέσει τελικά το παρόν ρομποτικό σύστημα.



Σχήμα. Περιγραφή της πρακτικής εφαρμογής του ρομπότ ακουστικών μετρήσεων.

Το ρομπότ προσεγγίζει το σημείο στο οποίο στέλνεται, υπολογίζοντας την απόσταση που διανύει. Με τον τρόπο αυτόν, το ρομπότ έχει συνεχώς απόλυτη επίγνωση της τρέχουσας θέσης του, καθώς επίσης και του προσανατολισμού του. Παράλληλα, στέλνει συνεχώς μηνύματα τα οποία καταγράφονται στην μνήμη του υπολογιστή, και παρέχουν πληροφορίες που έχουν να κάνουν με την τρέχουσα θέση του οχήματος, καθώς επίσης και με την περιγραφή των εμποδίων που συναντά. Σκοπός των μηνυμάτων αυτών είναι η καλύτερη κατανόηση του χώρου. Όταν το όχημα θα καταλήξει στο σημείο προορισμού, θα έχει πλέον καταγραφεί πλήρως η ακριβής διαδρομή που ακολούθησε, και έτσι θα υπάρχει η δυνατότητα χαρτογράφησης του χώρου μαζί με τα εμπόδια που

περιέχονται σε αυτόν. Η χαρτογράφηση του χώρου θα δίνει στο ρομπότ τη δυνατότητα να επιστρέψει με ευκολία στο αρχικό του σημείο.

Η μέτρηση της απόστασης γίνεται με τη βοήθεια ενσωματωμένου αυτοσχέδιου οδομέτρου. Η κίνηση του οχήματος πραγματοποιείται με τη χρήση σερβοκινητήρων, ενώ ο εντοπισμός και η αποφυγή των εμποδίων που συναντά γίνεται χρησιμοποιώντας αισθητήρες υπερύθρων. Όλη η λειτουργία του ρομπότ προγραμματίστηκε με τη χρήση του επεξεργαστή arduino.

Αντικείμενο της παρούσας εργασίας είναι η μεταφορά του ρομπότ – οχήματος από ένα σημείο του χώρου σε ένα άλλο, δηλαδή, η χάραξη της διαδρομής μεταξύ δύο διαδοχικών σημείων – τετραγώνων, η οποία και θα αποτελέσει την βάση μιας επαναληπτικής διαδικασίας για «σάρωση» όλου του χώρου. Η διαδικασία της λήψης των ακουστικών μετρήσεων δεν αποτελεί αντικείμενο της συγκεκριμένης διπλωματικής εργασίας.

Κλείνοντας, να σημειώσουμε ότι το όλο project αναφέρεται σε μεγάλους εσωτερικούς χώρους, όπως για παράδειγμα ένα μεγάλο club ή ένα κλειστό γυμναστήριο, όπου θέλουμε να καταγράψουμε τις διαφορές της ηχητικής από σημείο σε σημείο του χώρου.

## ΚΕΦΑΛΑΙΟ 3

### ΚΑΤΑΣΚΕΥΗ

#### 3.1 Εισαγωγή

Το ρομπότ ακουστικών μετρήσεων αποτελεί ένα αυτόνομο κινούμενο ρομποτικό σύστημα. Η κίνησή του πραγματοποιείται με τη βοήθεια τροχών (τροχοφόρο όχημα), ενώ ακολουθεί διαφορετικό τρόπο οδήγησης, τον οποίο και αναλύσαμε στο προηγούμενο κεφάλαιο. Ως ρομπότ διαφορικής οδήγησης, το σύστημα είναι μη-ελέγξιμο (μη-ολονομικό), καθώς όπως έχουμε αναφέρει, διαθέτει 3 βαθμούς ελευθερίας (2 για την μετατόπιση  $x$ ,  $y$ , 1 για την περιστροφή  $\varphi$ ), εκ των οποίων οι 2 είναι ελεγχόμενοι (μέσω των 2 κινητήρων) και ο ένας μη-ελεγχόμενος.

Το ρομποτικό σύστημα της παρούσας εργασίας αποτελεί μία προσωπική κατασκευή, καθώς έχει κατασκευαστεί εξολοκλήρου από την αρχή (από το σασί του ρομπότ, μέχρι τον επεξεργαστή, τα κυκλώματα των αισθητήρων και άλλες κυκλωματικές διατάξεις). Η κατασκευή πραγματοποιήθηκε με απλά υλικά, τα οποία και προμηθευτήκαμε κυρίως από καταστήματα ηλεκτρονικών ειδών, ενώ οι κινητήρες αγοράστηκαν μέσω του διαδικτύου. Το ρομπότ κατασκευάστηκε με τρόπο τέτοιο ώστε να τηρούνται συγκεκριμένες προδιαγραφές. Από τη μία πλευρά, έπρεπε να έχει σχετικά μεγάλες διαστάσεις, πρώτον ώστε να υπάρχει η δυνατότητα να τοποθετηθεί ένας φορητός υπολογιστής πάνω σε αυτό και δεύτερον, για να έχει μεγαλύτερη ισορροπία κατά την κίνησή του (όπως θα δούμε αργότερα το μικρόφωνο πρέπει να τοποθετηθεί σε αρκετά μεγάλο ύψος, συγκριτικά με τις υπόλοιπες διαστάσεις). Από την άλλη πλευρά, υπήρχε η ανάγκη για όσο το δυνατό ελαφρύτερη κατασκευή, προκειμένου να μην υπάρχει ιδιαίτερα μεγάλη επιβάρυνση της μπαταρίας του ρομπότ.







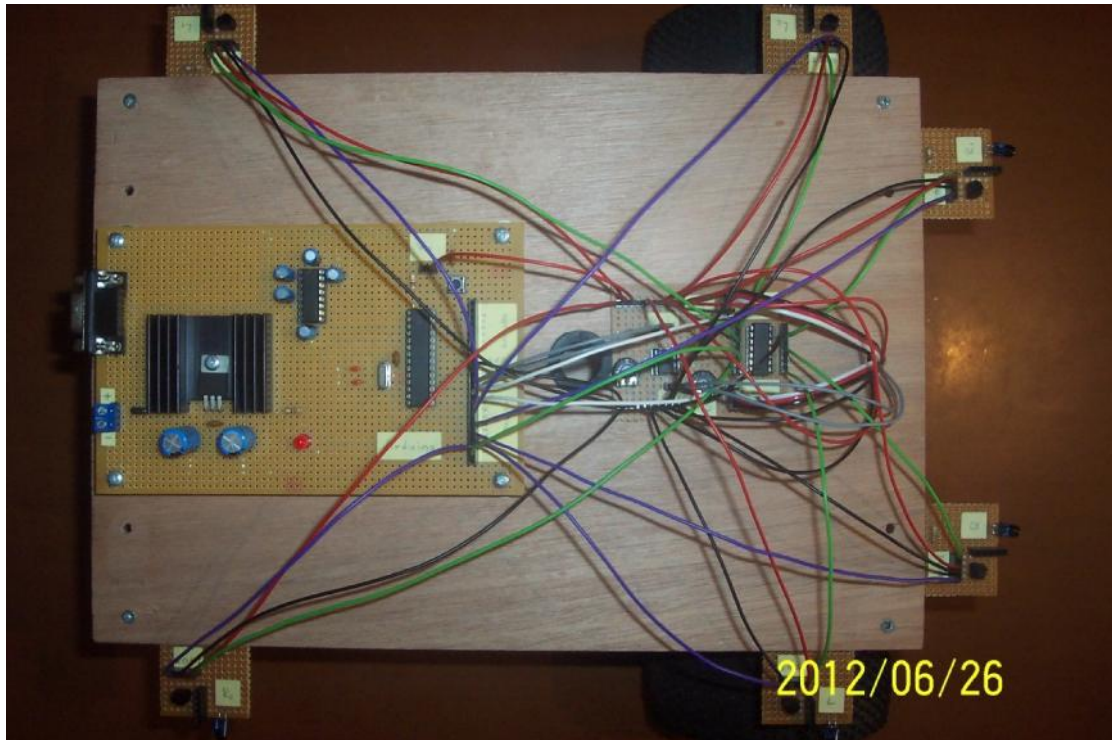
*Εικόνες 3.1.1. και 3.1.2. Το ρομπότ ακουστικών μετρήσεων.*

### **3.2 Το μηχανολογικό μέρος – σασί του ρομπότ**

Το ρομποτικό όχημα είναι μία, κατά κύριο λόγο, ξύλινη κατασκευή. Η επιλογή αυτή έγινε για λόγους απλότητας, οικονομίας, ευκολίας στην επεξεργασία του ξύλου, καθώς επίσης και εξαιτίας του μικρού βάρους του συγκεκριμένου υλικού που χρησιμοποιήθηκε. Η κατασκευή αποτελείται από 3 επίπεδα. Ο υπολογιστής του ρομπότ είναι τοποθετημένος στο πρώτο επίπεδο της κατασκευής, προκειμένου να υπάρχει όσο το δυνατόν χαμηλότερο κέντρο βάρους, και κατά συνέπεια μεγαλύτερη ισορροπία. Στο δεύτερο επίπεδο βρίσκεται η πλακέτα του επεξεργαστή, καθώς επίσης και άλλες κυκλωματικές διατάξεις, ενώ στο τρίτο επίπεδο, έχει συνδεθεί η μπαταρία του ρομπότ και έχει προσαρμοστεί κατάλληλα ο στύλος πάνω στον οποίο θα τοποθετηθεί το μικρόφωνο για την λήψη των ακουστικών μετρήσεων.

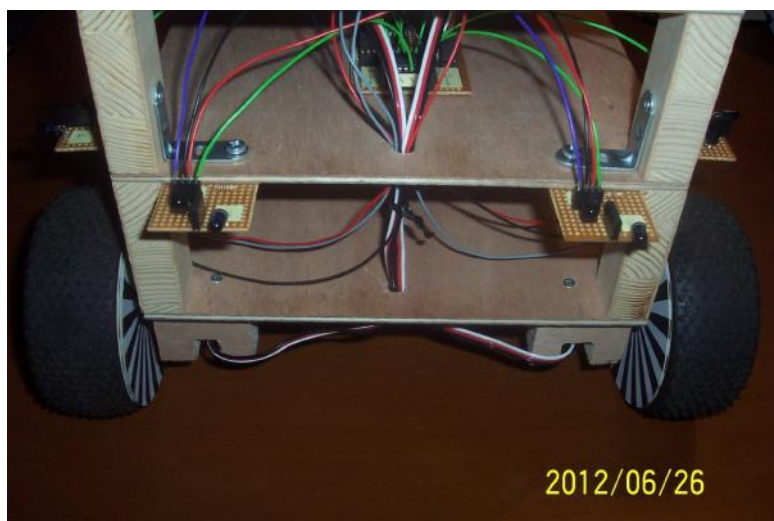


*Εικόνα 3.2.1. Προσαρμογή του στύλου για την στήριξη του μικροφώνου.*



*Εικόνα 3.2.2. Διασυνδέσεις μεταξύ των κυκλωματικών διατάξεων στο δεύτερο επίπεδο της κατασκευής.*

Το δεύτερο με το τρίτο επίπεδο μπορούν εύκολα να αποσυνδεθούν (όπως φαίνεται στις παρακάτω φωτογραφίες), προκειμένου να έχουμε τη δυνατότητα να επέμβουμε στις συνδέσεις μεταξύ των κυκλωμάτων.





Εικόνες 3.2.3. και 3.2.4. Τρόπος διασύνδεσης του δεύτερου και του τρίτου επιπέδου της κατασκευής.

Ως ρομπότ διαφορετικής οδήγησης, το όχημα κινείται πάνω σε 3 τροχούς, δύο κινητήριους τροχούς και έναν βοηθητικό. Οι δυο *κινητήριοι τροχοί* (οι τροχοί που περιστρέφονται από τους 2 κινητήρες του ρομπότ) βρίσκονται στο μπροστινό μέρος του οχήματος, ενώ στο πίσω μέρος είναι προσαρμοσμένος ένας μικρότερον διαστάσεων βοηθητικός τροχός, με δυνατότητα περιστροφής προς οποιαδήποτε κατεύθυνση, ο οποίος ονομάζεται *caster wheel* (εικόνα 3.2.5).



Εικόνα 3.2.5. Περιστρεφόμενος βοηθητικός τροχός – *caster wheel*.

Η επιλογή των διαστάσεων έγινε με τέτοιο τρόπο, ώστε (όπως έχει προαναφερθεί) στο πρώτο επίπεδο της κατασκευής να υπάρχει ο απαραίτητος χώρος για να τοποθετηθεί ένας μικρός φορητός Η/Υ. Ο υπολογιστής που χρησιμοποιήθηκε είναι ένα netbook τυπικών διαστάσεων, 25x17x4 cm περίπου (μήκος x πλάτος x ύψος), και βάρους της τάξης του 1,5kg. Η κατασκευή λοιπόν, έχει τις παρακάτω διαστάσεις :

- μήκος = 32 cm
- πλάτος = 22 cm
- ύψος = 24 cm

Το βάρος της κατασκευής, χωρίς να υπολογίζεται και το βάρος του netbook, είναι της τάξης των 2,5kg, κατά συνέπεια το συνολικό βάρος του συστήματος φτάνει τα 4kg περίπου. Το συγκεκριμένο βάρος αποτελεί μία σαφώς ικανοποιητική τιμή, καθώς όπως μπορούμε να αντιληφθούμε, το ρομπότ πρόκειται για μία αρκετά ελαφριά κατασκευή (ιδιαίτερα συγκριτικά με τις διαστάσεις της), γεγονός που αποτελούσε και τον αρχικό στόχο.

Όπως φαίνεται και στην παρακάτω εικόνα, η προσαρμογή των δυο μπροστινών τροχών έχει γίνει με τέτοιο τρόπο έτσι ώστε η κατασκευή να έχει συνολικά μεγάλη ευστάθεια.



*Εικόνα 3.2.6. Τοποθέτηση μπροστινών τροχών.*

Πρόκειται για τροχούς οι οποίοι χρησιμοποιούνται στον μοντελισμό, ενώ περιβάλλονται από καλής ποιότητας ελαστικό που δημιουργεί ισχυρή τριβή με το έδαφος, και μειώνει σε μεγάλο βαθμό την κύλιση με ολίσθηση, η οποία και εισάγει σφάλμα που προσπαθούμε να αποφύγουμε.



*Εικόνα 3.2.7. Λάστιχο τροχών.*

Οι δυο μπροστινοί τροχοί έχουν διάμετρο 11 cm, και πάχος 4 cm, ενώ ο περιστρεφόμενος τροχός έχει διάμετρο 4 cm, και πάχος 3 cm.





*Εικόνα 3.2.8. Μπροστινός τροχός.*

Τέλος, όπως έχει προαναφερθεί, το μικρόφωνο που λαμβάνει τις ακουστικές μετρήσεις, ήταν απαραίτητο να τοποθετηθεί σε σχετικά μεγάλο ύψος. Για τον λόγο αυτόν, χρησιμοποιήθηκε ένας στύλος (πλαστικός σωλήνας) πάνω στον οποίο έχει εφαρμοστεί κατάλληλα το μικρόφωνο, σε ύψος 1,20m περίπου από το έδαφος. Το μικρόφωνο έχει βάρος της τάξης των 300gr, ενώ ο στύλος στήριξης του μικροφώνου έχει προσαρμοστεί, κατά προσέγγιση, στο σημείο όπου βρίσκεται κέντρο βάρους ολόκληρου του συστήματος.

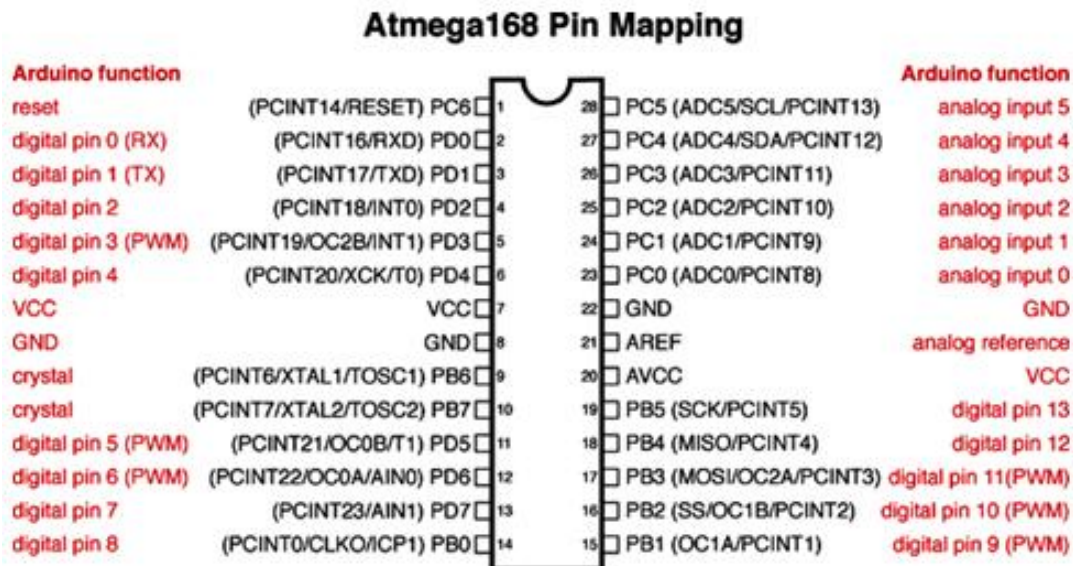


*Εικόνα 3.2.9. Ο στύλος για την στήριξη του μικροφώνου εκτείνεται σε ύψος 1,20m.*

### 3.3 Ο Επεξεργαστής

Ο επεξεργαστής που επιλέχθηκε για τον έλεγχο του συστήματος είναι ο arduino. Ο arduino βασίζεται σε έναν μικροελεγκτή AVR γενικής χρήσης. Η γλώσσα προγραμματισμού του στηρίζεται στην Wiring, μία παραλλαγή της C/C++ για μικροελεγκτές αρχιτεκτονικής AVR όπως ο ATmega, και υποστηρίζει όλες τις βασικές δομές της C καθώς και μερικά χαρακτηριστικά της C++. Στην γλώσσα του arduino όμως, εκτός από τις βασικές εντολές και συναρτήσεις της C, υπάρχουν και ειδικές εντολές που βοηθούν για τη διαχείριση του ειδικού του hardware.

Εξαιτίας της πολύ μικρής κατανάλωσης, και κυρίως της αρκετά μεγάλης μνήμης που διαθέτει (για τις απαιτήσεις της συγκεκριμένης εφαρμογής), επιλέχθηκε ο μικροελεγκτής AVR ATmega168, της εταιρίας ATMEL. Για την ακρίβεια, ο ATmega168 διαθέτει 16KB μνήμης, 2KB εκ των οποίων χρησιμοποιούνται από τον bootloader (βλ. Παράρτημα Α). Οι ακροδέκτες του ATmega168 με την σημασία τους, παρουσιάζονται στο παρακάτω σχήμα.



Σχήμα 3.3.1. Το ολοκληρωμένο ATmega168.

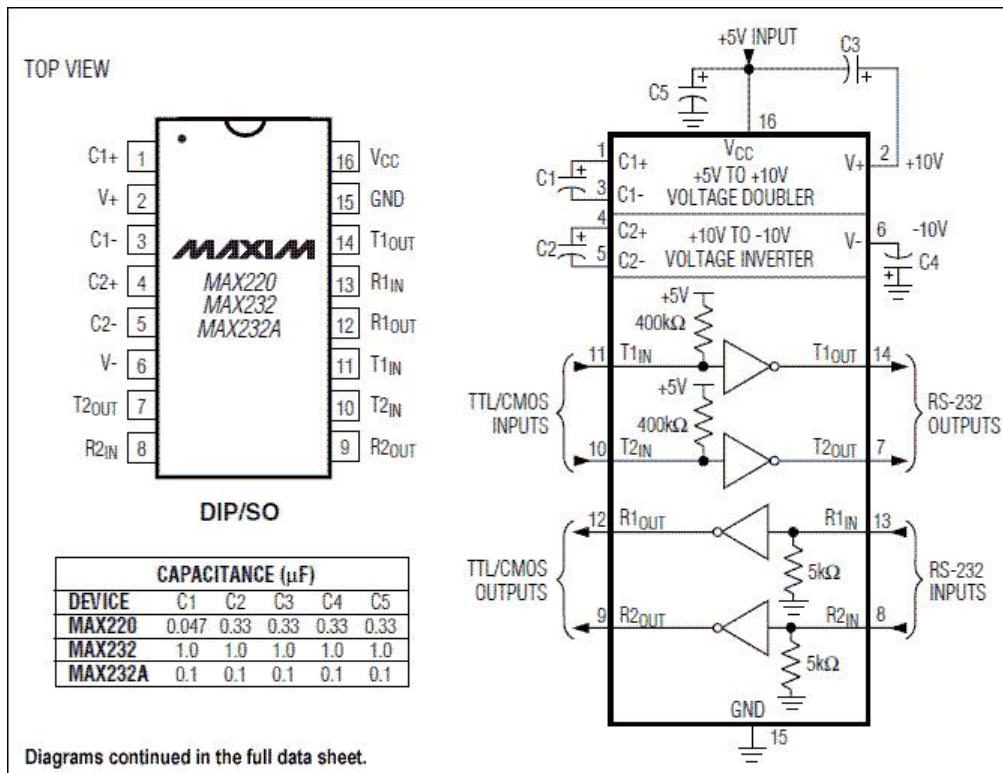
Εφόσον επιθυμούμε να χρησιμοποιήσουμε την θύρα USB για την σύνδεση και την επικοινωνία της συσκευής με τον υπολογιστή, ενώ ταυτόχρονα επιθυμούμε τη δυνατότητα προγραμματισμού της σειριακής θύρας, είμαστε αναγκασμένοι να καταφύγουμε στους μετατροπείς από USB σε σειριακή. Οι συγκεκριμένοι μετατροπείς κυκλοφορούν σε ξεχωριστά chip τα οποία στην πραγματικότητα κάνουν τη μετατροπή από USB σε λογικές στάθμες TTL. Δυστυχώς όμως, είναι σχετικά δύσκολο να βρεθούν στο εμπόριο.

Από την άλλη πλευρά, οι έτοιμοι μετατροπείς από USB σε RS232 (βλ. σχήμα 3.3.2), είναι ευρέως διαθέσιμοι στο εμπόριο, σε καταστήματα υπολογιστών. Οι συσκευές αυτές, ενσωματώνουν ένα chip USB σε TTL, καθώς επίσης και ένα επιπλέον ολοκληρωμένο που κάνει την απαραίτητη μετατροπή από TTL στάθμες, σε στάθμες RS232. Αυτό σημαίνει ότι για να καταφέρουμε να συνδέσουμε μία εξωτερική συσκευή με αυτόν τον μετατροπέα, απαιτείται η χρήση ενός μετατροπέα από RS232 σε στάθμες TTL.



Σχήμα 3.3.2. Ο μετατροπέας USB σε RS232 της Prolific.

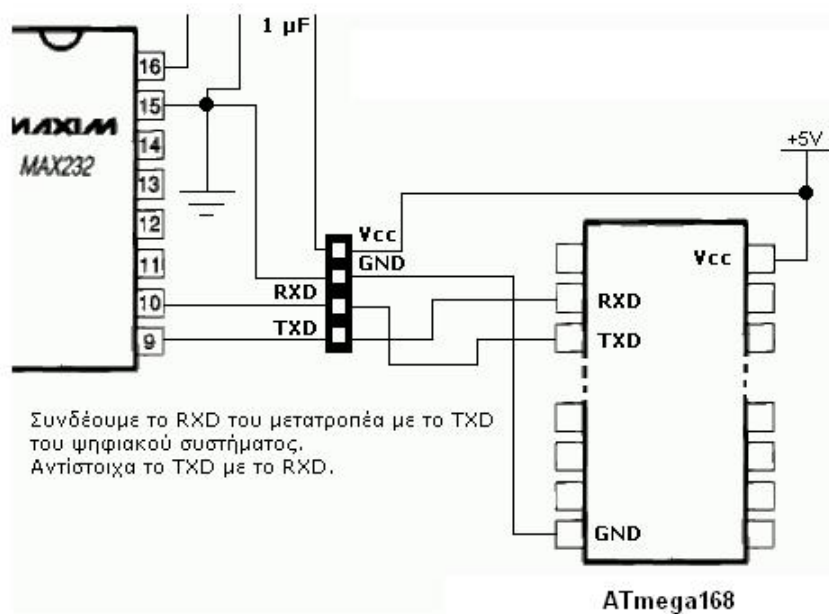
Το ολοκληρωμένο που χρησιμοποιήθηκε για την μετατροπή της στάθμης RS232 σε TTL, είναι το ολοκληρωμένο MAX232 της εταιρίας MAXIM. Αν χρησιμοποιήσουμε έναν μετατροπέα USB σε TTL, προφανώς το MAX232 δεν θα ήταν απαραίτητο. Ωστόσο, χρειάζεται στην περίπτωση που θέλουμε να συνδέσουμε τη συσκευή σε μία φυσική σειριακή θύρα του Η/Υ (εφόσον είναι διαθέσιμη).



Σχήμα 3.3.3. Το ολοκληρωμένο MAX232 και οι τιμές των πυκνωτών που πρέπει να συνδεθούν σε αυτό.

Ο ATmega168 διαθέτει έναν ενσωματωμένο προγραμματιζόμενο ελεγκτή USART, διευκολύνοντας έτσι την σύνδεσή του με τη σειριακή θύρα του υπολογιστή. Ο συγκεκριμένος ελεγκτής λειτουργεί με TTL στάθμες και για τον λόγο αυτόν, ήταν απαραίτητο να χρησιμοποιηθεί ένας μετατροπέας από TTL σε RS232 και το αντίστροφο (MAX232).

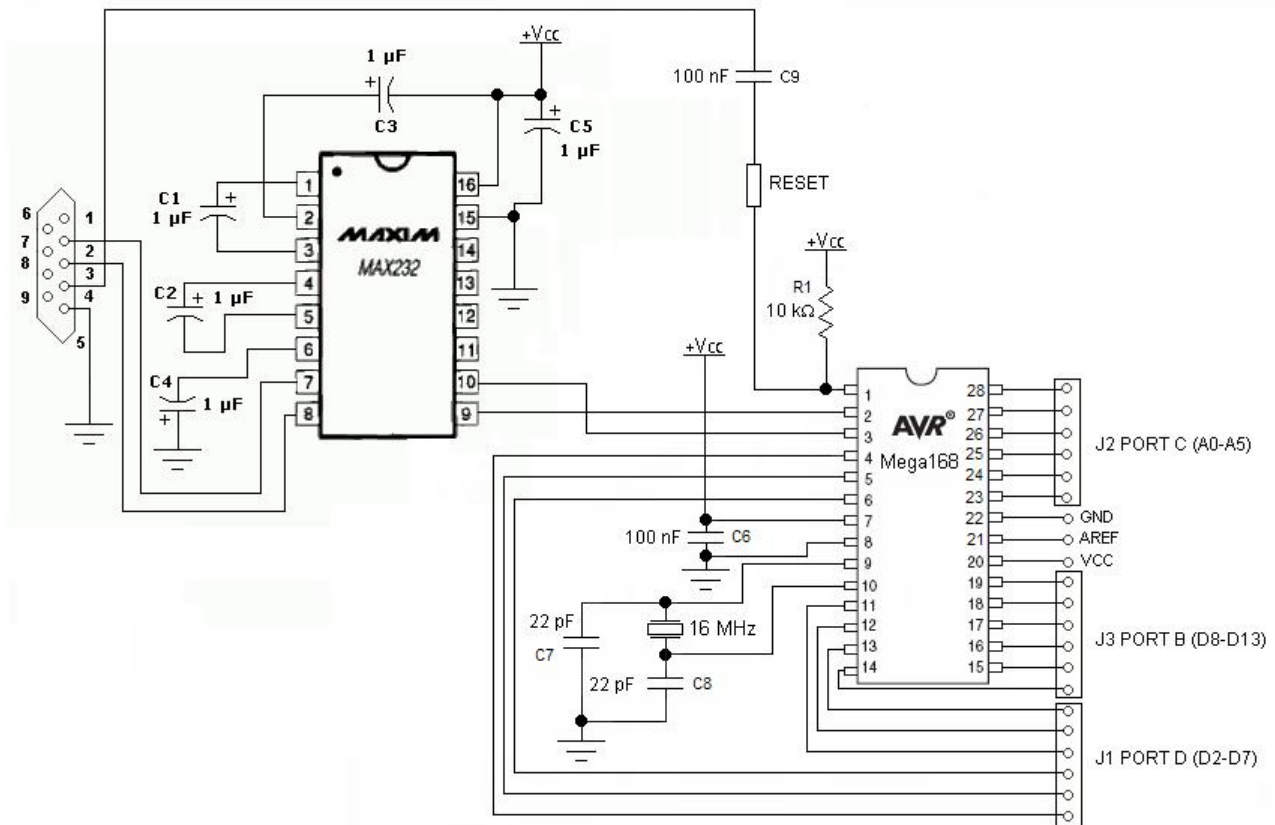
Για την σειριακή επικοινωνία χρησιμοποιούνται οι ακροδέκτες 2 και 3 (digital pin 0 και 1) του ATmega168, οι οποίοι και χρησιμοποιούνται για την λήψη (RX) και την εκπομπή (TX) TTL σειριακών δεδομένων αντίστοιχα. Οι ακροδέκτες αυτοί είναι συνδεδεμένοι με τους αντίστοιχους του ολοκληρωμένου MAX232, με τον τρόπο που δείχνει το παρακάτω σχήμα.



Σχήμα 3.3.4. Τρόπος σύνδεσης μετατροπέα.

Παρακάτω λοιπόν, παρουσιάζουμε το σχηματικό διάγραμμα του κυκλώματος επεξεργαστή, το οποίο και βασίζεται στον μικροελεγκτή ATmega168. Ο χρονισμός του μικροελεγκτή έγινε με τη χρήση ενός κρυσταλλικού ταλαντωτή συχνότητας 16MHz.

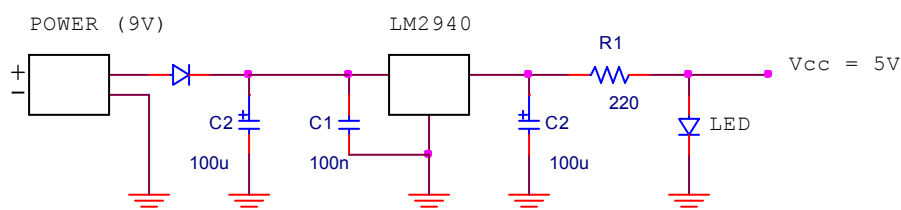




Σχήμα 3.3.5. Το σχηματικό διάγραμμα του κυκλώματος του arduino.

Ο ακροδέκτης 1 επιτελεί την επανεκκίνηση του μικροελεγκτή σε περίπτωση που τεθεί σε κατάσταση LOW. Για τον λόγο αυτόν, στον συγκεκριμένο ακροδέκτη έχει συνδεθεί ένας διακόπτης (RESET), προκειμένου να έχουμε τη δυνατότητα όταν επιθυμούμε να επανεκκινήσουμε τον ATmega.

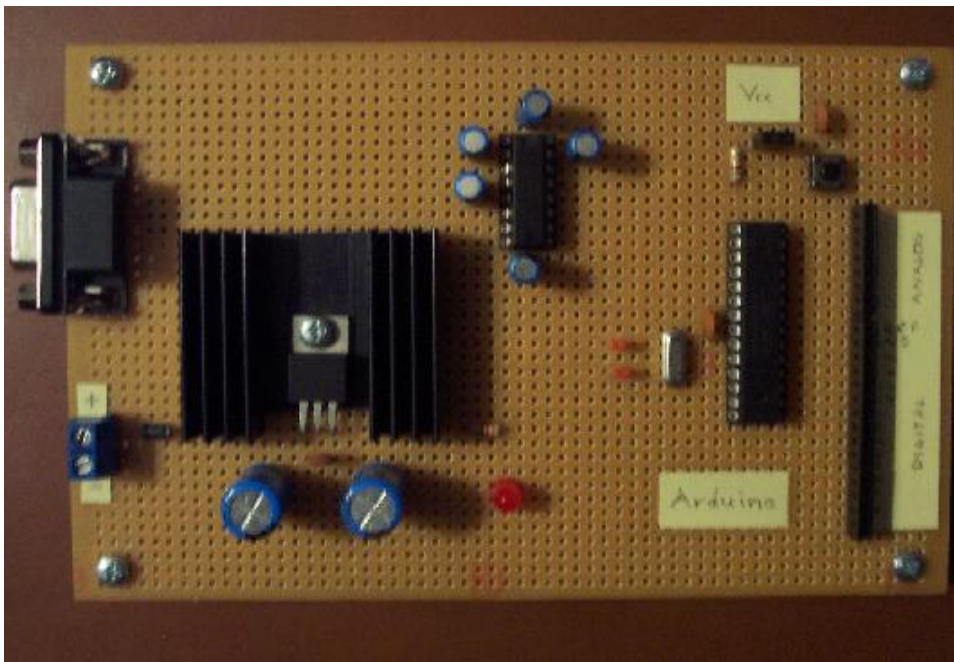
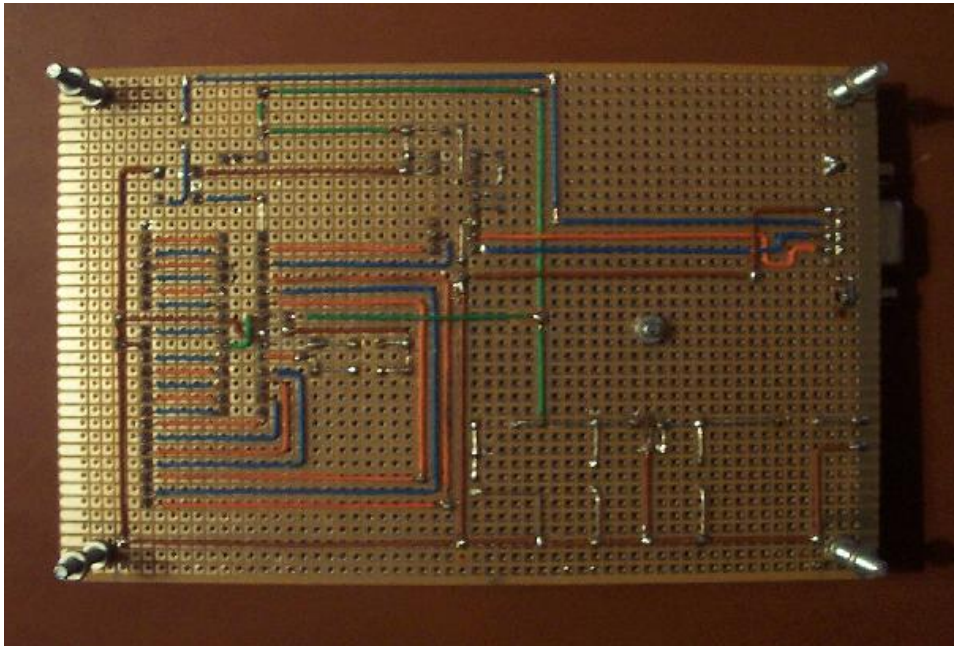
Τέλος, στο παρακάτω σχήμα παρουσιάζουμε το κύκλωμα τροφοδοτικού. Ο σταθεροποιητής LM2940 υποβιβάζει την εννιάβολτη τάση τροφοδοσίας του συστήματος (βλ. παράγραφο 3.7) και την διατηρεί στο επίπεδο των 5V, η οποία και αποτελεί την τάση λειτουργίας του μικροελεγκτή ATmega168.



Σχήμα 3.3.6. Κύκλωμα τροφοδοτικού.

Το κόκκινο LED χρησιμοποιείται ως ένδειξη για την έναρξη της λειτουργίας του συστήματος.

Παρακάτω λοιπόν, παρουσιάζουμε το κύκλωμα του επεξεργαστή που κατασκευάστηκε, σε διάτριτη πλακέτα πυριτίου.



Σχήματα 3.3.7 και 3.3.8. Ο επεξεργαστής του ρομποτικού συστήματος.

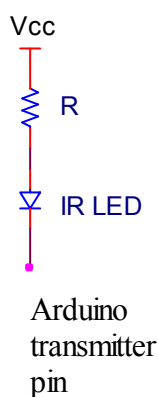
### 3.4 Αισθητήρες Υπερύθρων

Για την διαδικασία εντοπισμού και αποφυγής των εμποδίων χρησιμοποιήθηκαν αισθητήρες υπερύθρων. Οι αισθητήρες υπερύθρων είναι μία φθηνή και σχετικά αξιόπιστη λύση χωρίς να

απαιτούν πολύπλοκα κυκλώματα υποστήριξης. Είναι αρκετά δημοφιλείς στην κατηγορία των ρομποτικών οχημάτων. Τα σημαντικότερα μειονεκτήματά τους είναι ότι επηρεάζονται από τον φωτισμό του χώρου, τα χρώματα και την υφή των εμποδίων, καθώς επίσης και ότι αδυνατούν να μετρήσουν αποστάσεις μεγαλύτερες από 150cm.

Η αντιμετώπιση των εμποδίων γίνεται με τον εξής τρόπο : Όταν το ρομπότ εντοπίσει ένα εμπόδιο, τότε ελέγχει από ποια πλευρά βρίσκεται το εμπόδιο, και στρίβει επιτόπια προς την πλευρά αυτή, μέχρις ότου να ευθυγραμμιστεί με αυτό. Για να πετύχουμε τον παραπάνω τρόπο αντιμετώπισης των εμποδίων θα πρέπει να χρησιμοποιήσουμε 6 αισθητήρες (πομποδέκτες υπερύθρων). Δύο αισθητήρες πρέπει να τοποθετηθούν μπροστά προκειμένου να εντοπιστεί το εμπόδιο, καθώς επίσης και από ποια πλευρά του οχήματος βρίσκεται, ενώ δύο αισθητήρες πρέπει να τοποθετηθούν και σε κάθε πλάι, με σκοπό να πετύχουμε τη διαδικασία της ευθυγράμμισης. Ο τρόπος με τον οποίο πραγματοποιείται η όλη διαδικασία της αποφυγής των εμποδίων θα αναλυθεί στο επόμενο κεφάλαιο.

Για το κύκλωμα του πομπού που κατασκευάστηκε, χρησιμοποιήθηκαν LEDs υπερύθρων (IR LEDs – αναλυτική περιγραφή έχει γίνει στο Κεφάλαιο 2) τα οποία είναι συνδεδεμένα σε σειρά με μία αντίσταση, όπως φαίνεται στο παρακάτω σχήμα.



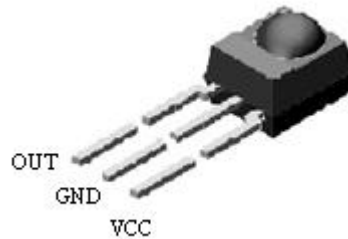
Σχήμα 3.4.1. Κύκλωμα πομπού.

Στο παραπάνω σχήμα, φαίνεται και η έννοια της «βύθισης» με την οποία λειτουργεί το κύκλωμα πομπού. Κάθε LED, όπως και όλες οι διόδοι, άγουν ρεύμα προς μία μόνο κατεύθυνση. Όπως φαίνεται στο σχήμα, η άνοδος του IR LED είναι συνδεδεμένη σε υψηλή τάση, οπότε για να συμπληρώσουμε το ηλεκτρικό κύκλωμα πρέπει να «βυθίσουμε» το ρεύμα στη γη, προκειμένου να ανάψει το LED. Κατά συνέπεια, θέτοντας στο λογικό επίπεδο 0 την έξοδο του μικροελεγκτή, ανάβουμε το IR LED, ενώ στην αντίθετη περίπτωση το IR LED σταματάει να εκπέμπει.

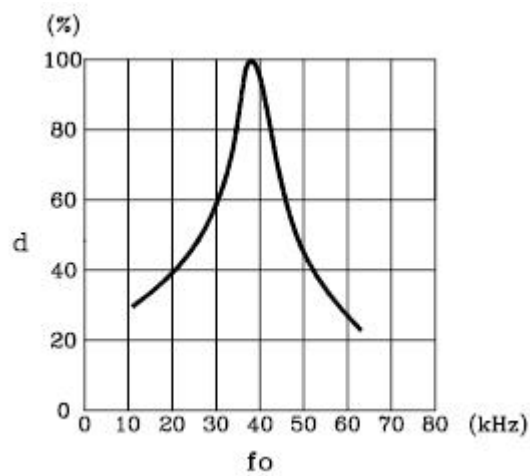
Γενικά, η αντίσταση R δεν είναι απαραίτητη στο παραπάνω κύκλωμα, όμως προτιμότερο είναι να μην παραλειφθεί αφενός για λόγους προστασίας του IR LED και αφετέρου, γιατί η παρουσία της μας δίνει τη δυνατότητα να περιορίσουμε την εμβέλεια του πομπού. Αυξομειώνοντας, λοιπόν, την αντίσταση R, μπορούμε να ρυθμίσουμε την εμβέλεια του IR LED, ενώ αν τη βραχυκυκλώσουμε αυξάνουμε στο μέγιστο την εμβέλεια. Για τις ανάγκες όμως της παρούσας εφαρμογής, χρειαζόμαστε ο πομπός να έχει τέτοια εμβέλεια, ώστε ο δέκτης να μπορεί (μέσω αντανάκλασης της υπέρυθρης ακτινοβολίας) να εντοπίσει αντικείμενα που βρίσκονται σε

σχετικά μικρές αποστάσεις της τάξης των 15 ως 30cm. Μετά από τις σχετικές δοκιμές, καταλήξαμε ότι ικανοποιητικές τιμές για την αντίσταση R αποτελούν τιμές της τάξης των 50 ως 220Ω περίπου, και επιλέξαμε αντίσταση  $R = 100\Omega$ .

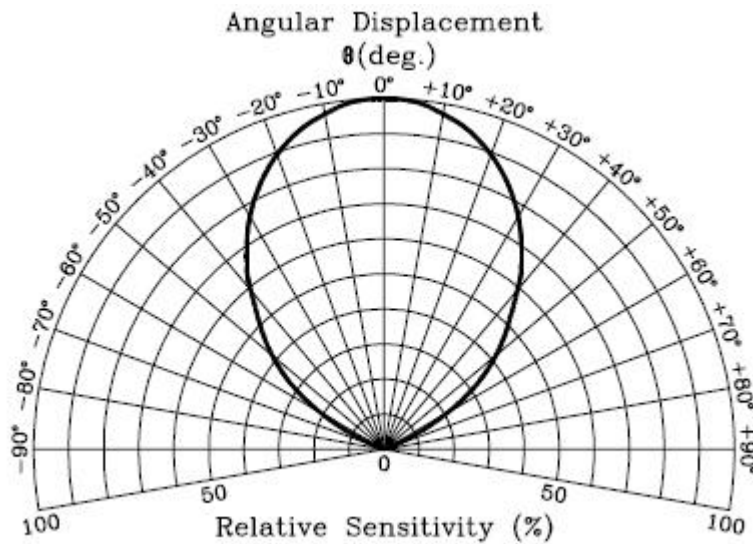
Το κύκλωμα του δέκτη βασίστηκε σε έναν κοινό ολοκληρωμένο δέκτη υπερύθρων, όπως είναι ο TSOP4840. Ο δέκτης υπερύθρων που χρησιμοποιήσαμε έχει μέγιστη ευαισθησία γύρω από την συχνότητα των 38,5KHz, όπως φαίνεται και στο διάγραμμα του σχήματος 3.4.2.



Εικόνα 3.4.1. Ο ολοκληρωμένος δέκτης υπερύθρων TSOP48XX.



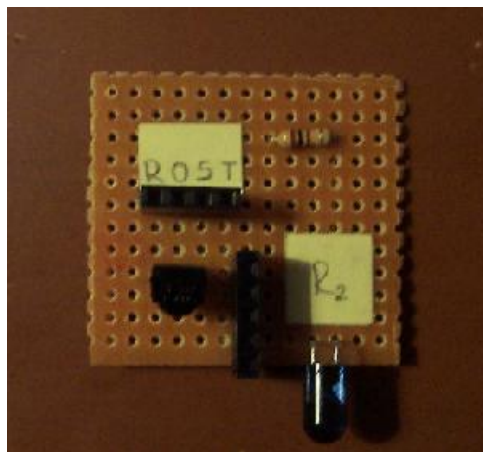
Σχήμα 3.4.2. Διάγραμμα σχετικής απόστασης λήψης συναρτήσει φέρουσας συχνότητας πομπού.



Σχήμα 3.4.3. Διάγραμμα ευαισθησίας.

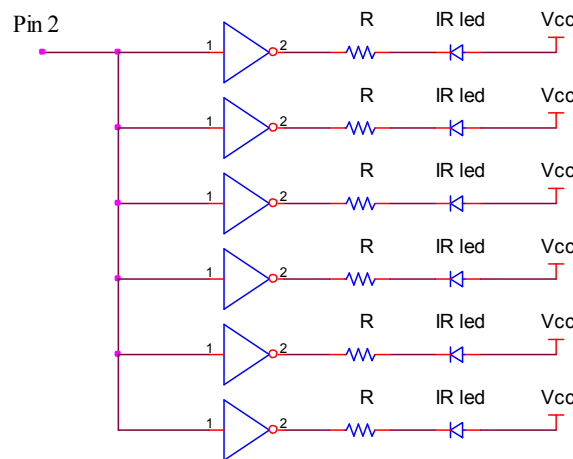
Στο διάγραμμα του σχήματος 3.4.3 παρουσιάζεται το γωνιακό εύρος λήψης του δέκτη υπέρυθρων. Όπως μπορούμε να διαπιστώσουμε από το παραπάνω διάγραμμα, εξαιτίας του μικρού γωνιακού εύρους του δέκτη (καθώς επίσης και εξαιτίας του μικρού ανοίγματος ημίσειας ισχύος του IR LED που είδαμε στο διάγραμμα της σχετικής ακτινικής έντασης του σχήματος 2.5.2.1), πομπός και δέκτης πρέπει να τοποθετηθούν σε μικρή απόσταση μεταξύ τους.

Αυτό που μας ενδιαφέρει είναι ο δέκτης να λαμβάνει σήμα μόνο μετά από την αντανάκλαση της υπέρυθρης ακτινοβολίας από κάποιο αντικείμενο. Όμως, εξαιτίας της μικρής απόστασης στην οποία βρίσκονται πομπός και δέκτης, ο τελευταίος θα λαμβάνει σήμα και απευθείας από το IR LED. Για τον λόγο αυτόν, τοποθετήσαμε ένα πλαστικό αντικείμενο μαύρου χρώματος ανάμεσα σε πομπό και δέκτη (συστοιχία από pins, όπως φαίνεται στην επόμενη εικόνα), προκειμένου να αποτρέπεται η απευθείας λήψη της υπέρυθρης ακτινοβολίας που εκπέμπει ο πομπός.



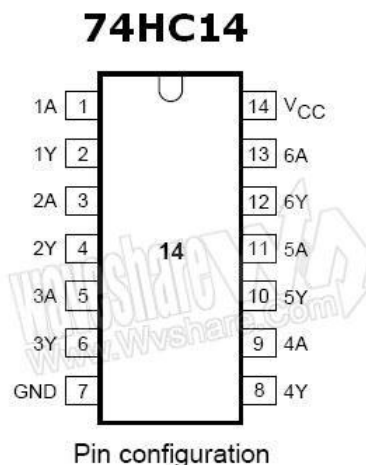
Εικόνα 3.4.2. Ένα από τα έξη κυκλώματα πομποδεκτών υπέρυθρων.

Γενικά, δεν μας ενδιαφέρει το πότε εκπέμπει υπέρυθρη ακτινοβολία το κάθε ένα από τα IR LEDs, αλλά το πότε λαμβάνει σήμα κάποιος από τους δέκτες. Για τον λόγο αυτόν, και εξαιτίας του περιορισμένου αριθμού των ακροδεκτών που έχει ο arduino σε συνδυασμό με τους πολλούς πομποδέκτες υπέρυθρων που χρησιμοποιούμε, θα διαθέσουμε μόνο έναν ακροδέκτη (digital pin 2) για την αποστολή παλμών στα IR LEDs. Κατά συνέπεια, όλα τα IR LEDs θα εκπέμπουν ταυτόχρονα και θα γίνεται έλεγχος, κάθε φορά, μόνο στους δέκτες υπέρυθρων που μας ενδιαφέρουν. Επειδή όμως, κάθε ακροδέκτης του ATmega μπορεί να «βυθίσει» το πολύ 20 mA, θα χρειαστεί να ενισχύσουμε το σήμα, με κάποιον τρόπο, ώστε να φωτοβολούν ικανοποιητικά και τα 6 IR LEDs. Για τον σκοπό αυτό θα χρησιμοποιήσουμε λογικές πύλες NOT, οι εισοδοί των οποίων βραχυκυκλώνονται και συνδέονται στον ακροδέκτη του arduino, ενώ οι έξοδοί τους οδηγούνται στα 6 IR LEDs, όπως φαίνεται στο παρακάτω σχήμα.

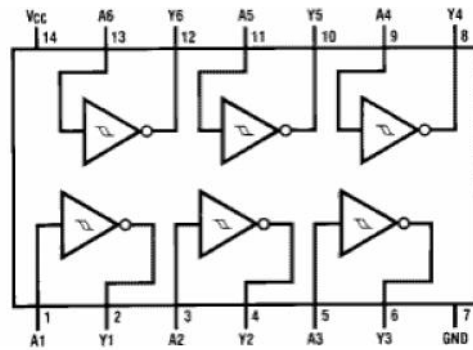


Σχήμα 3.4.4. Όλα τα IR LEDs οδηγούνται σε έναν ακροδέκτη του arduino με την χρήση αντιστροφών

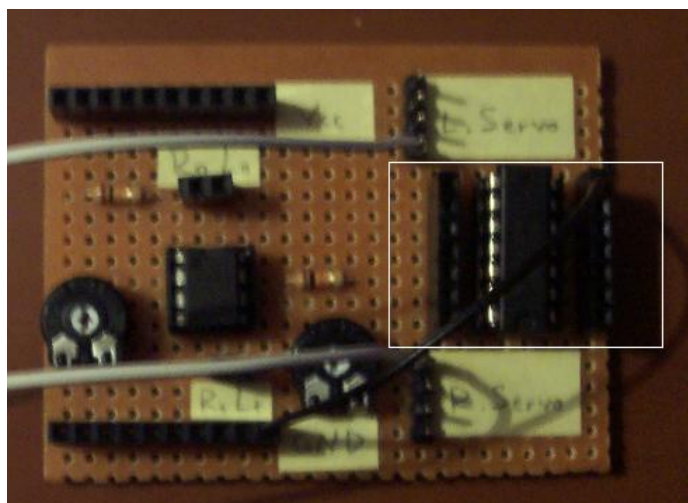
Το ολοκληρωμένο που χρησιμοποιήθηκε είναι το 74HC14, το οποίο αποτελείται από 6 πύλες NOT (όπως φαίνεται και στα παρακάτω σχήματα), κάθε μία από τις οποίες χρησιμοποιείται για ένα led.







Σχήματα 3.4.5. και 3.4.6. Το ολοκληρωμένο 74HC14.



Εικόνα 3.4.3. Το κύκλωμα που κατασκευάσαμε με τη χρήση του 74HC14.

## 3.5 Κωδικοποιητές

### 3.5.1 Εισαγωγή

Για την υλοποίηση του οδομέτρου χρησιμοποιήθηκαν *encoder discs* (κωδικοποιητές). Ο encoder disc είναι μία επιφάνεια που εναλλάσσει το χρώμα της από μαύρο σε λευκό (και το αντίστροφο) και έχει τοποθετηθεί στο πίσω μέρος του κάθε τροχού. Η υλοποίηση βασίζεται στην ανάκλαση υπέρυθρων ακτινών πάνω στην επιφάνεια του encoder disc, με αποτέλεσμα, κατά την περιστροφή του κάθε τροχού, να παίρνουμε στον δέκτη μια παλμοσειρά που είναι ανάλογη με τις συνολικές εναλλαγές του χρώματος.

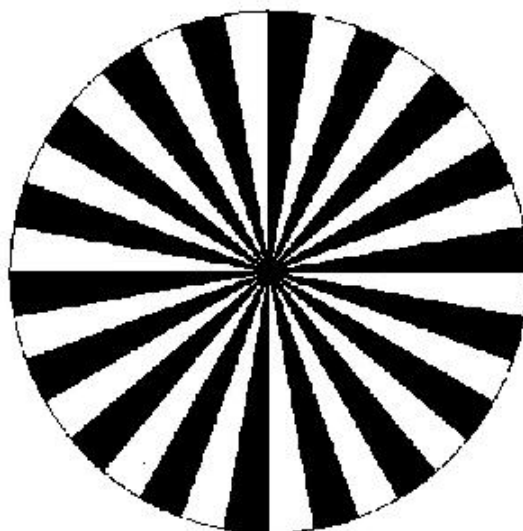
Ο encoder disc που χρησιμοποιήθηκε παρουσιάζεται στην επόμενη εικόνα, προσαρμοσμένος στο πίσω μέρος του τροχού.



*Εικόνα 3.5.1.1. Ο encoder disc προσαρμοσμένος πάνω στη ρόδα.*

### **3.5.2 Encoder discs**

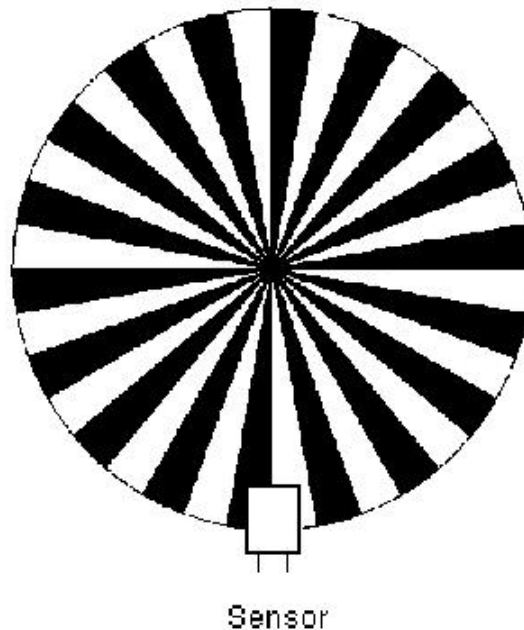
Η ακριβής θέση στην οποία βρισκόμαστε αποτελεί ένα από τα βασικότερα ζητήματα στον τομέα της ρομποτικής. Πολλές φορές χρειαζόμαστε να κινήσουμε ένα ρομπότ για συγκεκριμένη απόσταση ή να το κάνουμε να στρίψει για συγκεκριμένη γωνία. Οι κωδικοποιητές μας παρέχουν έναν τρόπο μέτρησης της θέσης του κινητήρα, ώστε να μπορούμε να πραγματοποιήσουμε έναν συνεχόμενο έλεγχο της κίνησης. Υπάρχουν πάρα πολλές μορφές κωδικοποιητών, για τις ανάγκες της εργασίας όμως, θα ασχοληθούμε με την πιο απλή μορφή κωδικοποιητών, η οποία παρουσιάζεται στην επόμενη εικόνα.



*Εικόνα 3.5.2.1. Ο οπτικός κωδικοποιητής 36 τμημάτων που χρησιμοποιήθηκε.*



Το παραπάνω σχέδιο προσαρμόζεται με τέτοιο τρόπο ώστε να περιστρέφεται παράλληλα με τον κινητήρα, ενώ χρησιμοποιείται ένας οπτικός αισθητήρας τοποθετημένος κοντά στην άκρη του τροχού, με τον τρόπο που φαίνεται στην παρακάτω εικόνα. Ο αισθητήρας δείχνει οποιαδήποτε στιγμή αν βλέπει άσπρο ή μαύρο.



*Εικόνα 3.5.2.2. Απλός οπτικός κωδικοποιητής με έναν αισθητήρα.*

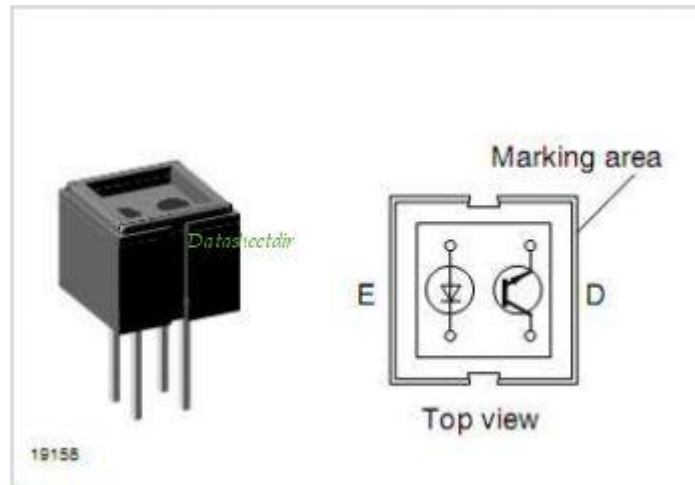
Η επιλογή τροχού μεγάλης διαμέτρου ( $d = 11\text{cm}$ ) έγινε προκειμένου να χωρέσει κωδικοποιητής με τον μεγαλύτερο δυνατό αριθμό τμημάτων. Ο κωδικοποιητής που χρησιμοποιήθηκε (εικόνες 3.5.2.1 και 3.5.2.2) αποτελείται από 36 τμήματα (18 λευκό, 18 μαύρο), δηλαδή 36 εναλλαγές χρώματος. Η επιλογή του αριθμού των τμημάτων έγινε με κριτήριο το κάθε τμήμα να είναι στην άκρη του (στο σημείο δηλαδή που βρίσκεται ο αισθητήρας) τόσο φαρδύ, όσο χρειάζεται ώστε να μπορεί ο αισθητήρας να δει την εναλλαγή του χρώματος κατά την περιστροφή του τροχού. Ο encoder disc που χρησιμοποιήθηκε έχει διάμετρο 10,2cm και άρα, το κάθε τμήμα έχει άνοιγμα 0,89cm. Το διάστημα αυτό είναι αρκετό για να ανιχνεύσει ο αισθητήρας την κάθε εναλλαγή. Με τον αισθητήρα που χρησιμοποιήσαμε όμως, έχουμε την δυνατότητα να μειώσουμε κι άλλο το άνοιγμα του κάθε τμήματος, και κατά συνέπεια, να αυξήσουμε ακόμα περισσότερο τον αριθμό των τμημάτων του κωδικοποιητή.

Η χρήση των encoder discs έγινε με σκοπό την μέτρηση της απόστασης που διανύει το ρομπότ, καθώς επίσης και για τον συντονισμό της περιστροφής των δύο τροχών (διαδικασία που θα περιγράψουμε σε επόμενο κεφάλαιο). Επιδιώκουμε κωδικοποιητή με τον μεγαλύτερο δυνατό αριθμό εναλλαγών για τους εξής λόγους. Πρώτον, για να έχουμε την μεγαλύτερη δυνατή ακρίβεια στη μέτρηση, και δεύτερον για μειώσουμε σε μεγάλο βαθμό την κύλιση με ολίσθηση. Όπως θα δούμε αργότερα, κατά τον συντονισμό των τροχών, οι κινητήρες σταματούν και επανεκκινούν μέσα σε πολύ μικρό χρονικό διάστημα την περιστροφή τους, γεγονός που μειώνει την κίνηση με ολίσθηση. Όσο περισσότερες είναι οι εναλλαγές του χρώματος, τόσο περισσότερο πετυχαίνουμε

κίνηση χωρίς ολίσθηση, ενώ παράλληλα, εκμηδενίζουμε την ταλάντευση του οχήματος που προκύπτει από την διαδικασία του συντονισμού των τροχών.

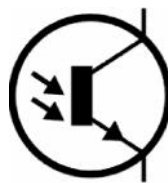
### 3.5.3 Ο αισθητήρας CNY70

Ο αισθητήρας που χρησιμοποιήθηκε για την παραπάνω διαδικασία είναι ο CNY70.



Εικόνα 3.5.3.1. Ο αισθητήρας CNY70.

Πρόκειται για έναν πομποδέκτη υπέρυθρων, όπου πομπός και δέκτης είναι ρυθμισμένοι προς την ίδια κατεύθυνση, προκειμένου ο αισθητήρας να αντιληφθεί την παρουσία ενός εμποδίου χρησιμοποιώντας την ανακλώμενη από το εμπόδιο υπέρυθρη ακτίνα. Το μήκος κύματος λειτουργίας είναι 950nm, ενώ ο δέκτης αποτελείται από ένα φωτοτρανζίστορ. Όπως βλέπουμε και από το κυκλωματικό διάγραμμα του αισθητήρα στην εικόνα 3.5.3.1, το φωτοτρανζίστορ στην συγκεκριμένη περίπτωση, λειτουργεί με την βάση ασύνδετη (βλ. παράγραφο 2.5.3).



Εικόνα 3.5.3.2. Φωτοτρανζίστορ.

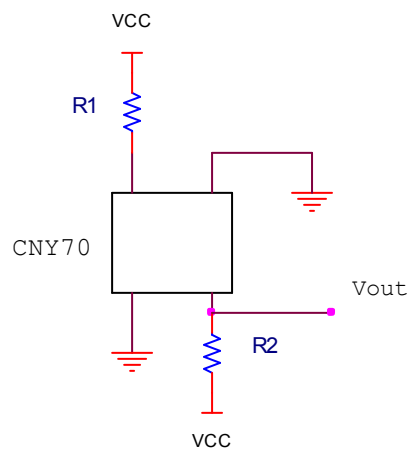
Ο συγκεκριμένος αισθητήρας είναι σχεδιασμένος προκειμένου να αντιλαμβάνεται την παρουσία του εμποδίου σε πολύ μικρή απόσταση, από 2 ως 10mm. Για τον λόγο αυτό, ο CNY70 έχει προσαρμοστεί πάρα πολύ κοντά στον encoder disc, όπως φαίνεται και στην επόμενη φωτογραφία.



Εικόνα 3.5.3.3. Απόσταση του αισθητήρα από τον encoder disc.

Ο αισθητήρας CNY70 χρησιμοποιήθηκε με σκοπό την ανάγνωση του encoder disc. Έχει την ικανότητα να διαχωρίζει ανοιχτές από σκούρες επιφάνειες, καθώς ανάλογα με το χρώμα της επιφάνειας που βρίσκεται μπροστά του, δίνει διαφορετικό σήμα στην έξοδό του. Για επιφάνειες λευκού χρώματος δίνει έξοδο περίπου 0V, ενώ για σκούρες επιφάνειες δίνει έξοδο της τάξης των 5V.

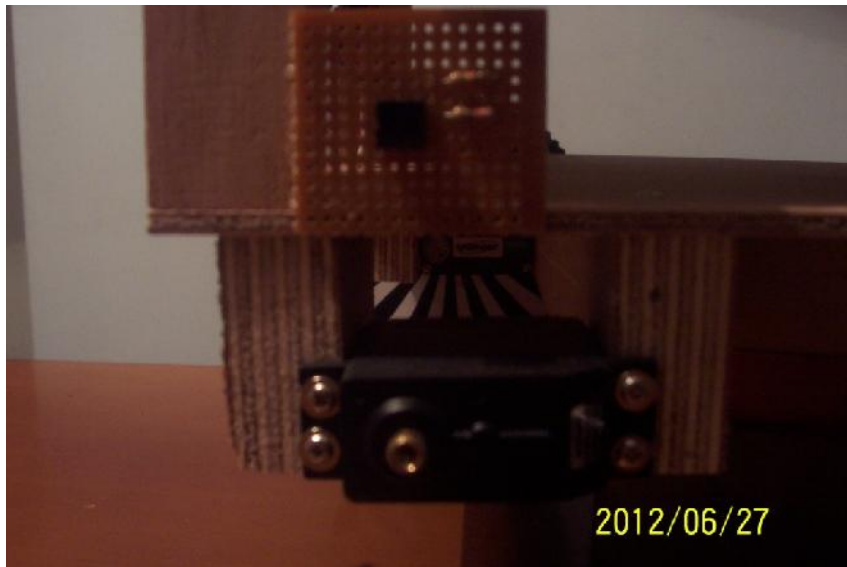
Ο αισθητήρας έχει τοποθετηθεί με τέτοιο τρόπο ώστε πομπός και δέκτης να είναι στην ίδια κατακόρυφη ευθεία, προκειμένου να βρίσκονται, ανά πάσα στιγμή, απέναντι από το ίδιο τμήμα του encoder disc. Το κυκλωματικό διάγραμμα του αισθητήρα CNY70 παρουσιάζεται στο παρακάτω σχήμα.



Σχήμα 3.5.3.1. Το κύκλωμα του αισθητήρα.

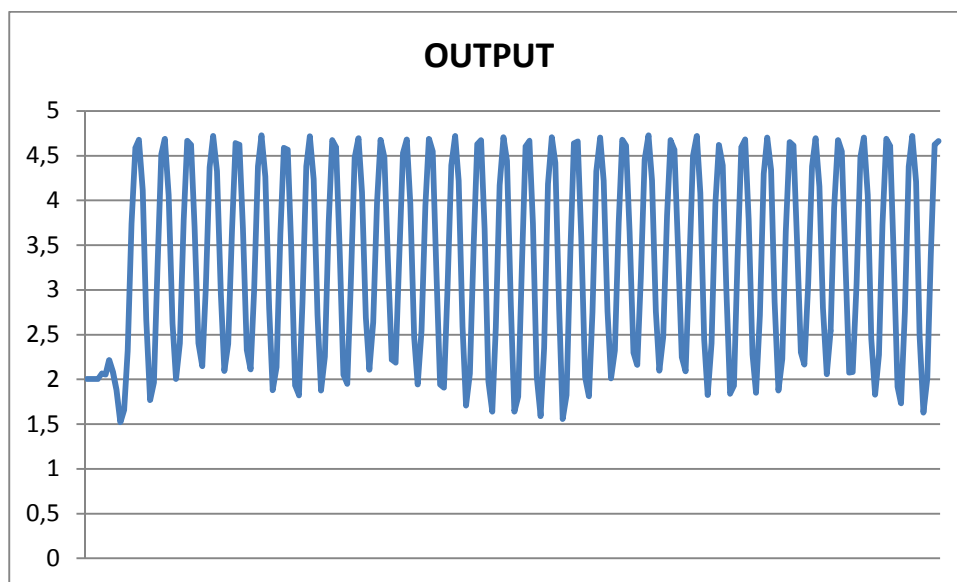
Οι τιμές που επιλέξαμε για τις παραπάνω αντιστάσεις είναι :  $R_1 = 220\Omega$  και  $R_2 = 10K$ .

Στην επόμενη εικόνα παρουσιάζεται το κύκλωμα του αισθητήρα και ο τρόπος με τον οποίο έχει προσαρμοστεί πάνω στην κατασκευή.



Εικόνα 3.5.3.4. Προσαρμογή του αισθητήρα πάνω στην κατασκευή.

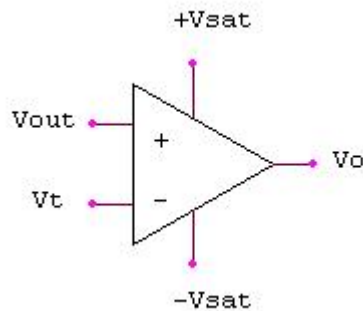
Κατά την περιστροφή του τροχού ο αισθητήρας διαβάζει τις εναλλαγές του χρώματος πάνω στον encoder disc, ενώ στην έξοδό του σχηματίζεται κυματομορφή που έχει την μορφή του παρακάτω διαγράμματος.



Σχήμα 3.5.3.2. Έξοδος του CNY70 στην μέγιστη ταχύτητα περιστροφής του κινητήρα.

Όπως παρατηρούμε και από το παραπάνω διάγραμμα, για άσπρη επιφάνεια η έξοδος του αισθητήρα είναι στα 1,78V (LOW) ενώ για μαύρη επιφάνεια στα 4,63V (HIGH).

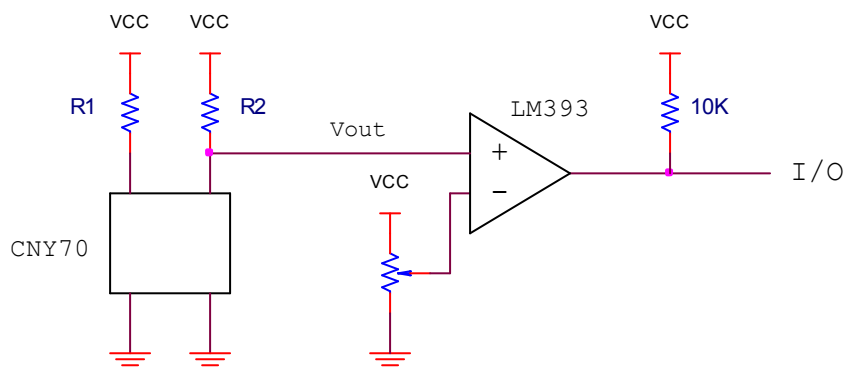
Για να μετατρέψουμε την παραπάνω κυματομορφή σε παλμοσειρά αρκεί να θέσουμε μια τάση κατωφλίου, κάτω από την οποία η έξοδος θα είναι στο λογικό 0 (0V), ενώ αλλιώς θα είναι στο λογικό 1 (5V). Μία επιλογή είναι να χρησιμοποιήσουμε έναν συγκριτή στην έξοδο του αισθητήρα. Ο συγκριτής που χρησιμοποιήθηκε είναι ο *LM393*.



Σχήμα 3.5.3.3. Μη αναστρέφων συγκριτής.

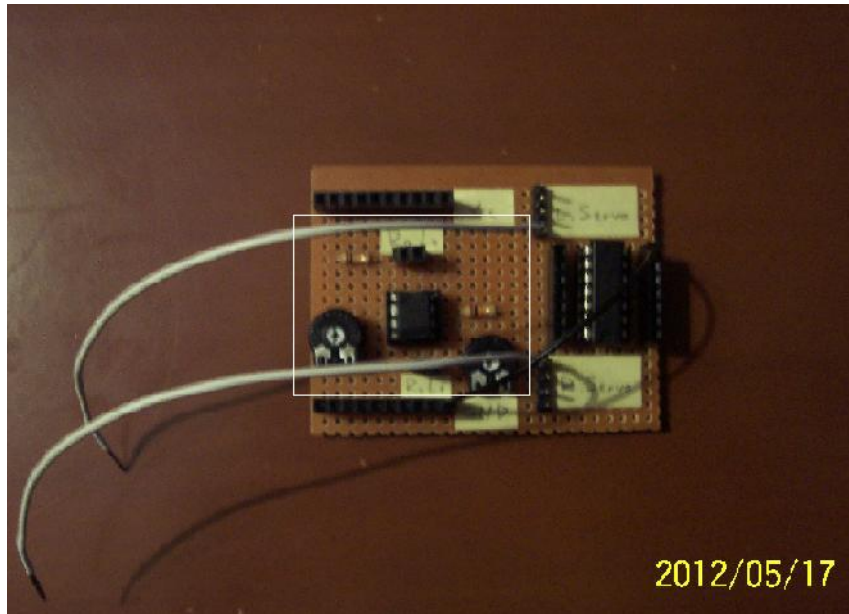
Ως κατώφλι επιλέγουμε την μέση τάση εξόδου του αισθητήρα, δηλαδή τα 3,2V ( $[V_H - V_L]/2$ ). Η τάση αυτή εφαρμόζεται στην αναστρέφουσα είσοδο του συγκριτή, ενώ στην μη αναστρέφουσα είσοδο εφαρμόζεται η τάση εξόδου του CNY70. Για  $V_{out} > V_{th}$  έχουμε τάση εξόδου του συγκριτή  $V_o = +V_{sat}$  (5V), ενώ για  $V_{out} < V_{th}$  έχουμε  $V_o = -V_{sat}$  (0V).

Η τάση κατωφλίου πραγματοποιείται με την κατάλληλη ρύθμιση της μεταβλητής αντίστασης (trimmer) που είναι συνδεδεμένη στην αναστρέφουσα είσοδο του συγκριτή, όπως φαίνεται στο παρακάτω σχήμα.



Σχήμα 3.5.3.4. Κυκλωματικό διάγραμμα με την τοποθέτηση του συγκριτή *LM393* στην έξοδο του αισθητήρα.

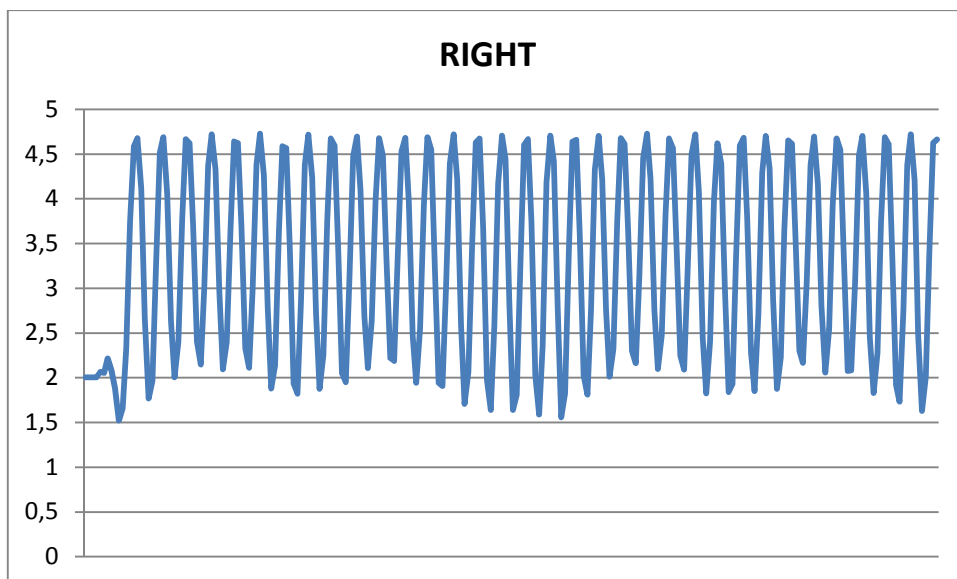
Στην παρακάτω φωτογραφία παρουσιάζεται το κύκλωμα που κατασκευάσαμε. Το ολοκληρωμένο *LM393* αποτελείται από 2 συγκριτές, κάθε ένας από τους οποίους χρησιμοποιήθηκε για τον κωδικοποιητή του κάθε τροχού.

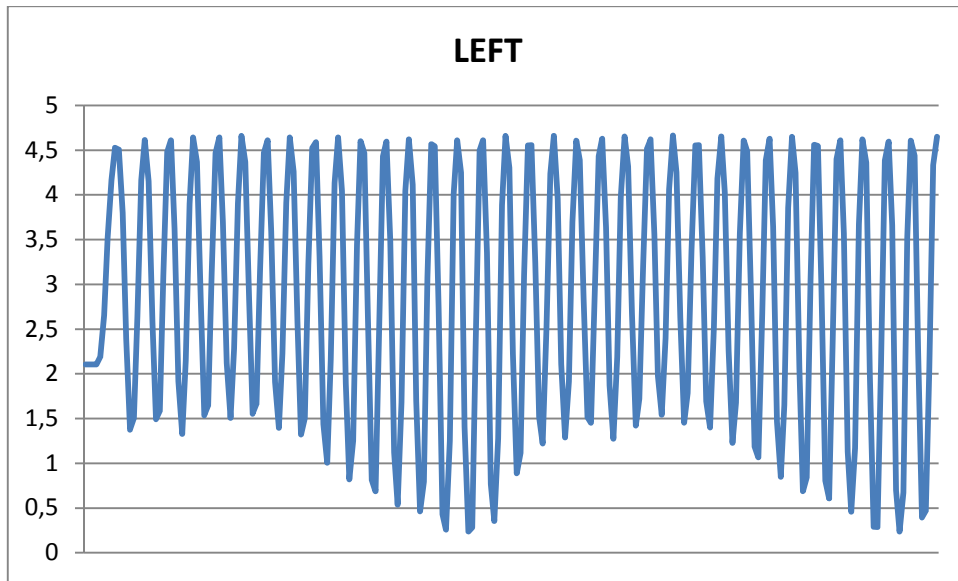


Εικόνα 3.5.3.5.

Πρέπει να σημειώσουμε, ότι το διάγραμμα του σχήματος 3.5.3.2 αναφέρεται στην έξοδο του αισθητήρα για τον κωδικοποιητή του δεξιού τροχού. Η έξοδος του αισθητήρα για τον αριστερό κωδικοποιητή παρουσιάζει διαφορετική συμπεριφορά. Για τον λόγο αυτόν, για τον αισθητήρα του αριστερού τροχού έχει επιλεγεί άλλη τάση κατωφλίου με διαφορετική ρύθμιση του trimmer.

Στα παρακάτω διαγράμματα, παρουσιάζονται συγκεντρωμένες οι κυματομορφές εξόδου για τους δύο αισθητήρες, σε μέγιστη ταχύτητα περιστροφής των κινητήρων. Η τάση εξόδου του αισθητήρα για τον αριστερό κωδικοποιητή κυμαίνεται από 0,75V (για λευκή επιφάνεια) ως 4,55V (για μαύρη επιφάνεια). Κατά συνέπεια, η τάση κατωφλίου που επιλέξαμε για τον αισθητήρα του αριστερού κωδικοποιητή είναι 2,65V ( $[V_H - V_L]/2$ ).





Σχήματα 3.5.3.5. και 3.5.3.6. Έξοδοι των αισθητήρων για τον δεξιό και τον αριστερό κωδικοποιητή.

Όπως θα αναλύσουμε αργότερα, εκτός από την επιλογή του συγκριτή για την διαμόρφωση των κυματομορφών εξόδου των παραπάνω αισθητήρων, ο arduino μας δίνει τη δυνατότητα να επεξεργαστούμε προγραμματιστικά τις εξόδους των αισθητήρων, διαβάζοντας στην ουσία, αναλογικά σήματα.

## 3.6 Κινητήρες

### 3.6.1 Εισαγωγή

Οι πιο απλοί και πιο συνηθισμένοι μηχανισμοί κίνησης στα ρομπότ είναι οι ηλεκτρικοί κινητήρες. Ο ηλεκτρικός κινητήρας είναι μία ηλεκτρομηχανική συσκευή, η οποία χρησιμοποιείται για την μετατροπή της ηλεκτρικής ενέργειας σε κινητική. Η λειτουργία του κινητήρα βασίζεται στη μαγνητική δύναμη μεταξύ ρευματοφόρων αγωγών. Η αλληλεπίδραση μεταξύ των ρευμάτων των αγωγών έχει σαν αποτέλεσμα την εμφάνιση μαγνητικής δύναμης και επομένως και ροπής. Με αυτόν τον τρόπο προκαλείται η περιστροφή του άξονα του κινητήρα.

Η αποδιδόμενη ισχύς  $P$  ενός κινητήρα (το έργο που παράγεται από τον στρόφαλο (άξονα), ανά μονάδα χρόνου) ισούται με το γινόμενο της ροπής  $\tau$  επί της γωνιακής ταχύτητας  $\omega$  του άξονα, δηλαδή :

$$P = \omega \times \tau$$

Συνήθως, η γωνιακή ταχύτητα ενός κινητήρα εκφράζεται σε στροφές  $n$  ανά μονάδα χρόνου. Άρα η αποδιδόμενη ισχύς σε  $HP$  είναι :

$$P = \tau \times n/K$$



όπου  $K = 1008400$ , αν χρησιμοποιούμε για μονάδες μέτρησης  $oz, in, min$ , ή  $K = 7600$ , αν χρησιμοποιούμε  $Kp, m, sec$ , ενώ η αποδιδόμενη ισχύς σε  $kW$  είναι επίσης :

$$P = \tau \times n / K$$

όπου  $K = 9550$ , αν χρησιμοποιούμε ως μονάδες  $N, m$  και  $min$ .

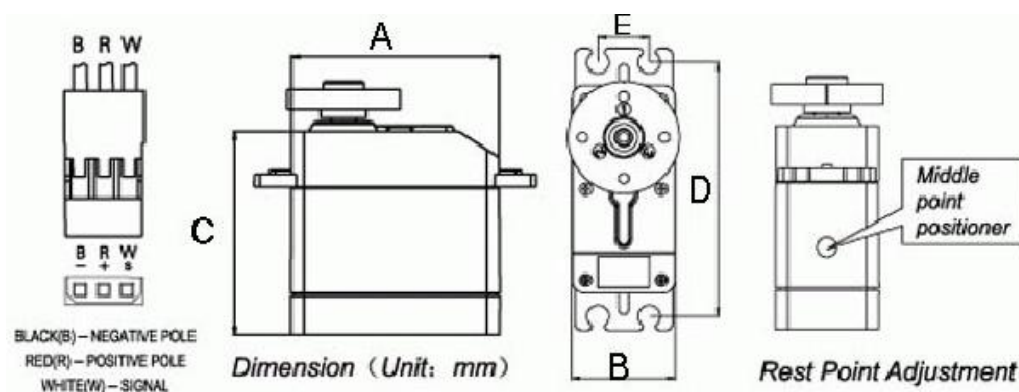
Ο κάθε κινητήρας μπορεί να αποδώσει έως μία δεδομένη μέγιστη ισχύ, η οποία και ορίζεται από τις τεχνικές προδιαγραφές του.

### 3.6.2 Ο servo κινητήρας SM-S4315R

Ο τύπος κινητήρων που συναντάται πιο συχνά σε ρομποτικές εφαρμογές είναι οι *σερβοκινητήρες* (*servo motors*). Οι σερβοκινητήρες συνήθως έχουν περιορισμένο εύρος κίνησης (για παράδειγμα  $0-180^\circ$ ), με αποτέλεσμα να χρησιμοποιούνται κατά κύριο λόγο ως μηχανισμοί κίνησης σε ρομποτικούς βραχίονες ή άλλες εφαρμογές όπου δεν απαιτείται συνεχόμενη περιστροφή. Με κατάλληλη ρύθμιση όμως, ένας servo περιορισμένου εύρους κίνησης μπορεί να μετατραπεί σε κινητήρα συνεχούς περιστροφής, προκειμένου να υπάρχει η δυνατότητα να χρησιμοποιηθεί και ως μηχανισμός κίνησης ενός τροχού. Παρόλα αυτά, στο εμπόριο κυκλοφορούν γενικά και σερβοκινητήρες συνεχούς περιστροφής ( $0-360^\circ$ ), όπως είναι και οι κινητήρες που προμηθευτήκαμε. Η περιστροφή του σερβοκινητήρα (φορά, γωνία) καθορίζεται μέσω ηλεκτρονικού σήματος που δέχεται το κύκλωμα ελέγχου.

Όσον αφορά λοιπόν την παρούσα εφαρμογή, οι κινητήρες που είναι υπεύθυνοι για την κίνηση των τροχών πρέπει να πληρούν, για τις ανάγκες της εργασίας, κάποιες βασικές προϋποθέσεις. Χρειαζόμαστε (όπως αναφέραμε) συνεχούς περιστροφής κινητήρες, οι οποίοι πρέπει να έχουν τάση λειτουργίας της τάξης των  $5V$  και ροπή ικανή να μεταφέρει το βάρος της κατασκευής χωρίς να υπάρχει μεγάλη επιβάρυνση της μπαταρίας.

Οι κινητήρες που χρησιμοποιήθηκαν για την κίνηση του οχήματος είναι servo κινητήρες συνεχούς περιστροφής (*continuous rotation*), ενώ το μοντέλο είναι το *SM-S4315R* της εταιρίας *Spring Model Electronic*. Ο συγκεκριμένος κινητήρας είναι ενδεδειγμένος ως μηχανισμός κίνησης τροχών, σε ρομποτικές εφαρμογές όπου απαιτείται η μεταφορά φορτίου με σχετικά μεγάλο βάρος.



Εικόνα 3.6.1. Μηχανολογικό σχέδιο κινητήρα.





Εικόνα 3.6.2. Ο κινητήρας.

Τα τεχνικά χαρακτηριστικά και οι προδιαγραφές του κινητήρα παρουσιάζονται στον παρακάτω πίνακα.

Products specification								Technical parameters						
Size (mm)					Weight		Wire	4.8V			6V			Rotation angle
A	B	C	D	E	g	oz		Speed	Torque		Speed	Torque		
							cm	rpm	kg-cm	oz-in	rpm	kg-cm	oz-in	
42.0	20.5	39.5	49.0	10.0	60	1.55	30.0	0.21	13	180.85	0.17	15.1	210.07	360°

(Specifications are subjected to change without notice.)

Πίνακας 3.6.1. Πίνακας προδιαγραφών κινητήρα SM-S4315R.

### 3.6.3 Ροπή κινητήρα

Όπως βλέπουμε στον παραπάνω πίνακα, ο συγκεκριμένος κινητήρας μπορεί να αποδώσει, για τάση λειτουργίας 4,8V, μηχανική ροπή ως και 13kg-cm, που (όπως θα περιγράψουμε στη συνέχεια αυτής της παραγράφου) αποτελεί μια ικανοποιητική τιμή για τα δεδομένα της συγκεκριμένης εφαρμογής.

Όπως θα δούμε περιληπτικά και στο επόμενο κεφάλαιο (κατά την οριακή περίπτωση για την οποία ο τροχός δεν ολισθαίνει), η τριβή που δημιουργείται μεταξύ τροχού και δαπέδου κατά την κίνηση του οχήματος δίνεται από τον τύπο :

$$T = \mu * N$$

όπου

$N$  η κάθετη αντίδραση, η οποία στην περίπτωση του οριζόντιου δαπέδου που μας ενδιαφέρει, ισούται με το βάρος, και

$\mu$  ο συντελεστής τριβής του δαπέδου.

Στον παρακάτω πίνακα παρουσιάζουμε ενδεικτικά τις τιμές του συντελεστή τριβής για μερικούς τύπους δαπέδου που μπορεί να μας απασχολήσουν.

Τύπος Δαπέδου	Συντελεστής Τριβής
Άσφαλτος	$0,80 < \mu < 1,00$
Τσιμέντο	$0,80 < \mu < 1,00$
Πλακάκια	$0,75 < \mu < 0,95$
Λίθινες Πλάκες	$0,75 < \mu < 1,00$

Πίνακας 3.6.1. Συντελεστής τριβής διάφορων τύπων δαπέδου.

Οι παραπάνω τιμές αναφέρονται σε τραχύ, ξηρό οδόστρωμα και σε ταχύτητα οχήματος μικρότερη των 50 km/h, συνθήκες δηλαδή, που επικρατούν και στην παρούσα εφαρμογή.

Όπως παρατηρούμε και στον παραπάνω πίνακα, ο συντελεστής τριβής για τις περιπτώσεις δαπέδων που μας ενδιαφέρουν, είναι κοντά στη μονάδα. Άρα, μπορούμε να θεωρήσουμε ότι προσεγγιστικά ισχύει :

$$T \cong N$$

Επίσης, όπως αναφέραμε προηγουμένως, η αντίδραση του εδάφους  $N$  είναι (για την περίπτωση του οριζόντιου επιπέδου) ίση με το βάρος στον κάθε τροχό, και κατά συνέπεια στον άξονα του κάθε κινητήρα. Όπως έχει αναφερθεί στην παράγραφο 3.2, το συνολικό βάρος της κατασκευής (μαζί με τον φορητό υπολογιστή) είναι της τάξης των 4kg. Αν υποθέσουμε ότι το βάρος της κατασκευής είναι ομοιόμορφα κατανεμημένο σε ολόκληρη τη βάση της, καθώς επίσης και ότι τα 3 σημεία στήριξης (οι 3 τροχοί) ισαπέχουν από το κέντρο βάρους του συστήματος (οι 2 παραπάνω υποθέσεις είναι πολύ κοντά στην πραγματικότητα), τότε μπορούμε να θεωρήσουμε ότι η δύναμη που ασκείται από το βάρος της κατασκευής στον κάθε τροχό είναι :

$$B_{wh} = B/3$$

όπου  $B = 4\text{kg}$  το συνολικό βάρος του ρομπότ.

Άρα :

$$T \cong N = B_{wh} = B/3$$

Όπως γνωρίζουμε από την επιστήμη της κινηματικής, η δύναμη της τριβής που δημιουργείται μεταξύ του τροχού και του εδάφους κατά την περιστροφή του κάθε τροχού, είναι και η δύναμη στην οποία οφείλει το όχημα την κίνησή του. Η ροπή της τριβής δίνεται από την παρακάτω σχέση :

$$\tau = r \times T$$

όπου  $r$  η ακτίνα του τροχού, άρα  $r = d/2 = 11/2 = 5,5\text{cm}$  ( $d$  η διάμετρος του τροχού).

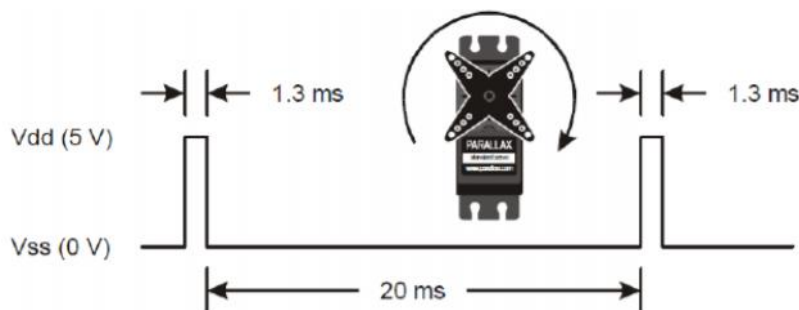
Άρα :

$$\tau = r \times T = r \times \frac{B}{3} = 5,5 \times \frac{4}{3} = \frac{22}{3} \cong 7,3 \text{kg} \cdot \text{cm}$$

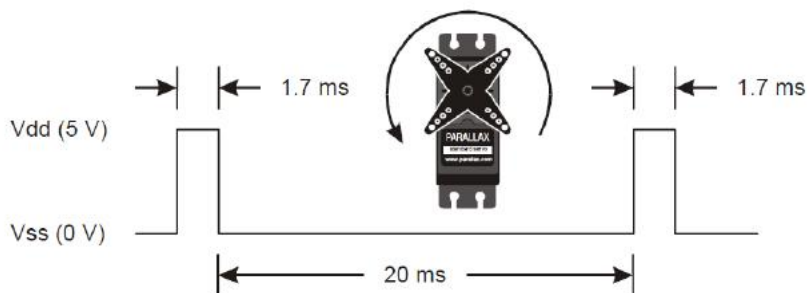
Κατά συνέπεια, με μέγιστη αποδιδόμενη μηχανική ροπή 13kg·cm, οι κινητήρες που επιλέχθηκαν έχουν τη δυνατότητα να μετακινήσουν το βάρος του συστήματος.

### 3.6.4 Οδήγηση κινητήρων

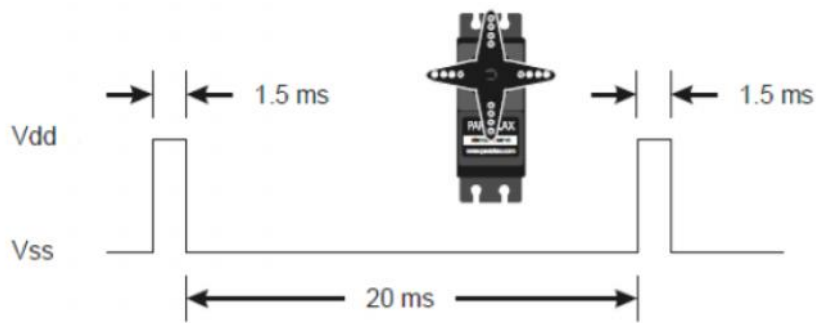
Για την οδήγηση των 2 servo κινητήρων του οχήματος, χρησιμοποιήθηκε ο timer1 του arduino (pins 9, 10) για την παραγωγή παλμών PWM (διαμόρφωση εύρους παλμού). Οι PWM είναι τετραγωνικοί παλμοί των οποίων η συχνότητα είναι σταθερή, το duty cycle όμως, μπορεί να κυμαίνεται από 0 ως 100%. Για την οδήγηση των κινητήρων του ρομπότ απαιτήθηκε παραγωγή παλμών PWM με σταθερή περίοδο 20 ms, και  $T_H$  (χρόνος κατά τον οποίο ο παλμός είναι HIGH) 1,5 ms για την θέση ισορροπίας. Η αύξηση του  $T_H$  του παλμού οδηγεί στην αριστερόστροφη κίνηση των κινητήρων από την θέση ισορροπίας ενώ η μείωση σε δεξιόστροφη περιστροφή, όπως φαίνεται χαρακτηριστικά στις περιπτώσεις των παρακάτω σχημάτων. Για  $T_H = 1$  ms έχουμε μέγιστη ταχύτητα κατά την δεξιόστροφη κίνηση, ενώ για  $T_H = 2$  ms μέγιστη ταχύτητα κατά την αριστερόστροφη κίνηση. Η μεταβολή της συνολικής περιόδου του PWM και του  $T_H$  ρυθμίζεται με βάση δυο καταχωρητές του arduino.



Σχήμα 3.6.1. Δεξιόστροφη κίνηση.



Σχήμα 3.6.2. Αριστερόστροφη κίνηση.



Σχήμα 3.6.3. Θέση ισορροπίας.

Για τον έλεγχο, λοιπόν, της ταχύτητας των κινητήρων πρέπει να δημιουργήσουμε προγραμματιστικά, τους παλμούς που περιγράψαμε προηγουμένως. Παρακάτω παρουσιάζουμε τον κώδικα της υπορουτίνας pulseout με την οποία δημιουργούμε έναν παλμό περιόδου 20ms. Η ρουτίνα αυτή δέχεται ως παραμέτρους το pin στο οποίο είναι συνδεδεμένος ο κινητήρας, καθώς επίσης και το  $T_H$  του παλμού σε  $\mu s$ , το οποίο και ορίζει την ταχύτητα και τη φορά περιστροφής του κινητήρα.

```
void pulseOut(int pin, int us)
{
  digitalWrite(pin, HIGH);
  delayMicroseconds(us);
  digitalWrite(pin, LOW);
  delayMicroseconds(20000 - us);
}
```

Όπως αναφέραμε, η παράμετρος us είναι το  $T_H$  του παλμού σε  $\mu s$ , άρα για  $us = 1500$  έχουμε ακινητοποίηση των κινητήρων, ενώ για τιμές 1000 και 2000 έχουμε μέγιστη ταχύτητα περιστροφής κατά ωρολογιακή και αντιωρολογιακή φορά αντίστοιχα.

Παρόλα αυτά, ο arduino έχει ειδική βιβλιοθήκη για τους servo κινητήρες, την <Servo.h>, και κατά συνέπεια μας δίνεται η δυνατότητα, να ελέγχουμε τη λειτουργία των κινητήρων με έτοιμες εντολές, όπως και θα δούμε αναλυτικά στα επόμενα κεφάλαια. Χαρακτηριστικά, με μία εντολή :

```
servo.write(angle);
```

μπορούμε να ρυθμίσουμε την ταχύτητα και τη φορά περιστροφής του κινητήρα, δίνοντας την κατάλληλη στο όρισμα angle, όπου angle η γωνία του άξονα σε μοίρες. Κατά συνέπεια, για γωνία κοντά στις  $90^\circ$  δεν έχουμε περιστροφή, ενώ για τιμές 0 και 180 έχουμε μέγιστη ταχύτητα προς τη μία ή προς την άλλη κατεύθυνση.

## 3.7 Τροφοδοσία

### 3.7.1 Μπαταρίες

Για την τροφοδοσία της κατασκευής χρησιμοποιήθηκαν επαναφορτιζόμενες μπαταρίες Νικελίου – Υδριδίου Μετάλλου (Ni-MH). Οι συγκεκριμένες μπαταρίες αποτελούν τις πιο δημοφιλείς επαναφορτιζόμενες μπαταρίες, και έχουν αντικαταστήσει τις μπαταρίες Νικελίου – Καδμίου (Ni-Cd) καθώς έχουν μεγαλύτερη πυκνότητα ενέργειας. Παράγονται σε τυπικά μεγέθη AAA, AA κτλ. Θεωρούνται εν γένει καλοί αντικαταστάτες των απλών αλκαλικών μπαταριών σε πολλές περιπτώσεις. Οι κυψέλες των μπαταριών αυτών είναι της τάξεως των 1.2Volt. Τέλος, με την τεχνολογία νικελίου-υδριδίου μετάλλου οι μπαταρίες μπορούν να επαναφορτιστούν πολλές φορές, και το γεγονός αυτό καθιστά σχεδόν περιττή την αντικατάστασή τους.

Πλεονεκτήματα: Υψηλής χωρητικότητας σε τυπικά μεγέθη - φορτίζουν εύκολα

Μειονεκτήματα: Ακριβότερες από τις Ni-Cd, αποφορτίζονται πολύ γρήγορα σε κατάσταση αδράνειας και δεν μπορούν να χρησιμοποιηθούν σε συστήματα με μεγάλες απαιτήσεις ενέργειας.

Πυκνότητα Ενέργειας: 100 Wh/kg

Οι μπαταρίες που χρησιμοποιήθηκαν είναι μεγέθους AA και χωρητικότητας 2500mAh και φαίνονται στην επόμενη εικόνα, ενώ οι τεχνικές προδιαγραφές τους παρουσιάζονται παρακάτω.



Εικόνα 3.7.1.1. AA 2A Rechargeable NIMH NI-MH Battery Cell 2500mAh 1.2V

## Τεχνικές Προδιαγραφές.

- Τάση μπαταρίας : 1,2V
- Χωρητικότητα : 2500mAh
- Τύπος μπαταριών : AA νικελίου-υδριδίου μετάλλου
- Χημική σύνθεση : Νικελίου-υδριδίου μετάλλου
- Διάρκεια ζωής εκτός λειτουργίας : 2 έτη
- Επαναφορτίζεται πλήρως έως και 1.000 φορές, χωρίς να απαιτείται πλήρης αποφόρτιση πριν την επαναφόρτιση.

Όπως έχει ήδη αναφερθεί, οι μπαταρίες τύπου Ni-MH φορτίζονται πολύ εύκολα. Παρακάτω παρουσιάζεται ένα φορτιστής για συγκεκριμένου τύπου μπαταρίες.



Εικόνα 3.7.1.2. Φορτιστής μπαταριών AA/AAA Ni-MH και Ni-Cd.

Η επιλογή της τροφοδοσίας έγινε με βάση τα datasheet του arduino, όπου απαιτείται τάση εισόδου από 7V ως 20V DC (ο σταθεροποιητής του τροφοδοτικού που κατασκευάσαμε και είδαμε προηγουμένως, θα υποβιβάζει, έτσι κι αλλιώς, την τάση στα 5V). Προτεινόμενη τροφοδοσία είναι από 9V ως 12V. Για λόγους οικονομίας, όμως, επιλέγουμε την χαμηλότερη δυνατή τροφοδοσία.

Επιδιώκουμε, λοιπόν, τροφοδοσία της τάξης των 9V, και για τον λόγο αυτόν χρησιμοποιήθηκαν 6 μπαταρίες τύπου Ni-MH. Στο σημείο αυτό, πρέπει να επισημανθεί ότι ενώ οι μπαταρίες που χρησιμοποιούμε έχουν τάση 1,2V, όταν είναι πλήρως φορτισμένες δίνουν 1,4V η κάθε μία. Κατά συνέπεια, οι 6 μπαταρίες δίνουν τάση της τάξης των 8,4V, που αποτελεί ικανοποιητική τροφοδοσία.

Η σύνδεση των μπαταριών μεταξύ τους, έγινε χρησιμοποιώντας 2 μπαταριοθήκες (των 3 μπαταριών η καθεμία) όπως φαίνεται στην παρακάτω εικόνα. Στην κάθε μπαταριοθήκη



προσαρμόστηκε ένα clip για εννιάβολτη μπαταρία. Επίσης, για την έναρξη και τον τερματισμό της τροφοδοσίας, χρησιμοποιήθηκε ένας διακόπτης 2 θέσεων (ON-OFF).



Εικόνα 3.7.1.3. Κύκλωμα μπαταρίας.

Προκειμένου να μην επιβαρύνουμε τις μπαταρίες με τον μεγάλο αριθμό των δοκιμών που πραγματοποιήθηκαν, κατά τη διάρκεια της εργασίας χρησιμοποιήθηκε για την τροφοδοσία του ρομπότ, ένα τροφοδοτικό ηλεκτρονικού υπολογιστή το οποίο και παρέχει στην κατασκευή σταθερή τάση της τάξης των 12V.



Εικόνα 3.7.1.4. Τροφοδοτικό.

### 3.7.2 Κατανάλωση Ισχύος

Στην παρούσα παράγραφο θα παρουσιάσουμε την συνολική κατανάλωση ισχύος του συστήματος κάνοντας έναν σύντομο υπολογισμό των επιμέρους καταναλώσεων της εκάστοτε διάταξης.

Όπως περιγράψαμε στην παράγραφο 3.4, το σύστημα αισθητήρων για την αποφυγή εμποδίων αποτελείται από 6 πομποδέκτες υπέρυθρης ακτινοβολίας. Ο δέκτης υπέρυθρων που χρησιμοποιήθηκε είναι ο TSOP4840, ο οποίος (σύμφωνα με το αντίστοιχο datasheet) καταναλώνει 5mA, και κατά συνέπεια η συνολική κατανάλωση των 6 δεκτών είναι στα 30mA. Από την άλλη πλευρά, όπως αναλύσαμε, όλα τα IR LEDs (που έχουν χρησιμοποιηθεί ως πομποί υπέρυθρης ακτινοβολίας) είναι συνδεδεμένα με τον τρόπο που δείξαμε μέσω των σχημάτων 3.4.1 και 3.4.4, σε έναν ακροδέκτη του arduino, και για τον λόγο αυτόν μπορούν να καταναλώσουν συνολικά μέχρι και 20mA (όσα mA μπορεί και να βυθίσει κάθε ακροδέκτης I/O του arduino). Τα 6 IR LEDs, λοιπόν, για την καλύτερη δυνατή φωτοβολία, καταναλώνουν 20mA. Τέλος, σε ότι αφορά τις αισθητήριες διατάξεις, σύμφωνα με το datasheet του CNY70 (που χρησιμοποιήθηκε για το διάβασμα των κωδικοποιητών), ο συγκεκριμένος αισθητήρας καταναλώνει ένα ρεύμα της τάξης των 20mA, και άρα οι 2 αισθητήρες CNY70 έχουν μία συνολική κατανάλωση που φτάνει τα 40mA.

Όσον αφορά τα ολοκληρωμένα που έχουν χρησιμοποιηθεί στο σύνολο της κατασκευής, ο μικροελεγκτής ATmega168 καταναλώνει περίπου 30mA, το ολοκληρωμένο 74HC14 50mA, ο μετατροπέας MAX232 10mA, ενώ ο συγκριτής LM393 καταναλώνει μία αμελητέα ποσότητα της τάξης του 1mA. Η κατανάλωση του τελευταίου ολοκληρωμένου δεν θα συμπεριληφθεί στον υπολογισμό της συνολικής κατανάλωσης του συστήματος καθώς αποτελεί ασήμαντο μέγεθος σε σύγκριση με τις παραπάνω τιμές. Όλες οι παραπάνω τιμές έχουν ληφθεί από τα datasheets των εκάστοτε ολοκληρωμένων.

Οι κινητήρες του οχήματος καταναλώνουν ένα ρεύμα το οποίο εξαρτάται από το συνολικό βάρος της κατασκευής. Όσο μεγαλύτερο είναι το βάρος του ρομπότ και κατ'επέκταση όσο μεγαλύτερη είναι η ροπή του κάθε σετνο, τόσο μεγαλύτερο είναι και το ρεύμα που «τραβάει», με αποτέλεσμα την περαιτέρω επιβάρυνση της μπαταρίας. Όπως αναλύσαμε στην παράγραφο 3.6, για συνολικό βάρος της κατασκευής στο επίπεδο των 4kg, η ροπή στον άξονα του κάθε κινητήρα είναι της τάξης των 7,3kg·cm. Με την συγκεκριμένη ροπή και κάτω από κανονικές συνθήκες λειτουργίας του ρομπότ, μετρήσαμε μία κατανάλωση της τάξης των 350mA για τον κάθε ένα από τους 2 κινητήρες. Η παραπάνω τιμή αποτελεί μία μέση κατανάλωση για τον κάθε κινητήρα κατά την διάρκεια της κίνησης του ρομπότ. Είναι χαρακτηριστικό πάντως, ότι κατά την εκκίνηση (κάθε φορά) του οχήματος, η συνολική κατανάλωση του συστήματος που μετράμε μέσω του πολυμέτρου, εκτοξεύεται, προς στιγμήν, στο επίπεδο του 1.4A και εν συνεχεία, πέφτει στα επίπεδα των 750-850mA. Το γεγονός αυτό οφείλεται στο ότι κατά την εκκίνηση οι κινητήρες προσπαθούν να «νικήσουν» την αδράνεια του οχήματος, με αποτέλεσμα να καταναλώσουν στιγμιαία πολύ μεγαλύτερη ποσότητα ρεύματος.



Τέλος, πρέπει να αναφέρουμε και το κόκκινο led που χρησιμοποιείται ως ένδειξη για την τροφοδοσία του συστήματος, το οποίο (όπως και κάθε απλό led) καταναλώνει ρεύμα της τάξης των 10mA.

Μετά τις παραπάνω επισημάνσεις υπολογίζουμε την συνολική κατανάλωση του συστήματος.

TSOP4840	$6 \times 5\text{mA} = 30\text{mA}$
IR LED	20mA
CNY70	$2 \times 20\text{mA} = 40\text{mA}$
ATMEGA168	30mA
74HC14	50mA
MAX232	10mA
RED LED	10mA
SM-S4315R SERVO	$2 \times 350\text{mA} = 700\text{mA}$
ΣΥΝΟΛΟ	890mA

Όπως περιγράψαμε στην προηγούμενη παράγραφο, οι μπαταρίες που χρησιμοποιήθηκαν έχουν χωρητικότητα 2500mAh. Με συνολική κατανάλωση του συστήματος της τάξης των 900mA περίπου, η μέγιστη διάρκεια λειτουργίας του ρομπότ θα είναι :

$$t_{max} = \frac{C_{BAT}}{I_{total}} = \frac{2500\text{mAh}}{900\text{mA}} \cong 2,7\text{h}$$

δηλαδή 2 ώρες και 40 λεπτά περίπου.

Αν χρησιμοποιηθούν μπαταρίες χωρητικότητας 2700mAh (οι οποίες και είναι οι πιο συνηθισμένες στο εμπόριο), τότε η μέγιστη διάρκεια λειτουργίας θα είναι :

$$t_{max} = \frac{C_{BAT}}{I_{total}} = \frac{2700\text{mAh}}{900\text{mA}} = 3\text{h}$$

Οι 3 ώρες λειτουργίας στην τελευταία περίπτωση, αποτελούν μία πολύ ικανοποιητική διάρκεια για την συγκεκριμένη εφαρμογή, καθώς το ρομπότ δεν έχει κατασκευαστεί με σκοπό να λειτουργεί επί πολλές ώρες, αλλά για όσο χρόνο χρειάζεται προκειμένου να σαρώσει τον χώρο, λαμβάνοντας παράλληλα ακουστικές μετρήσεις. Μετά από το παραπάνω χρονικό διάστημα, οι μπαταρίες αποσυνδέονται από το σύστημα και επαναφορτίζονται (με έναν φορτιστή σαν αυτόν της εικόνας 3.7.1.2 – χρόνος φόρτισης 7-10h περίπου) και ξαναχρησιμοποιούνται για την επαναλειτουργία του συστήματος.

## ΚΕΦΑΛΑΙΟ 4

### Η ΛΕΙΤΟΥΡΓΙΑ ΤΟΥ ΡΟΜΠΟΤ

#### 4.1 Εισαγωγή

Το σημαντικότερο ζήτημα της παρούσας διπλωματικής εργασίας είναι η πλοήγηση του κινούμενου ρομπότ που κατασκευάστηκε, και για την ακρίβεια, η αυτόνομη μετακίνησή του μεταξύ 2 δοθέντων σημείων του χώρου. Όπως έχει αναφερθεί, το παρόν ρομποτικό σύστημα ακολουθεί την μέθοδο διαφορικής οδήγησης, της οποίας η κινηματική έχει αναλυθεί στο σύνολο της παραγράφου 2.3. Η βασική λεπτομέρεια που πρέπει να αναφερθεί στο σημείο αυτό, είναι ότι ο τρόπος περιστροφής που έχει επιλεγεί για το συγκεκριμένο ρομπότ διαφορικής οδήγησης είναι η επιτόπια περιστροφή. Για να επιτευχθεί αυτός ο τρόπος, οι 2 κινητήριои τροχοί περιστρέφονται με την ίδια ταχύτητα αλλά αντίθετη φορά (βλ. περίπτωση 2 – παράγραφος 2.3.2). Στην περίπτωση αυτήν λοιπόν, το όχημα περιστρέφεται ως προς το μέσο του άξονα διαφορικής οδήγησης, το *ICC* δηλαδή, βρίσκεται στο μέσο της απόστασης μεταξύ των 2 τροχών.

Κατά τη διάρκεια της μετακίνησης του ρομπότ, πρωτεύον ζήτημα αποτελεί η συνεχόμενη εκτίμηση της θέσης του στον χώρο (θέση – προσανατολισμός), η οποία υλοποιείται μέσω τεχνικών οδομετρίας. Η πλοήγηση του ρομποτικού συστήματος επιτυγχάνεται με την συνεχόμενη μεταφορά της αρχής του συστήματος συντεταγμένων, στα σημεία αλλαγής κατεύθυνσης του οχήματος (το όχημα αλλάζει συνήθως κατεύθυνση, λόγω της παρουσίας εμποδίων, εν μέσω της προβλεπόμενης διαδρομής).

Στις επόμενες παραγράφους, θα γίνει μία εκτεταμένη ανάλυση των επιμέρους εργασιών που εκτελούνται, καθώς επίσης και της συνολικής λειτουργίας του παρόντος ρομποτικού συστήματος.

#### 4.2 Αποφυγή Εμποδίων

##### 4.2.1 Εισαγωγή

Όπως έχουμε ήδη αναφέρει, η διαδικασία αποφυγής εμποδίων θα πραγματοποιηθεί με τη χρήση αισθητήρων εκπομπής και λήψης υπέρυθρης ακτινοβολίας, η τεχνολογία των οποίων έχει παρουσιαστεί στην παράγραφο 2.5, η περιγραφή των κυκλωματικών τους διατάξεων δε, έχει γίνει στην παράγραφο 3.4. Οι 2 μπροστινοί πομποδέκτες υπέρυθρων R και L, έχουν ως αρμοδιότητα τον εντοπισμό του εμποδίου και της πλευράς στην οποία βρίσκεται, ενώ οι πλάγιοι αισθητήρες R1, R2, L1, L2, χρησιμοποιούνται με σκοπό την υλοποίηση της ευθυγράμμισης με το αντικείμενο το οποίο κάθε φορά συναντάται.

Όταν εντοπιστεί ένα εμπόδιο και η πλευρά από την οποία βρίσκεται, τότε το ρομπότ σταματάει, και στρίβει επιτόπια προς την πλευρά αυτήν. Το ρομπότ περιστρέφεται, μέχρις ότου να παραλληλιστεί με το αντικείμενο αυτό, δηλαδή μέχρι οι 2 πλάγιοι αισθητήρες να δουν το αντικείμενο στην ίδια απόσταση (στην ίδια ζώνη, όπως θα δούμε στην συνέχεια). Το όχημα σταματάει να περιστρέφεται και συνεχίζει να πορεύεται παράλληλα με το εμπόδιο που συνάντησε. Όταν οι 2 πλάγιοι αισθητήρες παύουν να βλέπουν το εμπόδιο, το όχημα σταματάει, αφού πρώτα έχει συνεχίσει στην ίδια πορεία για ένα μικρό προκαθορισμένο διάστημα. Τέλος, το ρομπότ περιστρέφεται για όσο απαιτείται, έτσι ώστε να κατευθυνθεί εκ νέου προς το σημείο προορισμού. Αν συναντήσει την άλλη πλευρά του εμποδίου, ή κάποιο άλλο αντικείμενο στην πορεία, τότε ακολουθείται η ίδια διαδικασία. Με τον τρόπο αυτόν, πραγματοποιείται η διαδικασία αποφυγής των αντικειμένων, τα οποία εμπεριέχονται στον χώρο εργασίας του συστήματος. Ο συγκεκριμένος τρόπος αποφυγής εμποδίων μπορεί να πραγματοποιηθεί, αν θεωρήσουμε ότι τα αντικείμενα που βρίσκονται στον χώρο πληρούν κάποιες προϋποθέσεις ως προς τη φύση τους (σχήμα, μέγεθος, κλίση, χρώμα, υφή, κτλ), όπως και θα αναλύσουμε στην παράγραφο 4.5.

## 4.2.2 Εντοπισμός εμποδίου

Για τον εντοπισμό ενός αντικειμένου που βρίσκεται στην πορεία του οχήματος, στέλνουμε συνεχείς παλμούς από τους πομπούς υπερύθρων (IR LEDs) των αισθητήρων R και L που βρίσκονται τοποθετημένοι στην μπροστινή πλευρά της κατασκευής, και αναμένουμε την ανίχνευση σήματος από τους αντίστοιχους δέκτες.

Γνωρίζουμε ότι, σε γενικές γραμμές, ο δέκτης υπερύθρων 40KHz που χρησιμοποιήθηκε, έχει μέγιστη ανταπόκριση σε συχνότητα της τάξης των 38,5KHz. Κατά συνέπεια, πρέπει να υλοποιηθεί η εκπομπή ενός σήματος υπέρυθρης ακτινοβολίας, στην συχνότητα των 38,5KHz, με duty cycle 50%. Για να κατασκευάσουμε λοιπόν έναν παλμό με τις παραπάνω προδιαγραφές, πρέπει να γνωρίζουμε, για την συγκεκριμένη συχνότητα, τα χρονικά διαστήματα για τα οποία θα θέσουμε σε HIGH και LOW αντίστοιχα, τον ακροδέκτη του arduino που συνδέεται με το IR LED.

Η περίοδος του παλμού αυτού θα είναι :

$$T = 1/f = 1/38,5KHz = 26\mu sec$$

Άρα  $T_H = T_L = 13\mu sec$ .

Γνωρίζουμε όμως ότι μία εντολή `digitalWrite` του arduino απαιτεί περίπου 3μsec για να ολοκληρωθεί η εκτέλεσή της.

Κατά συνέπεια, για τη δημιουργία του συγκεκριμένου παλμού χρησιμοποιήθηκε ο παρακάτω κώδικας.

```
//38.5 kHz is about 13 microseconds high and 13 microseconds low
```

```
digitalWrite(transmitter_pin, HIGH); //this takes about 3 μsec to happen
delayMicroseconds(10); //hang out for 10 μsec
```

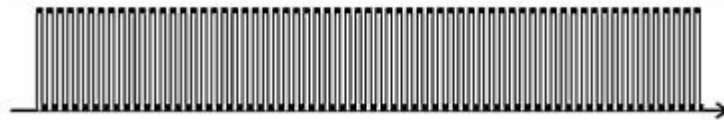
```
digitalWrite(transmitter_pin, LOW); //this also takes about 3 µsec
delayMicroseconds(10); //hang out for 10 µsec
```

Για την ανίχνευση του σήματος από τον δέκτη, πρέπει να στείλουμε επαναλαμβανόμενους παλμούς της παραπάνω μορφής, για ένα ορισμένο χρονικό διάστημα, της τάξης του 1msec. Για τον λόγο αυτόν, χρησιμοποιούμε ένα loop, όπως φαίνεται και στον παρακάτω κώδικα, μέσα από το οποίο, αποστέλλεται επαναλαμβανόμενος, ο παλμός συχνότητας 38,5KHz.

```
void pulse_sending()
{
  int i;

  for (i = 0; i < 1000; i += 1)
  {
    digitalWrite(transmitter_pin, LOW);
    delayMicroseconds(10);
    digitalWrite(transmitter_pin, HIGH);
    delayMicroseconds(10);
  }
}
```

Το φέρον κύμα που σχηματίζεται έχει την παρακάτω μορφή.



*Σχήμα 4.2.2.1. Η μορφή των κυμάτων υπέρυθρης ακτινοβολίας, που στέλνουν τα IR LEDs για την ανίχνευση κάποιου εμποδίου.*

Ο εντοπισμός ενός αντικειμένου, πραγματοποιείται με τη χρήση μίας απλής εντολής

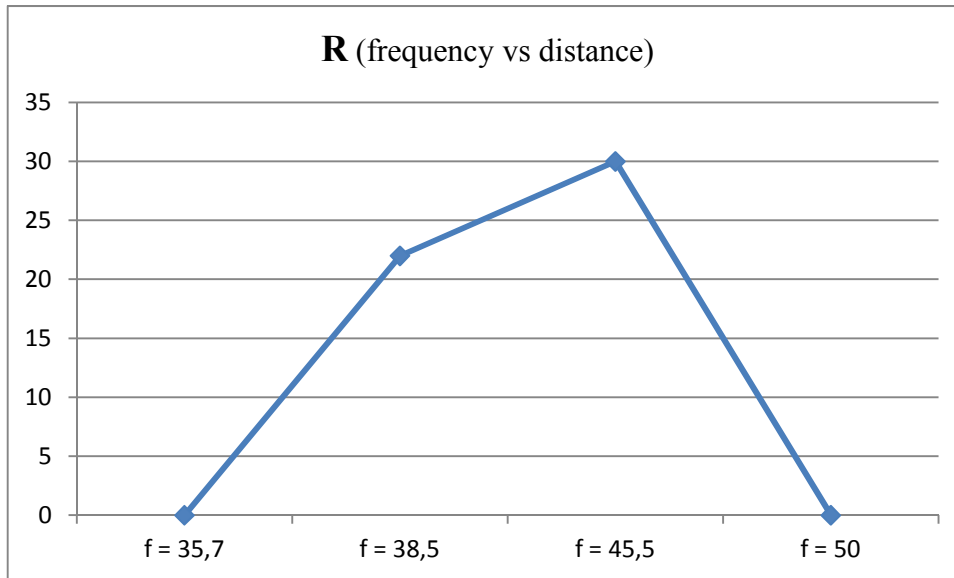
```
R = digitalRead(R_receiver_pin); // περίπτωση αισθητήρα R
```

Στην περίπτωση όπου ο ακροδέκτης εισόδου R\_receiver\_pin τίθεται σε λογικό επίπεδο LOW, έχουμε ανάκλαση του σήματος εκπομπής, γεγονός που σημαίνει την παρουσία ενός αντικειμένου στο οπτικό πεδίο του αισθητήρα R.

Οι δέκτες υπέρυθρων που χρησιμοποιήθηκαν, μπορούν να ανιχνεύσουν και σήματα με συχνότητες που βρίσκονται στην περιοχή γύρω από τα 38,5KHz. Λόγω κατασκευαστικών ατελειών, ο κάθε δέκτης υπέρυθρων μπορεί να έχει μέγιστη ευαισθησία σε συχνότητες διαφορετικές από τα 38,5KHz. Στα παρακάτω διαγράμματα, απεικονίζεται η ευαισθησία που έχουν οι 2 μπροστινοί αισθητήρες R και L, παρουσιάζοντας την απόσταση (σε cm) κάτω από την οποία εντοπίζεται ένα αντικείμενο, για διάφορες συχνότητες κοντά στα 38,5KHz.

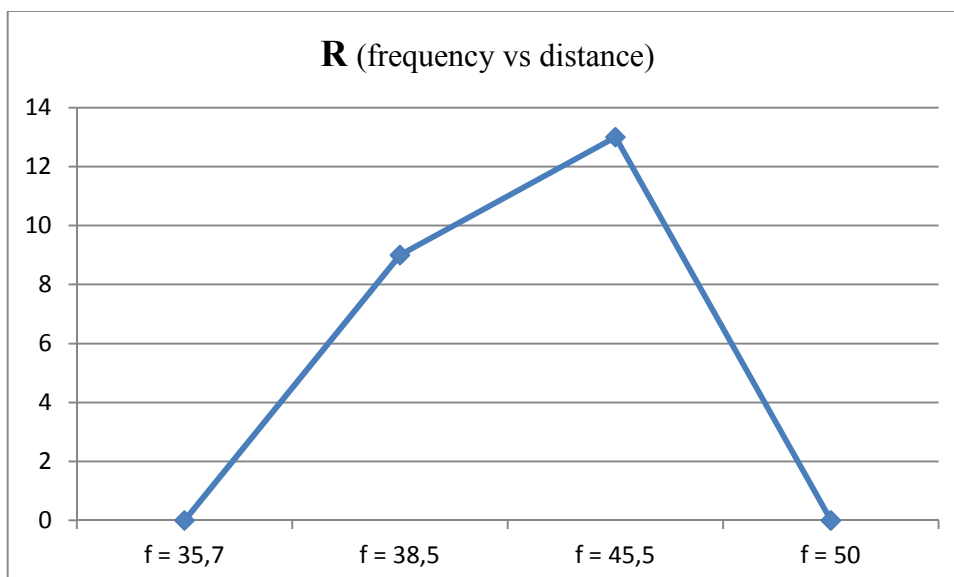
## ΑΙΣΘΗΤΗΡΑΣ R

f (KHz)	f = 35,7	f = 38,5	f = 41,5	f = 45,5	f = 50
d (cm)	0	22	$\infty$	30	0



Σχήμα 4.2.2.2. Εναισθησία δέκτη R για περίπτωση ανοιχτόχρωμων – λείων εμποδίων.

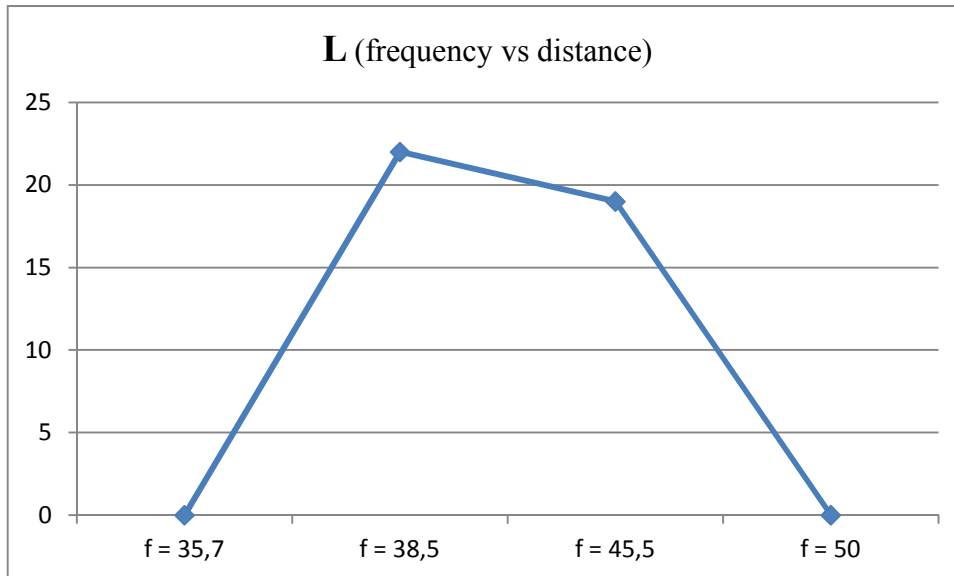
f (KHz)	f = 35,7	f = 38,5	f = 41,5	f = 45,5	f = 50
d (cm)	0	9	$\infty$	13	0



Σχήμα 4.2.2.3. Εναισθησία δέκτη R για περίπτωση σκουρόχρωμων – τραχέων εμποδίων.

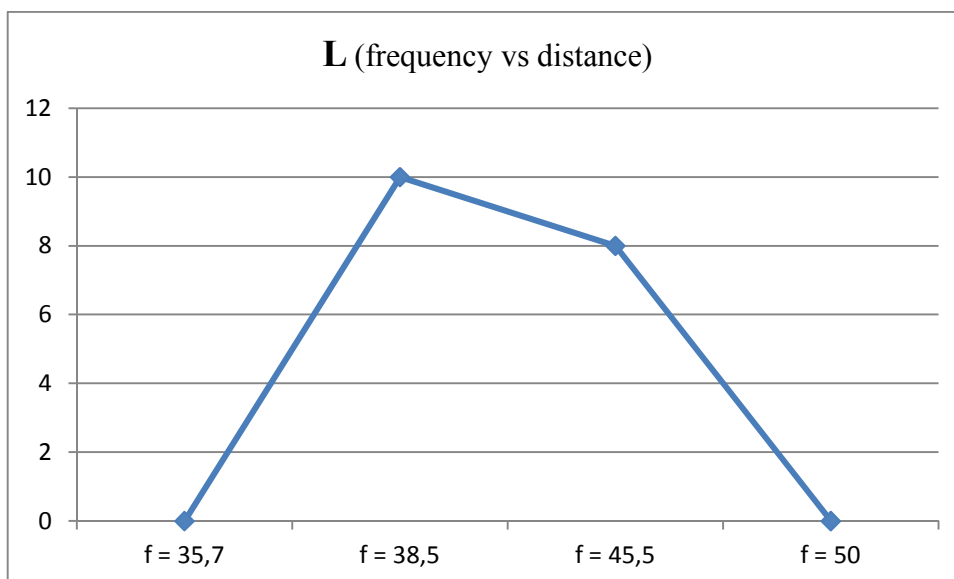
## ΑΙΣΘΗΤΗΡΑΣ L

f (KHz)	f = 35,7	f = 38,5	f = 41,5	f = 45,5	f = 50
d (cm)	0	22	$\infty$	19	0



Σχήμα 4.2.2.4. Εναισθησία δέκτη R για περίπτωση ανοιχτόχρωμων – λείων εμποδίων.

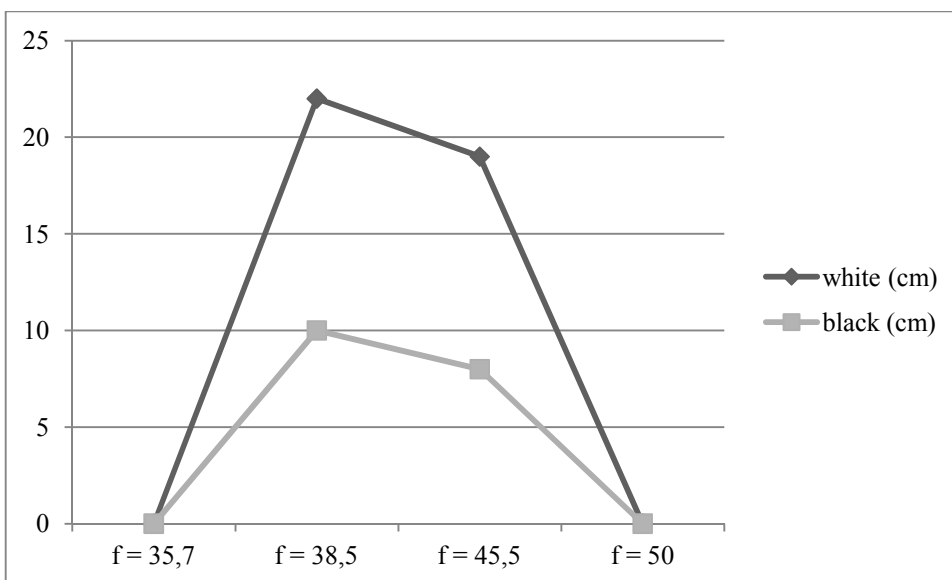
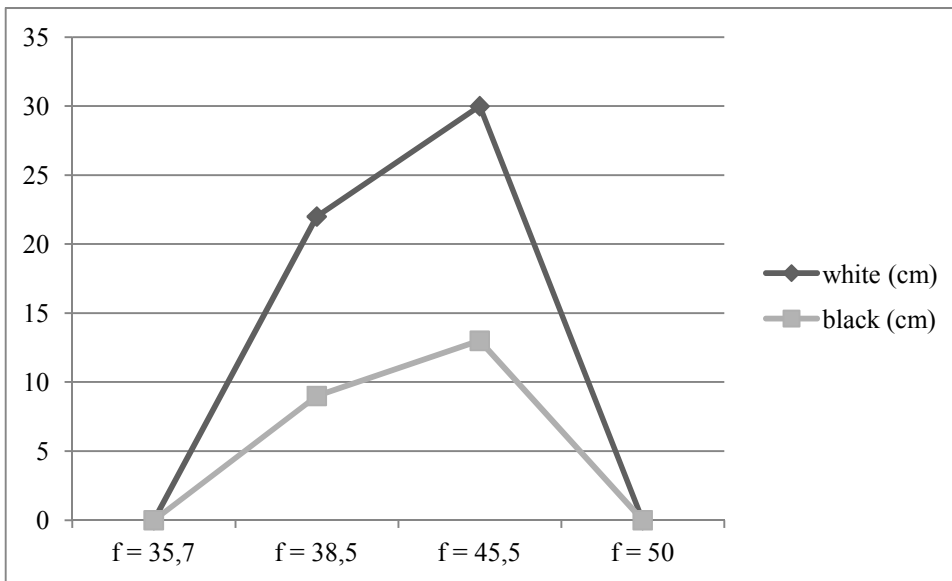
f (KHz)	f = 35,7	f = 38,5	f = 41,5	f = 45,5	f = 50
d (cm)	0	10	$\infty$	8	0



Σχήμα 4.2.2.5. Εναισθησία δέκτη L για περίπτωση σκουρόχρωμων – τραχέων εμποδίων.

Όπως βλέπουμε και από τους πίνακες των παραπάνω διαγραμμάτων, οι δέκτες των αισθητήρων R και L δεν έχουν μέγιστη ευαισθησία στα 38,5KHz, αλλά στην συχνότητα των 41,5KHz. Για τον εντοπισμό των εμποδίων πάντως, θα στείλουμε και από τους 2 αισθητήρες παλμούς συχνότητας 38,5KHz, για τους οποίους η ανίχνευση ενός αντικείμενου γίνεται στην ίδια απόσταση (22cm), όπως βλέπουμε από τα αντίστοιχα διαγράμματα.

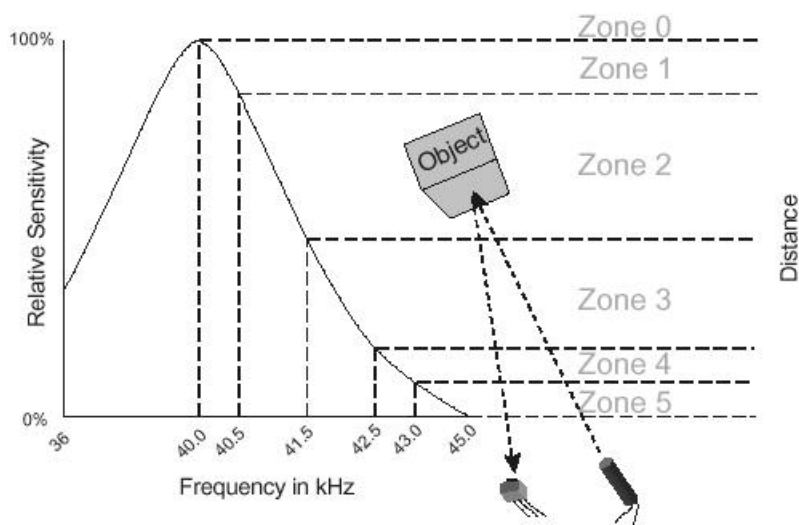
Επίσης, παρατηρούμε ότι υπάρχει μεγάλη διαφορά στην απόσταση που εντοπίζεται ένα ανοιχτόχρωμο από ένα σκουρόχρωμο αντικείμενο. Η διαφορά αυτή γίνεται ακόμα περισσότερο αντιληπτή, από τα συγκεντρωτικά διαγράμματα των παρακάτω σχημάτων. Παρόλα αυτά, θα θεωρήσουμε ότι τα αντικείμενα που βρίσκονται στον χώρο εργασίας, έχουν κατά βάση ανοιχτό χρώμα και λεία επιφάνεια (βλ. παράγραφο 4.5).



Σχήματα 4.2.2.6.a,b Διαφορά στην ανταπόκριση των δεκτών R και L αντίστοιχα, για ανοιχτόχρωμα – σκουρόχρωμα αντικείμενα..

### 4.2.3 Ευθυγράμμιση με εμπόδιο

Όπως αναφέραμε, ο δέκτης υπέρυθρης ακτινοβολίας μπορεί να ανιχνεύσει (με διαφορετική ανταπόκριση) και σήματα φέρουσας συχνότητας στην περιοχή γύρω από τα 38,5KHz. Κατά συνέπεια, αποσυντονίζοντας σταδιακά τον πομπό υπέρυθρων από αυτήν την τιμή, μπορούμε να προσεγγίσουμε την απόσταση ενός αντικειμένου στο οποίο ανακλάται η εκπεμπόμενη υπέρυθρη ακτινοβολία, χωρίζοντας με αυτόν τον τρόπο τον χώρο σε ζώνες, όπως φαίνεται και στο παρακάτω σχήμα.



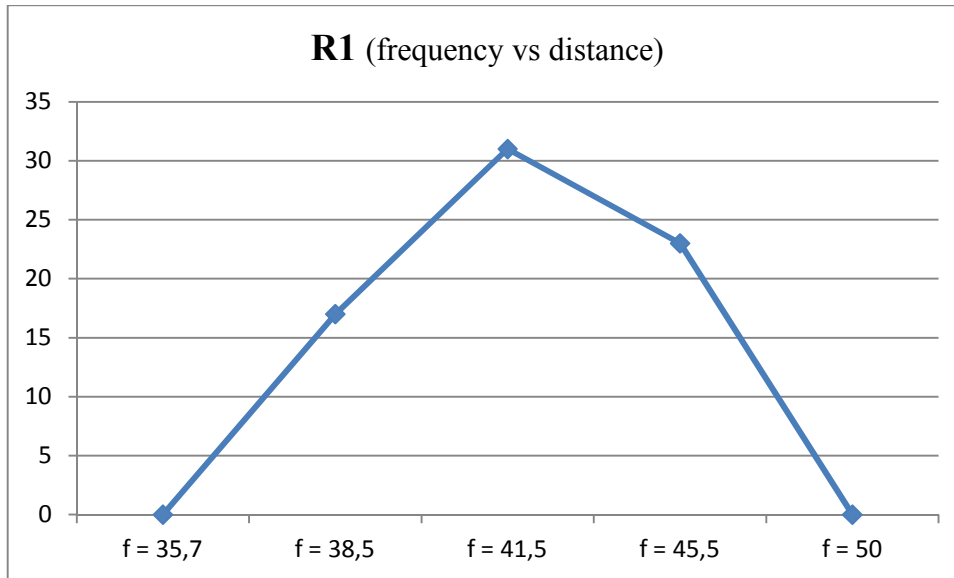
Σχήμα 4.2.3.1. Ο διαχωρισμός του χώρου σε ζώνες, με τη μεταβολή της φέρουσας συχνότητας των κυμάτων υπέρυθρης ακτινοβολίας που εκπέμπονται από τα IR LEDs.

Για την υλοποίηση της παραπάνω διαδικασίας, παραθέτουμε τα αντίστοιχα διαγράμματα (με αυτά της προηγούμενης παραγράφου) για τους πλάγιους αισθητήρες R1, R2, L1 και L2 (για περιπτώσεις ανοιχτόχρωμων αλλά και σκουρόχρωμων αντικειμένων).



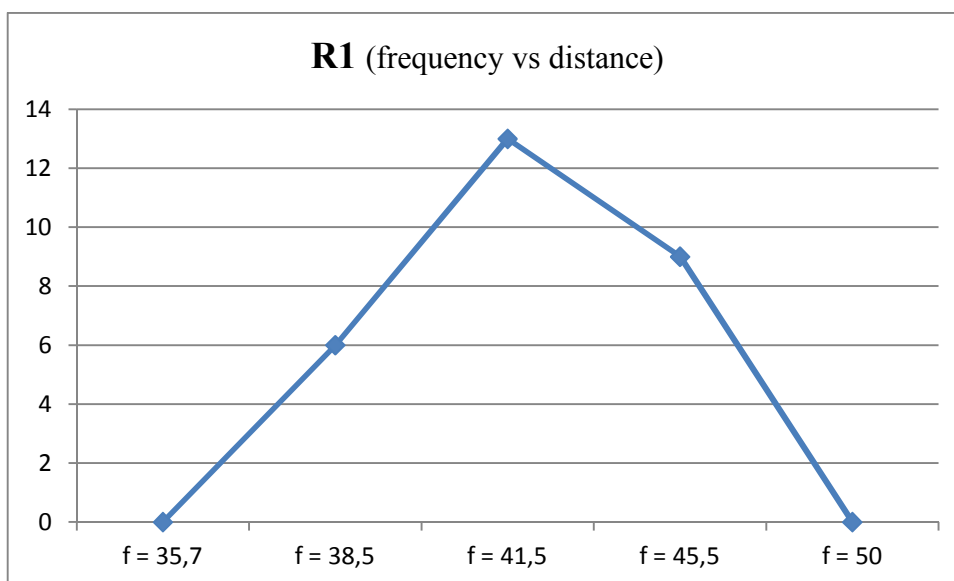
## ΑΙΣΘΗΤΗΡΑΣ R1

f (KHz)	f = 35,7	f = 38,5	f = 41,5	f = 45,5	f = 50
d (cm)	0	17	31	23	0



Σχήμα 4.2.3.2. Ανοιχτόχρωμα – λείας επιφάνειας αντικείμενα.

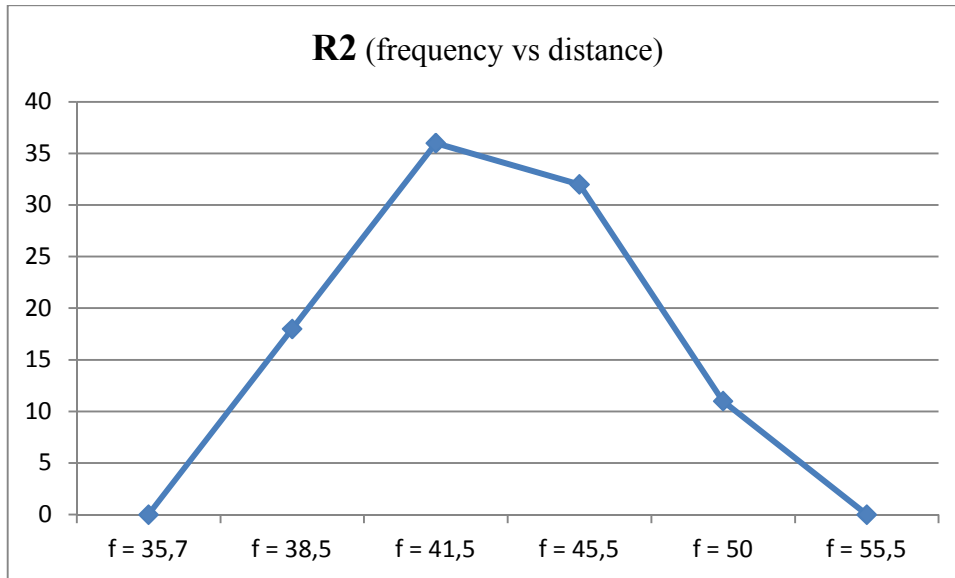
f (KHz)	f = 35,7	f = 38,5	f = 41,5	f = 45,5	f = 50
d (cm)	0	6	13	9	0



Σχήμα 4.2.3.3. Σκουρόχρωμα – τραχείας επιφάνειας αντικείμενα.

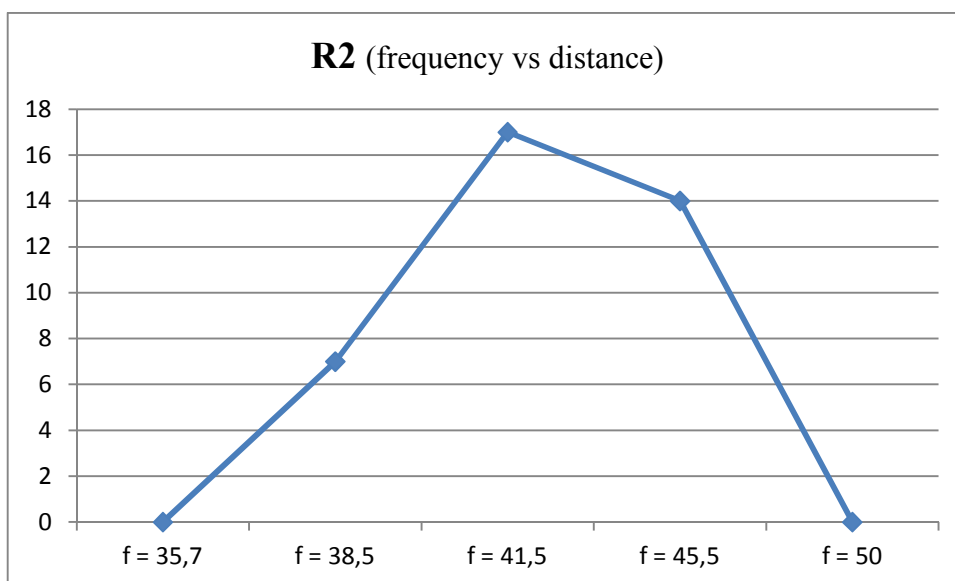
## ΑΙΣΘΗΤΗΡΑΣ R2

f (KHz)	f = 35,7	f = 38,5	f = 41,5	f = 45,5	f = 50	f = 55,5
d (cm)	0	18	36	32	11	0



Σχήμα 4.2.3.4. Ανοιχτόχρωμα – λείας επιφάνειας αντικείμενα.

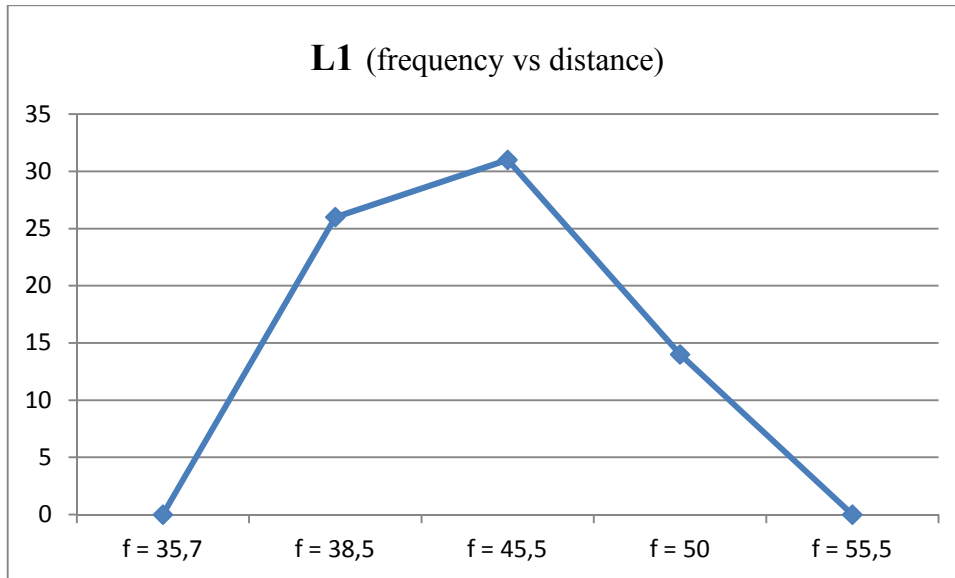
f (KHz)	f = 35,7	f = 38,5	f = 41,5	f = 45,5	f = 50
d (cm)	0	7	17	14	0



Σχήμα 4.2.3.5. Σκουρόχρωμα – τραχείας επιφάνειας αντικείμενα.

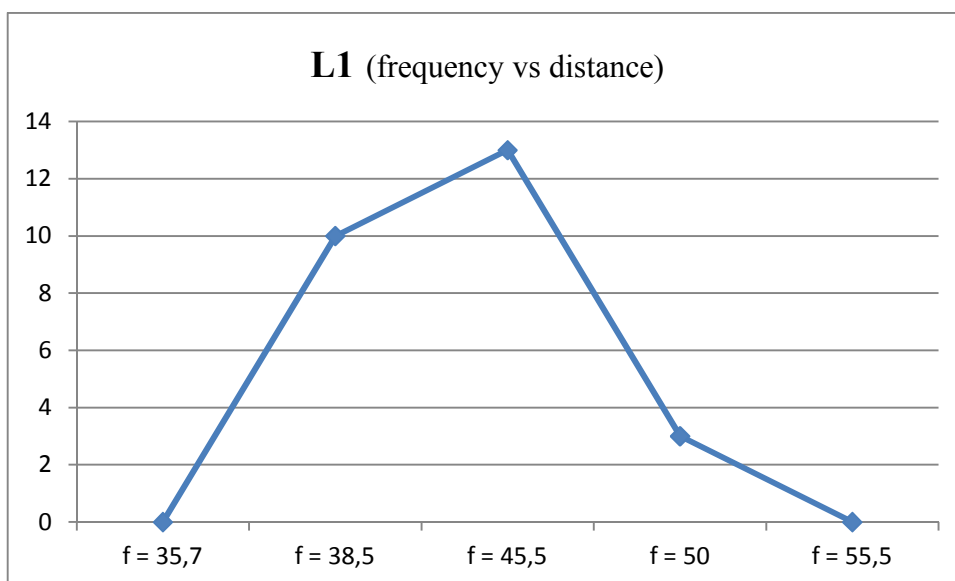
## ΑΙΣΘΗΤΗΡΑΣ L1

f (KHz)	f = 35,7	f = 38,5	f = 41,5	f = 45,5	f = 50	f = 55,5
d (cm)	0	26	$\infty$	31	14	0



Σχήμα 4.2.3.6. Ανοιχτόχρωμα – λείας επιφάνειας αντικείμενα.

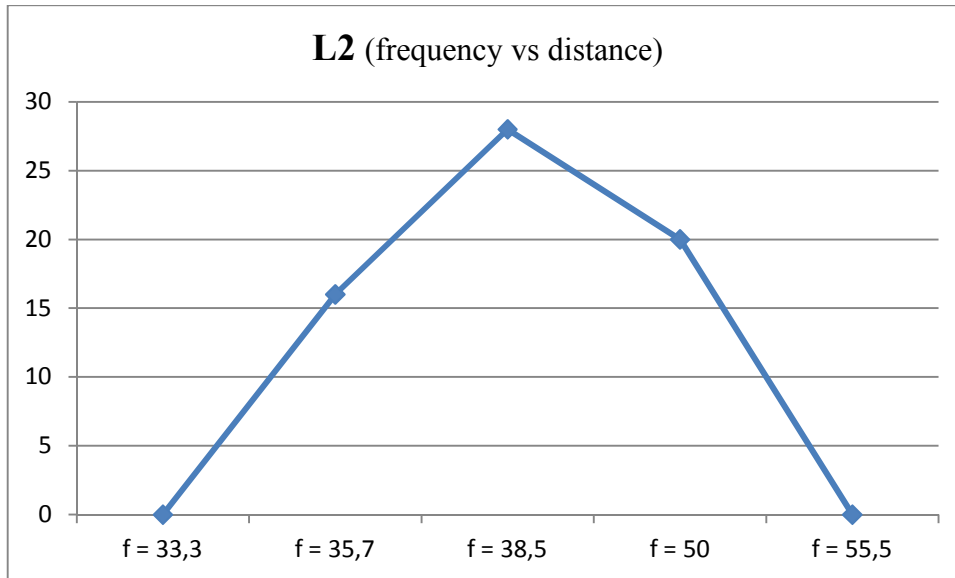
f (KHz)	f = 35,7	f = 38,5	f = 41,5	f = 45,5	f = 50	f = 55,5
d (cm)	0	10	$\infty$	13	3	0



Σχήμα 4.2.3.7. Σκουρόχρωμα – τραχείας επιφάνειας αντικείμενα.

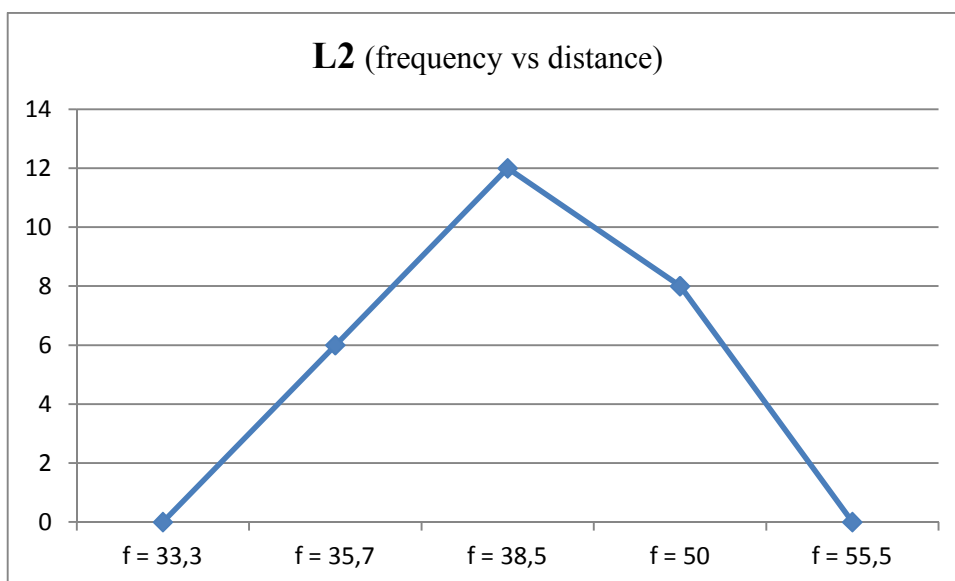
## ΑΙΣΘΗΤΗΡΑΣ L2

f (KHz)	f = 33,3	f = 35,7	f = 38,5	f = 45,5	f = 50	f = 55,5
d (cm)	0	16	28	$\infty$	20	0



Σχήμα 4.2.3.8. Ανοιχτόχρωμα – λείας επιφάνειας αντικείμενα.

f (KHz)	f = 33,3	f = 35,7	f = 38,5	f = 45,5	f = 50	f = 55,5
d (cm)	0	6	12	$\infty$	8	0



Σχήμα 4.2.3.9. Σκουρόχρωμα – τραχείας επιφάνειας αντικείμενα.

Με βάση τα παραπάνω διαγράμματα, πραγματοποιούμε τον προαναφερόμενο διαχωρισμό του χώρου (μπροστά από κάθε αισθητήρα) σε ζώνες απόστασης, οι οποίες παρουσιάζονται στους παρακάτω πίνακες (για κάθε έναν από τους πλάγιους αισθητήρες).

#### **ΑΙΣΘΗΤΗΡΑΣ R1**

d (cm)	d < 17	17 < d < 23	23 < d < 31	d > 31
zone	0	1	2	3

d (cm)	d < 6	6 < d < 9	9 < d < 13	d > 13
zone	0	1	2	3

*Πίνακες 4.2.3.1.a,b. Ζώνες εντοπισμού ανοιχτόχρωμων – σκουρόχρωμων εμποδίων αντίστοιχα, για τον αισθητήρα R1.*

#### **ΑΙΣΘΗΤΗΡΑΣ R2**

d (cm)	d < 11	11 < d < 18	18 < d < 32	32 < d < 36	d > 36
zone	0	1	2	3	4

d (cm)	d < 7	7 < d < 14	14 < d < 17	d > 17
zone	0	1	2	3

*Πίνακες 4.2.3.2.a,b. Ζώνες εντοπισμού ανοιχτόχρωμων – σκουρόχρωμων εμποδίων αντίστοιχα, για τον αισθητήρα R2.*

#### **ΑΙΣΘΗΤΗΡΑΣ L1**

d (cm)	d < 14	14 < d < 26	26 < d < 31	d > 31
zone	0	1	2	3

d (cm)	d < 3	3 < d < 10	10 < d < 13	d > 13
zone	0	1	2	3

*Πίνακες 4.2.3.3.a,b. Ζώνες εντοπισμού ανοιχτόχρωμων – σκουρόχρωμων εμποδίων αντίστοιχα, για τον αισθητήρα L1.*

## ΑΙΣΘΗΤΗΡΑΣ L2

d (cm)	d < 16	16 < d < 20	20 < d < 28	d > 28
zone	0	1	2	3

d (cm)	d < 6	6 < d < 8	8 < d < 12	d > 12
zone	0	1	2	3

*Πίνακες 4.2.3.4.a,b. Ζώνες εντοπισμού ανοιχτόχρωμων – σκουρόχρωμων εμποδίων αντίστοιχα, για τον αισθητήρα L2.*

Για τον υπολογισμό της ζώνης απόστασης στην οποία βρίσκεται ένα αντικείμενο, πραγματοποιούμε διαδοχικά την εκπομπή διαφορετικών σημάτων, στις φέρουσες συχνότητες των παραπάνω διαγραμμάτων (αναλόγως τον αισθητήρα). Η παραπάνω διαδικασία υλοποιείται με ρουτίνες της παρακάτω μορφής.

```
void R1_pulse_sending()
{
  int i;
  int j;
  int pulse[3] = {8,9,10};

  R1_distance = 0;

  for (i = 0; i <= 2; i += 1)
  {

    for (j = 0; j < num; j += 1)
    {
      digitalWrite(transmitter_pin, LOW);
      delayMicroseconds(pulse[i]);
      digitalWrite(transmitter_pin, HIGH);
      delayMicroseconds(pulse[i]);
    }

    R1 = digitalRead(R1_receiver_pin);
    R1_distance = R1_distance + R1;
  }
}
```

Το παραπάνω παράδειγμα αναφέρεται στον αισθητήρα R1, ενώ οι συχνότητες των σημάτων εκπομπής είναι αυτές των διαγραμμάτων 4.2.3.2 και 4.2.3.3 (38.5KHz, 41.5KHz, 45.5KHz).

Έστω λοιπόν, ότι ένα ανοιχτού χρώματος αντικείμενο βρίσκεται σε απόσταση 17cm έως 23cm μπροστά από τον αισθητήρα R1. Αρχικά, η τιμή της μεταβλητής `R1_distance` είναι 0. Εκπέμπεται ένα σήμα συχνότητας 45.5KHz, το οποίο (σύμφωνα με το διάγραμμα του σχήματος 4.2.3.2) εντοπίζει το αντικείμενο. Η εντολή `digitalRead` επιστρέφει την τιμή 0, η οποία και προστίθεται στην `R1_distance`. Εν συνεχεία, αποστέλλεται σήμα συχνότητας 41.5KHz, το οποίο επίσης αντιλαμβάνεται την ύπαρξη του εμποδίου. Η `digitalRead` επιστρέφει και πάλι 0, με αποτέλεσμα η `R1_distance` να παραμένει 0. Τέλος, στέλνουμε σήμα συχνότητας 38.5KHz, το οποίο δεν εντοπίζει το εμπόδιο με αποτέλεσμα η `digitalRead` να επιστρέφει την τιμή 1, η οποία προστίθεται στην `R1_distance`. Η τελική τιμή της μεταβλητής `R1_distance` είναι 1, και κατά συνέπεια το αντικείμενο βρίσκεται στην περιοχή 1 (zone 1). Με τον τρόπο αυτό, υλοποιείται ο υπολογισμός της ζώνης (απόστασης μπροστά από έναν αισθητήρα) στην οποία βρίσκεται ένα αντικείμενο.

Για την ευθυγράμμιση με ένα εμπόδιο, ελέγχουμε τις ζώνες στις οποίες εντοπίζεται το αντικείμενο από τους 2 πλάγιους αισθητήρες κατά την περιστροφή του οχήματος, και σταματάμε την περιστροφή του όταν το αντικείμενο βρεθεί στην ίδια ζώνη για τους 2 αισθητήρες.

Η βασική δυσκολία στην υλοποίηση της ευθυγράμμισης με ένα εμπόδιο, εντοπίζεται στον γεγονός ότι τα όρια των ζωνών απόστασης διαφέρουν από αισθητήρα σε αισθητήρα, και σε κάποιες περιπτώσεις η διαφορά αυτή είναι σημαντική (όπως παρατηρούμε και στα παραπάνω διαγράμματα). Για τον παραπάνω λόγο, σπάνια επιτυγχάνεται απόλυτος παραλληλισμός αλλά αντί αυτού, το ρομπότ συνήθως σχηματίζει κάποιας τάξης γωνία (σχετικά μικρή) ως προς το εμπόδιο. Η γωνία αυτή μπορεί να οδηγήσει σε σύγκρουση με το εμπόδιο, και κατά συνέπεια απαιτείται συνεχής έλεγχος των 2 πλάγιων αισθητήρων. Όταν οι 2 πλάγιοι αισθητήρες εντοπίσουν ξανά το εμπόδιο σε διαφορετική ζώνη, τότε το ρομπότ πρέπει να περιστραφεί μέχρις ότου οι αισθητήρες δουν και πάλι το εμπόδιο στην ίδια ζώνη. Στην περίπτωση όμως, όπου οι 2 αισθητήρες εντοπίσουν το εμπόδιο στην πρώτη ζώνη (zone 0), και το όχημα έχει σχηματίσει γωνία με κατεύθυνση προς το εμπόδιο, τότε η σύγκρουση θα είναι δύσκολο να αποφευχθεί.

Η εισαγωγή μίας προκαθορισμένης κίνησης προς τα πίσω (στην περίπτωση εντοπισμού εμποδίου) για ένα μικρό διάστημα, δεν εξυπηρετεί. Αυτό συμβαίνει διότι σε περίπτωση κατά την οποία το ρομπότ κατευθύνεται υπό μεγάλη γωνία προς το εμπόδιο (σχεδόν κάθετα), μία ενδεχόμενη κίνηση προς τα πίσω θα φέρει το ρομπότ σε μεγάλη απόσταση από το εμπόδιο. Εξαιτίας της σχετικά μικρής εμβέλειας των IR LEDs, όταν το ρομπότ περιστραφεί προκειμένου να ευθυγραμμιστεί με το αντικείμενο που συνάντησε, υπάρχει πολύ μεγάλη πιθανότητα οι πλάγιοι αισθητήρες να μην εντοπίσουν το αντικείμενο αυτό, με αποτέλεσμα το ρομπότ να περιστρέφεται συνεχόμενα στο σημείο όπου σταμάτησε.

Γενικά, ο συνεχόμενος έλεγχος των πλάγιων αισθητήρων κατά τη διάρκεια της κίνησης παραπλεύρως ενός εμποδίου, εμπεριέχει μία σημαντική δυσκολία. Οι ρουτίνες με τις οποίες υλοποιείται η εκπομπή του φέροντος κύματος από τα IR LEDs (και ειδικά οι ρουτίνες οι οποίες πραγματοποιούν τον υπολογισμό των ζωνών για τους πλάγιους αισθητήρες με την εκπομπή σημάτων διαφορετικής συχνότητας), απασχολούν τον επεξεργαστή του συστήματος για μεγάλα χρονικά διαστήματα (εξαιτίας του ότι χρησιμοποιούν μεγάλης διάρκειας loops). Κατά την διάρκεια εκτέλεσης των ρουτινών αυτών όμως, το σύστημα πρέπει να καταγράφει την απόσταση την οποία

διανύει, μέσω της μέτρησης των εναλλαγών του ενός κωδικοποιητή. Όσο ο επεξεργαστής είναι απασχολημένος εξαιτίας της εκτέλεσης κάποιας ρουτίνας εκπομπής υπέρυθρων σημάτων, «χάνονται» πολλές εναλλαγές, με αποτέλεσμα την μέτρηση μικρότερης διανυόμενης απόστασης από την πραγματική, και άρα την λανθασμένη λειτουργία ολόκληρου του συστήματος.

Για την αντιμετώπιση του προβλήματος, μειώσαμε τη χρονική διάρκεια εκτέλεσης των ρουτινών εκπομπής υπέρυθρης ακτινοβολίας, ενώ παράλληλα πραγματοποιήσαμε τη διαδικασία ευθυγράμμισης χρησιμοποιώντας μόνο τους οπίσθιους πλάγιους αισθητήρες (R2 και L2).

Όταν λοιπόν το ρομπότ συναντήσει ένα εμπόδιο, τότε θα σταματήσει, και στην συνέχεια θα περιστραφεί, μέχρις ότου ο αισθητήρας R2 ή ο αισθητήρας L2 (ανάλογα με την πλευρά του εμποδίου) εντοπίσει το αντικείμενο στην πρώτη ζώνη (zone 0), σχηματίζοντας με αυτόν τον τρόπο, γωνία απομάκρυνσης από αυτό.

Όλα τα παραπάνω προβλήματα παρακάμπτονται μέσω της μέτρησης των εναλλαγών του κωδικοποιητή χρησιμοποιώντας διακοπές (interrupts). Όπως θα περιγράψουμε παρακάτω, οι interrupts παραμένουν ανεπηρέαστες από την εκτέλεση του υπόλοιπου προγράμματος. Με τον τρόπο αυτόν, έχουμε τη δυνατότητα να εκτελούμε ιδιαίτερα πολύπλοκες και χρονοβόρες διαδικασίες, χωρίς να επηρεάζεται η καταμέτρηση της απόστασης που διανύεται. Όμως, όπως θα εξηγήσουμε στην συνέχεια, στην παρούσα φάση δεν ήταν δυνατή η χρήση διακοπών στην ανάγνωση των κωδικοποιητών.

## **4.3 Εφαρμογές Κωδικοποιητών**

### **4.3.1 Εισαγωγή – Κύλιση χωρίς ολίσθηση**

Βασική αρμοδιότητα των κωδικοποιητών (encoder discs) που έχουμε χρησιμοποιήσει στο πίσω μέρος καθενός από τους 2 κινητήριους τροχούς, είναι η υλοποίηση του οδομέτρου για την μέτρηση της καλυπτόμενης απόστασης.

Εκτός όμως από τον παραπάνω λόγο, η χρήση των κωδικοποιητών έχει παράλληλα σκοπό και τον συντονισμό των 2 κινητήριων τροχών. Λόγω κατασκευαστικών ατελειών οι κινητήρες μπορεί να κινηθούν με οριακά διαφορετική ταχύτητα από αυτήν που προβλέπει το σήμα που έχουμε στείλει. Το γεγονός αυτό όμως, μπορεί να δημιουργήσει μεγάλη απόκλιση στην προβλεπόμενη πορεία του οχήματος, ειδικά στην περίπτωση που καλύψει σχετικά μεγάλη απόσταση. Για τον λόγο αυτόν, η διαδικασία του συντονισμού των τροχών έχει ιδιαίτερη σημασία για την ορθότητα της συνολικής λειτουργίας του συστήματος.

Ο συντονισμός των τροχών πραγματοποιείται με συνεχόμενο έλεγχο του διαστήματος κατά το οποίο έχει κινηθεί ο κάθε ένας από τους 2 τροχούς. Η διαδικασία αυτή επιτυγχάνεται με τη βοήθεια των κωδικοποιητών. Αν ο αισθητήρας του ενός τροχού έχει ανιχνεύσει εναλλαγή χρώματος στον κωδικοποιητή, τότε διακόπεται η κίνηση του συγκεκριμένου τροχού, μέχρις ότου εντοπίσει και ο άλλος αισθητήρας την αντίστοιχη εναλλαγή. Αυτό λοιπόν που γίνεται στην ουσία, είναι συνεχόμενη διακοπή και επανέναρξη της κίνησης των 2 τροχών, όπως έχουμε ήδη αναφέρει.



Η επιλογή κωδικοποιητών με μεγάλο αριθμό τμημάτων, συμβάλει στον περιορισμό της ταλάντευσης του οχήματος που προκύπτει εξαιτίας της παραπάνω διαδικασίας.

Επιπλέον, όπως επίσης έχουμε αναφέρει, η συνεχόμενη διακοπή και επανέναρξη της κίνησης των τροχών έχει ως αποτέλεσμα την μείωση της κύλισης με ολίσθηση. Η ολίσθηση αποτελεί ένα φαινόμενο το οποίο προσπαθούμε να αποφύγουμε, καθώς μειώνει την ακρίβεια της απόστασης που μετράμε μέσω του οδομέτρου. Παρόλα αυτά, στην παρούσα εφαρμογή, η επιλογή των συγκεκριμένων ελαστικών σε συνδυασμό με το ιδιαίτερα μεγάλο βάρος της κατασκευής, καθιστούν αμελητέα την ολίσθηση κατά την κίνηση του οχήματος.

### Κύλιση χωρίς ολίσθηση

Ένας τροχός κυλιέται χωρίς να ολισθαίνει, όταν δεν υπάρχει σχετική κίνηση μεταξύ του σημείου K του τροχού που εφάπτεται στο έδαφος, και του εδάφους. Κατ συνέπεια, το σημείο K έχει μηδενική ταχύτητα

$$v_K = 0$$

και άρα η τριβή που ασκείται από το έδαφος στον τροχό κατά την κύλισή του, είναι στατική.

Στον τροχό λοιπόν, ασκούνται η δύναμη του βάρους ( $B_{wh}$ ), η στατική τριβή ( $\vec{T}_s$ ), και η κάθετη αντίδραση από το έδαφος ( $\vec{N}$ ). Το μέτρο της στατικής τριβής καθορίζεται από τις υπόλοιπες δυνάμεις που ασκούνται στο σώμα, και μπορεί να πάρει τιμές εντός του παρακάτω διαστήματος.

$$0 \leq T_s \leq \mu_s N$$

Όταν η στατική τριβή πάρει τη μέγιστη τιμή της τότε αρχίζει η ολίσθηση του τροχού. Άρα, προκειμένου να επιτυγχάνεται η κύλιση του τροχού χωρίς ταυτόχρονη ολίσθηση, πρέπει να ισχύει η επόμενη συνθήκη.

$$T_s < \mu_s N$$

Όταν  $T_s = \mu_s N$  τότε ο τροχός είναι έτοιμος να ολισθήσει.

Είναι προφανές, ότι η δύναμη στην οποία οφείλεται η μεταφορική κίνηση του τροχού (και άρα του οχήματος), είναι η τριβή. Αν δεν υπήρχε η τριβή, τότε δεν θα υπήρχε και ροπή ως προς τον άξονα συμμετρίας του τροχού, με αποτέλεσμα ο τροχός να ολισθαίνει χωρίς να κυλιέται.

Όταν ένας τροχός ακτίνας  $r$  κυλιέται χωρίς να ολισθαίνει, τότε κάθε σημείο της περιφέρειάς του έρχεται διαδοχικά σε επαφή με το έδαφος. Κατά συνέπεια, αν το κέντρο μάζας του τροχού έχει μετακινηθεί κατά διάστημα  $dx$ , ένα σημείο της περιφέρειας θα έχει στραφεί κατά γωνία  $d\theta$ , η οποία και αντιστοιχεί σε μήκος τόξου  $ds$ . Λόγω της μη ολίσθησης έχουμε  $dx = ds$ , και επειδή ισχύει

$$ds = r \cdot d\theta$$

έχουμε

$$dx = r \cdot d\theta$$

δηλαδή

$$x = r \cdot \theta$$

Η τελευταία σχέση είναι η αναγκαία συνθήκη ώστε ένας τροχός να κυλιέται χωρίς να ολισθαίνει, και αποτελεί την βάση πάνω στην οποία στηρίζεται η λογική της οδομετρίας. Αν δεν ισχύει η παραπάνω σχέση, τότε δεν είναι δυνατό να εφαρμοστεί η τεχνική του οδομέτρου. Βέβαια, όπως θα δούμε στο επόμενο κεφάλαιο, με μία στατιστική ανάλυση του σφάλματος που εισάγεται από την κύλιση με ολίσθηση των τροχών, μας δίνεται η δυνατότητα να εφαρμόσουμε τεχνικές οδομετρίας για την προσεγγιστική μέτρηση της απόστασης, και για περιπτώσεις όπου υπάρχει σχετικά μικρού βαθμού ολίσθηση.

### 4.3.2 Υλοποίηση οδομέτρου

Η βασικότερη εργασία που επιτελούν οι κωδικοποιητές είναι η μέτρηση της διανυόμενης απόστασης. Η υλοποίηση ενός τρόπου οδομετρίας είναι το σημαντικότερο μέρος στην προσπάθεια πλοήγησης του παρόντος ρομποτικού συστήματος. Μέσω της χρήσης του οδομέτρου, θα λυθεί το ζήτημα της εκτίμησης της θέσης του συστήματος στο χώρο, καθώς επίσης και του προσανατολισμού του.

Παρά την μεγάλη σημασία που έχει για την πλοήγηση η υλοποίηση του οδομέτρου, αποτελεί μία ιδιαίτερα απλή διαδικασία. Η μέτρηση της απόστασης πραγματοποιείται με μία απλή καταμέτρηση των εναλλαγών χρώματος (άσπρο – μαύρο) στην επιφάνεια των κωδικοποιητών, από τον αντίστοιχο αισθητήρα CNY70. Ο αριθμός των εναλλαγών πολλαπλασιάζεται κάθε φορά με το μήκος του τόξου κάθε τμήματος του κωδικοποιητή. Το συγκεκριμένο μήκος θα το αναφέρουμε ως *μήκος βήματος (step)*, και υπολογίζεται με τον παρακάτω τρόπο :

$$step = \frac{2\pi r}{n}$$

όπου

$2\pi r$  η περιφέρεια του τροχού ακτίνας  $r = d / 2 = 11 / 2 = 5,5cm$  ( $d$  η διάμετρος του τροχού)

$n = 36$  ο αριθμός των τμημάτων του κωδικοποιητή που χρησιμοποιήσαμε (βλ. παράγραφο 3.5).

Άρα,

$$step = \frac{2\pi r}{n} = \frac{2 \times 3,14 \times 5,5}{36} = \frac{34,54}{36} \cong 0,95945cm$$

Κατά συνέπεια, για κάθε εναλλαγή του χρώματος διανύεται απόσταση ίση με το μήκος βήματος step.

Ένας απλός τρόπος να μετρήσουμε τις εναλλαγές του χρώματος των encoder discs, είναι να χρησιμοποιήσουμε διακοπές (*interrupts*). Όπως είδαμε στην παράγραφο 3.5, η ανάγνωση των κωδικοποιητών από τον αισθητήρα CNY70, μεταφράζεται σε μία κυματομορφή, η οποία οδηγούμενη στην μη-αναστρέφουσα είσοδο του συγκριτή LM393, μας δίνει μία παλμοσειρά που αντιστοιχεί στις εναλλαγές χρώματος του κωδικοποιητή. Ρυθμίζοντας τη λειτουργία των διακοπών έτσι ώστε να «χτυπάνε» σε κάθε μετάβαση HIGH – LOW του παλμού (παράμετρος CHANGE), δηλαδή σε κάθε αλλαγή της κατάστασης του αντίστοιχου pin εισόδου του μικροελεγκτή, τότε για κάθε interrupt μετράμε εναλλαγή του χρώματος, με αποτέλεσμα την πολύ εύκολη καταμέτρηση της απόστασης.

Η χρήση διακοπών όμως, εμπεριέχει μία μεγάλη δυσκολία η οποία προκύπτει από το ότι «χτυπάνε» πάρα πολλές interrupts σε κάθε εναλλαγή. Το γεγονός αυτό οφείλεται στις μεγάλες διακυμάνσεις της τάσης που συμβαίνουν στιγμιαία κατά τη διάρκεια μίας εναλλαγής, δηλαδή κατά τη μετάβαση από υψηλή σε χαμηλή στάθμη και αντίστροφα, στην παλμοσειρά εξόδου. Για την αντιμετώπιση του συγκεκριμένου προβλήματος απαιτείται η εισαγωγή καθυστερήσεων, γεγονός που καθιστά ιδιαίτερα περίπλοκη την υλοποίηση του οδομέτρου με χρήση διακοπών.

Η μέτρηση της απόστασης που διανύεται λοιπόν, δεν θα πραγματοποιηθεί με διακοπές (*interrupts*), αλλά θα γίνει χρησιμοποιώντας ένα flag, με τη βοήθεια του οποίου, θα καταγράφεται η προηγούμενη κατάσταση του ενός αισθητήρα CNY70 (έξοδος του συγκριτή LM393), και θα συγκρίνεται με την τρέχουσα κατάσταση. Αν έχει αλλάξει η κατάσταση του αισθητήρα, σημαίνει ότι έχει εντοπιστεί εναλλαγή του χρώματος στην επιφάνεια του ενός κωδικοποιητή, η οποία και καταγράφεται.

Παρακάτω, παρουσιάζουμε τον κώδικα με τον οποίο υλοποιείται η καταμέτρηση των εναλλαγών (*changes*).

```
void distance_counter()
{
  //ανάγνωση της κατάστασης του αισθητήρα του δεξιού τροχού
  //χρησιμοποιούμε digitalRead καθώς διαβάζουμε την έξοδο του συγκριτή
  right_status = digitalRead(right_sens_pin);

  if (right_status == HIGH)
  {
    if (flag == 2)
    {
      changes += 1;
    }
    flag = 1;
  }
  else
  {
    if (flag == 1)
    {
```

```

    changes += 1;
}
flag = 2;
}
}

```

Για τον υπολογισμό της τρέχουσας απόστασης, απλά πολλαπλασιάζουμε τον αριθμό των μετρούμενων εναλλαγών με το μήκος βήματος (step), με την εντολή :

```
distance = changes * disc_step;
```

όπου  $disc\_step = 0.95945$  όπως έχουμε προκαθορίσει.

Όμως, η μέτρηση των εναλλαγών χρώματος στους κωδικοποιητές, δεν πραγματοποιείται μόνο για τον υπολογισμό της καλυπτόμενης απόστασης, αλλά και για τον υπολογισμό των γωνιών κατά τις οποίες το όχημα περιστρέφεται.

Για να δημιουργήσουμε μία σχέση μεταξύ του διαστήματος το οποίο διανύει ο ένας τροχός και της γωνίας κατά την οποία περιστρέφεται το όχημα, πρέπει καταρχήν, να υπολογίσουμε την ακτίνα περιστροφής. Όπως έχουμε αναφέρει, το ρομπότ στρίβει επιτόπια, δηλαδή περιστρέφεται ως προς το μέσο της ευθείας που συνδέει τους 2 κινητήριους τροχούς (το ICC είναι στο μέσο του άξονα των 2 τροχών). Κατά συνέπεια, το μήκος του άξονα των 2 τροχών αποτελεί τη διάμετρο του κύκλου που διαγράφεται κατά την περιστροφή του οχήματος. Το μήκος του άξονα διαφορικής οδήγησης (η απόσταση μεταξύ των 2 τροχών δηλαδή)  $D_{robot}$  – βλ. παράγραφο 2.3.5, είναι :

$$D_{robot} = 28.5cm$$

Η απόσταση αυτή μετρήθηκε μεταξύ των σημείων στα οποία ο κάθε τροχός έρχεται σε επαφή με το έδαφος (μέσο του κάθε τροχού).

Κατά συνέπεια, η ακτίνα περιστροφής είναι :

$$R_{robot} = D_{robot}/2 = 14.25cm$$

Η περιφέρεια του διαγραφόμενου κύκλου είναι :

$$\Pi = 2\pi R_{robot} = 2 \cdot 3.14 \cdot 14.25 = 89.5cm$$

Το διάστημα αυτό αντιστοιχεί σε  $360^\circ$  ή αλλιώς  $2\pi rad = 6.28 rad$ .

Με απλή αναγωγή, έχουμε :

$$\frac{89.5}{d} = \frac{6.28}{\varphi}$$

$$89.5\varphi = 6.28d$$

$$\varphi = 0.07d$$

Καταλήγουμε λοιπόν στην παραπάνω σχέση, η οποία μας δίνει τη γωνία  $\varphi$  κατά την οποία περιστρέφεται ολόκληρο το όχημα, όταν ο κάθε τροχός διανύει ένα διάστημα  $d$ .

Κατά συνέπεια, εκτός από την μέτρηση της διανυόμενης απόστασης, με τη χρήση του οδομέτρου πραγματοποιείται και ο υπολογισμός των γωνιών περιστροφής του ρομπότ. Με τον τρόπο αυτόν, θα υλοποιηθεί η συνολική διαδικασία εκτίμησης θέσης και προσανατολισμού του ρομποτικού συστήματος.

Όλοι οι υπολογισμοί γωνιών, γίνονται σε rad, κατά την εκτύπωση των μηνυμάτων όμως, όλες οι γωνίες εμφανίζονται σε μοίρες, για την καλύτερη κατανόηση της λειτουργίας.

Στο σημείο αυτό, είναι σημαντικό να αναφέρουμε ότι η χρήση διακοπών στο ιδιαίτερα κρίσιμο ζήτημα της μέτρησης της απόστασης, αποτελεί την καλύτερη δυνατή επιλογή, καθώς οι interrupts μένουν ανεπηρέαστες από άλλες λειτουργίες του προγράμματος, με αποτέλεσμα να καθιστούν απίθανη την μη καταμέτρηση κάποιας εναλλαγής. Μελλοντικά, μία ενδεχομένως περισσότερο πολύπλοκη λειτουργία του συστήματος, ίσως να καταστήσει απαραίτητη την χρήση διακοπών στην υλοποίηση του οδομέτρου.

### 4.3.3 Συντονισμός τροχών

Ο τρόπος με τον οποίο θα προσπαθήσουμε να συντονίσουμε τους 2 κινητήριους τροχούς περιγράφηκε στην παράγραφο 4.3.1. Η συγκεκριμένη διαδικασία λοιπόν, θα υλοποιηθεί με την παρακάτω ρουτίνα.

```
void rotate_motors()
{
    wait_right_servo = 0;
    wait_left_servo = 0;
    if ((left_status != previous_left_status) && (right_status ==
previous_right_status))
    {
        wait_left_servo = 1;
    }
    else if ((left_status == previous_left_status) && (right_status !=
previous_right_status))
    {
        wait_right_servo = 1;
    }
    else if ((left_status != previous_left_status) && (right_status !=
previous_right_status))
    {
        previous_right_status = right_status;
        previous_left_status = left_status;
    }

    if (wait_left_servo == 0)
    {
```

```

    left_servo.write(0);
}
else
{
    left_servo.write(90);
}
if (wait_right_servo == 0)
{
    right_servo.write(180);
}
else
{
    right_servo.write(90);
}
}

```

### όπου

```

right_status = digitalRead(right_sens_pin);
left_status = digitalRead(left_sens_pin);

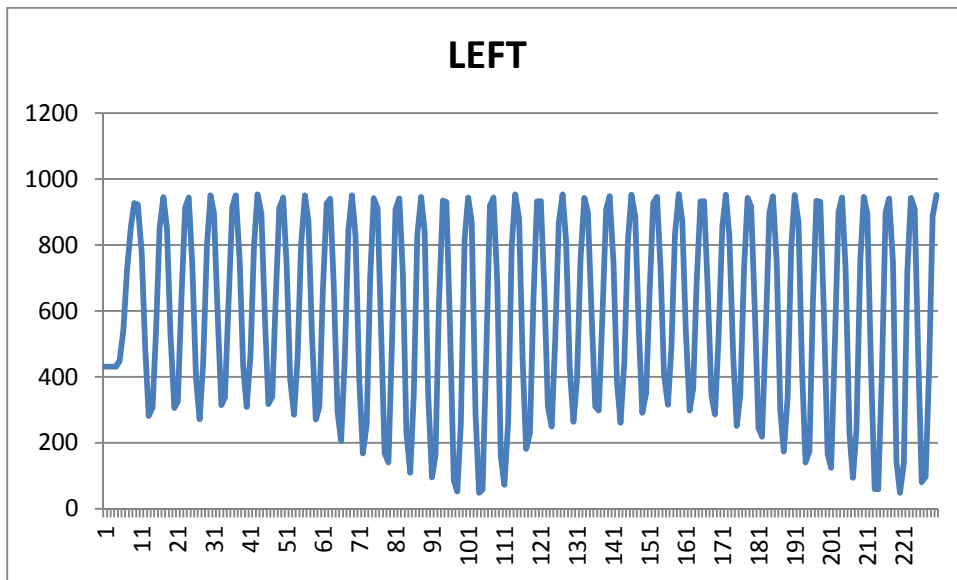
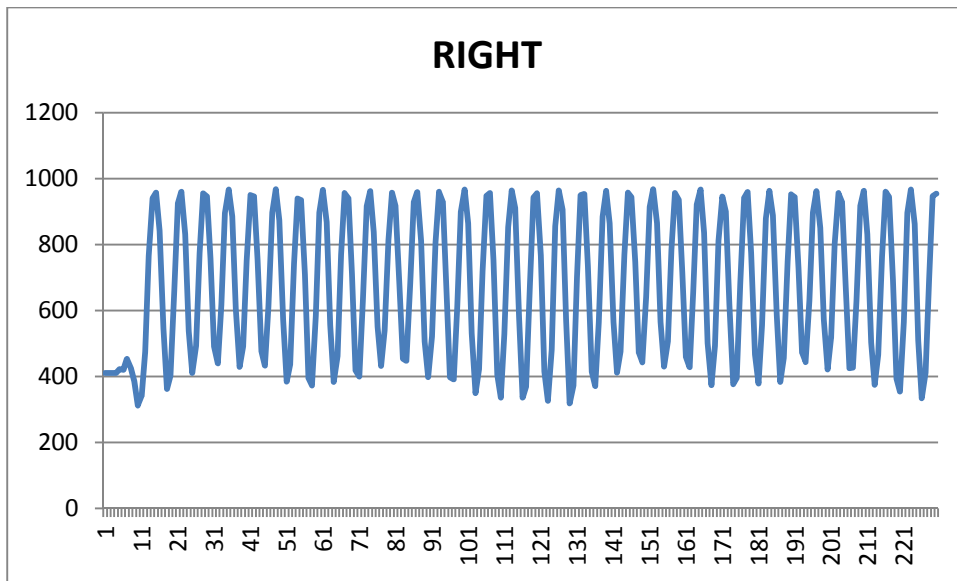
```

οι τιμές εισόδου των αισθητήρων CNY70 για δεξιό και αριστερό τροχό αντίστοιχα.

Η παραπάνω προσπάθεια για συντονισμό της κίνησης των 2 τροχών δεν ήταν επιτυχημένη. Το γεγονός αυτό υποθέτουμε ότι οφείλεται στις μεγάλες διακυμάνσεις της τάσης κατά την μετάβαση από χαμηλή σε υψηλή στάθμη (και αντίστροφα) των παλμών στις εξόδους του συγκριτή LM393 (δηλαδή, των παλμοσειρών εξόδου των αισθητήρων). Για να αντιμετωπίσουμε το συγκεκριμένο πρόβλημα, θα προσπαθήσουμε να εκμεταλλευτούμε τη δυνατότητα που παρέχει ο arduino, για αναλογική ανάγνωση του σήματος εξόδου του κάθε αισθητήρα. Κατά συνέπεια λοιπόν, θα παρακάμψουμε τον συγκριτή LM393, και θα οδηγήσουμε τις εξόδους των 2 αισθητήρων CNY70, κατευθείαν στους ακροδέκτες αναλογικής εισόδου του arduino (συγκεκριμένα, pins A0, A1 για δεξιό και αριστερό τροχό αντίστοιχα).

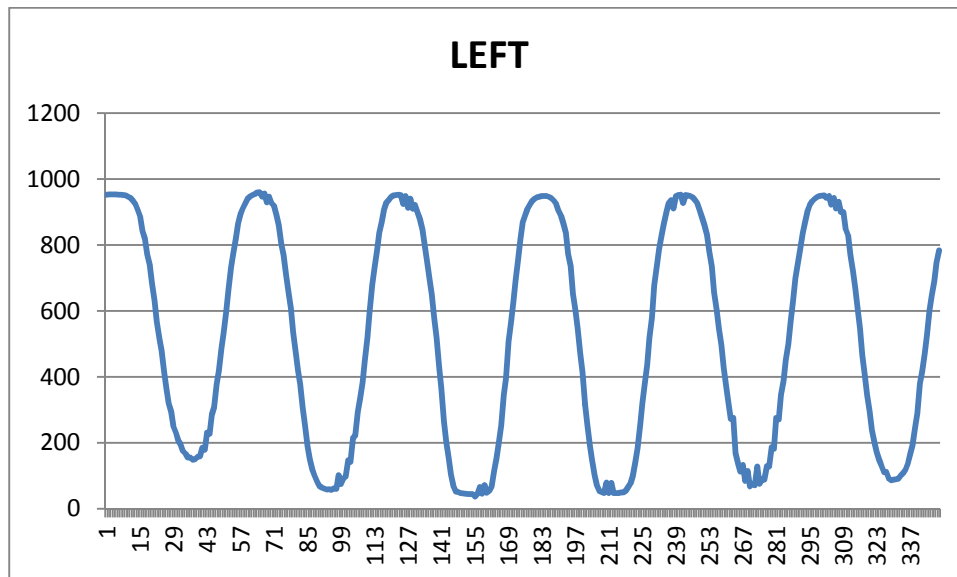
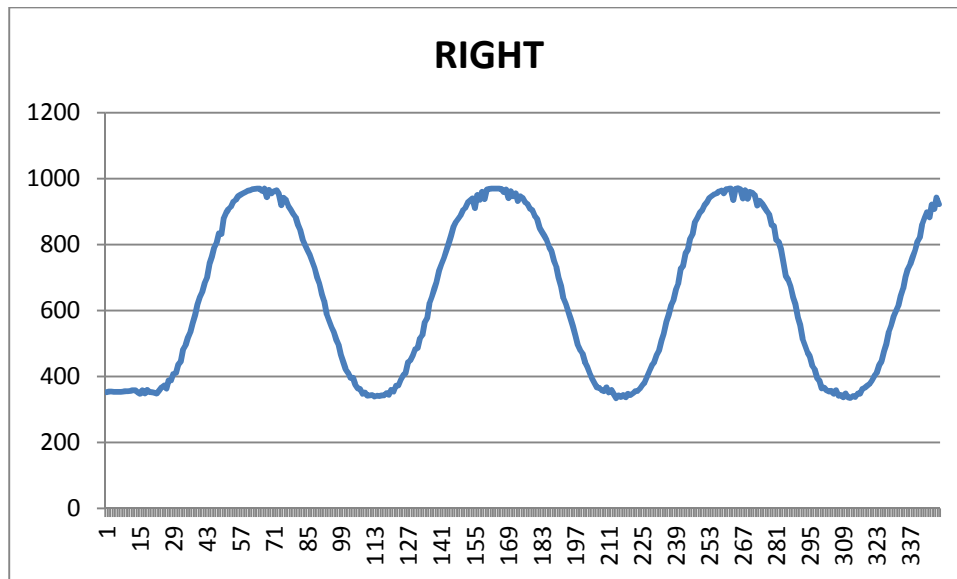
Σε γενικές γραμμές, η παρουσία του συγκριτή LM393 είναι πολύ σημαντική, καθώς παρέχει τη δυνατότητα για χρήση διακοπών (interrupts) κατά την μέτρηση των εναλλαγών χρώματος του κάθε κωδικοποιητή. Χωρίς την παρουσία ενός συγκριτή για την μετατροπή της κυματομορφής εξόδου των αισθητήρων σε παλμοσειρά, η χρήση διακοπών δεν είναι εφικτή.

Στα παρακάτω διαγράμματα, παρουσιάζουμε τις αναλογικές τιμές εξόδου των αισθητήρων CNY70, συγκεντρώνοντας ένα μεγάλο δείγμα τιμών, κατά την περιστροφή των τροχών σε μέγιστη ταχύτητα.



Σχήματα 4.3.3.1 και 4.3.3.2. Αναλογικές τιμές εξόδου αισθητήρων CNY70.

Αν καταγράψουμε ένα μεγάλο δείγμα τιμών εξόδου, αυτήν τη φορά όμως, για μικρή ταχύτητα περιστροφής των 2 τροχών, παίρνουμε τα παρακάτω διαγράμματα.



Σχήματα 4.3.3.1 και 4.3.3.2. Έξοδοι αισθητήρων CNY70 για μικρές ταχύτητες περιστροφής.

Στην περίπτωση των 2 παραπάνω διαγραμμάτων, έχει σταλεί ο ίδιος παλμός στους 2 κινητήρες προκειμένου να περιστραφούν με την ίδια ταχύτητα. Παρατηρώντας λοιπόν, τα 2 παραπάνω διαγράμματα φαίνεται χαρακτηριστικά το πόσο ασυντόνιστα περιστρέφονται οι κινητήρες (υπάρχει μεγάλη διαφορά μεταξύ των πραγματικών ταχυτήτων των κινητήρων για ίδιο σήμα ελέγχου).

Για την ανίχνευση, λοιπόν, της εναλλαγής χρώματος, θα θέσουμε ένα κατώφλι στην ανάγνωση της τιμής του κάθε αισθητήρα. Η επιλογή του κατωφλίου αυτού θα γίνει με βάση την μέση περίπου τιμή εξόδου του κάθε αισθητήρα. Κατά συνέπεια, παρατηρώντας τα διαγράμματα των σχημάτων 4.3.3.1 και 4.3.3.2 θα επιλέξουμε :



$$R_{TH} = 650$$

$$L_{TH} = 600$$

για τους αισθητήρες του δεξιού και του αριστερού τροχού αντίστοιχα.

Η διαφορά στον κώδικα είναι ότι πρέπει να θέσουμε την μεταβλητή που δείχνει την κατάσταση του κάθε αισθητήρα (`right_status`, `left_status`), με βάση τις παραπάνω τιμές. Άρα :

```
right_value = analogRead(right_sens_pin);
left_value = analogRead(left_sens_pin);
if (right_value > 650)
{
    right_status = 1;
}
else
{
    right_status = 0;
}
if (left_value > 600)
{
    left_status = 1;
}
else
{
    left_status = 0;
}
```

Ο υπόλοιπος κώδικας (ρουτίνα `rotate_motors` που περιγράψαμε παραπάνω) παραμένει ο ίδιος.

Επειδή και οι τελευταίες προσπάθειες δεν οδήγησαν στον απόλυτο συγχρονισμό των τροχών, καταλήξαμε στην πιο αποδοτική (τελικά) λύση, η οποία ήταν ο πειραματικός υπολογισμός του παλμού που πρέπει να σταλεί σε κάθε κινητήρα, προκειμένου οι 2 κινητήρες να περιστραφούν με την ίδια ταχύτητα. Μετά από δοκιμές λοιπόν, καταλήξαμε στη ρουτίνα :

```
void forward_move()
{
    right_servo.write(180);
    left_servo.write(63);
}
```

η οποία οδηγεί στην απολύτως ευθύγραμμη πορεία του οχήματος.

#### 4.4 Περιγραφή της λειτουργίας

Στις επόμενες παραγράφους θα περιγράψουμε αναλυτικά τον τρόπο με τον οποίο προγραμματίστηκε η συνολική πλοήγηση του συστήματος. Όπως έχουμε αναφέρει, η λειτουργία που εκτελεί στην παρούσα φάση το ρομπότ ακουστικών μετρήσεων, είναι η αυτόνομη μετακίνησή του μεταξύ 2 δοθέντων σημείων του χώρου, και η ταυτόχρονη καταγραφή των πληροφοριών που

αφορούν τη θέση του και τον περιβάλλον χώρο, και τις οποίες συλλέγει κατά την πλοήγησή του. Η διαδικασία αυτή αποτελεί τη βάση για επέκταση της λειτουργίας στην μετακίνηση, διαδοχικά, σε όλα τα δοθέντα σημεία στα οποία θέλουμε να ληφθούν ακουστικές μετρήσεις, ενώ παράλληλα παρέχεται η δυνατότητα της εύκολης χαρτογράφησης του χώρου.

Αρχικά λοιπόν, εισάγονται μέσω της σειριακής θύρας του υπολογιστή, τα σημεία έναρξης και προορισμού. Για λόγους καλύτερης κατανόησης, θεωρούμε αρχικά ότι το ρομπότ είναι προσανατολισμένο κατά τη θετική φορά του άξονα Y, και η εισαγωγή των σημείων έναρξης και προορισμού γίνεται με αυτό το δεδομένο. Για να είναι δυνατός, όμως, ο υπολογισμός των γωνιών περιστροφής, περιστρέφουμε το σύστημα συντεταγμένων, έτσι ώστε το ρομπότ να βρεθεί προσανατολισμένο κατά τη θετική φορά του άξονα X. Όταν το ρομπότ στρίβει δεξιά, η γωνία περιστροφής είναι αρνητική, ενώ στην αντίθετη περίπτωση, θετική. Τα πραγματικά σημεία έναρξης και προορισμού είναι, δηλαδή, συμμετρικά με τα δοθέντα, ως προς των άξονα X. Το αρχικό σημείο μεταφέρεται στην αρχή των αξόνων, ενώ προσδιορίζεται η θέση του σημείου προορισμού στο νέο σύστημα συντεταγμένων. Άρα :

$$x'_T = x_T - x_A$$

$$y'_T = y_T - y_A$$

Υπολογίζεται η απόσταση μεταξύ των 2 σημείων με βάση την σχέση :

$$d = \sqrt{x'^2_T + y'^2_T}$$

ενώ επίσης υπολογίζεται η γωνία του σημείου  $(x'_T, y'_T)$  με βάση το ανάπτυγμα Maclaurin της συνάρτησης τόξο εφαπτομένης :

$$\arctan(t) = \sum_{n=0}^{\infty} \frac{(-1)^n}{2n+1} t^{2n+1}$$

όπου  $t = y'_T / x'_T$

Ο παραπάνω τύπος ισχύει για  $|t| \leq 1$  δηλαδή για  $|y'_T / x'_T| \leq 1$ . Για την αντίθετη περίπτωση υπολογίζουμε το  $\arctan(1/t)$  και παίρνουμε την συμπληρωματική γωνία. Επίσης, ανάλογα με το πρόσημο των  $x'_T$  και  $y'_T$  παίρνουμε όλες τις περιπτώσεις, όπως φαίνεται και στον παρακάτω κώδικα υπολογισμού των γωνιών.

```
double angle_calculation(float x, float y)
{
    float t;
    double a;

    if (abs(x) >= abs(y))
    {
```

```

t = y / x;
a = arctan(t);
if ((x < 0) && (y > 0))
{
    a = a + 3.14;
}
}
else
{
    t = x / y;
    a = arctan(t);
    a = 1.57 - a;
    if ((y < 0) && (x > 0))
    {
        a = a - 3.14;
    }
}
if ((x <= 0) && (y <= 0))
{
    a = a - 3.14;
}
return a;
}

```

Γνωρίζοντας λοιπόν τη διεύθυνση του διανύσματος προορισμού, το ρομπότ περιστρέφεται όσο χρειάζεται ώστε να κινηθεί προς την συγκεκριμένη κατεύθυνση, και ξεκινάει την πορεία του, μετρώντας το διάστημα που διανύει, μέχρι να καλύψει την συνολική απόσταση που υπολογίστηκε παραπάνω.

Στην περίπτωση στην οποία το ρομπότ συναντήσει κάποιο εμπόδιο, τότε το ρομπότ σταματάει, υπολογίζονται οι συντεταγμένες του σημείου στο οποίο σταμάτησε (αφού απόσταση και γωνία είναι γνωστές), το σημείο αυτό γίνεται σημείο αναφοράς, και το σημείο προορισμού υπολογίζεται εκ νέου ως προς το σημείο εντοπισμού του εμποδίου (νέο σημείο αναφοράς).

Εν συνεχεία, το ρομπότ περιστρέφεται μέχρις ότου σχηματιστεί γωνία απομάκρυνσης από το εμπόδιο, δηλαδή όταν το εμπόδιο εντοπιστεί στην πρώτη ζώνη (zone 0) από τον αισθητήρα R2 ή τον αισθητήρα L2, ανάλογα με την πλευρά στην οποία βρίσκεται το αντικείμενο. Υπολογίζεται η γωνία κατά την οποία περιστράφηκε το ρομπότ, με τον τρόπο που περιγράφηκε στην παράγραφο 4.3.2, και το όχημα ξεκινάει και πάλι την κίνησή του, μετρώντας από την αρχή την απόσταση που διανύει, κινούμενο αυτήν τη φορά παραπλεύρως του εμποδίου.

Κατά τη διάρκεια της κίνησης αυτής, γίνεται συνεχόμενος έλεγχος της ζώνης στην οποία εντοπίζεται το εμπόδιο από τους πλάγιους αισθητήρες. Όταν το εμπόδιο βρεθεί στην τελευταία ζώνη του αισθητήρα R2 ή L2 ανάλογα (zone 3 συνήθως), το ρομπότ αντιλαμβάνεται το τέλος του εμποδίου, συνεχίζει να κινείται για ένα προκαθορισμένο διάστημα (20cm) και σταματάει, καταγράφοντας το σημείο στο οποίο σταμάτησε. Το σημείο αυτό γίνεται σημείο αναφοράς και υπολογίζεται η γωνία και η απόσταση του σημείου προορισμού ως προς το νέο σημείο αναφοράς. Από την γωνία του σημείου προορισμού αφαιρείται η γωνία της τρέχουσας κατεύθυνσης του

οχήματος, και σαν αποτέλεσμα παίρνουμε φορά και γωνία κατά την οποία πρέπει να περιστραφεί το ρομπότ ώστε να κατευθυνθεί εκ νέου προς το σημείο προορισμού. Το ρομπότ περιστρέφεται και ξεκινάει και πάλι την κίνησή του μέχρι να καλύψει την απόσταση που υπολογίστηκε προηγουμένως. Αν συναντηθεί ξανά κάποιο εμπόδιο, ακολουθείται η ίδια διαδικασία από την αρχή.

Κατά τη διάρκεια της κίνησης παραπλεύρως ενδεχόμενου εμποδίου, έχουμε καταστήσει απαραίτητο τον έλεγχο και των 2 μπροστινών αισθητήρων (R και L) για την ανίχνευση νέου εμποδίου. Ο εκ νέου εντοπισμός κάποιου εμποδίου έχουμε θεωρήσει ότι σημαίνει το τέλος των ορίων του χώρου. Στην περίπτωση αυτήν, το ρομπότ διακόπτει τη λειτουργία του και αναμένει την είσοδο νέου σημείου προορισμού. Όπως έχουμε αναλύσει όμως, για τους λόγους που περιγράφηκαν στην παράγραφο 4.2.3, στην παρούσα φάση δεν είναι δυνατός ο ταυτόχρονος έλεγχος μπροστινών και πλάγιων αισθητήρων. Σε θεωρητική βάση πάντως, αυτή είναι η δική μας πρόταση, όσον αφορά τον τρόπο με τον οποίο το ρομπότ θα αντιλαμβάνεται τα όρια του χώρου εργασίας.

## 4.5 Περιορισμοί

Η παραπάνω περιγραφή της λειτουργίας του συστήματος, και κυρίως η περιγραφή της διαδικασίας αποφυγής εμποδίων, μας οδηγούν στο συμπέρασμα ότι οφείλουμε να κάνουμε κάποιες παραδοχές. Για λόγους απλότητας λοιπόν, αλλά και καλύτερης κατανόησης της όλης εφαρμογής, κάναμε ορισμένες παραδοχές, κυρίως όσον αφορά στο περιβάλλον πλοήγησης του συστήματος, και ειδικά στη φύση των εμποδίων που συναντώνται.

Για την ορθή λειτουργία της εφαρμογής λοιπόν, θεωρούμε ότι τα εμπόδια που βρίσκονται στον χώρο εκτέλεσης πληρούν της παρακάτω προϋποθέσεις :

1. είναι αντικείμενα μεγάλων διαστάσεων, αρκετά μεγαλύτερων διαστάσεων, δηλαδή, από το μέγεθος της κατασκευής

Η παραπάνω παραδοχή γίνεται προκειμένου να επιτυγχάνεται η διαδικασία της ευθυγράμμισης με τα εμπόδια. Σε περίπτωση εμποδίων μικρότερων διαστάσεων από αυτές του ρομπότ, η διαδικασία της ευθυγράμμισης δεν είναι εφικτή. Εκτός από τον παραπάνω λόγο, αντικείμενα μικρών διαστάσεων, όπως για παράδειγμα ένας λεπτός στύλος, είναι δύσκολο να εντοπιστούν.

2. τα εμπόδια είναι αντικείμενα ορθογωνικού σχήματος και δεν δημιουργούν καμπυλώσεις, προκειμένου να είναι πραγματοποιήσιμη η ευθυγράμμιση με αυτά

Η διαδικασία της ευθυγράμμισης επιτυγχάνεται πάντως, για εμπόδια που σχηματίζουν αμελητέα καμπυλότητα, δηλαδή για αντικείμενα των οποίων η βασική διάσταση μπορεί προσεγγιστικά να θεωρηθεί ευθεία.

3. τα εμπόδια είναι κατά κύριο λόγο, ανοιχτόχρωμα αντικείμενα και έχουν σχετικά λεία επιφάνεια

Όπως είδαμε στην ενότητα της αποφυγής εμποδίων, για αντικείμενα με σκουρόχρωμη ή τραχιά επιφάνεια, οι αποστάσεις εντοπισμού των εμποδίων μειώνονται σημαντικά, και δεν εξυπηρετούν για την παρούσα εφαρμογή. Εκτός αυτού, οι ζώνες αποστάσεων για τις διάφορες συχνότητες εκπομπής των IR LEDs, συγκλίνουν αισθητά μεταξύ τους και είναι πλέον δύσκολο να διαχωριστούν όταν τα ρομπότ περιστρέφεται προκειμένου να παραλληλιστεί με το εμπόδιο.

4. τα εμπόδια είναι αντικείμενα τα οποία δεν σχηματίζουν εσωτερικές γωνίες

Όσον αφορά στην παραπάνω περίπτωση, η συγκεκριμένη παραδοχή, εκτός από λόγους απλότητας της λειτουργίας, έγινε και προκειμένου να έχουμε τη δυνατότητα να θέσουμε έναν τρόπο αναγνώρισης των ορίων του χώρου. Όπως περιγράψαμε και προηγουμένως, όταν το ρομπότ εντοπίσει ένα εμπόδιο, τότε θα ευθυγραμμιστεί με αυτό και θα κινηθεί παράλληλα με το εμπόδιο μέχρις ότου σταματήσει να το βλέπει από τους πλαϊνούς αισθητήρες. Αν το όχημα κινούμενο παράλληλα με το εμπόδιο, εντοπίσει μπροστά και άλλο αντικείμενο, τότε θεωρούμε ότι έχει βρεθεί σε κάποια γωνία, γεγονός που σημαίνει το τέλος των ορίων του χώρου. Με τον τρόπο αυτό, το ρομπότ έχει τη δυνατότητα αναγνώρισης των μη προσβάσιμων σημείων. Κατά συνέπεια, ο περιορισμός αυτός έγινε και με σκοπό τον καλύτερο ορισμό της λειτουργίας της όλης εφαρμογής.

5. τα εμπόδια δεν παρουσιάζουν κάποια κλίση

6. δεν υπάρχουν κινούμενα εμπόδια στον χώρο εκτέλεσης της εφαρμογής

Σε γενικές γραμμές λοιπόν, σύμφωνα με τους παραπάνω περιορισμούς, θεωρούμε ότι η εκτέλεση της λειτουργίας του συστήματος πραγματοποιείται σε έναν σχετικά μεγάλο, άδειο χώρο, όπου υπάρχει η πιθανότητα να βρίσκονται μερικά μεγάλων διαστάσεων αντικείμενα, όπως για παράδειγμα μεγάλες κολώνες (ή κάποια εσωτερική σκάλα) σε κεντρικά σημεία του χώρου. Όλες οι παραπάνω περιπτώσεις περιορισμών πάντως, είναι δυνατό να ξεπεραστούν με τις κατάλληλες κατασκευαστικές, και κυρίως, προγραμματιστικές μετατροπές.

Τέλος, θεωρούμε ότι η πλοήγηση πραγματοποιείται :

- σε οριζόντιο επίπεδο, αν και οι συγκεκριμένη κινητήρες μπορούν να δώσουν στο όχημα τη δυνατότητα να ανέλθει και σε ελαφρώς κεκλιμένο επίπεδο, παράλληλα όμως, με την έντονη επιβάρυνση των μπαταριών

- σε σχετικά ομαλό δάπεδο, αν και η επιλογή των συγκεκριμένων ελαστικών, σε συνδυασμό με της μεγάλης διαμέτρου τροχούς, δίνει στο όχημα τη δυνατότητα να κινηθεί και σε πιο ανώμαλες επιφάνειες (ή να διέρθει από πάνω από σχετικά μικρών διαστάσεων αντικείμενα), χωρίς να εισάγεται ένα απαγορευτικού μεγέθους σφάλμα.

## 4.6 Ο κώδικας

Παρακάτω λοιπόν, παρουσιάζεται τον τελικό κώδικα του προγράμματος συνολικής λειτουργίας του συστήματος.

```
#include <Servo.h>

const int transmitter_pin = 2;
const int R2_receiver_pin = 3;
const int R1_receiver_pin = 4;
const int R_receiver_pin = 5;
const int L_receiver_pin = 6;
const int L1_receiver_pin = 7;
const int L2_receiver_pin = 8;
const int right_sens_pin = 11;
const int left_sens_pin = 12;
const int led_pin = 13;

const int num_1 = 500;
const int num_2 = 300;
const float disc_step = 0.95945;

int R = 0;
int R1 = 0;
int R2 = 0;
int L = 0;
int L1 = 0;
int L2 = 0;
int R_distance = 0;
int R1_distance = 0;
int R2_distance = 0;
int L_distance = 0;
int L1_distance = 0;
int L2_distance = 0;
int flag = 0;
int cont = 0;
int count = 0;
int changes = 0;
int turning = 0;
```

```

int new_point = 0;
int destination = 0;
int right_status = 0;
int left_status = 0;

int ch;
int i = 0;
int j = 0;
int pros[4];
int points[4];
int numbers[10];

double distance = 0;
double arc = 0;
double degrees_angle;
double turning_angle;
double turning_arc;
double final_distance;
double final_angle;
double final_arc;
double final_x;
double final_y;
double given_x;
double given_y;
double current_distance = 0;
double current_angle = 0;
double current_arc;
double current_x = 0;
double current_y = 0;

unsigned long time = 0;
unsigned long previous_time = 0;

Servo right_servo;
Servo left_servo;

void setup()
{
  Serial.begin(9600);
  pinMode(transmitter_pin, OUTPUT);
  pinMode(R_receiver_pin, INPUT);
  pinMode(R1_receiver_pin, INPUT);
  pinMode(R2_receiver_pin, INPUT);
  pinMode(L_receiver_pin, INPUT);
  pinMode(L1_receiver_pin, INPUT);
  pinMode(L2_receiver_pin, INPUT);
  pinMode(right_sens_pin, INPUT);
  pinMode(left_sens_pin, INPUT);
  right_servo.attach(9);
  left_servo.attach(10);
}

```

```

void loop()
{
  if (destination == 1)
  {
    stop_motors();
    i = 0;
    j = 0;
    for (i = 0; i < 4; i += 1)
    {
      pros[i] = 1;
      points[i] = 0;
    }
    while (j < 4)
    {
      ch = Serial.read();
      if (ch != -1)
      {
        ch -= 48;
        if (ch != -16)
        {
          if (ch == -3)
          {
            pros[j] = -1;
          }
          else
          {
            numbers[i] = ch;
            points[j] = 10*points[j] + numbers[i];
            i++;
          }
        }
        else
        {
          i = 0;
          j++;
        }
      }
    }
    destination = 0;
    for (i = 0; i < 4; i += 1)
    {
      points[i] = pros[i] * points[i];
    }
    current_x = points[0];
    current_y = points[1];
    given_x = points[2];
    given_y = points[3];
    given_x -= current_x;
    given_y -= current_y;
    final_x = given_y;
    final_y = (-1) * given_x;
    final_distance = distance_calculation(final_x, final_y);
    Serial.println("IP");
  }
}

```



```

}

if (current_distance < final_distance)
{
  if (new_point == 0)
  {
    new_point = 1;
    final_distance = distance_calculation(final_x, final_y);
    final_angle = angle_calculation(final_x, final_y);
    turning_angle = final_angle - current_angle;
    turning_arc = abs(turning_angle) / 0.07;
    if (turning_angle < 0)
    {
      do
      {
        turn_right();
        distance_counter();
        arc = changes * disc_step;
      } while (arc < turning_arc);
      Serial.print("RT ");
    }
    else
    {
      do
      {
        turn_left();
        distance_counter();
        arc = changes * disc_step;
      } while (arc < turning_arc);
      Serial.print("LT ");
    }
    degrees_angle = abs(turning_angle) * 57.3;
    Serial.println(degrees_angle);
    changes = 0;
    stop_motors();
    delay(1500);
    Serial.println("DC 0");
  }
  else
  {
    count += 1;
    front_pulse_sending();
    if (((R_distance == 0) || (L_distance == 0)) && (count > 20))
    {
      count = 0;
      stop_motors();
      changes = 0;
      current_x = current_distance * cos(final_angle);
      current_y = current_distance * sin(final_angle);
      final_x -= current_x;
      final_y -= current_y;
      current_x = 0;
      current_y = 0;
    }
  }
}

```

```

delay(1500);
if ((R_distance == 0) && (L_distance != 0))
{
  Serial.println("ROF");
  Serial.print("DC ");
  Serial.println(current_distance);
  turning = 1;
  while (cont == 0)
  {
    count += 1;
    R2_pulse_sending();
    if ((R2_distance == 0) && (count > 20))
    {
      stop_motors();
      cont = 1;
      count = 0;
    }
    else
    {
      turn_left();
      distance_counter();
    }
  }
  arc = changes * disc_step;
  current_angle = 0.07 * arc;
  if (current_angle < 0.82)
  {
    changes = 0;
    step_left();
    current_angle += 0.07 * arc;
  }
  degrees_angle = abs(current_angle) * 57.3;
  Serial.print("LT ");
  Serial.println(degrees_angle);
  current_angle = final_angle + current_angle;
}
else if (L_distance == 0)
{
  Serial.println("LOF");
  Serial.print("DC ");
  Serial.println(current_distance);
  turning = 2;
  while (cont == 0)
  {
    count += 1;
    L2_pulse_sending();
    if ((L2_distance == 0) && (count > 20))
    {
      stop_motors();
      cont = 1;
      count = 0;
    }
    else

```

```

        {
            turn_right();
            distance_counter();
        }
    }
    arc = changes * disc_step;
    current_angle = 0.07 * arc;
    if (current_angle < 0.90)
    {
        changes = 0;
        step_right();
        current_angle += 0.07 * arc;
    }
    degrees_angle = abs(current_angle) * 57.3;
    Serial.print("RT ");
    Serial.println(degrees_angle);
    current_angle = final_angle - current_angle;
}
cont = 0;
stop_motors();
changes = 0;
delay(1500);
Serial.println("DC 0");
}
else
{
    forward_move();
    distance_counter();
    distance = changes * disc_step;
    current_distance = distance;
    sending_messages();
}

if (turning != 0)
{
    current_distance = 0;
    if (turning == 1)
    {
        do
        {
            forward_move();
            distance_counter();
            distance = changes * disc_step;
            current_distance = distance;
            sending_messages();
            R2_pulse_sending();
        } while (R2_distance != 2);
        Serial.println("ROE");
    }
    else if (turning == 2)
    {
        do
        {

```

```

        forward_move();
        distance_counter();
        distance = changes * disc_step;
        current_distance = distance;
        sending_messages();
        L2_pulse_sending();
    } while (L2_distance != 2);
    Serial.println("LOE");
}
Serial.print("DC ");
Serial.println(current_distance);
turning = 0;
changes = 0;
step_forward();
stop_motors();
changes = 0;
current_distance += distance;
Serial.print("DC ");
Serial.println(current_distance);
current_x = current_distance * cos(current_angle);
current_y = current_distance * sin(current_angle);
final_x -= current_x;
final_y -= current_y;
current_x = 0;
current_y = 0;
current_distance = 0;
new_point = 0;
delay(1500);
}
}
else
{
    stop_motors();
    if (destination == 0)
    {
        Serial.print("DC ");
        Serial.println(current_distance);
        Serial.println("FP");
        destination = 1;
        new_point = 0;
        count = 0;
        changes = 0;
        current_distance = 0;
        current_angle = final_angle;
    }
}
}

void front_pulse_sending()
{
    int i;

```

```

int j;

R_distance = 0;
L_distance = 0;

for (j = 0; j < num_1; j += 1)
{
    digitalWrite(transmitter_pin, LOW);
    delayMicroseconds(10);
    digitalWrite(transmitter_pin, HIGH);
    delayMicroseconds(10);
}

R = digitalRead(R_receiver_pin);
R_distance = R_distance + R;
L = digitalRead(L_receiver_pin);
L_distance = L_distance + L;
}

void R1_pulse_sending()
{
    int i;
    int j;
    int p[2] = {8.5,9};
    int pulseSelect;

    R1_distance = 0;

    for (i = 0; i < 2; i += 1)
    {
        pulseSelect = p[i];
        for (j = 0; j < num_2; j += 1)
        {
            digitalWrite(transmitter_pin, LOW);
            delayMicroseconds(pulseSelect);
            digitalWrite(transmitter_pin, HIGH);
            delayMicroseconds(pulseSelect);
        }

        R1 = digitalRead(R1_receiver_pin);
        R1_distance = R1_distance + R1;
    }
}

void R2_pulse_sending()
{
    int i;
    int j;
    int p[2] = {9,10};
    int pulseSelect;

    R2_distance = 0;

```

```

for (i = 0; i < 2; i += 1)
{
    pulseSelect = p[i];
    for (j = 0; j < num_2; j += 1)
    {
        digitalWrite(transmitter_pin, LOW);
        delayMicroseconds(pulseSelect);
        digitalWrite(transmitter_pin, HIGH);
        delayMicroseconds(pulseSelect);
    }

    R2 = digitalRead(R2_receiver_pin);
    R2_distance = R2_distance + R2;
}
}

```

```

void L1_pulse_sending()
{
    int i;
    int j;
    int p[2] = {8,10};
    int pulseSelect;

    L1_distance = 0;

    for (i = 0; i < 2; i += 1)
    {
        pulseSelect = p[i];
        for (j = 0; j < num_2; j += 1)
        {
            digitalWrite(transmitter_pin, LOW);
            delayMicroseconds(pulseSelect);
            digitalWrite(transmitter_pin, HIGH);
            delayMicroseconds(pulseSelect);
        }

        L1 = digitalRead(L1_receiver_pin);
        L1_distance = L1_distance + L1;
    }
}

```

```

void L2_pulse_sending()
{
    int i;
    int j;
    int p[2] = {10,11};
    int pulseSelect;

    L2_distance = 0;

    for (i = 0; i < 2; i += 1)
    {
        pulseSelect = p[i];

```

```

    for (j = 0; j < num_2; j += 1)
    {
        digitalWrite(transmitter_pin, LOW);
        delayMicroseconds(pulseSelect);
        digitalWrite(transmitter_pin, HIGH);
        delayMicroseconds(pulseSelect);
    }

    L2 = digitalRead(L2_receiver_pin);
    L2_distance = L2_distance + L2;
}

void distance_counter()
{
    right_status = digitalRead(right_sens_pin);
    left_status = digitalRead(left_sens_pin);

    if (right_status == HIGH)
    {
        if (flag == 2)
        {
            changes += 1;
        }
        flag = 1;
    }
    else
    {
        if (flag == 1)
        {
            changes += 1;
        }
        flag = 2;
    }
}

void sending_messages()
{
    time = millis();
    if (time - previous_time >= 1000)
    {
        previous_time = time;
        Serial.print("DC ");
        Serial.println(current_distance);
    }
}

void forward_move()
{
    right_servo.write(180);
    left_servo.write(63);
}

```

```

void step_forward()
{
  do
  {
    forward_move();
    distance_counter();
    distance = changes * disc_step;
  } while (distance < 20);
}

void backwards_move()
{
  right_servo.write(80);
  left_servo.write(100);
}

void step_back()
{
  do
  {
    backwards_move();
    distance_counter();
    distance = changes * disc_step;
  } while (distance < 5);
}

void turn_right()
{
  right_servo.write(80);
  left_servo.write(80);
}

void step_right()
{
  do
  {
    turn_right();
    distance_counter();
    arc = changes * disc_step;
  } while (arc < 5);
}

void turn_left()
{
  right_servo.write(100);
  left_servo.write(100);
}

void step_left()
{
  do
  {
    turn_left();

```



```

        distance_counter();
        arc = changes * disc_step;
    } while (arc < 4.5);
}

void stop_motors()
{
    right_servo.write(90);
    left_servo.write(90);
}

double arctan(float x)
{
    int n;
    float s;
    double r;
    double p;
    double result;

    const float b = -1;

    result = 0;
    for (n = 0; n < 100; n += 1)
    {
        s = 2*n + 1;
        p = pow(x, s) * pow(b, n);
        r = p / s;
        result += r;
    }
    return result;
}

double angle_calculation(float x, float y)
{
    float t;
    double a;

    if (abs(x) >= abs(y))
    {
        t = y / x;
        a = arctan(t);
        if ((x < 0) && (y > 0))
        {
            a = a + 3.14;
        }
    }
    else
    {
        t = x / y;
        a = arctan(t);
        a = 1.57 - a;
        if ((y < 0) && (x > 0))
        {

```

```

        a = a - 3.14;
    }
}
if ((x <= 0) && (y <= 0))
{
    a = a - 3.14;
}
return a;
}

double distance_calculation(float x, float y)
{
    float s;
    double r;

    s = sq(x) + sq(y);
    r = sqrt(s);
    return r;
}

```

## 4.7 Μηνύματα Εξόδου

Όπως έχουμε τονίσει, η χαρτογράφηση του περιβάλλοντος χώρου είναι ένα από τα σημαντικότερα ζητήματα σε εφαρμογές πλοήγησης κινούμενων ρομποτικών συστημάτων. Μέσα στον κώδικα με τον οποίο προγραμματίστηκε η λειτουργία του ρομπότ, υπάρχουν εντολές εκτύπωσης μηνυμάτων, τα οποία παρέχουν σημαντικές πληροφορίες για την τρέχουσα κατάσταση του συστήματος καθώς επίσης και του χώρου εργασίας του. Η συλλογή και η ανάλυση των πληροφοριών αυτών, θα καταστήσουν εφικτή την κατάρτιση χάρτη του περιβάλλοντος χώρου.

Τα μηνύματα εξόδου αφορούν :

- την συνεχόμενη καταγραφή την διανυόμενης απόστασης, ανά τακτά χρονικά διαστήματα
- τον εντοπισμό ενός εμποδίου, καθώς και την πλευρά ως προς το ρομπότ κατά την οποία εντοπίστηκε
- το τέλος του εμποδίου
- την γωνία και την φορά κατά την οποία το όχημα στρίβει σε ανάλογη περίπτωση

Τα μηνύματα εξόδου τυπώνονται στην σειριακή θύρα του, προσαρμοσμένου πάνω στο ρομπότ, φορητού υπολογιστή (εντολές `Serial.print`), μέσω της οποίας πραγματοποιείται η επικοινωνία με τον `arduino`.

Η καταγραφή της απόστασης γίνεται κάθε 1sec, με τύπωση των αποτελεσμάτων του οδομέτρου σε cm. Τυπώνονται, δηλαδή, τα cm που έχουν καλυφθεί από το όχημα, μετά από ένα σημείο αλλαγής κατεύθυνσης. Για λόγους απλότητας, δεν χρησιμοποιήσαμε κάποιον από τους διαθέσιμους `timers` του `arduino` για την μέτρηση του χρόνου, αλλά υλοποιήσαμε έναν αυτοσχέδιο `timer`, εκμεταλλευόμενοι τις δυνατότητες που παρέχει ο `arduino` με την εντολή `millis()` στην συγκεκριμένη περίπτωση.

Η εντολή `millis()` επιστρέφει σε msec τον χρόνο από τον οποίο ο επεξεργαστής ξεκίνησε να τρέχει το συγκεκριμένο πρόγραμμα. Κατά συνέπεια, παίρνοντας τη διαφορά μεταξύ της τρέχουσας τιμής του μετρούμενου χρόνου και της τελευταίας στιγμής που ξεπεράστηκε η χρονική διαφορά του ενός δευτερολέπτου, υλοποιούμε έναν timer για την μέτρηση 1sec, ο κώδικας του οποίου παρουσιάζεται παρακάτω.

```
void sending_messages()
{
  time = millis();
  if (time - previous_time >= 1000)
  {
    previous_time = time;
    Serial.print("distance covered = ");
    Serial.print(current_distance);
    Serial.println(" cm");
  }
}
```

Στον παραπάνω κώδικα, οι εντολές εκτύπωσης περιέχουν ολοκληρωμένα τα μηνύματα (και όχι σε κωδικοποιημένη μορφή, όπως θα δούμε στη συνέχεια), προκειμένου να υπάρχει μεγαλύτερη κατανόηση της εργασίας που επιτελεί η συγκεκριμένη ρουτίνα.

Στην περίπτωση κατά την οποία το ρομπότ εντοπίσει ένα εμπόδιο, τότε το σύστημα ενημερώνεται αμέσως για τον εντοπισμό του εμποδίου και την πλευρά στην οποία βρίσκεται, το ρομπότ σταματάει και καταγράφεται η απόσταση την οποία διένυσε μέχρι να σταματήσει.

Όταν το ρομπότ περιστρέφεται για να αλλάξει κατεύθυνση, τότε καταγράφεται η φορά προς την οποία στρίβει (προς τα δεξιά ή προς τα αριστερά), καθώς και η γωνία περιστροφής σε μοίρες (όλοι οι υπολογισμοί γίνονται, όπως έχουμε δει σε rad, όμως γίνεται μετατροπή σε μοίρες για την καλύτερη κατανόηση των μηνυμάτων).

Τέλος, στην περίπτωση όπου το ρομπότ έχει εντοπίσει ένα εμπόδιο και αφού έχει ήδη κινηθεί παράλληλα με αυτό, τότε θα εμφανιστεί το ανάλογο μήνυμα όταν ξεπεραστεί το εμπόδιο, ενώ θα ακολουθήσει η καταγραφή της απόστασης που καλύφθηκε μέχρι να αναγνωριστεί το τέλος του εμποδίου. Το ρομπότ, όπως έχουμε ορίσει, θα συνεχίσει την κίνησή του για επιπλέον 20cm (εντολή `step_forward`) όπου θα σταματήσει και θα καταγραφεί η νέα απόσταση που έχει διανύσει, ώστε να είναι κάθε φορά εύκολος ο υπολογισμός των σημείων αλλαγής κατεύθυνσης.

Παρακάτω παραθέτουμε την τύπωση μηνυμάτων στην σειριακή θύρα του φορητού υπολογιστή, για ένα παράδειγμα προσομοίωσης της λειτουργίας του συστήματος. Το δοθέν σημείο προορισμού (σε cm) είναι το σημείο (160, -140), ενώ το ρομπότ είναι αρχικά τοποθετημένο στην αρχή των αξόνων, προσανατολισμένο κατά τη θετική φορά του άξονα X.

Στην πραγματικότητα, για λόγους κατανόησης, έχουμε θεωρήσει ότι αρχικά το ρομπότ είναι τοποθετημένο στην αρχή των αξόνων κάθετα και όχι οριζόντια, είναι δηλαδή προσανατολισμένο κατά τη θετική φορά του άξονα Y, ενώ οι υπολογισμοί συνεχίζονται αφού πρώτα περιστρέψουμε το

σύστημα συντεταγμένων. Στην συγκεκριμένη περίπτωση δηλαδή, το δοθέν σημείο προορισμού είναι το (140, 160) το οποίο όμως (όπως έχουμε ορίσει) μεταφράζεται στο συμμετρικό ως προς τον άξονα X σημείο (160, -140), με αποτέλεσμα να θεωρηθεί ότι το ρομπότ είναι πλέον αρχικά προσανατολισμένο κατά τη θετική φορά του άξονα X. Η περιστροφή του συστήματος συντεταγμένων γίνεται για διευκόλυνση στον υπολογισμό των γωνιών όταν το ρομπότ αλλάζει κατεύθυνση.

Στην έξοδο της σειριακής θύρας λοιπόν (επιλογή πεδίου Serial Monitor, στο μενού του περιβάλλοντος προγραμματισμού του arduino), εμφανίζονται για το συγκεκριμένο παράδειγμα, τα παρακάτω μηνύματα.

```
initial position
right turn for 41.19 degrees
distance covered = 0 cm
distance covered = 0.96 cm
distance covered = 27.82 cm
distance covered = 56.61 cm
left object found
distance covered = 75.80 cm
right turn for 53.88 degrees
distance covered = 0 cm
distance covered = 0.96 cm
distance covered = 26.86 cm
distance covered = 53.73 cm
left object no more
distance covered = 76.76 cm
distance covered = 96.90 cm
left turn for 98.37 degrees
distance covered = 0 cm
distance covered = 0.96 cm
distance covered = 27.82 cm
distance covered = 54.69 cm
distance covered = 81.55 cm
distance covered = 108.42 cm
final position
```

Και στην περίπτωση αυτή, τα μηνύματα παρουσιάζονται ολοκληρωμένα για την καλύτερη κατανόηση της λειτουργίας.

Χρησιμοποιώντας τα παραπάνω μηνύματα εξόδου, μπορούμε να προχωρήσουμε στην ανάλυση της διαδρομής που ακολουθήθηκε, καθώς επίσης και στην προσεγγιστική περιγραφή του περιβάλλοντος χώρου.

Το ρομπότ, λοιπόν, ξεκινάει από αρχική θέση την αρχή των αξόνων, προσανατολισμένο πάνω στον άξονα X. Αρχικά, στρίβει δεξιά κατά  $41.19^\circ$ , άρα η νέα κατεύθυνση έχει γωνία  $-41.19^\circ$ . Το ρομπότ καλύπτει απόσταση 75.8cm και σταματάει, επειδή εντοπίστηκε ένα αντικείμενο από την αριστερή του πλευρά. Κατά συνέπεια, το σημείο A (βλ. σχήμα 4.6.1) στο οποίο σταμάτησε το ρομπότ προκειμένου να αλλάξει κατεύθυνση έχει τις εξής συντεταγμένες :

$$x_A = 75.8 \cos(-41.19) = 56.85 \text{ cm}$$

$$y_A = 75.8 \sin(-41.19) = -50 \text{ cm}$$

Το εμπόδιο εντοπίστηκε από την αριστερή πλευρά του οχήματος, οπότε ανιχνεύτηκε πρώτα από τον αισθητήρα L. Σύμφωνα με το διάγραμμα του αισθητήρα L που παρουσιάσαμε στην παράγραφο 4.2, ο συγκεκριμένος αισθητήρας εντοπίζει ένα (ανοιχτόχρωμο, όπως έχουμε θεωρήσει) αντικείμενο σε απόσταση της τάξης των 22cm. Αυτό σημαίνει ότι αν επεκτείνουμε την απόσταση των 75.8cm, που κάλυψε το ρομπότ μέχρι να σταματήσει, κατά 22cm, παίρνουμε τις συντεταγμένες ενός σημείου του εμποδίου, οι οποίες και είναι :

$$x_E = (75.8 + 22) \cos(-41.19) = 73.35 \text{ cm}$$

$$y_E = (75.8 + 22) \sin(-41.19) = -64.5 \text{ cm}$$

Στην συνέχεια το ρομπότ στρίβει προς τα δεξιά κατά  $53.88^\circ$ , δηλαδή περιστρέφεται κατά γωνία  $-53.88^\circ$ . Η νέα κατεύθυνση λοιπόν του οχήματος είναι  $(-41.19^\circ) + (-53.88^\circ) = -95^\circ$ . Η γωνία αυτή ταυτίζεται με τη γωνία που σχηματίζει και το εμπόδιο στον χώρο, καθώς το ρομπότ έστριψε όσο χρειαζόταν ώστε να παραλληλιστεί με το εμπόδιο.

Το σημείο A έχει γίνει σημείο αναφοράς και συνεχίζεται να καταγράφεται η απόσταση η οποία διανύεται. Αφού καλυφθεί απόσταση 76.76cm από το σημείο A, το σύστημα ενημερώνεται για το τέλος των ορίων του εμποδίου. Το σημείο όπου έγινε η ανάλογη ενημέρωση είναι το εξής:

$$x'_B = 76.76 \cos(-95) = -6.9 \text{ cm}$$

$$y'_B = 76.76 \sin(-95) = -76 \text{ cm}$$

Οι παραπάνω συντεταγμένες όμως, είναι ως προς το σημείο A, άρα οι συντεταγμένες του σημείου B είναι :

$$x_B = x_{B'} + x_A = -6.9 + 56.85 = 49.95 \text{ cm}$$

$$y_B = y_{B'} + y_A = -76 - 50 = -126 \text{ cm}$$

Ανάλογα με τη ζώνη στην οποία είδε από το σημείο B (για τελευταία φορά) το εμπόδιο ο αισθητήρας L2, μπορούμε να προσεγγίσουμε το σημείο που βρίσκεται το όριο του συγκεκριμένου αντικειμένου.

Εν συνεχεία, το ρομπότ συνεχίζει στην ίδια κατεύθυνση για επιπλέον 20cm, όπου και σταματάει, έχοντας καλύψει συνολικά 96.9cm από το σημείο A. Το σημείο Γ, λοιπόν, όπου και πραγματοποιηθεί η επόμενη αλλαγή κατεύθυνσης, έχει ως προς το A τις παρακάτω συντεταγμένες :

$$x'_\Gamma = 96.9 \cos(-95) = -8.7 \text{ cm}$$

$$y'_\Gamma = 96.9 \sin(-95) = -95.9 \text{ cm}$$

Άρα, ως προς την αρχή των αξόνων, το σημείο Γ είναι :

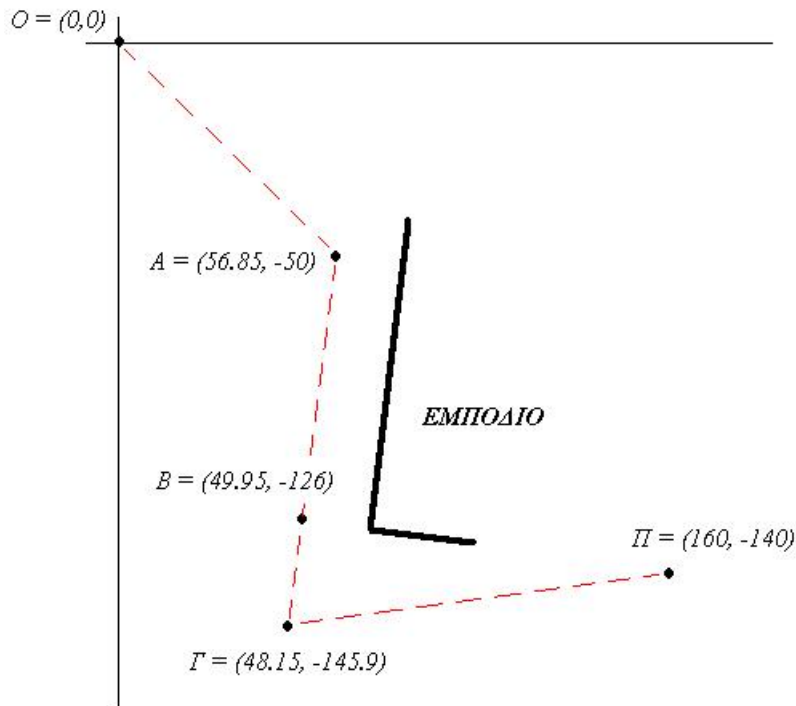
$$x_\Gamma = x'_\Gamma + x_A = -8.7 + 56.85 = 48.15 \text{ cm}$$

$$y_\Gamma = y'_\Gamma + y_A = -95.9 - 50 = -145.9 \text{ cm}$$

Το σημείο Γ γίνεται το νέο σημείο αναφοράς.

Τέλος, το ρομπότ στρίβει αριστερά κατά  $98.37^\circ$  με αποτέλεσμα η νέα κατεύθυνση να έχει γωνία  $-95^\circ + 98.37^\circ = 3.37^\circ$ . Το όχημα κινείται σε αυτήν την κατεύθυνση κατά 108.42cm, όπου και καταλήγει στο σημείο προορισμού Π, το οποίο και επαληθεύεται.

Παρακάτω, περιγράφουμε σχηματικά όλη την παραπάνω διαδικασία, αποδεικνύοντας την επάρκεια των πληροφοριών που παρέχουν τα μηνύματα εξόδου, στην προσπάθεια κατανόησης του περιβάλλοντος χώρου.



Σχήμα 4.7.1. Παράδειγμα προσομοίωσης – Ανάλυση διαδρομής και προσεγγιστικός σχεδιασμός του χώρου.

Φυσικά, είναι δεδομένο ότι η θέση του παραπάνω εμποδίου στον χώρο εμπεριέχει σφάλμα, καθώς :

- ο υπολογισμός του ορίου του εμποδίου είναι προσεγγιστικός, αφού δεν γνωρίζουμε ακριβή απόσταση από τον αισθητήρα L2, παρά μόνο τη ζώνη στην οποία το είδε ο αισθητήρας για τελευταία φορά (παρόλα αυτά, θεωρούμε ότι από το σημείο εντοπισμού του εμποδίου και έπειτα, το ρομπότ έχει κινηθεί απολύτως παράλληλα με το εμπόδιο)
- όπως έχουμε αναφέρει, οι αισθητήρες υπέρυθρων επηρεάζονται από παράγοντες όπως είναι ο φωτισμός του χώρου, με αποτέλεσμα οι τιμές που μας παρέχουν τα διαγράμματα τις παραγράφου 4.2 να μην είναι δεδομένες, αλλά να διαφέρουν ανάλογα με τις συνθήκες που επικρατούν
- δεν μπορούμε να γνωρίζουμε τη φύση του εμποδίου (χρώμα, υφή)
- παρά το γεγονός ότι ένα IR LED παρουσιάζει έντονη κατευθυντικότητα, η γωνία εκπομπής της υπέρυθρης ακτινοβολίας μπορεί να έχει ως αποτέλεσμα την εισαγωγή σφάλματος στην ακριβή θέση όπου εντοπίστηκε το εμπόδιο.

Όπως αναφέραμε, τα μηνύματα εξόδου παρουσιάστηκαν προηγουμένως σε περιγραφική μορφή. Ο μελλοντικός χρήστης του συστήματος όμως, θα χρησιμοποιήσει όλα τα δεδομένα εξόδου με σκοπό τον προγραμματισμό (σε γραφικό περιβάλλον) της αυτόματης κατάρτισης χάρτη του χώρου εργασίας. Για τον λόγο αυτόν, τα μηνύματα εξόδου (όπως είδαμε στον κώδικα της προηγούμενης παραγράφου), έχουν γραφεί σε κωδικοποιημένη μορφή η οποία και περιγράφεται αναλυτικά στον επόμενο πίνακα.

Λειτουργία	Μήνυμα	Επεξήγηση	Παράμετρος
Αρχική θέση – Επιβεβαίωση λήψης δεδομένων	IP	Initial Position	
Τελική θέση – Αίτηση αποστολής νέων δεδομένων	FP	Final Position	
Καταγραφή της απόστασης	DC	Distance Covered	Απόσταση (cm)
Στροφή δεξιά	RT	Right Turn	Γωνία (μοίρες)
Στροφή αριστερά	LT	Left Turn	Γωνία (μοίρες)
Εντοπισμός εμποδίου - δεξιά	ROF	Right Object Found	
Εντοπισμός εμποδίου - αριστερά	LOF	Left Object Found	
Τέλος δεξιού εμποδίου	ROE	Right Object End	
Τέλος αριστερού εμποδίου	LOE	Left Object End	
Τέλος ορίων του χώρου	EOS	End Of Space	

*Πίνακας 4.7.1*

Παρακάτω λοιπόν, παρουσιάζεται η προσομοίωση του ίδιου παραδείγματος, με εκτύπωση των μηνυμάτων σε κωδικοποιημένη μορφή αυτή τη φορά, όπως δηλαδή θα εμφανίζονται κατά την διάρκεια της λειτουργίας του συστήματος.

```

IP
RT 41.19
DC 0
DC 0.00
DC 25.91
DC 53.73
DC 82.51
LOF
DC 92.11
RT 46.18
DC 0
DC 0.96
DC 26.86
DC 55.65
LOE
DC 69.08
DC 89.23
LT 93.82
DC 0

```

DC 0.96  
DC 26.86  
DC 55.65  
DC 85.39  
FP

## 4.8 Είσοδος Δεδομένων

Τα παραπάνω μηνύματα εξόδου, εκτός από τις πληροφορίες που παρέχουν για τον περιβάλλον χώρο και την διαδρομή που ακολουθήθηκε, έχουν και ως σκοπό την κατάθεση αίτησης στο σύστημα (ή στον χρήστη) για αποστολή νέων δεδομένων (μήνυμα FP), καθώς επίσης και την επιβεβαίωση για τη λήψη των δεδομένων αυτών (μήνυμα IP).

Καταρχήν, το μήνυμα IP του πίνακα 4.7.1 θα εμφανιστεί όταν ολοκληρωθεί η λήψη των νέων κάθε φορά δεδομένων. Η συγκεκριμένη λειτουργία, έχει κυρίως σημασία για την περίπτωση όπου αναβαθμίσουμε το σύστημα, προσθέτοντας και δυνατότητα ασύρματης αποστολής των δεδομένων εισόδου. Στην περίπτωση αυτήν, αν για παράδειγμα το ρομπότ βρίσκεται σε σχετικά μεγάλη απόσταση, είναι πιθανό να μην ληφθούν τα δεδομένα που στέλνονται, και κατά συνέπεια το μήνυμα IP έχει στόχο να ενημερώσει τον χρήστη ότι τα δεδομένα που έστειλε ελήφθησαν σωστά από το σύστημα.

Έτσι κι αλλιώς πάντως, αν δεν ληφθούν σωστά τα νέα δεδομένα ή αν γίνει κάποιο λάθος κατά την εισαγωγή τους, το σύστημα δεν επιτρέπει εκκίνηση του οχήματος. Όπως είναι προφανές λοιπόν, η εμφάνιση του συγκεκριμένου μηνύματος υποδηλώνει την έναρξη της κίνησης του ρομπότ, από την αρχική θέση που ορίσαμε.

Από την άλλη πλευρά, το μήνυμα FP έχει ιδιαίτερη σημασία, καθώς υποδηλώνει την άφιξη στο σημείο προορισμού και καταθέτει αίτηση στο σύστημα ή στον χρήστη, για εισαγωγή νέων δεδομένων (νέου σημείου προορισμού).

Όπως παρουσιάσαμε στην εισαγωγή του Μέρους Β, η τελική μορφή της λειτουργίας του συστήματος, εμπεριέχει την λήψη ακουστικών μετρήσεων μετά την άφιξη στο σημείο προορισμού. Βέβαια, τα σημεία προορισμού δεν είναι απαραίτητα και σημεία λήψης ακουστικών μετρήσεων. Τα τελικά σημεία που ορίζουμε κάθε φορά, μπορεί να αποτελούν ενδιάμεσους προορισμούς, προκειμένου να ακολουθηθεί μία επιθυμητή διαδρομή. Στην πραγματικότητα λοιπόν, το μήνυμα FP δηλώνει την άφιξη σε ένα τελικό σημείο, και μπορεί να θεωρηθεί ως εντολή έναρξης είτε της λήψης των μετρήσεων, είτε της εισαγωγής των νέων δεδομένων.

Στην παρούσα φάση πάντως, το μήνυμα FP σημαίνει την αίτηση για εισαγωγή νέου σημείου προορισμού. Για την επεξήγηση της διαδικασίας, επαναφέρουμε την μορφή του πίνακα δεδομένων εισόδου που παρουσιάσαμε στην σελίδα 45.



No.	Αρχικό Σημείο	Τελικό Σημείο
1	$(x_0, y_0)$	$(x_1, y_1)$
2	$(x_1, y_1)$	$(x_2, y_2)$
3	$(x_2, y_2)$	$(x_3, y_3)$
.	.	.
.	.	.
N	$(x_{N-1}, y_{N-1})$	$(x_N, y_N)$

Πίνακας 4.8.1

Αρχικά λοιπόν, εισάγονται σημείο έναρξης  $(x_0, y_0)$  και σημείο προορισμού  $(x_1, y_1)$ , και μόλις ολοκληρωθεί σωστά η εισαγωγή των δεδομένων αυτών, εμφανίζεται το μήνυμα IP και αρχίζει η κίνηση του οχήματος. Το ρομπότ πλοηγείται αυτόνομα στον χώρο και προσεγγίζει το σημείο προορισμού. Όταν το ρομπότ φτάσει στο τελικό σημείο  $(x_1, y_1)$ , εμφανίζεται το μήνυμα FP και το σύστημα αναμένει την εισαγωγή του νέου σημείου προορισμού. Εισάγεται ως νέο σημείο προορισμού το σημείο  $(x_2, y_2)$ , και με τον ίδιο τρόπο το ρομπότ ξεκινάει την πλοήγησή του με σκοπό να φτάσει στο σημείο που ορίσαμε. Η ίδια διαδικασία ακολουθείται επαναληπτικά, μέχρις ότου το ρομπότ φτάσει στο σημείο N που έχουμε ορίσει ως τελικό προορισμό.

Στο σημείο αυτό, πρέπει να αναφέρουμε τον ακριβή τρόπο εισαγωγής των δεδομένων εισόδου. Σύμφωνα με τον τρόπο που γράφτηκε ο κώδικας λειτουργίας του ρομπότ, η εισαγωγή αρχικού και τελικού σημείου πραγματοποιείται σε μία γραμμή εντολών ως εξής :

```
"x0 y0 x1 y1 "
```

δηλαδή εισάγονται διαδοχικά τετμημένα – τεταγμένα του αρχικού σημείου, και εν συνεχεία τετμημένα – τεταγμένα του τελικού σημείου. Όλες οι παραπάνω τιμές είναι σε cm. Μεταξύ όλων των τιμών αφήνουμε κενό (SPACE), ενώ απαιτείται ιδιαίτερη προσοχή καθώς πρέπει να πληκτρολογηθεί χαρακτήρας κενού και στο τέλος, μετά δηλαδή την τελευταία τιμή, και εν συνεχεία να πατηθεί το πλήκτρο SEND, για αποστολή των δεδομένων εισόδου. Αν δεν αφήσουμε κενό μετά την τελευταία τιμή, δεν θα ληφθούν τα δεδομένα που στείλαμε και δεν θα εμφανιστεί το μήνυμα IP για επιβεβαίωση.

## ΚΕΦΑΛΑΙΟ 5

### ΣΦΑΛΜΑΤΑ ΚΑΙ ΒΕΛΤΙΩΣΕΙΣ

#### 5.1 Σφάλματα

##### 5.1.1 Εισαγωγή

Τα σημαντικότερα σφάλματα που εισάγονται κατά την κίνηση του ρομπότ, οφείλονται κυρίως σε κάποιες κατασκευαστικές λεπτομέρειες. Τα σφάλματα που εισάγονται κατά την κίνηση έχουν σαν αποτέλεσμα το όχημα να μην βρεθεί ακριβώς στο ζητούμενο σημείο, αλλά η τελική του θέση να έχει μία μικρή απόκλιση από το σημείο αυτό. Μετά από δοκιμές και μετρήσεις που έγιναν καταλήξαμε στο συμπέρασμα ότι η απόκλιση αυτή είναι της τάξης του 1 έως 2%, όμως σε κάποιες περιπτώσεις μπορεί να είναι αρκετά μεγαλύτερη και να φτάσει ακόμα και το +/-5%. Ακόμα και στην χειρότερη περίπτωση πάντως, η απόκλιση δεν είναι ιδιαίτερα μεγάλη και στην παρούσα φάση, το ρομπότ λειτουργεί ικανοποιητικά. Πάντως, όπως θα περιγράψουμε στην επόμενη παράγραφο, υπάρχουν πάρα πολλά περιθώρια βελτίωσης, που μπορούν να ελαχιστοποιήσουν σε πολύ μεγάλο βαθμό την παραπάνω απόκλιση.

##### 5.1.2 Είδη Σφαλμάτων

#### ΣΦΑΛΜΑ ΠΕΡΙΣΤΡΕΦΟΜΕΝΟΥ ΤΡΟΧΟΥ

Το σημαντικότερο σφάλμα εισάγεται από τον βοηθητικό περιστρεφόμενο τροχό (caster wheel), και οφείλεται στο γεγονός ότι ο τροχός είναι έκκεντρος. Το συγκεκριμένο σφάλμα δημιουργείται κυρίως κατά τις αλλαγές κατεύθυνσης του οχήματος. Για παράδειγμα, την στιγμή που το όχημα έχει ήδη πραγματοποιήσει μια επιτόπια στροφή, ο πίσω τροχός θα είναι στραμμένος προς την τελευταία κατεύθυνση του οχήματος, με αποτέλεσμα, όταν ξεκινήσει ξανά το ρομπότ και μέχρι ο τροχός να έρθει στη ζητούμενη κατεύθυνση, να έχει δημιουργήσει μια σημαντική απόκλιση στην νέα πορεία του οχήματος. Η απόκλιση αυτή αν και μπορεί να εκτιμηθεί σε έναν βαθμό, δεν μπορεί να υπολογιστεί με ακρίβεια και μόνο μια αλλαγή στην κατασκευή του οχήματος μπορεί να διορθώσει το συγκεκριμένο πρόβλημα.

Ένα σφάλμα που επίσης μπορεί να εισάγει ο περιστρεφόμενος τροχός, είναι αυτό που δημιουργείται εξαιτίας ορισμένων ανωμαλιών στην επιφάνεια του δαπέδου (όπως για παράδειγμα είναι οι αυλακώσεις μεταξύ πλακών). Μια ανωμαλία του εδάφους μπορεί προς στιγμήν να στέψει

τον πίσω τροχό, αλλάζοντας, έστω σε πολύ μικρό βαθμό, την κατεύθυνση του οχήματος, δημιουργώντας μία απόκλιση στην τελική θέση του ρομπότ, η οποία για μεγάλες αποστάσεις μπορεί να αποδειχθεί σημαντική. Η απόκλιση αυτή είναι μη υπολογίσιμη καθώς οι ανωμαλίες του δαπέδου αποτελούν αστάθμητο παράγοντα.

## ΣΦΑΛΜΑ ΕΛΑΣΤΙΚΟΥ

Ένα ακόμα σφάλμα κατασκευαστικών αιτιών, είναι αυτό που οφείλεται στην συμπίεση του ελαστικού των 2 μπροστινών τροχών και κατά συνέπεια, ουσιαστικά στην λανθασμένη μέτρηση της διαμέτρου των τροχών. Όπως περιγράψαμε στο προηγούμενο κεφάλαιο, οι τροχοί που επιλέχθηκαν περιβάλλονται από λάστιχο το οποίο είναι τέτοιας φύσεως (είναι μαλακό και συμπιέζεται λόγω του βάρους του ρομπότ) προκειμένου να ελαχιστοποιείται όσο το δυνατόν περισσότερο η κύλιση με ολίσθηση. Όπως έχουμε ήδη αναφερθεί κατά την παρουσίαση των διαστάσεων της κατασκευής, η διάμετρος των 2 μπροστινών τροχών έχει μετρηθεί 11cm. Όμως, κατά την κίνηση του οχήματος, τα ελαστικά συμπιέζονται εξαιτίας του συνολικού βάρους της κατασκευής, με αποτέλεσμα η διάμετρος των τροχών να είναι στην πραγματικότητα ελαφρώς μικρότερη από αυτήν που έχει μετρηθεί, κατά ένα διάστημα που είναι λίγο μικρότερο από 1mm. Αυτή η διαφορά προσδίδει ένα μικρό σφάλμα και στο βήμα του κάθε τμήματος του κωδικοποιητή (το οποίο έχει μετρηθεί 0,95945cm) και άρα δημιουργείται ένα σφάλμα στην τελική καλυπτόμενη απόσταση, που είναι της τάξης του 0,5%. Το συγκεκριμένο σφάλμα δεν αποτελεί σημαντικό πρόβλημα καθώς μπορεί να προβλεφθεί και να ληφθεί υπόψη από προηγουμένως.

## ΣΦΑΛΜΑ ΣΤΡΟΓΓΥΛΟΠΟΙΗΣΗΣ

Ένα σφάλμα το οποίο είναι άξιο αναφοράς, αν και είναι ιδιαίτερα μικρό, είναι αυτό που οφείλεται στην στρογγυλοποίηση κάποιων μετρήσεων. Όπως αναλύσαμε προηγουμένως, για κάθε τμήμα του κωδικοποιητή διανύεται απόσταση της τάξης των 0,95945cm. Ο αριθμός αυτός προέρχεται από τον εξής υπολογισμό :

$$step = \frac{2\pi r}{n}$$

όπου

$2\pi r$  η περιφέρεια του τροχού ακτίνας  $r = d / 2 = 11 / 2 = 5,5cm$  ( $d$  η διάμετρος του τροχού)

$n = 36$  ο αριθμός των τμημάτων του κωδικοποιητή.

Άρα,

$$step = \frac{2\pi r}{n} = \frac{2 \times 3,14 \times 5,5}{36} = \frac{34,54}{36} = 0,959444444$$

και κατά συνέπεια παίρνουμε προσεγγιστικά μήκος βήματος 0,95945cm.

Όπως παρατηρούμε, στον παραπάνω υπολογισμό εμπεριέχεται και ο αριθμός  $\pi = 3,14$  ο οποίος προέρχεται επίσης από στρογγυλοποίηση και κατά συνέπεια, προσθέτει και αυτός κάποια ανακρίβεια (έστω και απειροελάχιστη) στον υπολογισμό του step.

Το παραπάνω σφάλμα αθροίζεται συνεχώς στην καλυπτόμενη απόσταση, και συνεπώς είναι πιθανό να αποδειχθεί σημαντικό. Το σφάλμα στρογγυλοποίησης όμως, όντας επαναλαμβανόμενο, μπορεί να συνυπολογιστεί στην τελική απόσταση που διανύει το ρομπότ. Παρόλα αυτά, δεν λάβαμε υπόψη το συγκεκριμένο σφάλμα στον υπολογισμό της απόστασης, καθώς (όπως και το σφάλμα που δημιουργείται από την συμπίεση του ελαστικού) δεν έχει μέγεθος συγκρίσιμο με τα υπόλοιπα είδη σφαλμάτων, και ειδικά με την απόκλιση που οφείλεται στον περιστρεφόμενο τροχό. Είναι όμως σημαντικό να γίνει αναφορά σε όλα τα είδη σφαλμάτων, από την στιγμή μάλιστα που συνεισφέρουν (έστω και σε διαφορετικό βαθμό) στην συνολική απόκλιση τα τελικής θέσης του ρομπότ.

## ΣΦΑΛΜΑ ΚΩΔΙΚΟΠΟΙΗΤΩΝ

Ένα ακόμα ιδιαίτερα αξιοσημείωτο σφάλμα είναι αυτό που οφείλεται στη διαφορά της ζητούμενης τελικής απόστασης από την μετρούμενη απόσταση. Η τελευταία, όπως έχουμε περιγράψει, προκύπτει από τον πολλαπλασιασμό μεταξύ του αριθμού των εναλλαγών του κωδικοποιητή και του μήκους του κάθε τμήματός του (*step*). Κατά συνέπεια, η απόσταση που μετράμε είναι ένα ακέραιο πολλαπλάσια του αριθμού  $step = 0,95945\text{cm}$ , και άρα η καλυπτόμενη απόσταση θα διαφέρει από τη ζητούμενη κατά ένα διάστημα από 0 έως  $0,95945\text{cm}$ . Θεωρητικά, για μικρές αποστάσεις, αυτό το διάστημα ίσως είναι σχετικά σημαντικό. Το σφάλμα αυτό όμως, εισάγεται μόνο μία φορά, στο τέλος της διαδρομής που το ρομπότ έχει να διανύσει, και για τις αποστάσεις της παρούσας εφαρμογής, είναι πρακτικά αμελητέο.

Επίσης, παρά τις πολλές προσπάθειες για την επίτευξη συντονισμού των 2 τροχών, αποδείχθηκε αδύνατο να πετύχουμε τον απόλυτο συγχρονισμό, με αποτέλεσμα να δημιουργείται μια επιπλέον (μικρή) απόκλιση στην τελική θέση του ρομπότ.

## ΣΦΑΛΜΑ ΟΛΙΣΘΗΣΗΣ

Το σημαντικότερο είδος σφάλματος το οποίο συναντάται συνήθως σε τέτοιου είδους εφαρμογές, είναι το σφάλμα που εισάγεται εξαιτίας της κίνησης με ολίσθηση. Η συχνή αναφορά του συγκεκριμένου όρου στην παρούσα διπλωματική εργασία, έχει να κάνει με το γεγονός ότι η λογική του οδομέτρου έχει νόημα μόνο αν το όχημα κινείται με παράλληλη κύλιση χωρίς ολίσθηση των 2 κινητήριων τροχών (όπου και είναι προσαρμοσμένοι οι encoder discs). Η μέτρηση της διανυόμενης απόστασης επιτυγχάνεται, όπως έχουμε εκτενώς αναφερθεί, με την μέτρηση των εναλλαγών του χρώματος κατά την περιστροφή των κωδικοποιητών, δηλαδή κατά την περιστροφή των τροχών. Αν οι τροχοί ολισθήσουν, δηλαδή κινηθούν χωρίς να περιστραφούν, τότε το όχημα θα έχει κινηθεί χωρίς όμως να έχουν μετρηθεί εναλλαγές του χρώματος των κωδικοποιητών, και αυτό έχει ως αποτέλεσμα την εισαγωγή σφάλματος. Η χρήση, όμως, τραχέως ελαστικού σε συνδυασμό με τη συνεχόμενη διακοπή και επανέναρξη της κίνησης των τροχών (που συμβαίνει μέσα σε πολύ μικρό χρονικό διάστημα κατά τη διαδικασία του συντονισμού των τροχών), μειώνει στο μέγιστο το σφάλμα στο οποίο αναφερόμαστε. Μοναδική εξαίρεση αποτελεί η αντίθετη περίπτωση, δηλαδή το «σπινιάρισμα» κατά την εκκίνηση του οχήματος (κατά το οποίο πραγματοποιείται μόνο περιστροφική και όχι μεταφορική κίνηση), όπου γίνεται καταμέτρηση περισσότερων εναλλαγών, και κατά συνέπεια μεγαλύτερης απόστασης από αυτήν που στην πραγματικότητα έχει διανύσει το ρομπότ.

## 5.2 Αντιμετώπιση Σφαλμάτων

### 5.2.1 Εισαγωγή

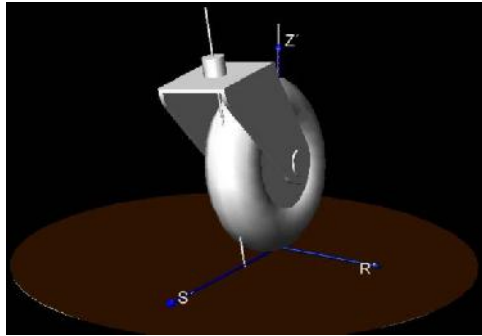
Βασικό μέλημα για την μελλοντική βελτίωση και τελειοποίηση της λειτουργίας του ρομπότ αποτελεί η αντιμετώπιση των σφαλμάτων που αναλύθηκαν στην προηγούμενη παράγραφο. Κάποια από τα σφάλματα που περιγράφηκαν προηγουμένως, οφείλονται σε απρόβλεπτους παράγοντες (είναι *μη-συστηματικά*) και είναι δύσκολο να αντιμετωπιστούν. Αυτού του τύπου τα σφάλματα όμως, δεν θα μας απασχολήσουν ιδιαίτερα, καθώς δεν επηρεάζουν σημαντικά την πλοήγηση του συστήματος.

Τα περισσότερα σφάλματα όμως, ακόμα και αν δεν μπορούν να υπολογιστούν με ακρίβεια, είναι επαναλαμβανόμενα (*συστηματικά*). Τα σφάλματα του συγκεκριμένου τύπου μπορούν να αντιμετωπιστούν με μία καλή στατιστική ανάλυση. Καταγράφοντας σε πίνακες δεδομένων μετρήσεις από διάφορες δοκιμές πλοήγησης σε ένα δεδομένο περιβάλλον, μας δίνεται η δυνατότητα να συγκρίνουμε, για παράδειγμα, διαφορετικές δοκιμές της ίδιας διαδρομής, και να εξάγουμε μέσους όρους σφαλμάτων που οφείλονται σε παράγοντες όπως η κύλιση με ολίσθηση, οι στρογγυλοποιήσεις παραμέτρων κ.α. Οι μέσοι όροι αυτοί, μπορούν να προστίθενται επαναληπτικά ανά δεδομένα διαστήματα, ελαχιστοποιώντας κατά τον τρόπο αυτόν, τα συστηματικά σφάλματα. Η ανάλυση μπορεί αργότερα να επεκταθεί και σε διαφορετικές συνθήκες, εξάγοντας χρήσιμα συμπεράσματα και βελτιώνοντας σημαντικά τη λειτουργία του συστήματος.

Παρόλα αυτά, μια στατιστική ανάλυση δεν θα μπορούσε να λύσει πολλά προβλήματα στην παρούσα φάση και δεν θα ήταν ιδιαίτερα ουσιαστική, καθώς τα σφάλματα του παραπάνω τύπου, δεν συμβάλουν ιδιαίτερα στην συνολική απόκλιση, και είναι σχετικά αμελητέα σε σύγκριση με σφάλματα που οφείλονται σε κατασκευαστικούς λόγους, και ειδικά στην απόκλιση που δημιουργεί ο περιστρεφόμενος τροχός (caster). Πρωτεύον μέλημα λοιπόν, είναι η αντικατάσταση του βοηθητικού περιστρεφόμενου τροχού που βρίσκεται στο πίσω μέρος της κατασκευής, και η εύρεση ενός τρόπου, ώστε να υπάρχει η δυνατότητα να ακολουθηθεί, οποιαδήποτε στιγμή, ακριβώς η πορεία στην οποία οδηγούν οι 2 κινητήριοι τροχοί.

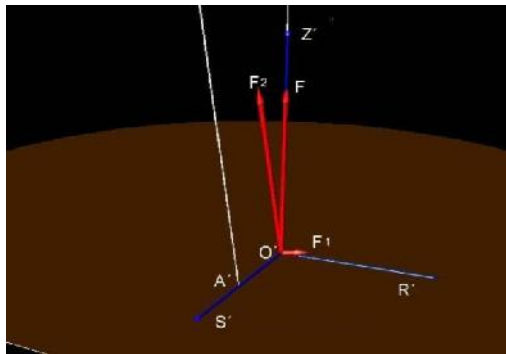
### 5.2.2 Αντικατάσταση Περιστρεφόμενου Τροχού

Στην παράγραφο αυτήν, θα εξετάσουμε τον βοηθητικό περιστρεφόμενο τροχό (caster wheel) και θα θεωρήσουμε ότι αποτελεί πηγή συστηματικών σφαλμάτων για τους λόγους που θα περιγράψουμε παρακάτω. Μετά από προσεκτική παρατήρηση και ανάλυση, γίνεται σαφές ότι τα συστηματικά σφάλματα οδομετρίας που προέρχονται από τη χρήση του βοηθητικού τροχού, εμφανίζονται εξαιτίας του γεγονότος ότι ο άξονας περιστροφής του τροχού (όχι ο άξονας του τροχού) δεν είναι απολύτως κάθετος, όπως θα έπρεπε να είναι (βλ. σχήμα 5.2.2.1).

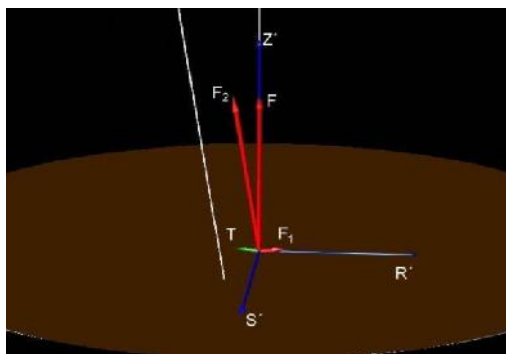


Σχήμα 5.2.2.1. Λόγω κατασκευαστικών ατελειών ο άξονας του caster δεν είναι κάθετος στο έδαφος.

Η αντίδραση του εδάφους  $F$  στον τροχό αναλύεται σε 2 συνιστώσες, τις  $F_1$  και  $F_2$ , όπως βλέπουμε στο σχήμα 5.2.2.2. Η δύναμη  $F_2$  είναι παράλληλη στον άξονα του caster, ενώ η  $F_1$  είναι κάθετη σε αυτόν και έχει την τάση να περιστρέψει τον τροχό. Καθώς ο τροχός στρίβει, μία δύναμη τριβής  $T$  εμφανίζεται (όπως βλέπουμε και στο σχήμα 5.2.2.3), η οποία τείνει να κάνει τον τροχό παράλληλο με το επίπεδο κίνησης.

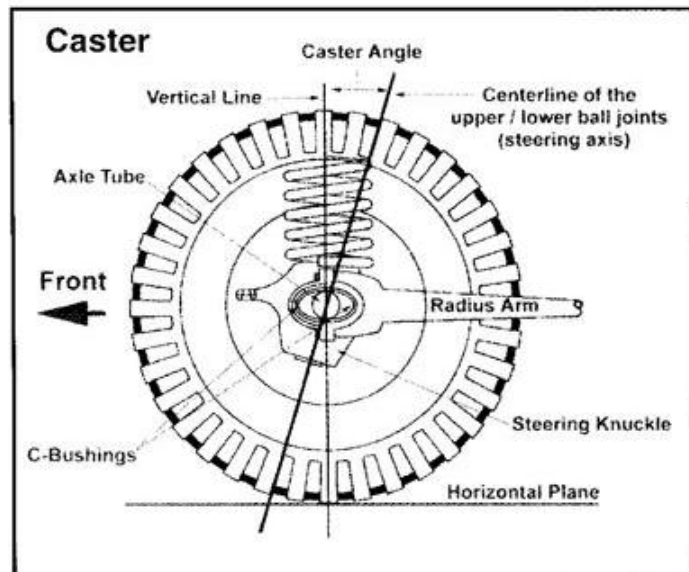


Σχήμα 5.2.2.2. Η αντίδραση του εδάφους  $F$  αναλύεται σε παράλληλη και κάθετη δύναμη.



Σχήμα 5.2.2.3. Η κάθετη δύναμη έχει την τάση να στρίψει τον τροχό, και η τριβή  $T$  εμφανίζεται προκειμένου να κάνει το επίπεδο του caster κάθετο στην κατεύθυνση της κίνησης.

Παρότι η δύναμη  $T$  είναι μικρή, επηρεάζει την τροχιά του ρομπότ και το αναγκάζει να παρεκκλίνει από την πορεία του. Το μέγεθος και η κατεύθυνση της δύναμης  $T$  εξαρτάται από την γωνία μεταξύ του άξονα περιστροφής και του άξονα του τροχού (*caster angle* – βλ. σχήμα 5.2.2.4).



Σχήμα 5.2.2.4. Caster Angle.

Επειδή η μοντελοποίηση αυτής της επίδρασης του περιστρεφόμενου τροχού στην προβλεπόμενη τροχιά του ρομπότ, αποτελεί ένα πολύ σύνθετο θέμα με αβέβαια πρακτικά κέρδη, αποφασίσαμε να βελτιώσουμε την λειτουργία του συστήματος, αντικαθιστώντας τον βοηθητικό τροχό με κάποιο άλλο εξάρτημα, που θα μπορούσε να αποτελέσει μία προτιμότερη λύση για μία εφαρμογή αυτού του τύπου.

## Omniwheel

Μία πρόταση είναι η αντικατάσταση του βοηθητικού περιστρεφόμενου τροχού με ένα *omniwheel* (σχήμα 5.2.2.5), το οποίο είναι και αυτό ένα είδος τροχού με δυνατότητα κίνησης προς οποιαδήποτε κατεύθυνση. Οι τροχοί αυτού του τύπου περιέχουν μικρούς κυλίνδρους κατά μήκος της περιφέρειάς τους, οι οποίοι είναι τοποθετημένοι κάθετα στην κατεύθυνση περιστροφής.

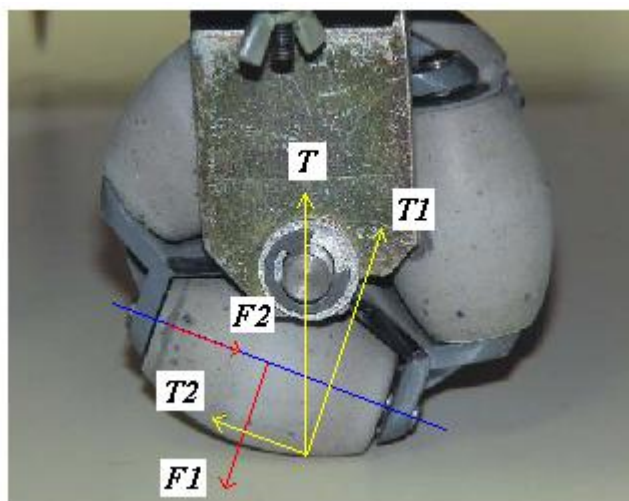


Σχήμα 5.2.2.5. Omniwheel.

Τα omniwheels αντιστέκονται στην αποσταθεροποίηση και είναι κατάλληλα για εφαρμογές, στο περιβάλλον εκτέλεσης των οποίων είναι πιθανή η παρουσία σκόνης ή υγρασίας. Δεν απαιτούν λίπανση των αξόνων ή κάποιου άλλου είδους συντήρησης, και έχουν ιδιαίτερα χαμηλό κόστος (ακόμα και της τάξης των 2\$).

Το βασικότερο πλεονέκτημα αυτών των τροχών είναι ότι δρουν σαν ένα σημείο στήριξης με μία αμελητέα μετατόπιση, με αποτέλεσμα οι επιδράσεις του caster που περιγράφηκαν παραπάνω να ελαχιστοποιούνται. Ένα άλλο αξιοσημείωτο πλεονέκτημα είναι ότι οι δυνάμεις που δημιουργούν διαταραχές και οφείλονται στο omniwheel είναι τυχαίες, και κατά συνέπεια τα σφάλματα που εισάγονται είναι μη-συστηματικά, και άρα λιγότερο σημαντικά για την οδομετρία.

Από την άλλη πλευρά, το σημαντικότερο μειονέκτημα του συγκεκριμένου τύπου περιστρεφόμενου τροχού, είναι η χαμηλή του ποιότητα. Πρώτον, η προβολή του omniwheel δεν είναι απολύτως κυκλική, ενώ δεύτερον, οι κύλινδροί του επιτρέπουν την κίνηση και κατά μήκος του οριζώντιου άξονά του. Αυτή η κίνηση είναι αρκετά απότομη και συμβαίνει όταν ο άξονας του κυλίνδρου αλλάζει κλίση, όπως περιγράφεται και στα σχήματα 5.2.2.6 και 5.2.2.7.

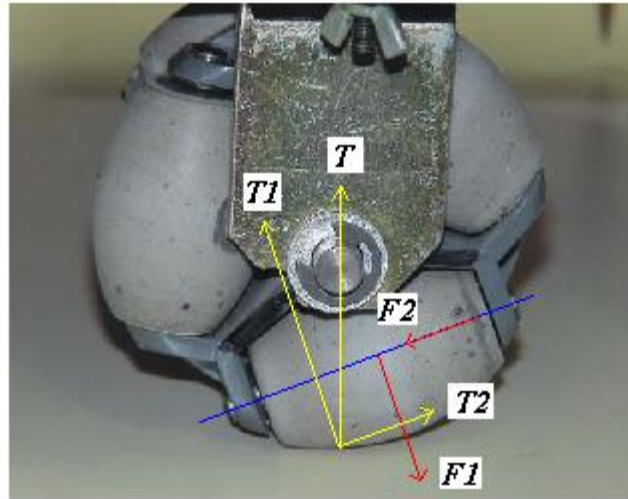


Σχήμα 5.2.2.6. Η συνιστώσα της αντίδρασης του εδάφους ωθεί τον κύλινδρο αριστερά.

Όταν η κλίση του άξονα του κυλίνδρου είναι όπως φαίνεται στο σχήμα 6, η συνιστώσα  $T_2$  της αντίδρασης του εδάφους σπρώχνει τον κύλινδρο αριστερά μέχρι να πλησιάσει την αριστερή άκρη. Καθώς το omniwheel περιστρέφεται, η κλίση του άξονα του κυλίνδρου αλλάζει, και η δύναμη  $F_2$  είναι απαραίτητη προκειμένου να αντισταθμίσει την νέα δύναμη  $T_2$ . Αυτή η δύναμη δεν μπορεί να ασκηθεί μέχρι ο κύλινδρος να φτάσει στην δεξιά άκρη, και κατά συνέπεια ο άξονας του κυλίνδρου (και κατά μήκος με αυτόν όλο το omniwheel) ολισθαίνει, μέχρι ο κύλινδρος να έρθει σε επαφή με την δεξιά άκρη.

Αυτές οι ατέλειες του τροχού προκαλούν δυνάμεις διαταραχής οι οποίες και αντιτίθενται στην ακρίβεια της οδομετρίας. Παρόλα αυτά, δεν εμφανίζονται συνεχώς, παρά μόνο όταν αλλάζει η κλίση του άξονα του κυλίνδρου. Επιπλέον, η εμφάνισή τους είναι αρκετά τυχαία, αντίθετα με τις δυνάμεις που εμφανίζονται εξαιτίας των ατελειών του βοηθητικού περιστρεφόμενου τροχού (caster) που είχαμε χρησιμοποιήσει προηγουμένως, οι οποίες και επηρεάζουν με συστηματικό τρόπο ολόκληρη την κίνηση του ρομπότ.



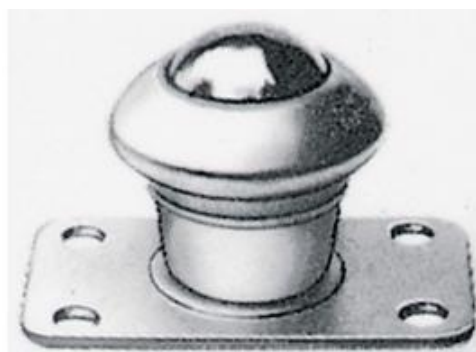


Σχήμα 5.2.2.7. Καθώς το omniwheel περιστρέφεται, η δύναμη  $T_2$  αλλάζει φορά και σπρώχνει τον κύλινδρο δεξιά.

### Ball Transfer Unit

Η αντικατάσταση του caster με ένα omniwheel, λοιπόν, βελτιώνει κατά πολύ ολόκληρη τη λειτουργία του συστήματος ελαχιστοποιώντας σε μεγάλο βαθμό τα συστηματικά σφάλματα που εισήγαγε ο περιστρεφόμενος τροχός. Όπως αναφέραμε όμως, ακόμα και με τη χρήση ενός omniwheel (εξαιτίας των ατελειών του τροχού), δεν έχουμε την επιδιωκόμενη ακρίβεια στην οδομετρία του συστήματος. Η ιδανική λύση θα ήταν η χρήση ενός εξαρτήματος, με μια μικρή σφαίρα (μπίλια) προσαρμοσμένη στο κέντρο, έτσι ώστε να υπάρχει δυνατότητα κίνησης προς οποιαδήποτε κατεύθυνση ανά πάσα στιγμή. Η λογική αυτού του τρόπου κίνησης, είναι παρόμοια με τον τρόπο χειρισμού ενός παλιάς τεχνολογίας mouse ηλεκτρονικού υπολογιστή, με προσαρμοσμένη μπίλια.

Σύμφωνα με τα παραπάνω λοιπόν, μία προτεινόμενη επιλογή είναι η χρήση ενός εξαρτήματος που ονομάζεται *ball transfer unit* (σφαιρική μονάδα μεταφοράς), το οποίο κινείται ακριβώς με τον τρόπο που περιγράψαμε και παρουσιάζεται στο παρακάτω σχήμα.



Σχήμα 5.2.2.8. *Ball Transfer Unit*.

Τα ball transfer units είναι χειριζόμενα συστήματα πολλαπλών κατευθύνσεων, τα οποία χρησιμοποιούνται σε εφαρμογές όπου ένα φορτίο απαιτείται να κινηθεί ομαλά, με ακρίβεια και με το ελάχιστο δυνατό έργο προς οποιαδήποτε κατεύθυνση. Όπως αναφέραμε, περιέχουν μία σφαιρική μπίλια προσαρμοσμένη εφαρμοστά σε ένα ειδικό (συνήθως μεταλλικό) αντικείμενο.

Τα συγκεκριμένα εξαρτήματα έχουν (όπως και τα omnivheels) ιδιαίτερα χαμηλό κόστος, ακόμα και της τάξης των 5\$. Αποτελούν την πλέον ιδανική επιλογή για ένα σύστημα όπως αυτό της παρούσας εφαρμογής, καθώς εξαλείφουν πλήρως τα συστηματικά σφάλματα (αλλά και αυτά που οφείλονται σε απρόβλεπτες μικρές ανωμαλίες του εδάφους) τα οποία εισάγονται από τον οποιονδήποτε βοηθητικό τροχό που χρησιμοποιείται για την περιστροφή του οχήματος.

### 5.2.3 Αναβάθμιση Κωδικοποιητών

Μία πολύ σημαντική μετατροπή που μπορούμε να πραγματοποιήσουμε προκειμένου να τελειοποιήσουμε τη λειτουργία του ρομπότ, είναι η αντικατάσταση των κωδικοποιητών, με αντίστοιχους κωδικοποιητές με μεγαλύτερο, όμως, αριθμό τμημάτων. Η επιλογή τροχών μεγάλης διαμέτρου έγινε (όπως έχουμε αναφέρει στην παράγραφο 3.5.2) προκειμένου να υπάρχει η δυνατότητα προσαρμογής κωδικοποιητών με τη μεγαλύτερη δυνατή διάμετρο. Οι κωδικοποιητές 36 τμημάτων που χρησιμοποιήθηκαν, έχουν διάμετρο 10,2cm και κατά συνέπεια, το άνοιγμα του κάθε τμήματος είναι της τάξης των 0,89cm. Η επιλογή, όμως, τροχών μεγάλης διαμέτρου, σε συνδυασμό με τις προδιαγραφές του αισθητήρα CNY70, μας δίνει τη δυνατότητα να χρησιμοποιήσουμε κωδικοποιητές ακόμα περισσότερων τμημάτων. Ο αισθητήρας CNY70 έχει την δυνατότητα να εντοπίσει την εναλλαγή των χρώματος μεταξύ τμημάτων που έχουν μήκος έως και 0,5cm. Ο αισθητήρας μπορεί να χρησιμοποιηθεί και για τμήματα (λευκό – μαύρο) ακόμα μικρότερου μήκους, δεν θα λάβουμε όμως την καλύτερη δυνατή περίπτωση, καθώς εξαιτίας διάφορων κατασκευαστικών ατελειών, σε συνδυασμό κυρίως με την υψηλή ταχύτητα περιστροφής των κινητήρων, υπάρχει η πιθανότητα να χαθεί κάποια εναλλαγή, γεγονός το οποίο είναι πάρα πολύ κρίσιμο και ιδιαίτερα σημαντικό να αποτρέψουμε. Με χρήση, λοιπόν, κωδικοποιητών της ίδιας διαμέτρου ( $d_{enc} = 10,2cm$ ) και τμημάτων ανοίγματος της τάξης του 0,5cm τότε μπορούμε να αυξήσουμε τον αριθμό των τμημάτων σε :

$$n' = \frac{2\pi r_{enc}}{a'}$$

όπου :

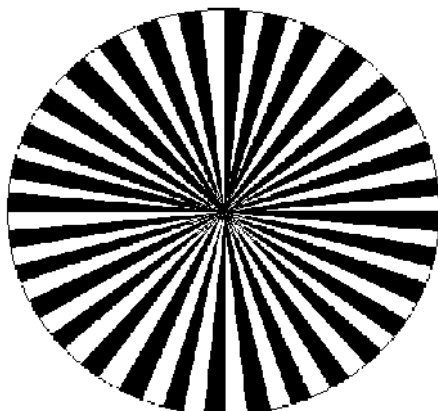
$2\pi r_{enc}$  η περιφέρεια του encoder disc ακτίνας  $r_{enc} = d_{enc} / 2 = 10,2 / 2 = 5,1cm$

$a'$  το νέο άνοιγμα των τμημάτων του κωδικοποιητή.

Άρα,

$$n' = \frac{2 \times 3,14 \times 5,1}{0,5} = \frac{32,028}{0,5} = 64,056 \approx 64 \tau\mu.$$

Μπορούμε να χρησιμοποιήσουμε, δηλαδή, κωδικοποιητή έως και 64 τμημάτων.



Σχήμα 5.2.3.1. Encoder disc 64 τμημάτων.

Με κωδικοποιητή 64 τμημάτων πλέον, έχουμε

$$step' = \frac{2\pi r}{n'} = \frac{2 \times 3,14 \times 5,5}{64} = \frac{34,54}{64} = 0,5396875cm$$

άρα, για κάθε εναλλαγή του χρώματος, διανύεται απόσταση ίση με το μήκος του βήματος, δηλαδή 0,5396875cm.

Η αύξηση του αριθμού των τμημάτων του κωδικοποιητή, και κατά συνέπεια, η μεγάλη μείωση του μήκους βήματος, έχει ιδιαίτερη σημασία. Με τον παραπάνω μετατροπή, μειώνεται ακόμα περισσότερο το σφάλμα που οφείλεται στη διαφορά της ζητούμενης τελικής απόστασης από την μετρούμενη απόσταση και το οποίο περιγράψαμε αναλυτικά στην προηγούμενη παράγραφο. Το σφάλμα αυτό είναι μικρότερο ή ίσο από το μήκος βήματος. Κατά συνέπεια, ενώ για κωδικοποιητή 36 τμημάτων κυμαίνεται από 0 έως 0,95945cm, για κωδικοποιητή 64 τμημάτων θα κυμαίνεται από 0 έως 0,5396875cm, δηλαδή μειώνεται σχεδόν κατά το ήμισυ. Εκτός από την ελαχιστοποίηση του παραπάνω σφάλματος, η αύξηση των τμημάτων του κωδικοποιητή, βελτιώνει τη διαδικασία συντονισμού των τροχών, μειώνοντας ακόμα περισσότερο και την ταλάντευση που οφείλεται στην συνεχόμενη διακοπή και επανέναρξη της κίνησης των κινητήρων.

## 5.3 Κατασκευαστικές Μετατροπές

### 5.3.1 Εισαγωγή

Οι σημαντικότερες κατασκευαστικές μετατροπές που θα μπορούσαν να γίνουν, αφορούν κυρίως τους αισθητήρες υπερύθρων, που χρησιμοποιήθηκαν για τις διαδικασίες αποφυγής εμποδίων. Οι περιορισμένες δυνατότητες (κυρίως οικονομικές) που είχαμε όσον αφορά στην προμήθεια υλικών και γενικότερα στον εξοπλισμό του συστήματος, μας οδήγησαν σε φθηνές και τεχνολογικά προσιτές λύσεις, όπως η χρήση των αισθητήρων υπερύθρων που έχουν περιγραφεί στην ενότητα 3.4. Παράλληλα, έγινε χρήση του μικρότερου δυνατού αριθμού των συγκεκριμένων αισθητήρων, προκειμένου να εξυπηρετήσουμε απλά τις ανάγκες της παρούσας εργασίας.

Οι πομποδέκτες υπερύθρων που χρησιμοποιήθηκαν λειτουργούν ικανοποιητικά όσον αφορά τον εντοπισμό των εμποδίων και για το λόγο αυτό, οι 2 αισθητήρες που είναι προσαρμοσμένοι στην μπροστινή πλευρά της κατασκευής, δεν είναι απαραίτητο να αντικατασταθούν. Παρόλα αυτά, το γεγονός ότι έχουν χρησιμοποιηθεί μόνο 2 αισθητήρες στην πλευρά αυτή, σε συνδυασμό με το μικρό γωνιακό εύρος εκπομπής και λήψης των αισθητήρων, έχουν ως αποτέλεσμα να μην καλύπτεται όλη η (έτσι κι αλλιώς μεγάλου μήκους) μπροστινή πλευρά και κατά συνέπεια να εμφανίζονται «νεκρά» σημεία. Το γεγονός αυτό σημαίνει ότι δεν υπάρχει η δυνατότητα να εντοπιστούν όλα τα εμπόδια, όπως για παράδειγμα, ένας στενός στύλος που κατευθύνεται προς το μέσο της πλευράς αυτής. Για τον λόγο αυτόν, η προσθήκη τουλάχιστον ενός ακόμα πομποδέκτη υπερύθρων στην μπροστινή πλευρά του ρομπότ θα έλυνε αυτό το πρόβλημα, με αποτέλεσμα να μπορούν πλέον να αντιμετωπιστούν και περιπτώσεις εμποδίων όπως η προηγούμενη.

Το σημαντικότερο πρόβλημα, όμως, των πομποδεκτών υπερύθρων που χρησιμοποιήθηκαν, έχει να κάνει με τη διαδικασία ευθυγράμμισης του οχήματος με τα εμπόδια που συναντώνται. Οι συγκεκριμένοι αισθητήρες δεν προσφέρονται για εφαρμογές στις οποίες απαιτείται υπολογισμός της απόστασης από κάποιο αντικείμενο, όπως και επιδιώκουμε για να πετύχουμε την ευθυγράμμιση. Όπως παρουσιάσαμε στην ενότητα 4.2, με τους συγκεκριμένους αισθητήρες υπερύθρων μπορούμε απλά να χωρίσουμε τον χώρο σε ζώνες, χωρίς να έχουμε ακρίβεια στις μετρήσεις. Έχουμε δηλαδή μόνο τη δυνατότητα, να γνωρίζουμε αν ένα εμπόδιο είναι πιο κοντά ή πιο μακριά από μια απόσταση αναφοράς, χωρίς όμως να γνωρίζουμε σε ποια ακριβώς απόσταση βρίσκεται. Εκτός από το γεγονός αυτό, όπως είδαμε στην συγκεκριμένη ενότητα, λόγω κατασκευαστικών διαφορών μεταξύ των δεκτών, προκύπτουν σημαντικές διαφορές και μεταξύ της ίδιας ζώνης για τους 2 αισθητήρες της κάθε πλευράς, και αυτό μας αποτρέπει από το να πετύχουμε απόλυτη ευθυγράμμιση με το εμπόδιο.

Για τους παραπάνω λόγους, θα ήταν προτιμότερο να αντικαταστήσουμε τους 4 πομποδέκτες υπερύθρων που είναι προσαρμοσμένοι στις δύο πλαϊνές πλευρές του ρομπότ, με αισθητήρες οι οποίοι είναι κατάλληλοι για μία τέτοιου είδους διαδικασία και θα προσφέρουν μεγαλύτερη ακρίβεια στην μέτρηση απόστασης από ένα εμπόδιο.

### 5.3.2 Αισθητήρες Υπερήχων

Μία πολύ καλή επιλογή είναι να χρησιμοποιηθούν αισθητήρες υπερήχων, οι οποίοι και προσφέρονται για τέτοιες εφαρμογές. Οι αισθητήρες υπερήχων παρέχουν μια οικονομικά αποδοτική μέθοδο ανίχνευσης, με ιδιότητες που δεν υπάρχουν σε άλλες τεχνολογίες. Με τη χρήση μιας ευρείας ποικιλίας μετατροπέων υπερήχων και διάφορα φάσματα συχνοτήτων, ένας αισθητήρας υπερήχων μπορεί να σχεδιαστεί για να λύσει πολλά προβλήματα εφαρμογών που είναι απαγορευτικά στο κόστος ή απλά δεν μπορούν να λυθούν από άλλους αισθητήρες. Οι αισθητήρες υπερήχων εκπέμπουν συνεχώς ηχητικούς παλμούς υψηλής συχνότητας προς την επιφάνεια του στόχου και ανακλώνται πίσω στον αισθητήρα. Τα ηλεκτρονικά του αισθητήρα μετρούν το χρόνο λήψης του σήματος και τον μετατρέπουν σε μονάδα μήκους. Δεδομένου ότι η ταχύτητα του ήχου επηρεάζεται από τη θερμοκρασία αέρα, οι αισθητήρες μας υπερήχων περιλαμβάνουν έναν ενσωματωμένο αισθητήρα θερμοκρασίας. Οι μετρήσεις στάθμης/απόστασης αντισταθμίζονται

αυτόματα σε όλη την κλίμακα λειτουργίας του αισθητήρα. Κλασικές εφαρμογές περιλαμβάνουν την ανίχνευση προσέγγισης, την παρουσία ή την απουσία αντικειμένου, την ανίχνευση εμποδίων σε αυτοματοποιημένα οχήματα, την μέτρηση απόστασης, μέτρηση στάθμης κλπ. Οι διαφορετικοί τρόποι λειτουργίας και συνθέσεις των αισθητήρων και συστημάτων υπερήχων επιτρέπουν τη χρήση στις περισσότερες εφαρμογές αυτοματισμού.

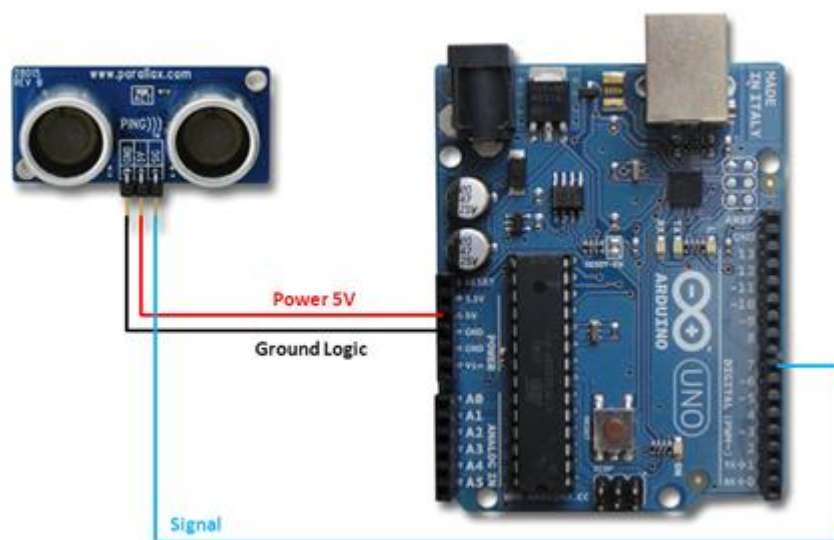
Ένα προτεινόμενο μοντέλο αισθητήρα είναι ο PING))) Ultrasonic Distance Sensor της εταιρίας Parallax, ο οποίος και παρουσιάζεται παρακάτω.



Σχήμα 5.3.2.1. Ο αισθητήρας PING)))<sup>TM</sup> Ultrasonic Distance Sensor της Parallax.

Ο αισθητήρας PING))) παρέχει ακριβείς μετρήσεις απόστασης, ενώ έχει την ικανότητα να αναγνωρίσει εμπόδια σε αποστάσεις από 2cm έως 3m περίπου, γεγονός που τον καθιστά κατάλληλο για μία εφαρμογή όπως η συγκεκριμένη. Είναι πολύ απλό να συνδεθεί με μικροελεγκτές όπως ο Basic Stamp ή ο Javelin Stamp, και φυσικά με τον Arduino, απαιτώντας μόνο έναν ακροδέκτη I/O, όπως φαίνεται και στο σχήμα 5.3.2.2. Η κατανάλωση ισχύος του αισθητήρα είναι της τάξης των 30mA έως 35mA το πολύ, ενώ η τάση λειτουργίας του είναι στα 5V.

Ο αισθητήρας PING))) λειτουργεί εκπέμποντας ριπές υπερήχου και παράγοντας παλμούς εξόδου, κάθε ένας από τους οποίους αντιστοιχεί στον χρόνο που απαιτείται ώστε να επιστρέψει στον αισθητήρα η ηχώ της κάθε ριπής. Μετρώντας το εύρος του παλμού αυτού, η απόσταση από τον στόχο μπορεί εύκολα να υπολογιστεί. Ο αισθητήρας στέλνει ριπές παλμών στα 40kHz, ενώ η κάθε ριπή ταξιδεύει στον αέρα (και σε θερμοκρασία δωματίου 20-25°C) με ταχύτητα 344m/s.



Σχήμα 5.3.2.2. Ο τρόπος σύνδεσης του αισθητήρα υπερήχων Parallax PING με τον arduino.

Παρακάτω παρουσιάζεται ο κώδικας (για arduino) που θα μπορούσε να χρησιμοποιηθεί για τον υπολογισμό της απόστασης με τον αισθητήρα υπερήχων Parallax Ping)).

```
int PingPin = 7;

void setup()
{
  Serial.begin(9600);
}

void loop()
{
  //Αποστολή παλμού.

  pinMode(PingPin, OUTPUT);
  digitalWrite(PingPin, LOW);
  delayMicroseconds(2);
  digitalWrite(PingPin, HIGH);
  delayMicroseconds(5);
  digitalWrite(PingPin, LOW);

  //Διαβάζουμε πόσο χρόνο το pin βρίσκεται σε κατάσταση HIGH, ώστε να μετρήσουμε
  //την διάρκεια του προκύπτοντος παλμού.

  pinMode(PingPin, INPUT);

  unsigned long Duration = pulseIn(PingPin, HIGH);

  if (Duration == 0)
  {
    Serial.println ("No Pulse received from the sensor");
  }
  else
  {
    Serial.print ("Distance : ");
    Serial.print (Convert_Time_Space(Duration));
    Serial.println (" cm");
  }
  delay(1000);
}

//Μετατροπή της διάρκειας του παλμού σε απόσταση.

unsigned long Convert_Time_Space(const unsigned long fnDuration)
{
  //Η συνάρτηση θα μπορούσε να είναι πιο ακριβής με τη χρήση floats.
  //Λαμβάνοντας υπόψη θερμοκρασία και υγρασία χρησιμοποιήσαμε 29μs ανά cm.
  //Διαιρούμε δια 2 αφού χρόνος που μετράμε προκύπτει από αντανάκλαση στο
  //εμπόδιο.

  return fnDuration / 29 / 2 ;
}
```

Ο αισθητήρας Parallax Ping Ultrasonic Distance Sensor έχει κόστος της τάξης των 30\$ τουλάχιστον, και κατά συνέπεια η αντικατάσταση των 4 πομποδεκτών υπέρυθρων που χρησιμοποιούνται για την ευθυγράμμιση με 4 αισθητήρες υπέρυθρων αντιστοίχως, θα αύξανε κατακόρυφα το κόστος της κατασκευής. Παρά, λοιπόν, τις μεγάλες δυνατότητές του, ο συγκεκριμένος αισθητήρας δεν αποτελεί και την πλέον συμφέρουσα επιλογή.

### 5.3.3 Ο αισθητήρας υπέρυθρων Sharp GP2D12

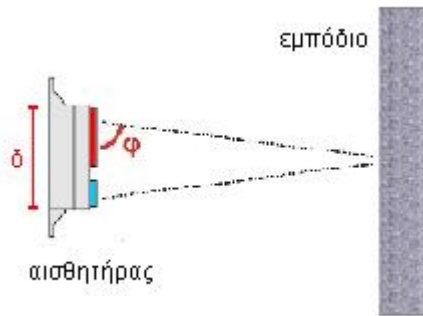
Μία εναλλακτική πρόταση αποτελεί η επιλογή του αισθητήρα υπέρυθρων GP2D12 Analog Distance Sensor της εταιρίας Sharp, ο οποίος παρουσιάζεται στο παρακάτω σχήμα.



*Σχήμα 5.3.3.1. Ο αισθητήρας υπέρυθρων Sharp GP2D12.*

Ο αισθητήρας Sharp GP2D12 είναι ένας αναλογικός αισθητήρας μέτρησης απόστασης, ο οποίος χρησιμοποιεί υπέρυθρη ακτινοβολία για τον εντοπισμό ενός εμποδίου σε απόσταση από 10 έως 80cm. Για τον λόγο αυτόν, ο συγκεκριμένος αισθητήρας αποτελεί ιδανική επιλογή για την παρούσα εφαρμογή, όπου απαιτείται ακριβής μέτρηση μικρών αποστάσεων από ένα εμπόδιο.

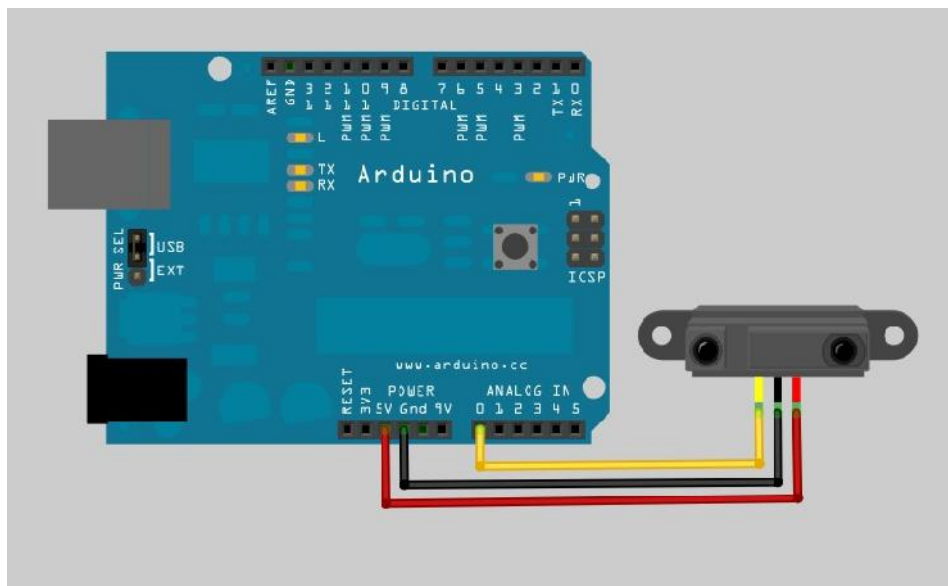
Οι αισθητήρες αυτής της κατηγορίας διαθέτουν έναν πομπό υπέρυθρων ακτινών και έναν δέκτη. Οι υπέρυθρες ακτίνες εκπέμπονται από τον πομπό του αισθητήρα, στην συνέχεια αντανακλώνται από το εμπόδιο και τέλος συλλέγονται από το δέκτη του. Ο δέκτης του αισθητήρα με τη βοήθεια ειδικού φακού μπορεί να μετρήσει τη γωνία  $\phi$  που σχηματίζουν οι ακτίνες (βλ. σχήμα 5.3.3.2), καθώς αυτές προσπίπτουν πάνω του μετά από την ανάκλασή τους από το εμπόδιο. Στην συνέχεια, αφού η απόσταση  $\delta$  είναι σταθερή και γνωστή από τα κατασκευαστικά χαρακτηριστικά του αισθητήρα, μπορεί με χρήση τριγωνομετρίας να υπολογιστεί η απόσταση μεταξύ αισθητήρα και εμποδίου. Στην πραγματικότητα, ο αισθητήρας παράγει στην έξοδό του μια τάση η οποία εξαρτάται από την απόσταση αυτήν.



Σχήμα 5.3.3.2. Η λειτουργία του αισθητήρα.

Ο αισθητήρας GP2D12 λοιπόν, παράγει μια μη γραμμική τάση εξόδου σε σχέση με την απόσταση ενός εμποδίου από τον αισθητήρα. Είναι συμβατός με όλους τους μικροελεγκτές Basic Stamp και SX, όπως φυσικά και με τον arduino. Όσον αφορά στον τελευταίο, η ύπαρξη αναλογικών θυρών εισόδου, μας δίνει τη δυνατότητα να αποφύγουμε τη χρήση κάποιου μετατροπέα analog to digital (ADC), και απλά να διαθέσουμε έναν ακροδέκτη I/O του arduino (μίας αναλογικής θύρας εισόδου) για την σύνδεση με τον GP2D12, όπως φαίνεται και στο σχήμα 5.3.3.3.

Η τάση λειτουργίας του αισθητήρα είναι στα 5V, ενώ το ρεύμα που καταναλώνει είναι της τάξης των 30-35mA περίπου.

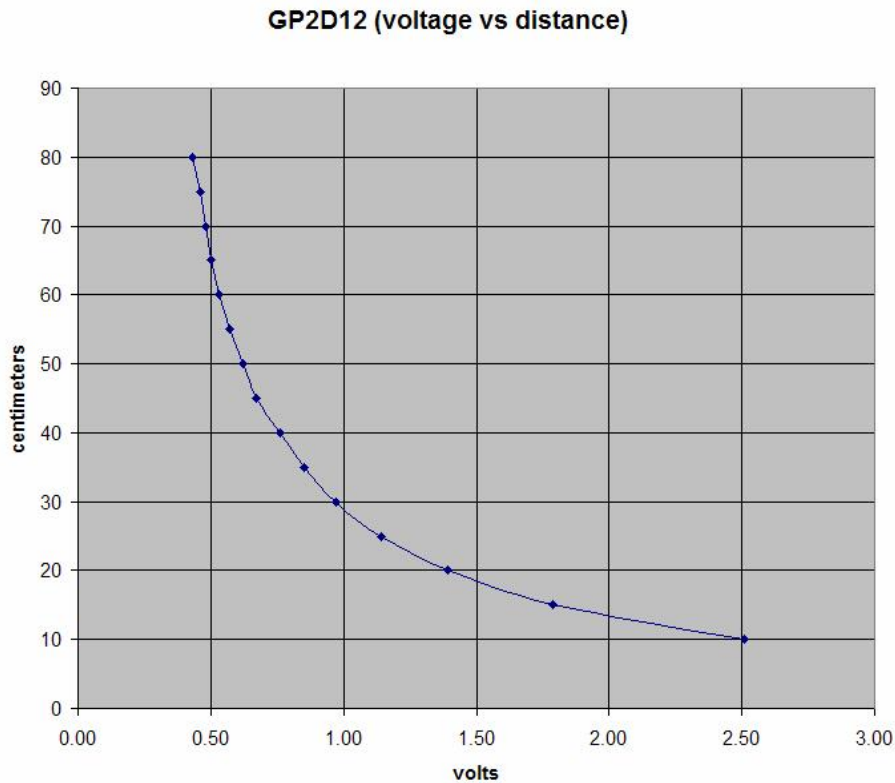


Σχήμα 5.3.3.3. Σύνδεση του Sharp GP2D12 με τον arduino.

Εξαιτίας του ότι η έξοδος του GP2D12 δεν είναι γραμμική, χρειάζεται να βρεθεί ένας τρόπος καθορισμού της απόστασης που αντιστοιχεί στην κάθε τάση εξόδου. Ένας τρόπος ρύθμισης του αισθητήρα μπορεί να υλοποιηθεί μετρώντας την τάση εξόδου του GP2D12 σε κάποιες δεδομένες αποστάσεις (σε cm) όπως φαίνεται στο διάγραμμα του σχήματος 5.3.3.4. Το παρακάτω διάγραμμα



σχηματίστηκε πραγματοποιώντας μετρήσεις της τάσης εξόδου ανά 5cm, για αποστάσεις από τα 80 έως τα 10cm.



*Σχήμα 5.3.3.4. Τάση εξόδου του GP2D12 συναρτήσει της απόστασης από το εμπόδιο.*

Παρακάτω παρουσιάζεται ένα παράδειγμα κώδικα για την ανάγνωση των τιμών της τάσης εξόδου του αισθητήρα Sharp GP2D12 και εκτύπωση των τιμών αυτών.

```
int sensorpin = 0; //Χρησιμοποιείται analog pin για τη σύνδεση με τον αισθητήρα
int val = 0;      //Μεταβλητή για την αποθήκευση τιμών από τον αισθητήρα

void setup()
{
  Serial.begin(9600);
}

void loop()
{
  val = analogRead(sensorpin); // reads the value of the sharp sensor
  Serial.println(val);        // prints the value of the sensor to the
                              // serial monitor
  delay(100);                 // wait for this much time before printing
                              // next value
}
```

Ο αισθητήρας Sharp GP2D12 έχει κόστος της τάξης των 10\$, γεγονός που τον καθιστά μία πολύ πιο οικονομική επιλογή από τον αισθητήρα υπερήχων της Parallax που περιγράψαμε νωρίτερα. Οποιοσδήποτε από τους 2 αισθητήρες που περιγράψαμε στο κεφάλαιο αυτό, παρέχει πολύ μεγάλη ακρίβεια στις εφαρμογές μέτρησης της απόστασης από ένα εμπόδιο. Βασιζόμενοι στο στοιχείο αυτό, μπορούμε να προτείνουμε τη χρήση 2 αισθητήρων συνολικά (οποιοδήποτε από τους παραπάνω τύπους αισθητήρων), οι οποίοι και θα τοποθετηθούν μόνο στην μπροστινή πλευρά της κατασκευής. Η μεγάλη ακρίβεια στην μέτρηση της απόστασης του εμποδίου από τον κάθε έναν από τους δύο αισθητήρες, σε συνδυασμό με την γνώση των διαστάσεων της κατασκευής, μας δίνει τη δυνατότητα να υπολογίσουμε εύκολα τη γωνία υπό την οποία το ρομπότ κατευθύνεται προς ένα εμπόδιο. Η γνώση της γωνίας αυτής είναι αρκετή ώστε να αντιληφθεί το ρομπότ το πόσο (και από ποιά πλευρά) πρέπει να στρίψει, προκειμένου να ευθυγραμμιστεί με το εμπόδιο.

Με τον παραπάνω τρόπο μας δίνεται η δυνατότητα να αποφύγουμε τη χρήση πολλών αισθητήρων σε όλη την περιφέρεια της κατασκευής, όπως και έχουμε πραγματοποιήσει. Η παραπάνω πρόταση αποτελεί έναν οικονομικό και πολύ ακριβή τρόπο για την υλοποίηση πλήρους ευθυγράμμισης με τα εμπόδια που συναντώνται.

### 5.3.4 Άλλες κατασκευαστικές τροποποιήσεις

#### ΤΡΟΠΟΠΟΙΗΣΗ ΒΑΣΗΣ ΜΙΚΡΟΦΩΝΟΥ

Μια άλλη κατασκευαστική μετατροπή που θα μπορούσε να πραγματοποιηθεί με σκοπό να τελειοποιηθεί η συνολική λειτουργία του συστήματος, είναι η κατάλληλη προσαρμογή ενός κινητήρα, προκειμένου να υπάρχει δυνατότητα περιστροφής της βάσης του μικροφώνου που λαμβάνει ακουστικές μετρήσεις. Με τον τρόπο αυτόν, θα υπάρχει η δυνατότητα λήψης ακουστικών μετρήσεων προς οποιαδήποτε κατεύθυνση. Διαφορετικά, η σταθερή θέση του μικροφώνου ενδεχομένως να οδηγήσει στην λήψη μη αντιπροσωπευτικού δείγματος μετρήσεων για ένα συγκεκριμένο σημείο του χώρου. Το γεγονός αυτό μπορεί να οφείλεται στις διαφορετικές συνθήκες του περιβάλλοντος που υπάρχουν στο ίδιο σημείο, για διαφορετικές όμως γωνίες λήψης. Για παράδειγμα, οι ανακλάσεις που οφείλονται στην παρουσία ενός μεγάλου αντικειμένου κοντά σε μια πλευρά του ρομπότ, μπορεί να ενισχύουν ή να αποσβένουν τον ήχο που λαμβάνεται από μια συγκεκριμένη γωνία του μικροφώνου (σε σύγκριση με μία άλλη). Για τους παραπάνω λόγους, η μετατροπή αυτή μπορεί να αποδειχθεί ιδιαίτερα σημαντική για την τελική λειτουργία του συστήματος.

#### ΠΡΟΣΑΡΜΟΓΗ ΚΑΤΕΥΘΥΝΤΗΡΩΝ

Τέλος, μία ακόμα τροποποίηση που θα μπορούσαμε να πραγματοποιήσουμε, είναι η κατάλληλη προσαρμογή ειδικών κατευθυντήρων, γύρω από τους πομπούς υπέρυθρης ακτινοβολίας. Η συγκεκριμένη επιλογή, έχει ως σκοπό την ελαχιστοποίηση του σφάλματος στην εκτίμηση της θέσης ενός εμποδίου, ένα ζήτημα το οποίο και αναλύσαμε στο τέλος του προηγούμενου κεφαλαίου. Εξαιτίας του γωνιακού εύρους εκπομπής των IR LEDs, κατά τον εντοπισμό ενός εμποδίου, καθώς

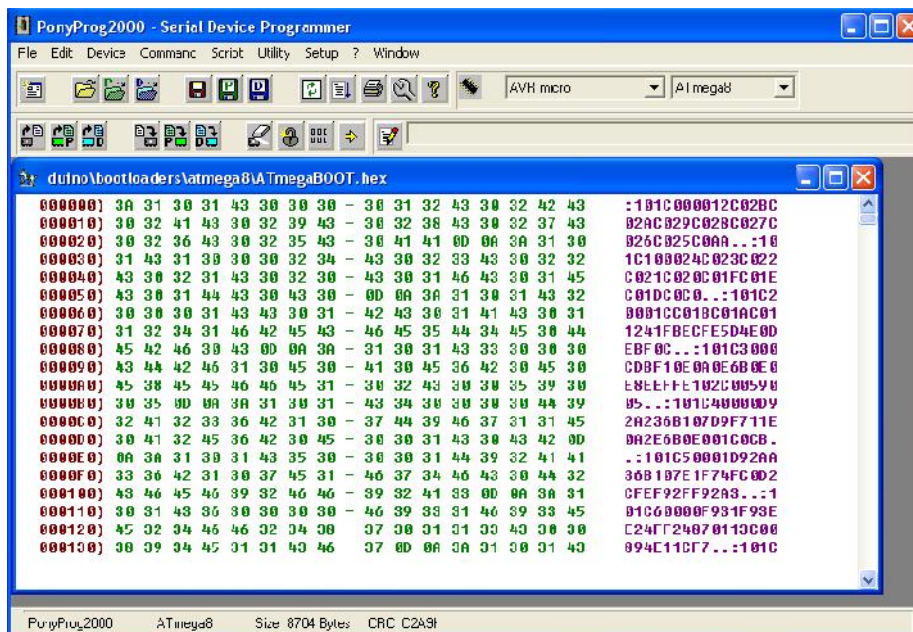
επίσης και κατά την προσπάθεια ευθυγράμμισης με αυτό, υπάρχει η πιθανότητα λήψης του σήματος υπέρυθρης ακτινοβολίας από τον αντίστοιχο δέκτη, μετά από ανάκλαση υπό κάποιας τάξης γωνία, σε κάποιο αντικείμενο. Το γεγονός αυτό εισάγει ένα σφάλμα στην εκτίμηση της θέσης του εμποδίου, κατά την προσπάθεια χαρτογράφησης του περιβάλλοντος χώρου. Η κατάλληλη προσαρμογή ενός (μαύρου κατά προτίμηση) μονωτικού περιβλήματος γύρω από τα IR LEDs, έχει ως αποτέλεσμα την αύξηση της κατευθυντικότητάς τους, και κατά συνέπεια την περισσότερο εστιασμένη προσέγγιση των εμποδίων.

# ΠΑΡΑΡΤΗΜΑ Α

## ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΣ ΜΙΚΡΟΕΛΕΓΚΤΗ ΑΤΜΕΓΑ168

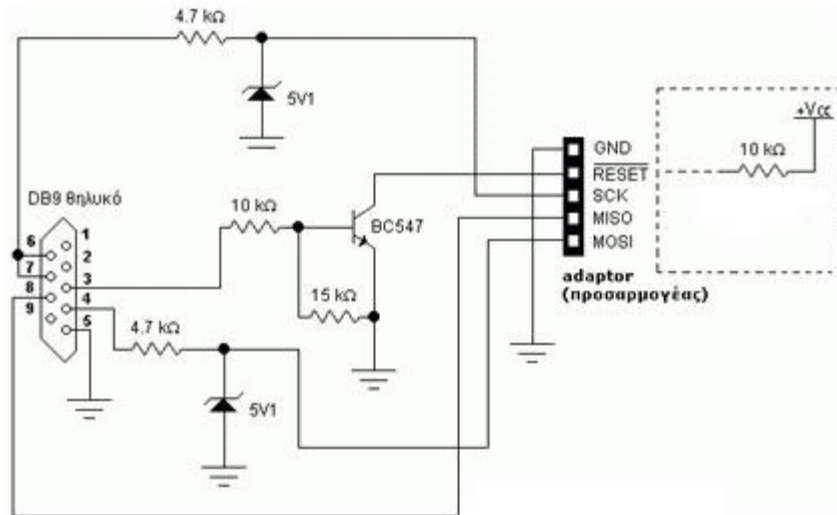
Γενικά, υπάρχουν πολλά κυκλώματα προγραμματισμού των μικροελεγκτών AVR (σαν αυτόν που χρησιμοποιούμε και στο δικό μας κύκλωμα). Ο προγραμματιστής είναι συνήθως ένα μικρό κύκλωμα, στο οποίο διασυνδέεται ο μικροελεγκτής, προκειμένου να φορτωθεί το προς εκτέλεση πρόγραμμα.

Ωστόσο, αυτό δεν είναι και το βασικό εξάρτημα. Την περισσότερη δουλειά την κάνει το πρόγραμμα που τρέχει στον Η/Υ και επικοινωνεί με αυτό το κύκλωμα. Υπάρχουν πολλά προγράμματα που χρησιμοποιούνται για τον προγραμματισμό μικροελεγκτών. Στην παρούσα περίπτωση, χρησιμοποιήσαμε το πρόγραμμα PonyProg2000 (βλ. εικόνα 1). Το συγκεκριμένο πρόγραμμα υποστηρίζει ένα μεγάλο πλήθος μικροελεγκτών, μεταξύ των οποίων και ο ΑΤmega168, τον οποίο και χρησιμοποιήσαμε.



Εικόνα 1. Το κεντρικό παράθυρο του προγράμματος PonyProg2000.

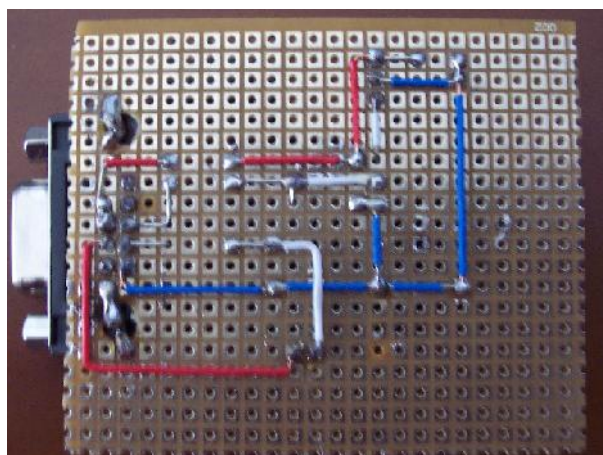
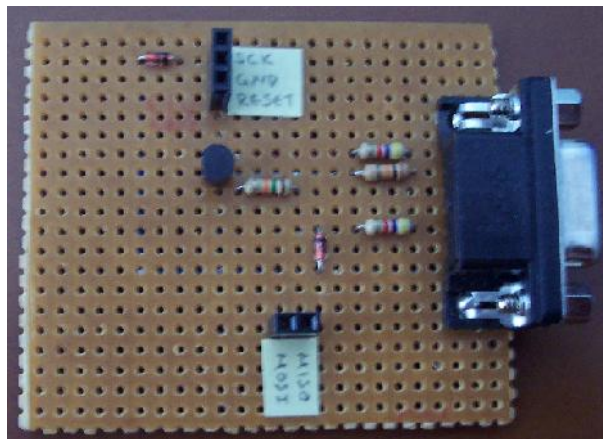
Στο παρακάτω σχήμα, παρουσιάζεται το κυκλωματικό διάγραμμα του προγραμματιστή που υλοποιήθηκε, και ο οποίος προορίζεται μόνο για μικροελεγκτές AVR (στους οποίους ανήκει και ο ΑΤmega168).



Σχήμα 2. Κύκλωμα προγραμματιστή του μικροελεγκτή ATmega168.

Στον ακροδέκτη RESET του ATmega είναι απαραίτητη η παρουσία της pull-up αντίστασης των 10KΩ, η οποία όμως είναι ήδη συνδεδεμένη, όπως έχουμε δείξει άλλωστε και στο σχήμα 3.3.5.

Η κατασκευή του παραπάνω κυκλώματος πραγματοποιήθηκε σε μία διάτρητη πλακέτα πυριτίου, όπως φαίνεται και στις παρακάτω εικόνες.



Εικόνες 2 και 3. Το κύκλωμα του προγραμματιστή που κατασκευάστηκε.

Στο σημείο αυτό πρέπει να τονίσουμε ότι χρειάζεται ιδιαίτερη προσοχή, καθώς ο συγκεκριμένος προγραμματιστής δεν λειτουργεί σωστά με έναν μετατροπέα από USB σε RS232. Για τον λόγο αυτό, απαιτείται η χρήση μίας κανονικής σειριακής θύρας του υπολογιστή, για την σύνδεση με το κύκλωμα του προγραμματιστή.

Το PonyProg2000 προκειμένου να προγραμματίσει με επιτυχία τον μικροελεγκτή, χρειάζεται να φορτωθεί σε αυτό (από αρχείο HEX), ο κώδικας του προγράμματος (File → Open Device File → ATmega168 → ATmegaBOOT.hex).

Εκτελείται λοιπόν, το PonyProg2000, με φορτωμένο το πρόγραμμα, έτοιμο να μεταφερθεί στον μικροελεγκτή. Πριν φορτώσουμε το πρόγραμμα, όμως, πρέπει να κάνουμε κάποιες επιπλέον ρυθμίσεις.

Καταρχήν, επιλέγουμε τον σωστό μικροελεγκτή επιλέγοντάς τον από το μενού, Device → AVR micro → ATmega168. Συνδέουμε το κύκλωμα του προγραμματιστή στην σειριακή θύρα του υπολογιστή (για παράδειγμα COM1). Κάνουμε τις κατάλληλες συνδέσεις μεταξύ των pin του κυκλώματος του προγραμματιστή με τους αντίστοιχους ακροδέκτες του μικροελεγκτή, και τέλος παρέχουμε τροφοδοσία στην συσκευή.

Αφού κάναμε τις απαραίτητες συνδέσεις και τροφοδοτήσαμε τα κυκλώματα, προγραμματίζουμε τα configuration bits του μικροελεγκτή. Από το μενού το PonyProg2000, επιλέγουμε Command → Security and Configuration Bits, και εμφανίζεται το παρακάτω παράθυρο.



Εικόνα 4. Το παράθυρο ρύθμισης των Security και Configuration Bits.

Στο παραπάνω παράθυρο, τσεκάρουμε όσα bits θέλουμε να είναι 0, δηλαδή ενεργοποιημένα. Αφήνουμε όλα τα security και configuration bits ανενεργά, και ελέγχουμε τις περιπτώσεις των memory lock bits, όπως αναγράφονται στον παρακάτω πίνακα του datasheet του ATmega168.



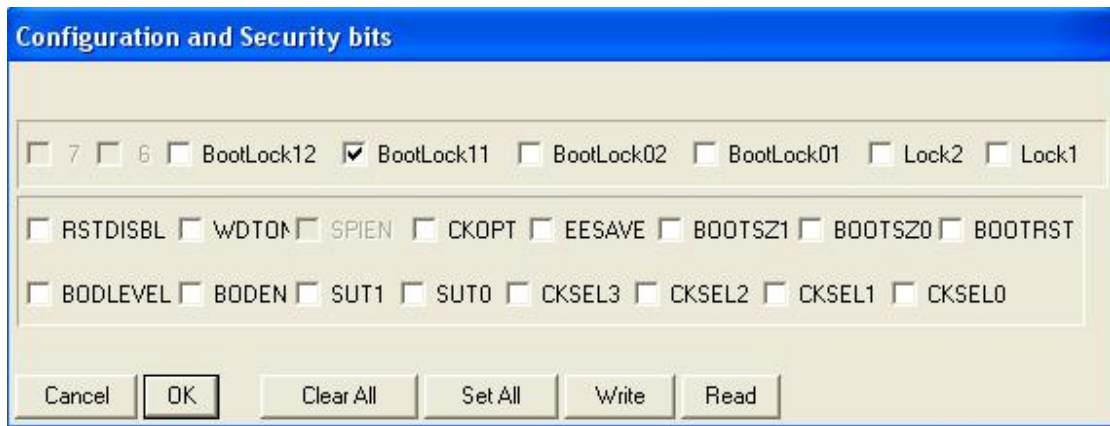
Memory lock bits			Protection type
LB mode	LB2	LB1	
1	1	1	No memory lock features enabled.
2	1	0	Further programming of the flash and EEPROM is disabled in parallel and serial programming mode. The fuse bits are locked in both serial and parallel programming mode. <sup>(1)</sup>
3	0	0	Further programming and verification of the flash and EEPROM is disabled in parallel and serial programming mode. The boot lock bits and fuse bits are locked in both serial and parallel programming mode. <sup>(1)</sup>

BLB0 mode	BLB02	BLB01	
1	1	1	No restrictions for SPM or LPM accessing the application section.
2	1	0	SPM is not allowed to write to the application section.
3	0	0	SPM is not allowed to write to the application section, and LPM executing from the boot loader section is not allowed to read from the application section. If interrupt vectors are placed in the boot loader section, interrupts are disabled while executing from the application section.
4	0	1	LPM executing from the boot loader section is not allowed to read from the application section. If interrupt vectors are placed in the boot loader section, interrupts are disabled while executing from the application section.
BLB1 mode	BLB12	BLB11	
1	1	1	No restrictions for SPM or LPM accessing the boot loader section.
2	1	0	SPM is not allowed to write to the boot loader section.
3	0	0	SPM is not allowed to write to the boot loader section, and LPM executing from the application section is not allowed to read from the boot loader section. If interrupt vectors are placed in the application section, interrupts are disabled while executing from the boot loader section.
4	0	1	LPM executing from the application section is not allowed to read from the boot loader section. If interrupt vectors are placed in the application section, interrupts are disabled while executing from the boot loader section.

Notes: 1. Program the fuse bits and boot lock bits before programming the LB1 and LB2.  
2. "1" means unprogrammed, "0" means programmed.

Πίνακας 1. Lock bit protection modes.

Σύμφωνα με τον παραπάνω πίνακα, μας αφορούν οι περιπτώσεις όπου  $LB_2LB_1 = 11$ ,  $BLB_{02}BLB_{01} = 11$ , ενώ  $BLB_{12}BLB_{11} = 10$  καθώς μας ενδιαφέρει να μην υπάρχει η πιθανότητα να γραφτεί κάποιο πρόγραμμα πάνω στον bootloader του μικροελεγκτή. Κατά συνέπεια, στο παράθυρο ρύθμισης των Security και Configuration Bits τσεκάρουμε μόνο το bit  $BLB_{11}$ , ενώ αφήνουμε τα υπόλοιπα ως έχουν, όπως βλέπουμε και στην παρακάτω εικόνα.



*Εικόνα 5. Ρύθμιση των Security και Configuration Bits.*

Εν συνεχεία πατάμε το πλήκτρο Write του παραθύρου και περιμένουμε μέχρι να ολοκληρωθεί η εγγραφή. Κλείνουμε το παράθυρο πατώντας OK, και είμαστε πλέον έτοιμοι να φορτώσουμε το πρόγραμμα.

Από το μενού το PonyProg2000, επιλέγουμε Command → Write All, και περιμένουμε για ένα μικρό χρονικό διάστημα, μέχρι να ολοκληρωθεί η εγγραφή και να εμφανιστεί το μήνυμα “Write Successful”. Στην περίπτωση όπου εμφανιστεί το μήνυμα “Write Failed” (ή κάποιο άλλο μήνυμα), τότε έχει γίνει κάποιο λάθος στη διαδικασία (η πιθανότητα να ευθύνεται το ολοκληρωμένο, εφόσον αυτό είναι καινούριο, είναι μικρή).



## ΠΑΡΑΡΤΗΜΑ Β

### ΠΕΡΙΕΧΟΜΕΝΑ ΣΥΝΟΔΕΥΤΙΚΟΥ ΔΙΣΚΟΥ (CD)

Το CD που συνοδεύει τη διπλωματική εργασία περιέχει τόσο τα απαραίτητα προγράμματα, όσο και επιπλέον πληροφοριακά στοιχεία, όπως για παράδειγμα τα φύλλα προδιαγραφών των διατάξεων που χρησιμοποιήθηκαν. Επίσης περιέχονται κάποιες χαρακτηριστικές φωτογραφίες των βασικότερων μερών της κατασκευής, όπως επίσης και ένα βίντεο από παράδειγμα εκτέλεσης της λειτουργίας του ρομπότ. Τέλος, περιέχεται σε ηλεκτρονική μορφή ολόκληρο το κείμενο της διπλωματικής εργασίας.

Αναλυτικά :

Κώδικας	Περιέχει τον τελικό κώδικα του προγράμματος στην γλώσσα του arduino. Υπάρχουν 2 αρχεία. Στο ένα αρχείο περιέχεται ο κώδικας με προκαθορισμένα σημεία έναρξης και προορισμού (χωρίς δηλαδή εισαγωγή δεδομένων), ενώ στο άλλο αρχείο περιέχεται η τελική μορφή του κώδικα, με είσοδο δεδομένων.
Διαγράμματα	Περιέχει σε αρχεία excel όλα τα διαγράμματα των αισθητήρων που χρησιμοποιήθηκαν, και τα οποία σχεδιάστηκαν μετά από συλλογή μεγάλου αριθμού μετρήσεων.
Φύλλα προδιαγραφών	Περιέχει τα datasheets όλων των ολοκληρωμένων κυκλωμάτων και στοιχείων που χρησιμοποιήθηκαν, όλων των αισθητήρων, καθώς επίσης και τα φύλλα προδιαγραφών κάποιων επιπλέον υλικών.
Φωτογραφίες	Περιέχονται κάποιες χαρακτηριστικές φωτογραφίες από την όλη κατασκευή, την πλακέτα του arduino, των κυκλωματικών διατάξεων των αισθητήρων κτλ.
Βίντεο	Περιέχει βίντεο ενός χαρακτηριστικού παραδείγματος εκτέλεσης της λειτουργίας του ρομπότ, σε μικρό χώρο.
Διπλωματική	Περιέχει το κείμενο της διπλωματικής εργασίας σε αρχείο pdf.

## ΒΙΒΛΙΟΓΡΑΦΙΑ

1. Μικροηλεκτρονικά κυκλώματα. SENDRA/SMITH. Εκδόσεις Παπασωτηρίου.
2. Μικροελεγκτές AVR και PIC. Κ. Πεκμεστζή. Έκδοση Ε.Μ.Π.
3. Ψηφιακή Σχεδίαση. Μ. Morris Mano. Εκδόσεις Παπασωτηρίου.
4. <http://www.arduino.cc/>
5. <http://arduino.cc/en/Tutorial/PWM>
6. <http://arduino.cc/en/Main/ArduinoBoardSerial>
7. <http://arduino.cc/en/Main/ArduinoBoardSerialSingleSided3>
8. <http://deltahacker.gr/2009/08/01/arduino-intro/>
9. <http://electronicslab.eu/el/psifiaka-hlektronika/14-oloklhromena-kyklomata>
10. <http://electronicslab.eu/el/avr-mikroelektes/324-hlektronika-ypologistika-systimata.html>
11. <https://ccrma.stanford.edu/~gary/controllers/ir.html>
12. <http://www1.cs.columbia.edu/~allen/F11/NOTES/icckinematics.pdf>
13. [http://nereus.mech.ntua.gr/pdf\\_ps/ecc07\\_3.pdf](http://nereus.mech.ntua.gr/pdf_ps/ecc07_3.pdf)
14. <http://users.ntua.gr/dpiperid/MyWebPage/Constructions/converters/rs232totl.htm>
15. <http://www.piperidis.co.nr/>
16. <http://www.grobot.gr/>
17. <http://www.mechatronics.gr/>
18. <http://www.microplanet.gr/>
19. <http://users.softlab.ntua.gr/~sgt>
20. [http://en.wikipedia.org/wiki/Mobile\\_robot](http://en.wikipedia.org/wiki/Mobile_robot)
21. <http://yourforum.gr/InvisionBoard/--t174790.html>