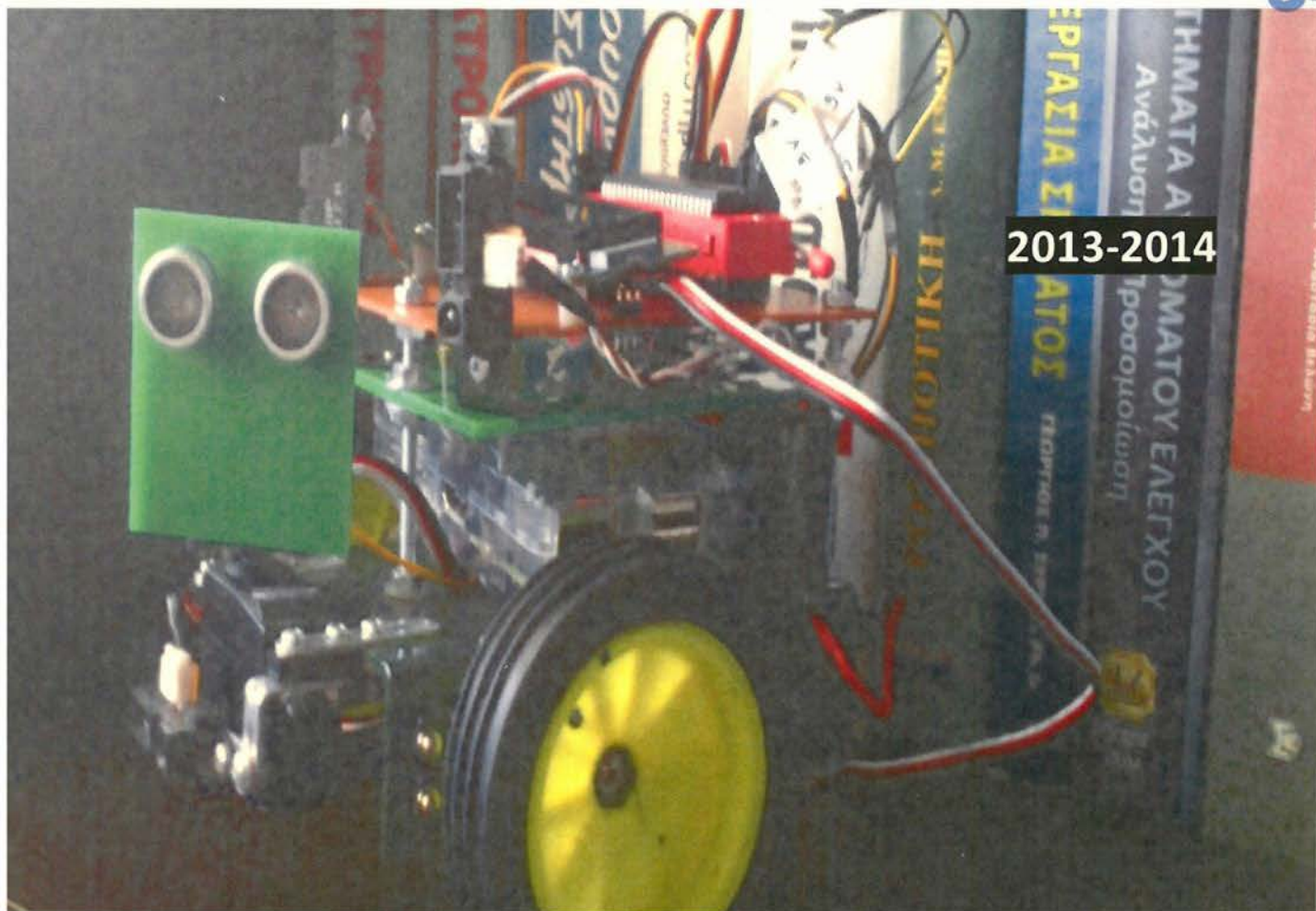


HY/S
84



2013-2014

Α.Τ.Ε.Ι ΠΕΙΡΑΙΑ,
Σ.Τ.ΕΦ, Η.Υ.Σ

ΡΟΜΠΟΤΙΚΟ ΟΧΗΜΑ ΑΠΟΦΥΓΗΣ ΕΜΠΟΔΙΩΝ

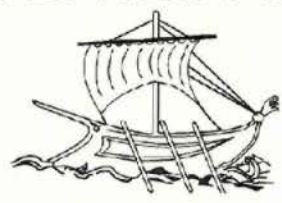
ΑΜ : 39194 Καμαρατάκης Γεώργιος

Εισηγητής : Δρ. Σταμάτης Αλατσαθιανός

ΑΙΓΑΛΕΩ 2014

ΒΙΒΛΙΟΘΗΚΗ
ΤΕΙ ΠΕΙΡΑΙΑ

ΑΤΕΙ ΠΕΙΡΑΙΑ



ΣΧΟΛΗ ΤΕΧΝΟΛΟΓΙΚΩΝ ΕΦΑΡΜΟΓΩΝ
ΤΜΗΜΑ ΗΛΕΚΤΡΟΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΙΚΩΝ
ΣΥΣΤΗΜΑΤΩΝ

ΕΥΧΑΡΙΣΤΙΕΣ

Η παρούσα πτυχιακή εργασία ολοκληρώθηκε μετά από επίμονες προσπάθειες, σε ένα αρκετά ενδιαφέρον αντικείμενο. Την προσπάθειά μου αυτή υποστήριξε ο επιβλέπων καθηγητής μου κ. Σταμάτης Αλατσαθανός, τον οποίο θα ήθελα να ευχαριστήσω.

Ακόμα θα ήθελα να ευχαριστήσω την οικογένειά μου, τους φίλους μου και ιδιαίτερα τον θείο μου και την θεία μου για την στήριξη τους όλα αυτά τα χρόνια.

ΠΕΡΙΛΗΨΗ

Η παρούσα πτυχιακή εργασία ασχολείται με την ανάπτυξη ενός ρομποτικού οχήματος σχεδιασμένο να αποφεύγει εμπόδια. Σε αυτό το πεδίο αυτοματισμού έχουν αναπτυχθεί αρκετές τεχνικές, η κάθε μία από αυτές έχει ορισμένα πλεονεκτήματα και μειονεκτήματα. Η προτεινόμενη τεχνική λαμβάνει υπ' όψη του ο μικροελεγκτής μετρήσεις απόστασης από συγκεκριμένες θέσεις για συγκεκριμένες διαδρομές, και διαλέγει την πιο μεγάλη μέτρηση (μεγαλύτερη απόσταση από το εμπόδιο) ακολουθώντας εκείνη την διαδρομή. Έτσι επιτυγχάνεται η αποφυγή εμποδίων σε βάθος χρόνου.

Περιεχόμενα

ΕΥΧΑΡΙΣΤΙΕΣ.....	2
ΠΕΡΙΛΗΨΗ.....	4
1. Εισαγωγή.....	9
1.1 Είδη οχημάτων.....	9
1.1.2 Όχημα με ερπύστριες.....	10
1.1.3 Ρομποτικό όχημα τεσσάρων τροχών.....	11
1.1.4 Δύο τροχών - “self-balancing robots”.....	13
1.1.5 Όχημα με παντοκατευθυντικούς τροχούς.....	14
1.2 Μέθοδοι αναγνώρισης περιβάλλοντος.....	18
1.2.1 Αισθητήρες απόστασης.....	18
1.2.1.1 Αισθητήρες υπέρυθρων ακτινών.....	18
1.2.1.2 Αισθητήρες υπερηχητικών παλμών.....	19
1.2.1.3 Γεωμετρικός υπολογισμός εκπομπής ακτίνας Laser.....	20
1.2.2 Αισθητήρας επαφής.....	21
1.3 Μέθοδοι αποφυγής εμποδίων.....	22
1.3.1 Μέθοδος αλλαγής κατεύθυνσης.....	22
1.3.2 Μέθοδος επιλογής κατεύθυνσης.....	23
2 Ρομποτικό όχημα αποφυγής εμποδίων.....	24
2.1 Υλικά και κόστος κατασκευής.....	25
2.2 Φιλοσοφία του ρομποτικού οχήματος.....	26
2.3 Ηλεκτρονικά υλικά κατασκευής.....	27
2.3.1 Κύκλωμα κατασκευής.....	27
2.3.2 Κινητήρες servo συνεχής περιστροφής.....	28
2.3.3 Κινητήρας servo περιστροφής 180 μοιρών.....	30
2.3.4 Μικροελεγκτής PIC 16F887.....	32
2.3.5 LCD οθόνη 16x2 χαρακτήρων.....	33
2.3.6 Αισθητήρες μέτρησης αποστάσεων.....	34
2.4 Σχέδια κατασκευής οχήματος.....	39
2.5 Κώδικας.....	40

2.5.1	Κώδικας project.....	40
3	Σενάρια αποτυχίας	54
4	Extra.....	55
4.1	Πορεία εργασίας (φωτογραφικό υλικό).....	55
5	Βιβλιογραφία	60

1.Εισαγωγή

Ένα όχημα για να γίνει αυτόνομο και να εκτελεί αλγόριθμους επιλογής πορείας χρειάζεται αισθητήρια όργανα με τα οποία θα γίνετε η αναγνώριση του περιβάλλοντος γύρω από το όχημα. Για να επιτύχουμε την αναγνώριση του χώρου γύρω απο το όχημα ο πιο κλασσικός τρόπος είναι να χρησιμοποιήσουμε ένα συνδυασμό αισθητήρων απόστασης. Το όχημα θα οδηγείται υπο τις υποδείξεις του εκάστοτε μικροελεγκτή, ο μικροελεγκτής θα υπολογίζει τις μετρήσεις των αισθητήρων, θα κάνει συγκρίσεις και θα ελέγχει το όχημα.

1.1 Είδη οχημάτων

1.1.1 Ρομποτικό όχημα τύπου αράχνης

Τα εντομοειδές οχήματα τύπου αράχνης διαφοροποιούνται ως προς τον αριθμό των άκρων τους γνωστά ως «πόδια». Τα πιο γνωστά οχήματα αυτού του είδους είναι τα **Hexapod** – ρομποτικό όχημα αράχνης έξι άκρων – ποδιών . Βασικό χαρακτηριστικό τους είναι η δυνατότητα να κινούνται σε ανομοιομορφα επίπεδα καθώς η γεωμετρία τα βοηθά να κρατάνε την ισοροπία τους και να υπερβαίνουν τις ανομοιομορφίες του εδάφους.



Εικ. 1.1.1 Hexapod

Μειονεκτήματα:

- Χαμηλή ταχύτητα μετακίνησης του οχήματος.
- Απαιτεί συγχρονισμό κινήσεων όλων των ακρών για την επιτυχή κίνηση του οχήματος. Οποιαδήποτε αποτυχία του συγχρονισμού θα επιφέρει αρνητικό αποτέλεσμα στην ισοροπία του οχήματος.
- Υψηλή κατανάλωση ρεύματος έχοντας 6 κινητήρες το ελάχιστο για την κίνηση.

Πλεονεκτήματα:

- Κίνηση σε ανώμαλο επίπεδο.
- Εύκολη κίνηση προς όλες τις κατευθύνσεις.
- Σταθερότητα

1.1.2 Όχημα με ερπύστριες

Αυτό το είδος οχήματος πρωτοεμφανίστηκε το 1770 και εμφανίζονται ακόμα στις μέρες μας. Η πιο γνώστη εμφάνιση αυτού του είδους είναι κυρίως οχήματα που έχουν να κάνουν με ανώμαλο έδαφος. Μερικά απο αυτά είναι τα τάνκς, οι μπουλντόζες, τα οχήματα εξακφής και καλλιεργητές χωραφίων – τρακτερ. Είναι ενδιαφέρον τρόπος λειτουργίας του οχήματος καθώς έχει δύο κινητήρες για την κίνηση και μέσο μεταφοράς της κίνησης ειδικά γρανάζια τα οποία μεταδίδουν την κίνηση στις ερπύστριες.

Για ευθύγραμμη κίνηση του οχήματος, οι κινητήρες κινούνται προς την ίδια κατεύθυνση δηλαδή έχουν περιστρέφονται ομόστροφα.

Για μεταβολή σε οποιαδήποτε άλλη κατεύθυνση εκτός της ευθύγραμμης, οι κινητήρες κινούνται αντίστροφα.



Εικ. 1.1.2 Ρομποτικό όχημα με ερπύστριες

1.1.3 Ρομποτικό όχημα τεσσάρων τροχών

Όχημα τεσσάρων τροχών εκ των οποίων οι δυο μπροστινοι τροχοί είναι περιστρέφόμενοι δίνοντας έτσι την κατεύθυνση στο όχημα και οι δυο όπισθεν τροχοί δίνουν την κίνηση του οχήματος. Αυτό το είδος είναι το πιο διαδεδομένο καθώς αυτήν την αρχιτεκτονική έχουν τα αυτοκίνητα. Αυτή η αρχιτεκτονική απαιτεί δύο κινητήρες για την διεκπεραίωση της κατασκευής του. Ο όπισθεν κινητήρας είναι συνδεδεμένος με ένα άξονα ο οποίος μεταδίδει την κίνηση στους δυο όπισθεν τροχούς. Ο μπροστινός κινητήρας ελέγχει τους δυο τροχούς ως προς την κλίση μεταβολής των τροχών.



1.1.3 α) RC car 4 τροχών

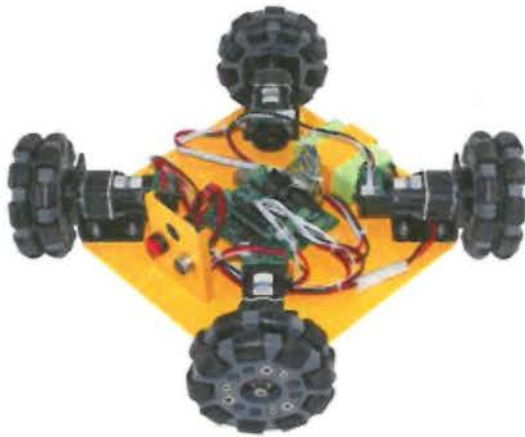
Μειονεκτήματα:

- Χαμηλή απόδοση κίνησης σε ανώμαλο επίπεδο
- Αδυναμία άμεσης μεταβολής σε οποιοδήποτε σημείο του χωρου—χρειάζεται πολλές αλλαγές κατεύθυνσης για να επιτευχθεί ο επιθυμητός στόχος.

Πλεονεκτήματα:

- Ειδικά σχεδιασμένο για ανάπτυξη υψηλών ταχυτήτων
- Σταθερότητα

Ένα ακόμα είδος ρομποτικού οχήματος είναι αυτό του παντοκατευθυντικού τύπου 4 τροχών. Αυτός ο τύπος έχει σχήμα ρόμβου όπου σε κάθε γωνία έχει κινητήρα για τον έλεγχο του εκάστοτε τροχού. Αυτός ο τύπος οχήματος έχει παντόκατευθυντική συμπεριφορά σε αντίθεση με το προηγούμενο τύπο 4 τροχών που αναφερθήκαμε.



1.1.3 β) Όχημα 4 τροχών παντοκατευθυντικού τύπου

Μειονεκτήματα:

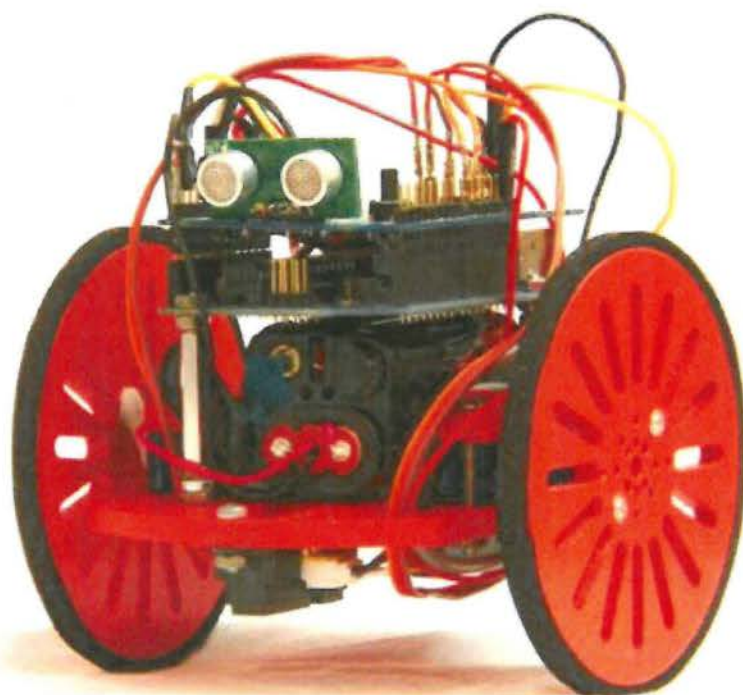
- Σχετικά μικρή ταχύτητα κίνησης οχήματος
- Υψηλή κατανάλωση ρεύματος έχοντας 4 κινητήρες για την κίνηση.

Πλεονεκτήματα:

- Ελευθερία κατεύθυνσης
- Σταθερότητα

1.1.4 Δύο τροχών - “self-balancing robots”

Αυτοί οι τύποι οχημάτων απαιτούν ισορροπία για να κινηθούν. Μιλώντας για ρομποτικό όχημα δύο τροχών μιλάμε για “self-balancing robots”. Αυτός ο τύπος οχήματος είναι ιδιαίτερα πολύπλοκος καθώς εκτός απ’ το να επιτευχθεί η κίνηση μας ενδιαφέρει και να επιτευχθεί η ευστάθεια του όλου του σώματος. Αμέσως οι παράγοντες της κατασκευής πληθαίνουν, έχουμε ανάγκη για πιο γρήγορη επεξεργαστική ισχύη και παραπάνω αισθητήρες.



Εικ. 1.1.4 Self-balancing robot

Στην παραπάνω εικόνα βλέπουμε ένα ρομποτικό όχημα τύπου self-balancing όπου για να κρατήσει την ισορροπία του χρησιμοποιεί δυο υπέρυθρους αισθητήρες στην βάση του οχήματος εμπρός και πίσω. Αναλόγως το επίπεδο των Volt που δείχνουν οι αισθητήρες και με την χρήση PID ελεγκτή εξισορροπείται το όχημα.

1.1.5 Όχημα με παντοκατευθυντικούς τροχούς

Αυτός ο τύπος οχήματος υλοποιείται συνήθως με ένα από τους δύο παρακάτω παντοκατευθυντικούς τροχούς:

Omni



Mecanum



Τα παρακάτω οχήματα έχουν την δυνατότητα να κινηθούν προς όλες τις κατευθύνσεις κάνοντας αλλαγή περιστροφής του κινητήρα, αλλαγή ταχύτητας του κινητήρα όπως επίσης παράμετρος είναι και ποιοί κινητήρες λειτουργούν για την εκάστοτε κίνηση .

Αυτός ο τύπος αποτελεί έναν απο τους πιο σύγχρονους τύπους οχημάτων καθώς έχει εξαιρετική συμπεριφορά στην κίνηση καθώς μπορεί να μεταβεί σε οποιαδήποτε κατεύθυνση στο επίπεδο.



Εικ. 1.1.5 α) Ρομποτικό όχημα με τροχούς τύπου Mecanum.



Εικ. 1.1.5 β) Όχημα βιομηχανικού τύπου με τροχούς τύπου Mecanum.

Πλεονεκτήματα :

- Ελευθερία κίνησης προς όλες τις κατευθύνσεις
- Σταθερότητα

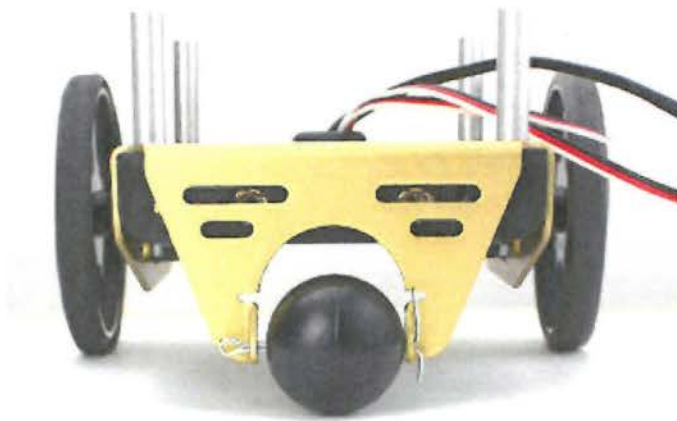
Μειονεκτήματα :

- Πολυπλοκότητα κατασκευής
- Υψηλή κατανάλωση ρεύματος έχοντας 4 κινητήρες για την κίνηση.

1.1.6 Τριών τροχών

Αυτός ο τύπος οχήματος είναι αυτός που επιλέχτηκε για την κατασκευή του οχήματος αποφυγής εμποδίων στο επίπεδο.

Περιλαμβάνει τρεις τροχούς εκ των οποίων οι δυο κινούνται με ξεχωριστούς κινητήρες ο καθένας και ο τρίτος τροχός ακολουθεί το όχημα δίνοντας ισορροπία. Ο τρίτος τροχός συνήθως είναι κάποιου είδους σφαίρα ή ρόδα τύπου trolley.



Εικ. 1.1.6 α) Τριών τροχών με βοηθητικό σφαιρικό τροχό.



Εικ. 1.1.6 β) Τριών τροχών με βοηθητική trolley ρόδα

Μειονεκτήματα :

- Χαμηλή ταχύτητα κίνησης οχήματος

Πλεονεκτήματα :

- Εύκολη κατασκευή
- Ελευθερία κίνησης προς όλες τις κατευθύνσεις
- Σταθερότητα
- Χαμηλή κατανάλωση ρεύματος

1.2 Μέθοδοι αναγνώρισης περιβάλλοντος

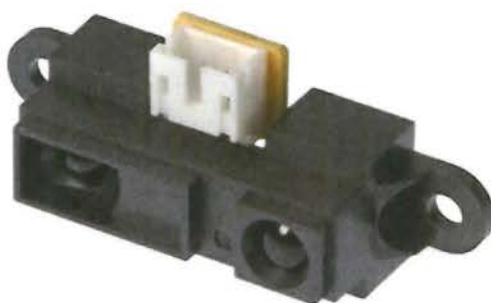
Για να μπορούμε να επιτύχουμε την αποφυγή εμποδίων χρειάζεται να αναπτυχθεί η δυνατότητα της αναγνώρισης του χώρου. Χωρίς την αίσθηση του χώρου το ρομποτικό όχημα είναι φυσικά αδύνατο να αποφύγει τα εμπόδια στο χώρο.

1.2.1 Αισθητήρες απόστασης

1.2.1.1 Αισθητήρες υπέρυθρων ακτινών

Αυτός ο τύπος αισθητήρων έχει ένα πομπό υπέρυθρων ακτινών και ένα δέκτη. Η ακτίνα ξεκινάει απ'τον πομπό συγκρούεται με το εμπόδιο, αντανακλάει και το λαμβάνει η δέκτης.

Η έξοδος του αισθητήρα δίνει την μέτρηση σε Volt. Αυτή η μέτρηση για να γίνει πραγματική τιμή χρειάζεται να υποστεί μετατροπή απο μικροελεγκτή βάσει του έντυπου χαρακτηριστικών του εκάστοτε αισθητήρα , ή εναλλακτικά να γίνει βαθμονόμηση (callibration) από τον χρήστη του αισθητήρα.



Εικ. 1.2.1.1 Αισθητήρας απόστασης υπερύθρων

Πλεονεκτήματα:

- Καλή ακρίβεια σε σχέση με άλλους αισθητήρες
- Απλό στην χρήση

Μειονεκτήματα:

- Προβλήματα αντανάκλασης της ακτίνας αναλόγως το χρώμα του αντικειμένου

1.2.1.2 Αισθητήρες υπερηχητικών παλμών

Αυτός ο τύπος αισθητήρων έχει ένα πομπό υπερηχητικών κυμάτων και ένα δέκτη. Το κύμα ξεκινάει απ'τον πομπό συγκρούεται με το εμπόδιο, αντανακλάει και το λαμβάνει η δέκτης.

Για να πάρουμε μέτρηση κάνοντας χρήση αυτού του αισθητήρα θα χρειαστούμε ένα μικροελεγκτή για να χρησιμοποιήσουμε το χρονόμετρο (timer). Πυροδοτώντας τον αισθητήρα να στείλει παλμό ξεκινάμε μαζί και το χρονόμετρο. Μόλις ο δέκτης λάβει τον παλμό σταματάμε το χρονόμετρο. Σαν μέτρηση έχουμε τον χρόνο που ταξίδεψε ο παλμός μέχρι το εμπόδιο και να γυρίσει πίσω. Για να μετατρέψουμε τον αυτόν χρόνο σε απόσταση πρέπει να γίνουν κάποιες αντικαταστάσεις στον τύπο που περιγράφει την ταχύτητα του ήχου ($v = 343.2 \text{ m/s}$) ως προς τις μονάδες μέτρησης .

Για να γίνει ο τύπος συμβατός με την μέτρηση μετατρέπουμε τα **seconds** σε **milliseconds** και τα **meters** σε **centimeters** . Αντικαθιστώντας τα milliseconds με την μέτρηση του χρονομέτρου ο τύπος μας δίνει την συνολική απόσταση που διάνυσε ο παλμός απ'την εκπομπή μέχρι την λήψη. Για την απόσταση μεταξύ του εμποδίου και του αισθητήρα απλώς διαιρούμε δια δυο την απόσταση που μας έδωσε ο τύπος.



Εικ. 1.2.1.2 Αισθητήρας υπερηχητικών παλμών

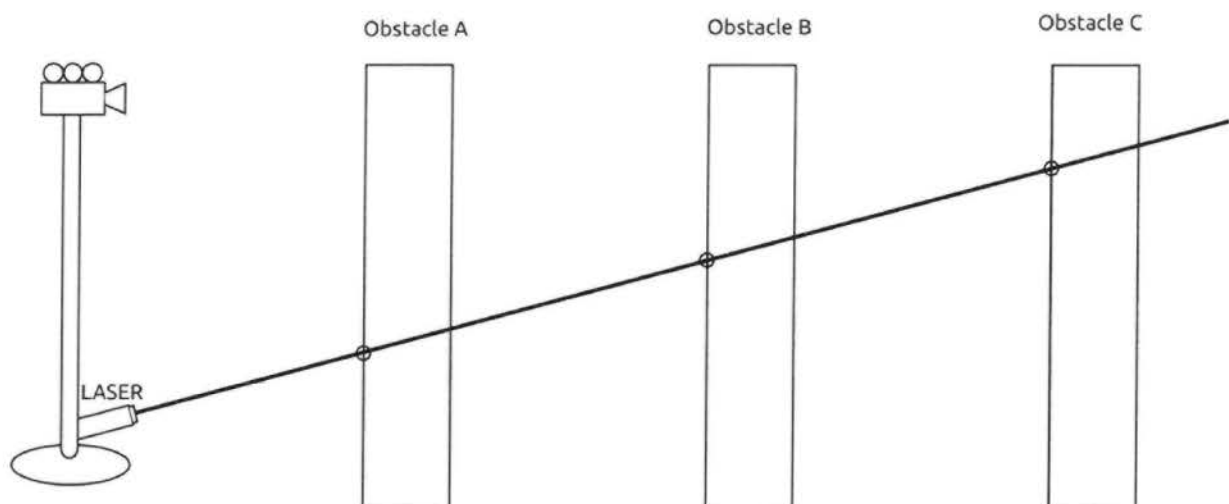
Πλεονεκτήματα:

- Μεγάλη εμβέλεια (2 cm έως 5 m)

Μειονεκτήματα:

- Προβλήματα αντάνακλασης της ακτίνας αναλόγως το υλικό του αντικειμένου
- Πολύπλοκο στην χρήση απλών εφαρμογών

1.2.1.3 Γεωμετρικός υπολογισμός εκπομπής ακτίνας Laser



Ένας πομπός υπέρυθρου laser είναι τοποθετημένος να στοχεύει τις ακτίνες του ύπο γωνία . Στην ίδια κάθετο που είναι τοποθετημένος ο πομπός laser είναι τοποθετημένη και μια ειδική κάμερα υπερήθρων.

Ο πομπός στοχεύει υπό γωνία με αποτέλεσμα το στίγμα του laser να εντοπίζεται σε διαφορετικό ύψος αναλόγως την απόσταση του αντικειμένου. Η κάμερα βλέπει μόνο το υπέρυθρο στίγμα. Με μία βαθμονόμηση (calibration) και γεωμετρικούς υπολογισμούς μπορούμε να διακρίνουμε μετρήσεις. Η γκάμα των αποστάσεων εξαρτάται από την γωνία που θα τοποθετηθεί ο πομπός.

Πλεονεκτήματα:

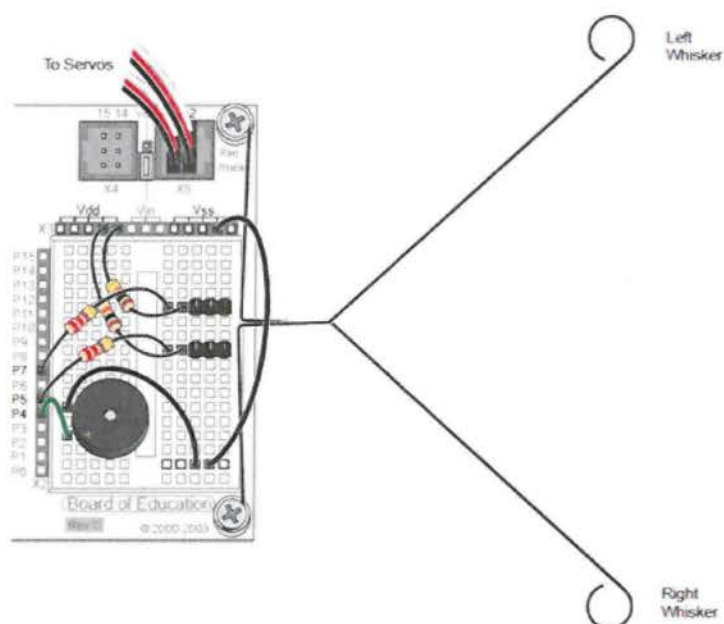
- Μεγάλη εμβέλεια εκπομπής

Μειονεκτήματα:

- Πολυπλοκότητα κατασκευής
- Ανακριβείς μετρήσεις

Συνήθως χρησιμοποιείτε σε εφαρμογές αναγνώρισης περιβάλλοντος στις οποίες δεν μας νοιάζει η ακρίβεια της μέτρησης .

1.2.2 Αισθητήρας επαφής



Αυτός το είδος αισθητήρα είναι αισθητήρας επαφής και είναι σχετικά απλό. Όταν κάποιο εμπόδιο ασκήσει δύναμη επάνω σε κάποιο **whisker** στέλνεται ένα λογικό 0 ή 1 στον μικροελεγκτή και αναλόγως διαχειρίζεται το συμβάν. Η εφαρμογή αυτού του αισθητήρα σ'ένα ρομποτικό όχημα αποφυγής εμποδίων είναι ένα επίπεδο ασφαλείας πριν από την σύγκρουση του οχήματος.

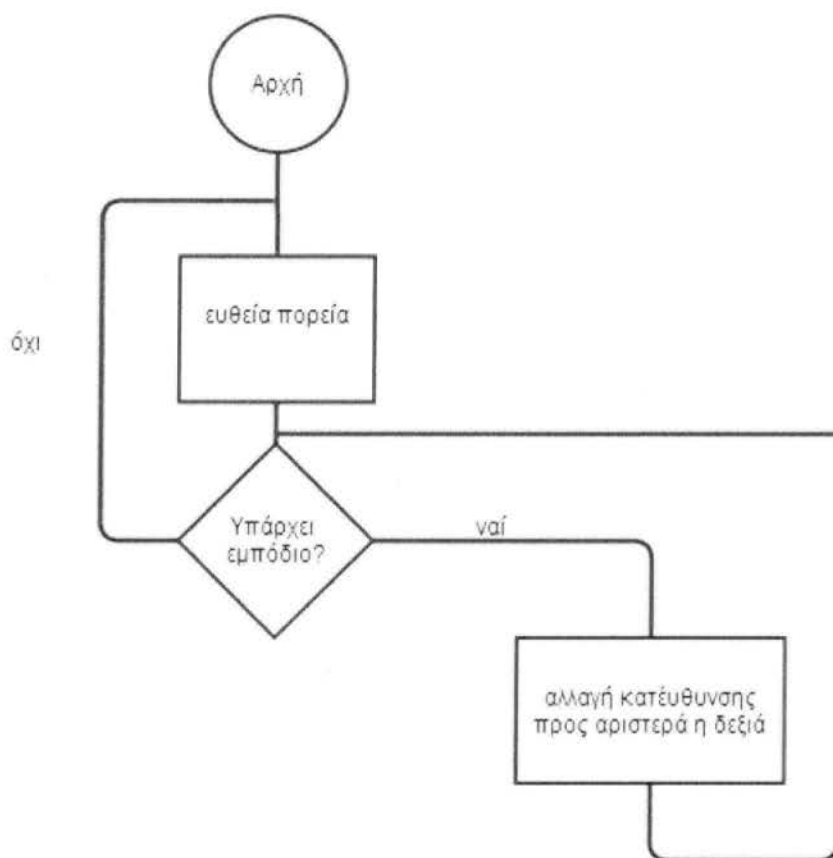
1.3 Μέθοδοι αποφυγής εμποδίων

1.3.1 Μέθοδος αλλαγής κατεύθυνσης

Ένας τρόπος για να επιτευχθεί η αποφυγή εμποδίων είναι η συνεχής μετατόπιση του οχήματος έως ότου εντοπιστεί το αντικείμενο και κάνει αλλαγή κατεύθυνσης μέχρι να εντοπιστεί ξανά νέο εμπόδιο.

Αυτή η μέθοδος είναι αρκετά απλή όσον αφορά την προγραμματιστική υλοποίηση, όσο και την κατασκευαστική υλοποίηση. Προγραμματιστικά απλά ελέγχουμε μέσω κάποιου αισθητήρα εάν υπάρχει εμπόδιο εάν όχι συνεχίζει στην πορεία του.

Διάγραμμα ροής μεθόδου



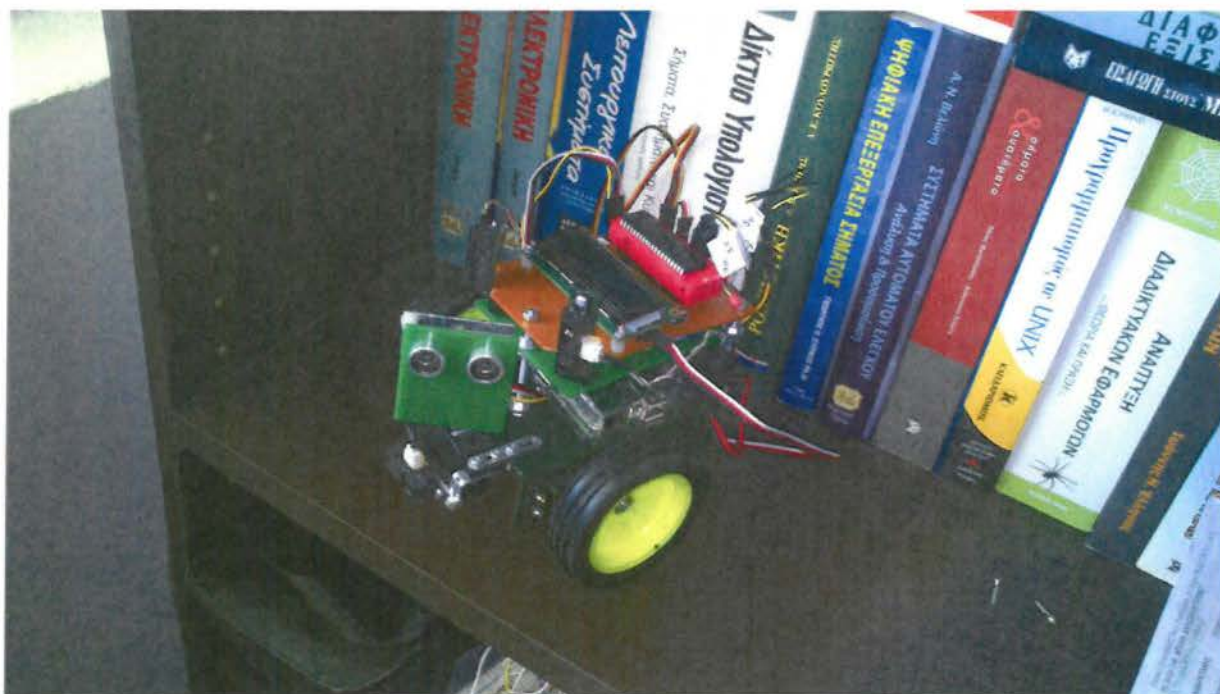
1.3.2 Μέθοδος επιλογής κατεύθυνσης

Αυτή η μέθοδος υπερέχει της προηγούμενης μεθόδου διότι αποφεύγει άσκοπες κινήσεις χώρο. Η λογική της τεχνικής αυτής είναι ότι πέρνονται μετρήσεις σε διακριτές θέσεις σταθερές ως προς το όχημα και επιλέγεται η καλύτερη (πιο απομακρυσμένη) πορεία. Αυτό προϋποθέτει παραπάνω μηχανικά εξαρτήματα σε σχέση με την προηγούμενη μέθοδο. Χρειάζεται ένα κινητήρα για περιστροφή του αισθητήρα μέτρησης απόστασης.

Διάγραμμα ροής μεθόδου



2. Ρομποτικό όχημα αποφυγής εμποδίων



2.1 Υλικά και κόστος κατασκευής

Φύλλο αλουμινίου πάχους 1mm και επεξεργασία του (κόψιμο και στρατζάρισμα) - 8 €

Ρόδες οχήματος απο παλίο τηλεκατευθυνόμενο όχημα.

Διάφορες βίδες - 6 €

Τα Plexiglass που χρησιμοποιήθηκαν ήταν χορηγία του καταστήματος PLEXIGLASS - ΚΩΣΤΑΚΗ ΑΦΟΙ ΟΕ στην Αργυρούπολη Αττικής.

Τα διάφορα καλώδια προυπήρχαν απο παλιότερες κατασκευές.

Το τροφοδοτικό προυπήρχε από παλιότερη κατασκευή.

1 μικροελεγκτής PIC 16F887 χορηγία απο την Microchip.

1 x Ultrasonic Module HC-SR04 Distance Sensor – 6 €

1 x PING Mounting Bracket Kit - 20 €

2 x Hitec Continuous Servo HSR-1425CR - 2 x 27 = 54 €

3 x Infrared Proximity Sensor - Sharp GP2Y0A21YK - 3 x 14 = 42 €

1 x Polyethylene Ball 1" – 5 €

1 x LCD 16x2 Chars – 10 €

1 x Ποτενσιόμετρο 47K826M – 0.10 €

1 x PIC 40-pin socket – 5 €

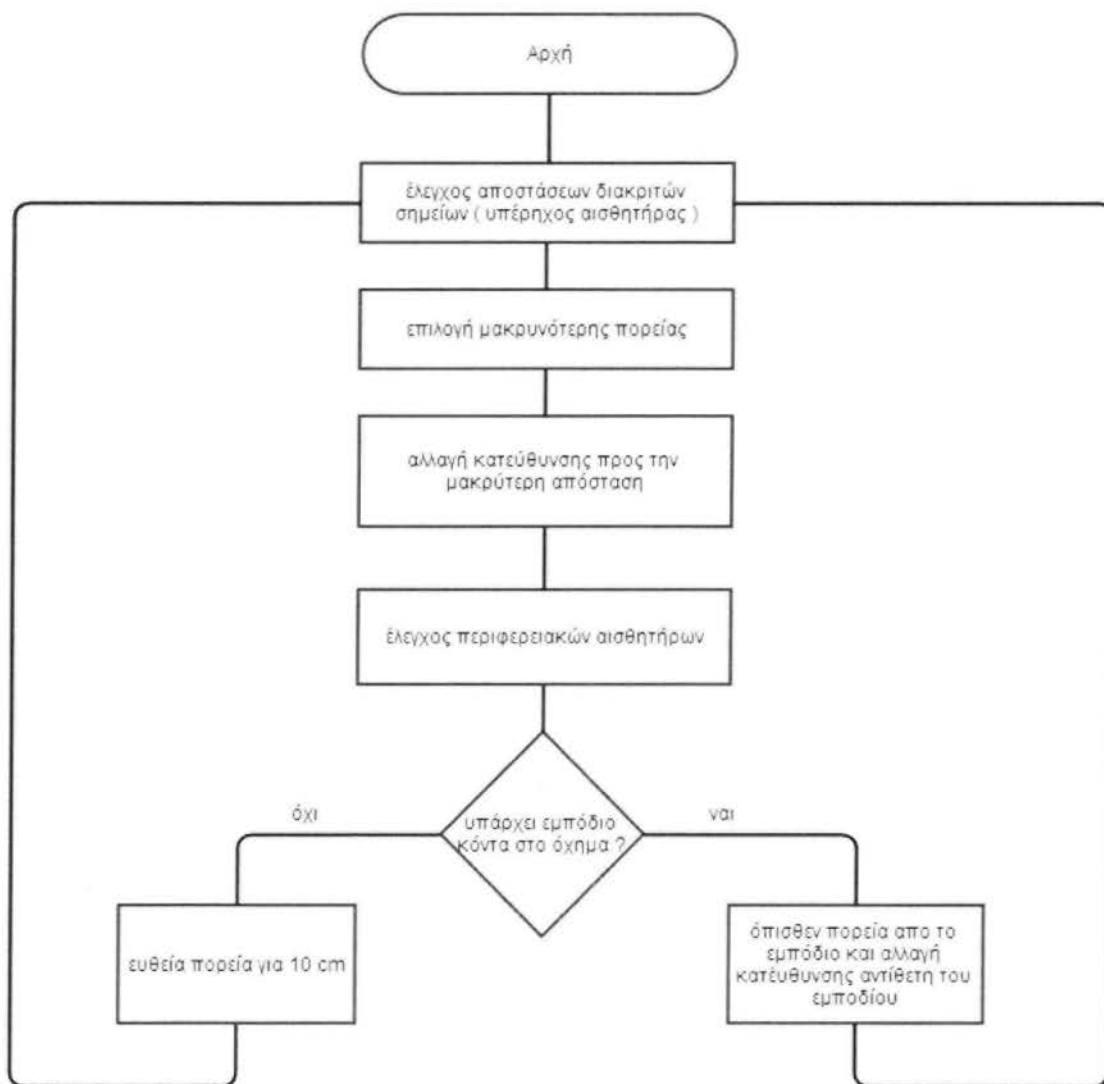
1 x DFRobot Lithium Polymer Battery 2200mAh 7.4V – 35 €

Σύνολο : 191.10 €

2.2 Φιλοσοφία του ρομποτικού οχήματος

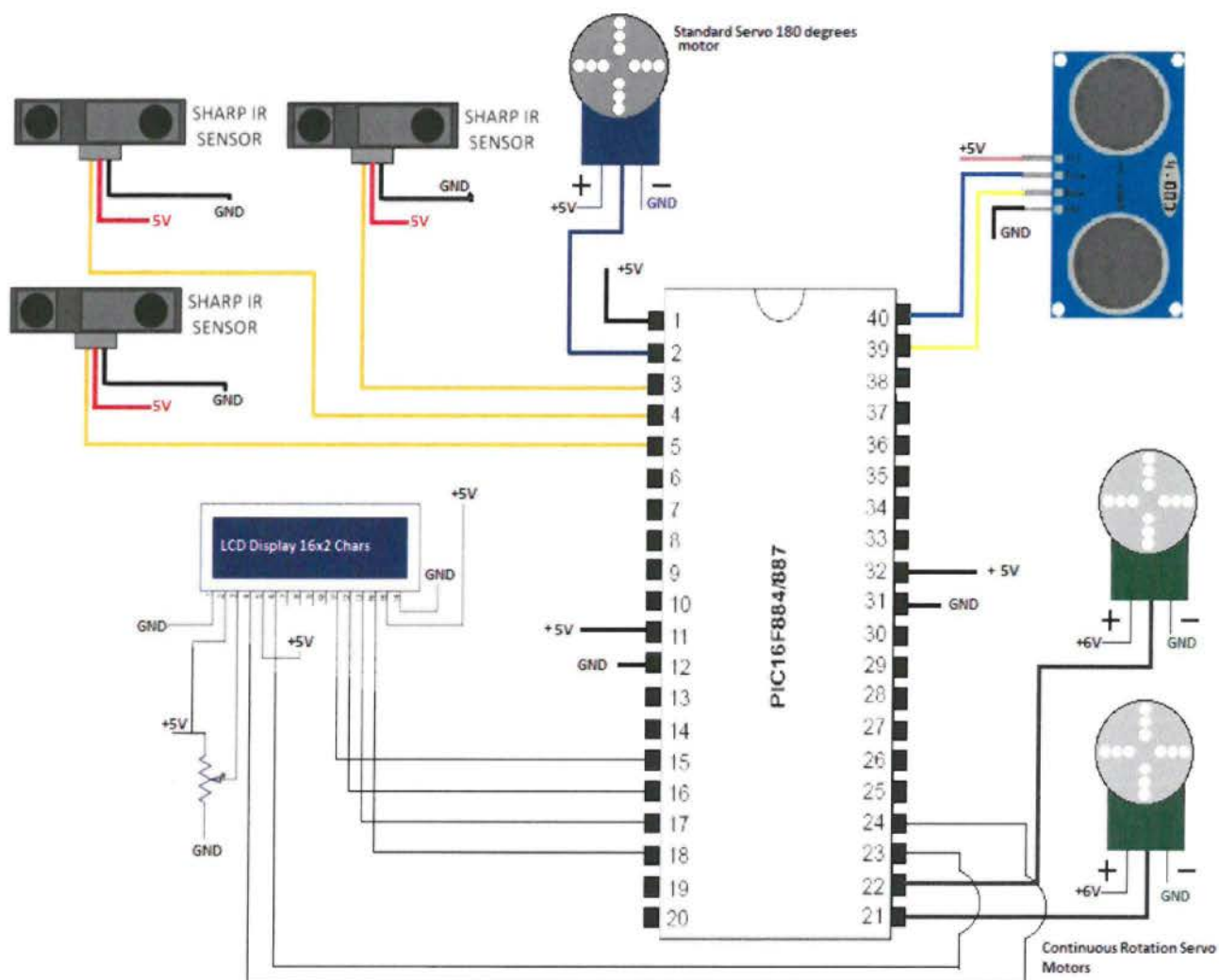
Το ρομποτικό όχημα λειτουργεί σύμφωνα με την μέθοδο επιλογής κατεύθυνσης. Ο αισθητήρας υπέρηχων παλμών δεν αρκεί για να αποφευχθούν τα τυχόν εμπόδια που μπορεί να βρεθούν στην διαδρομή του οχήματος που δεν τα είχε ανιχνεύσει ο αισθητήρας. Για αυτόν τον λόγο χρειάστηκε να προστεθούν και αισθητήρες μικρής εμβέλειας για να ελέγχεται η ύπαρξη αντικειμένων στις επικίνδυνες γωνίες σύγκρουσης. Εξαιτίας του μικροελεγκτή αυτοί οι αισθητήρες δεν μπορούν να δουλέυσουν παράλληλα με την λειτουργία του συστήματος οπότε η λειτουργία θα πρέπει να είναι γραμμική. Για αυτόν λόγο το ρομπότ ακολουθεί την παρακάτω ακολουθία:

Διάγραμμα ροής



2.3 Ηλεκτρονικά υλικά κατασκευής

2.3.1 Κύκλωμα κατασκευής



2.3.2 Κινητήρες servo συνεχής περιστροφής



Hitec Continuous Servo HSR-1425CR

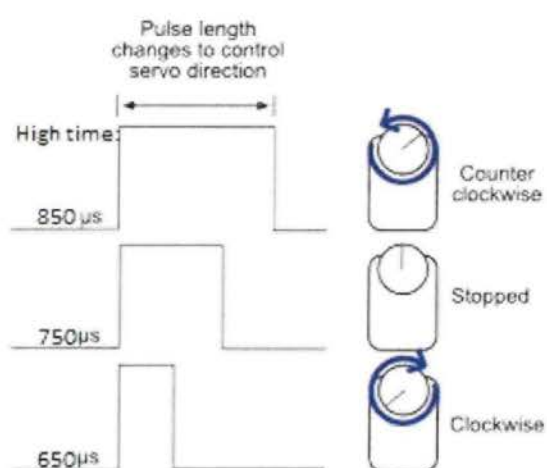
- Τύπος κινητήρα : “Brushed”
- Επικοινωνία : Με διαμόρφωση πλάτους παλμού - PWM
- Τάσεις λειτουργίας : 4.8 - 6.0 Volts
- Θερμοκρασίες λειτουργίας : -20 to +60 Degree C (-4F to +140F)
- Ταχύτητα στα 4.8V : 44 RPM (χωρίς φορτίο)
- Ταχύτητα στα 6.0V : 52 RPM (χωρίς φορτίο)
- Ροπή στα 4.8V : 38.8 oz-in (2.8 kg-cm)
- Ροπή στα 6.0V : 42 oz-in (3.1 kg-cm)

Οι σερβοκινητήρες δουλεύουν με είσοδο παλμών. Ο παλμός έχει συνολικό μήκος 20ms. Κάθε παλμός - κίνηση είναι διαφορετικός αλλά έχει συνολικό μήκος 20ms. Αυτός ο τύπος σερβοκινητήρα μοιάζει με βηματικό κινητήρα ως προς τον τρόπο λειτουργίας. Στον βηματικό δίνεται μια ακολουθία για να γίνει ένα βήμα. Σ' αυτόν το τύπο σερβοκινητήρα δίνεται ένας παλμός για να κάνει κάποια κίνηση. Κάνοντας βαθμονόμηση και μερικές δοκιμές μαθαίνουμε με πόσες επαναλήψεις ο κινητήρας πραγματοποιεί μισό και ένα κύκλο και πάνω σ' αυτό στηριζόμαστε για την λειτουργία του.

Για την λειτουργία του κινητήρα υπάρχουν τρεις παλμοί που καθορίζουν την κίνηση που θα κάνει και είναι οι παρακάτω:

	Διάρκεια high παλμού
Αριστερόστροφη κίνηση	850 μ s
Ακινητοποίηση κινητήρα	750 μ s
Δεξιόστροφη κίνηση	650 μ s

Πίνακας 2.3.2



Εικ. 2.3.2 Πλάτος παλμών

Παρακάτω φαίνεται ένα παράδειγμα πως σταματάνε και οι δυο κινητήρες :

```
SERV2=1;  
SERV1=1;  
delay_us(750);  
SERV2=0;  
SERV1=0;  
delay_ms(20);
```

2.3.3 Κινητήρας servo περιστροφής 180 μοιρών

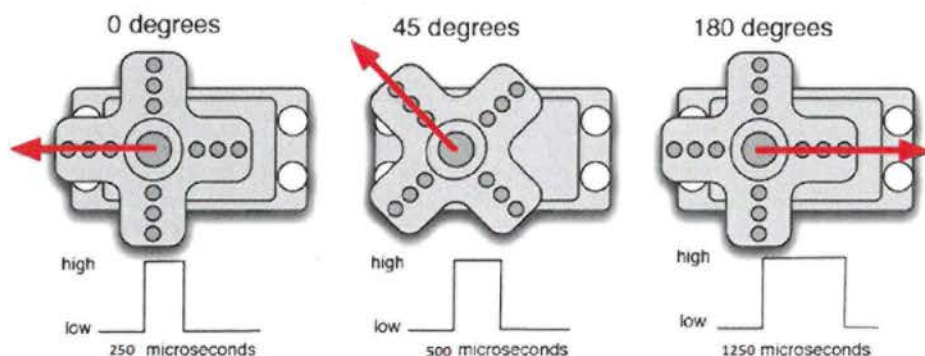


Parallax Standard Servo

- Μπορεί να πάει σε οποιαδήποτε θέση από 0 έως 180 μοίρες
- Τροφοδοσία : 4 έως 6 V
- 38 oz-in ροπή στα 6 V
- Operating temp range: +14 to +144 °F (-10 to +50 °C)
- Βάρος μόνο 1.55 oz (44 g)
- Δέχεται βίδες
- Επικοινωνία : Με διαμόρφωση πλάτους παλμού

Οι σερβοκινητήρες δουλεύουν με είσοδο παλμών. Κάθε παλμός αντιπροσωπεύει μια γωνία, αυτό σημαίνει ότι ανεξάρτητα την θέση στην οποία βρίσκεται ο κινητήρας εισάγοντας έναν συγκεκριμένο παλμό θα πάει σε αυτήν την θέση.

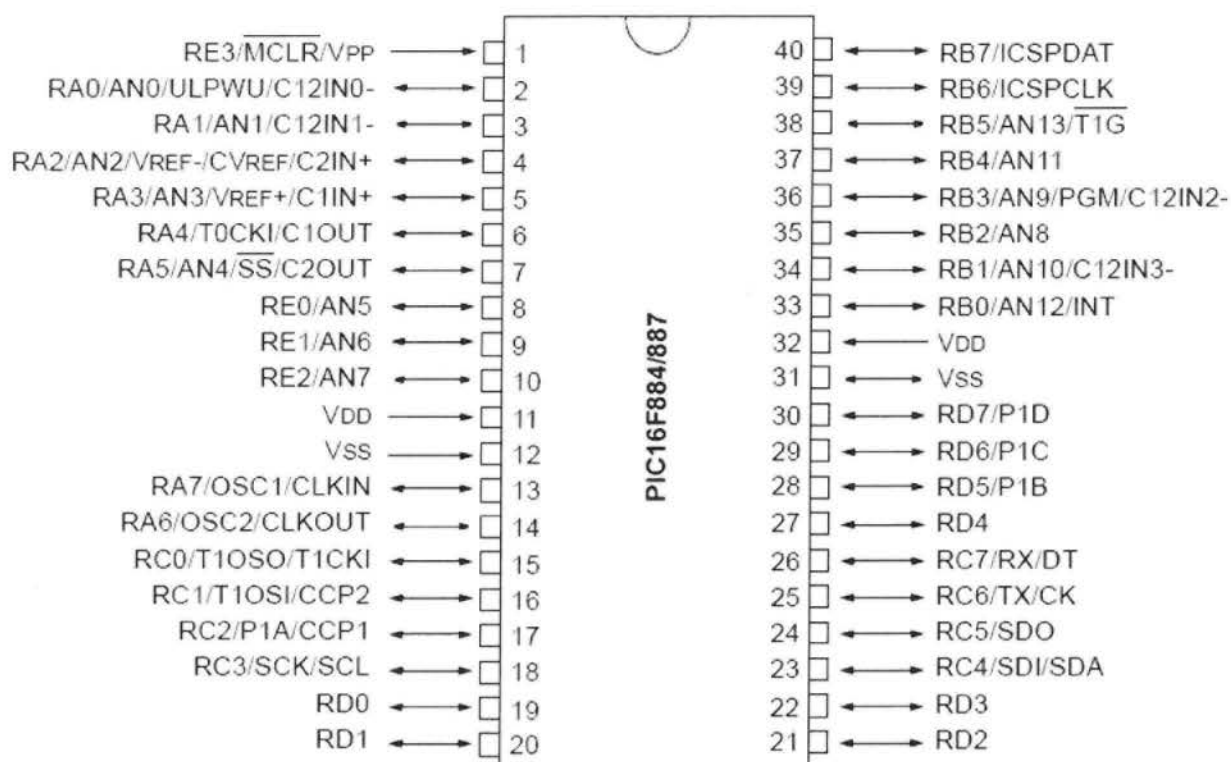
Η πρώτη θέση (-90 μοίρες) ξεκινώντας από αριστερά είναι ο παλμός που είναι high για 250 μs. Η τελευταία θέση (90 μοίρες) είναι ο παλμός που είναι high για high 1250 μs. Υπολογίζοντας τον χρόνο που πρέπει να είναι high βρίσκουμε και τα τις υπόλοιπες γωνίες.



Παρακάτω φαίνεται ένα παράδειγμα με τις γωνίες και την δημιουργία των παλμών:

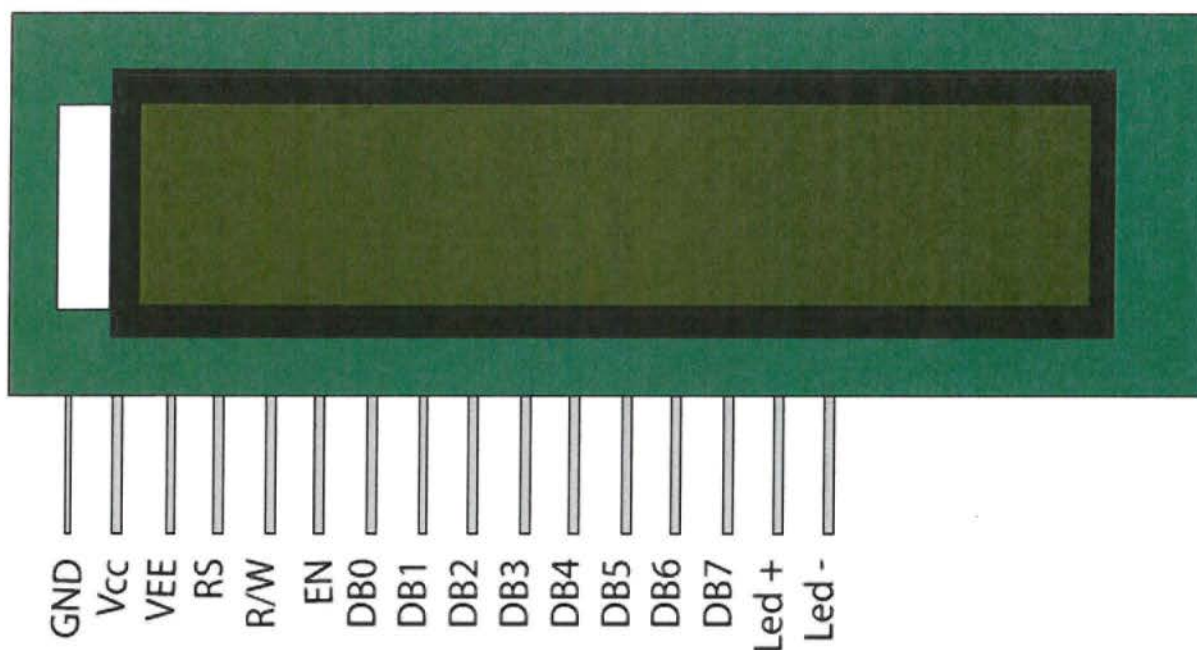
```
if(angle == - 90 ){
    SCANNER=1;
    delay_us(250);
    SCANNER=0;
    delay_ms(20);
}
if(angle == - 45 ){
    SCANNER=1;
    delay_us(500);
    SCANNER=0;
    delay_ms(20);
}
if(angle == 0 ){
    SCANNER=1;
    delay_us(750);
    SCANNER=0;
    delay_ms(20);
}
if(angle == 45 ){
    SCANNER=1;
    delay_us(1000);
    SCANNER=0;
    delay_ms(20);
}
if(angle == 90 ){
    SCANNER=1;
    delay_us(1250);
    SCANNER=0;
    delay_ms(20);
}
```


2.3.4 Μικροελεγκτής PIC 16F887



Ισχυρός και εύκολος στον προγραμματισμό CMOS FLASH-based 8-bit μικροελεγκτής. Ο PIC16F887 διαθέτει 256 bytes της μνήμης EEPROM, κρύσταλλο χρονισμού 8MHz, 2 συγκριτές, 14 κανάλια 10-bit αναλογικό σε ψηφιακό (A/D) μετατροπέα, 1 PWM, σύγχρονη σειριακή πόρτα και την λειτουργία Enhanced Universal Asynchronous Receiver Transmitter (EUSART).

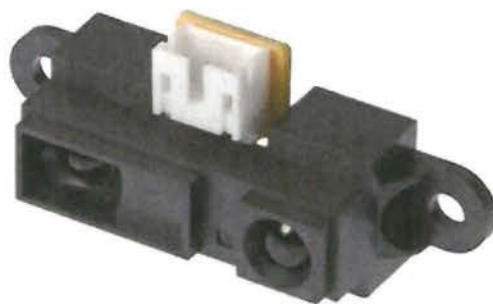
2.3.5 LCD οθόνη 16x2 χαρακτήρων



#	Λειτουργία	Όνομα
1	Γείωση (0V)	Ground
2	Πηγή τάσεως ; 5V (4.7V – 5.3V)	Vcc
3	Τάση επιπέδου φωτεινότητας , μέσω αντίστασης	V _{EE}
4	Selects command register when low; and data register when high	Register Select
5	Low για εγγραφή στον καταχωρητή; High για να διάβασμα απο τον καταχωρητή	Read/write
6	Ακροδέκτης ενεργοποίησης λειτουργίας οθόνης	Enable
7	8-bit ακροδέκτες δεδομένων	DB0
8		DB1
9		DB2
10		DB3
11		DB4
12		DB5
13		DB6
14		DB7
15	Πηγή τάσης Backlight V _{CC} (5V)	Led+
16	Γείωση Backlight Ground (0V)	Led-

Για την χρήση της LCD οθόνης έχει φροντίσει η πλατφόρμα της microC και έχει βιβλιοθήκες για την οδήγηση της.

2.3.6 Αισθητήρες μέτρησης αποστάσεων



Αισθητήρας απόστασης SHARP INFRARED SENSOR GP2D120XJ00F

- Τάση λειτουργίας: 4,5 V έως 5,5 V
- Μέση κατανάλωση ρεύματος: 33 mA
- Εύρος απόστασης μέτρησης: 4 εκ. έως 30 εκ. (1,5 " έως 12")
- Τύπος εξόδου: Αναλογική τάση
- Χρόνος απόκρισης: 38 ± 10 ms
- Μεγέθοι: 29,5 × 13,0 × 13,5 χιλιοστά (1,16 × 0,5 × 0,53 ")
- Βάρος: 3,5 g (0,12 oz)

Ο αισθητήρας αυτός έχοντας έξοδο αναλογική χρειάζεται για τις διάφορες εφαρμογές μετατροπή σε ψηφιακή μορφή. Για να γίνει αυτό χρειαζόμαστε έναν A/D converter. Επόμενο βήμα για την ορθή λειτουργία του είναι να γίνει βαθμονόμηση βάση των παρακάτω χαρακτηριστικών που δίνονται απ'τον κατασκευαστή.

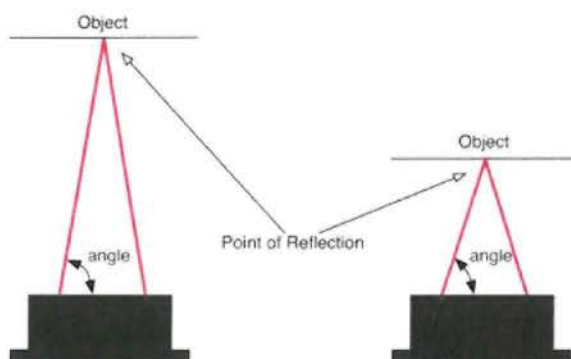
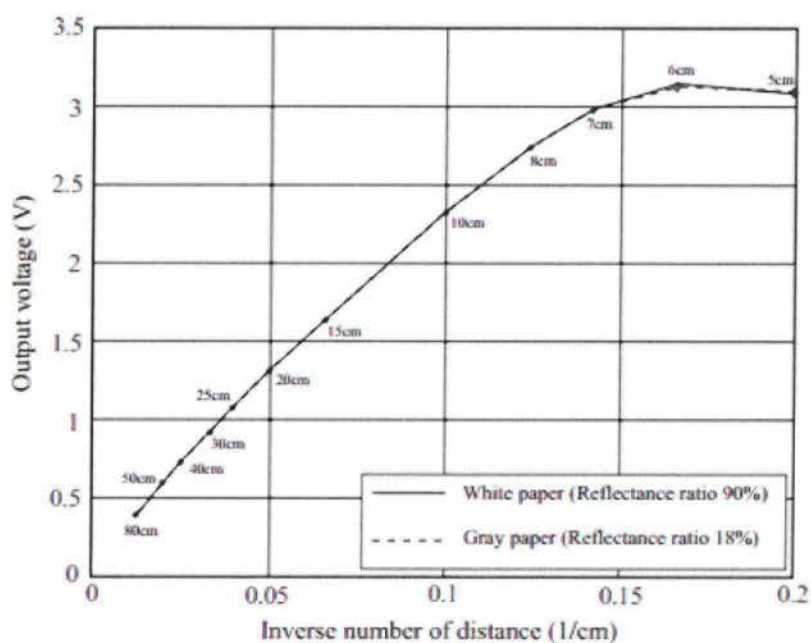
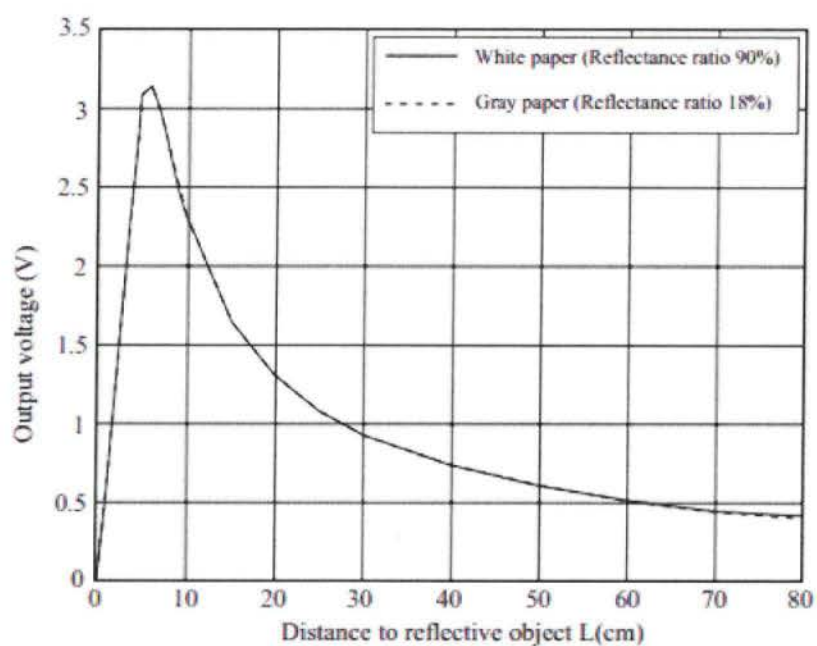
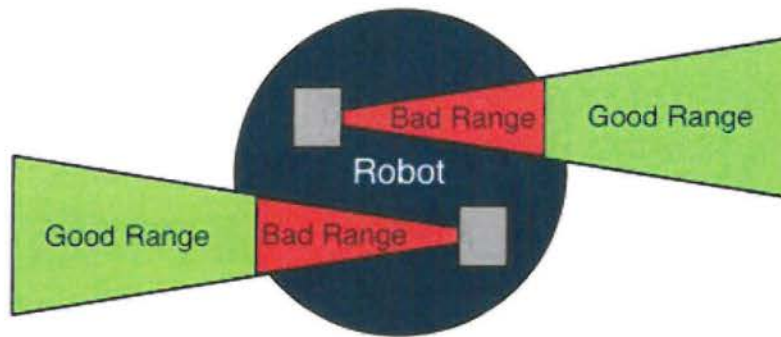


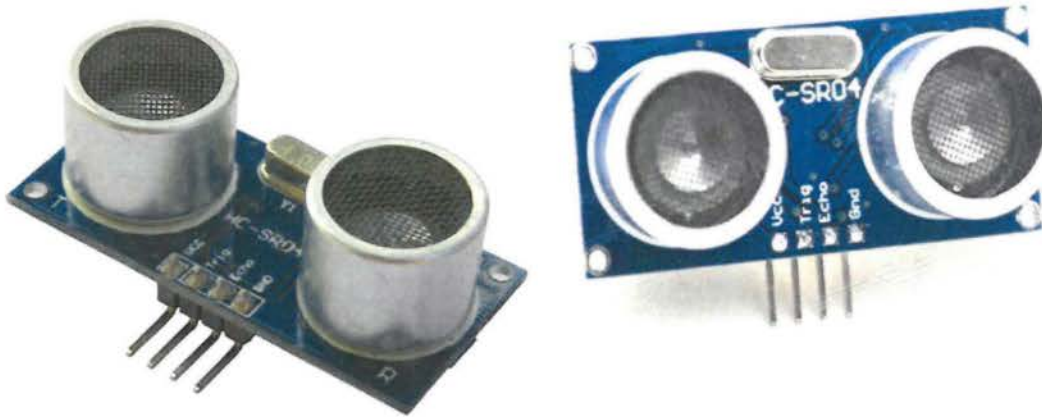
Fig. 2 Example of distance measuring characteristics(output)



Εικ. 2.3.6 Χαρακτηριστική αποστάσεων - τάσεων

Απο ότι παρατηρείται απο τις χαρακτηριστικές του κατασκευαστή οι υπέρυθροι αισθητήρες έχουν κακή ακρίβεια στις κοντινές αποστάσεις . Αυτό είναι μια πληροφορία που δίνεται στους χρήστες του αισθητήρα για να μεριμνήσουν αναλόγως. Για παράδειγμα τοποθέτηση του αισθητήρα πιο μακριά απο την περιοχή των μετρήσεων που ενδιαφέρει την εκάστοτε εφαρμογή.





Αισθητήρας απόστασης Ultrasonic Ranging Distance Sensor HC-SR04

- Τάση λειτουργίας : 5V DC
- Κατανάλωση ρεύματος : < 2mA
- Αποτελεσματική γωνία : < 15 °
- Εύρος απόστασης μέτρησης: 2cm - 500 εκατοστά
- Ανάλυση : 0,3 cm (δυνατότητα χρήσης μέχρι και αυτής της ανάλυσης)

Ο αισθητήρας έχει 4 ακροδέκτες : **Vcc** (τάση λειτουργίας), **Trigger**, **Echo**, **GND** (γείωση)

Για την λειτουργία του αισθητήρα χρειάζεται να γίνει χρήση του χρονομέτρου του μικροελεγκτή.

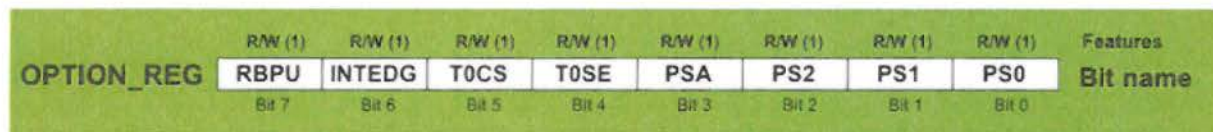
Όταν ο αισθητήρας πάρει τάση High στο trigger στέλνει το ηχητικό παλμό. Όταν γίνει το trigger στον αισθητήρα δίνεται low τάση στο trigger για να μην συνεχίσει επ'άπειρον να στέλνει παλμούς. Μόλις ο παλμός βρεί το εμπόδιο ,επιστρέφει και τον λάβει ο δέκτης του αισθητήρα τότε ο ακροδέκτης echo γίνεται high .

Στον μικροελεγκτή πρέπει να χρονομετρηθεί αυτή η διάρκεια μεταξύ της εκπομπής του παλμού μέχρι την λήψη του παλμού. Ο χρόνος αυτός αντιστοιχεί στην διπλάσια απόσταση μεταξύ αισθητήρα και εμποδίου αφού ο παλμός διανίει δυο φορές την απόσταση αυτή με το να πάει ως εκεί και να επιστρέψει. Ο γενικός τύπος που δίνει την απόσταση έχοντας την διάρκεια αυτή δίνεται παρακάτω:

$$\text{Απόσταση} = (\text{Διάρκεια λήψης}) * (\text{Ταχύτητα ήχου : } 340\text{m/s}) / 2$$

Η ανάλυση των μετρήσεων εξαρτάται απο τον prescaler που θα χρησιμοποιήσουμε στην χρονομέτρηση. Παρακάτω φαίνεται ο πίνακας που τα εξηγεί.

Ο καταχωρητής με τον οποίο ρυθμίζουμε τον prescaler είναι ο OPTION_REG και φαίνεται παρακάτω .



Legend

R/W (1)	Readable/Writable bit After reset, bit is set
---------	--

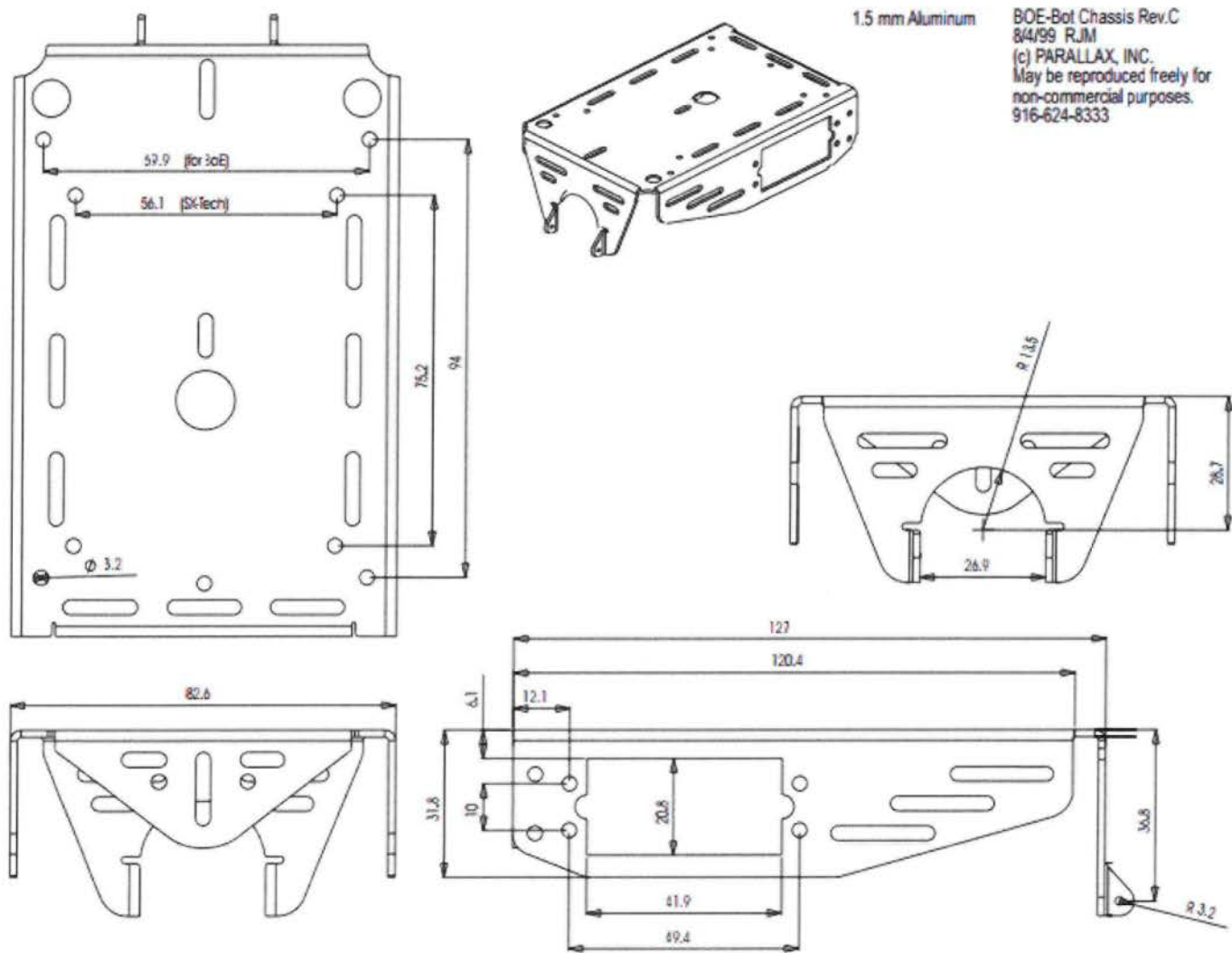
PS2	PS1	PS0	TMRO
0	0	0	1:2
0	0	1	1:4
0	1	0	1:8
0	1	1	1:16
1	0	0	1:32
1	0	1	1:64
1	1	0	1:128
1	1	1	1:256

Στην ουσία όσο μεγαλώνει ο prescaler αλλάζει η διακριτότητα των μετρήσεων (ανάλυση) με μειονέκτημα ότι ελατώνεται η εμβέλεια του αισθητήρα. Όσο μικραίνει ο prescaler τόσο μεγαλώνει η εμβέλεια αλλά χαλάει η διακριτότητα των μετρήσεων.

Άρα η ανάλυση με την εμβέλεια είναι αντιστρόφως ανάλογες.

2.4 Σχέδια κατασκευής οχήματος

Τα παρακάτω σχέδια ανήκουν στην Parallax για τα ρομποτικά οχήματα τύπου Boe Bot. Αυτά τα σχέδια χρησιμοποιήθηκαν και για την κατασκευή του οχήματος με μερικές τροποποιήσεις.



2.5 Κώδικας

2.5.1 Κώδικας project

Ο κώδικας αναπτύχθηκε σε γλώσσα microC και έγινε compile σε hex μορφή μέσα απ'την πλατφόρμα της microC .

Κώδικας <functions.h>

```
void init(){  
    TRISA.ra1 = 1;        // PORTA pins that are input for Infrared Sensors  
    TRISA.ra2 = 1;  
    TRISA.ra3 = 1;  
    adc_init();          // Initialize Analog-Digital Convertor  
    TRISA.ra0=0;  
    TRISD=0;  
    ANSEL = 0;           // Configure AN pins as digital I/O  
    ANSELH = 0;  
    C1ON_bit = 0;        // Disable comparators  
    C2ON_bit = 0;  
    TRISb.RB7 = 1;  
    TRISb. RB6 = 0;  
    delay_ms(50);  
    Lcd_Init();          // Initialize LCD  
    Lcd_Cmd(_LCD_CLEAR); // Clear display  
    Lcd_Cmd(_LCD_CURSOR_OFF); // Cursor off  
}
```

```
void CustomChar(char pos_row, char pos_char) {
```

```
    Lcd_Cmd(64);  
    for (ic = 0; ic<=7; ic++) {  
        Lcd_Chr_CP(character[ic]);  
        Lcd_Cmd(_LCD_RETURN_HOME);  
        Lcd_Chr(pos_row, pos_char, 0);  
    }  
}
```

```
void timerSet(){
```

```
    INTCON = 0x00;  
    OPTION_REG.T0CS =0;  
    OPTION_REG.PSA =0;  
    OPTION_REG.PS2 = 1;    //1  
    OPTION_REG.PS1 = 0;    //0  
    OPTION_REG.PS0 = 0;    //1  
}
```

```
double opt_sense(){
```

```
    TRIG = 1;  
    delay_us(10);  
    TRIG = 0;  
    while(ECHO==0){  
        TMR0=0; }  
    while(ECHO==1){  
        time = (double)TMR0;}  
    time=time*0.017*32;    //64    //4.352
```

```

wordtostr(time,buff);
//Lcd_Out(1,1,buff); // Write text in first row
return time;
}

void stopmotor(int choicestop){ //choicestop  0--- serv1  1---serv2  2 --- stop BOTH motors
    if(choicestop==0){
        SERV1=1;
        delay_us(750);    //stop serv1
        SERV1=0;
        delay_ms(20);
    }
    if(choicestop==1){
        SERV2=1;
        delay_us(750);    //stop serv2
        SERV2=0;
        delay_ms(20);
    }
    if(choicestop==2){
        SERV2=1;
        SERV1=1;
        delay_us(750);    //stop both
        SERV2=0;
        SERV1=0;
        delay_ms(20);
    }
}
}

```

```
void vehicle_right_rotation(){  
    SERV1=1;  
    SERV2=1;  
    delay_us(650);    //oxima paei deksiostrofa - teleio  
    portd=00000000;  
    delay_ms(20);  
}
```

```
void vehicle_left_rotation(){  
    SERV1=1;  
    SERV2=1;  
    delay_us(850);    //oxima paei deksiostrofa - teleio  
    portd=00000000;  
    delay_ms(20);  
}
```

```
void move_forward(){  
    SERV1=1;  
    SERV2=1;  
    delay_us(650);    //oxima paei eftheia - teleio  
    SERV1=0;  
    delay_us(200);  
    SERV2=0;  
    delay_ms(20);  
}
```

```

void move_backward(){
    SERV1=1;
    SERV2=1;
    delay_us(650);    //oxima paei eftheia    - teleio
    SERV2=0;
    delay_us(200);
    SERV1=0;
    delay_ms(20);
}

```

```

void scanner_set_position(int angle){
    if(angle == -90){
        SCANNER=1;
        delay_us(250);    //turn aristerostrofa serv1
        SCANNER=0;
        delay_ms(20);
    }
    if(angle == -45){
        SCANNER=1;
        delay_us(500);    //turn aristerostrofa serv1
        SCANNER=0;
        delay_ms(20);
    }
    if(angle == 0){
        SCANNER=1;
        delay_us(750);    //turn aristerostrofa serv1
    }
}

```

```

    SCANNER=0;
    delay_ms(20);
}
if(angle == 45){
    SCANNER=1;
    delay_us(1000);    //turn aristerostrofa serv1
    SCANNER=0;
    delay_ms(20);
}
if(angle == 90){
    SCANNER=1;
    delay_us(1250);    //turn aristerostrofa serv1
    SCANNER=0;
    delay_ms(20);
}
}

```

```

void scan(){
    for(i=0;i<20;i++){
        scanner_set_position(-90);
    }
    delay_ms(100); //wait to get position

    m1=opt_sense(); //astoxi
    m1=opt_sense(); //swsti
    pinak[0]=m1;
}

```

```

wordtostr(m1,buff);
Lcd_Out(1,1,buff);

for(i=0;i<20;i++){
    scanner_set_position(-45);
}
delay_ms(100);    //wait to get position

m2=opt_sense();
pinak[1]=m2;
wordtostr(m2,buff);
Lcd_Out(1,1,buff);

for(i=0;i<20;i++){
    scanner_set_position(0);
}
delay_ms(100);    //wait to get position

m3=opt_sense();
pinak[2]=m3;
wordtostr(m3,buff);
Lcd_Out(1,1,buff);

for(i=0;i<20;i++){
    scanner_set_position(45);
}

```

```

delay_ms(100); //wait to get position

m4=opt_sense();
pinak[3]=m4;
wordtostr(m4,buff);
Lcd_Out(1,1,buff);

for(i=0;i<20;i++){
    scanner_set_position(90);
}
delay_ms(100); //wait to get position

m5=opt_sense();
pinak[4]=m5;
wordtostr(m5,buff);
Lcd_Out(1,1,buff);

for(i=0;i<20;i++){
    scanner_set_position(0); //reset scanner to center
}

}

```



```

void check(){

    in1=0;
    in2=0;
    in3=0;
    in1= ADC_read(2); //mprosta aisthitiras
    delay_ms(50);

    in2= ADC_read(1); //deksia aisthitiras
    delay_ms(50);

    in3= ADC_read(3); //aristera aisthitiras
    delay_ms(50);

    if(in1>200){ //mprosta aisthitiras – elegxos empodiou
        for(i=0;i<10;i++){
            move_backward();
        }
        scan();
    }
    if(in2>400){ //deksia aisthitiras – elegxos empodiou
        for(i=0;i<10;i++){
            move_backward();
        }
        for(i=0;i<7;i++){
            vehicle_right_rotation();
        }
        scan();
    }
    if(in3>400){ //aristeros aisthitiras – elegxos empodiou
        for(i=0;i<10;i++){
            move_backward();
        }
        for(i=0;i<7;i++){
            vehicle_left_rotation();
        }
        scan();
    }
    in1=0;
    in2=0;
    in3=0;
}

```

Κώδικας <main.c>

```
#define TRIG PORTB.RB6 //ULTRASONIC RANGE SENSOR [ TRIGGER pin ]
#define ECHO PORTB.RB7 //-----//----- [ ECHO pin ]
#define SCANNER PORTA.RA0 //standard servo holding sensor
#define SERV1 PORTD.RD2 //first servo at rd2
#define SERV2 PORTD.RD3 //second servo at rd3

//LCD module connections
sbit LCD_RS at RC5_bit;
sbit LCD_EN at RC4_bit;
sbit LCD_D4 at RC0_bit;
sbit LCD_D5 at RC1_bit;
sbit LCD_D6 at RC2_bit;
sbit LCD_D7 at RC3_bit;

sbit LCD_RS_Direction at TRISC5_bit;
sbit LCD_EN_Direction at TRISC4_bit;
sbit LCD_D4_Direction at TRISC0_bit;
sbit LCD_D5_Direction at TRISC1_bit;
sbit LCD_D6_Direction at TRISC2_bit;
sbit LCD_D7_Direction at TRISC3_bit;

char txt[6];
char buff[3];
char bf1[4];
char bf2[4];
char bf3[4];

char error[]="BLOCKED!";

double time,timo,m1,m2,m3,m4,m5;
```

```

int i,j,k,n,maxi,in,ic,in1,in2,in3;
double pinak[4];
double eval = 0;
const char character[] = {0,31,4,10,17,17,14,0};
#include "functions.h"
void main() {
    init();
    timerSet();
    while(1){

        scan();
        //compare which is the best way to get there
        maxi=0;
        eval=pinak[0];
        for(n=1;n<=4;n++){
            if(eval<pinak[n]){
                eval=pinak[n];
                maxi=n;
            }
        }

        switch(maxi){ //START MAKING CHOISE OF DESTINATION AND ROLL!!!
        case 0:
            check();//stopmotor(2);
            for(i=0;i<17;i++){ //14
                TMRO=0;

```

```

    vehicle_right_rotation(); //rotate to destination [-90 degrees]
    TMRO=0;
}
check();
for(i=0;i<17;i++){ //14
    TMRO=0;
    move_forward(); //move straight forward 14~15cm
    TMRO=0;
}
check();
break;
case 1:
    check();
    for(i=0;i<7;i++){
        TMRO=0;
        vehicle_right_rotation(); //rotate to destination [-45 degrees]
        TMRO=0;
    }
    check();
    for(i=0;i<17;i++){
        TMRO=0;
        move_forward(); //move straight forward 14~15cm
        TMRO=0;

    }
    check();

```

```

break;
case 2:
    check();//stopmotor(2);
    for(i=0;i<17;i++){
        TMRO=0;
        move_forward(); //move straight forward 14~15cm [0 degrees] GO STRAIGHT
        TMRO=0;
    }
break;
case 3:
    check();
    for(i=0;i<7;i++){
        TMRO=0;
        vehicle_left_rotation(); //rotate to destination [45 degrees]
        TMRO=0;
    }
    check();
    for(i=0;i<17;i++){
        TMRO=0;
        move_forward(); //move straight forward 14~15cm
        TMRO=0;
    }
    check();
break;
case 4:
    check();

```

```
for(i=0;i<17;i++){
    TMRO=0;
    vehicle_left_rotation(); //rotate to destination [90 degrees]
    TMRO=0;
}
check();
for(i=0;i<17;i++){
    TMRO=0;
    move_forward(); //move straight forward 14~15cm
    TMRO=0;
}
check();
break;
} //END OF SWITCH
}
}
```

3. Σενάρια αποτυχίας

Σενάρια αποτυχίας είναι τα σενάρια τα οποία το ρομποτικό όχημα αποτυγχάνει να αποφύγει κάποιο εμπόδιο δηλαδή συγκρούεται με εκείνο. Το ποσοστό επιτυχίας του εξαρτάται από τον αριθμό των αισθητήρων καθώς και την ορθή λειτουργία τους. Έχοντας λοιπόν τρεις αισθητήρες θα μελετήσουμε τα σενάρια αποτυχίας με αυτόν τον αριθμό των αισθητήρων. Τα σενάρια της αποτυχίας πηγάζουν απ'το κομμάτι της αναγνώρισης περιβάλλοντος.

Οι αισθητήρες απόστασης παίρνουν μετρήσεις σε ευθεία , το οποίο σημαίνει ότι σε ένα κάθετο επίπεδο είναι διακριτά σημεία. Έτσι υπάρχουν αρκετά νεκρά σημεία τα οποία οι αισθητήρες δεν μπορούν να μετρήσουν. Σε αυτά τα νεκρά σημεία υπάρχουν πιθανότητες σύγκρουσης με κάποιο εμπόδιο.

Σενάρια αποτυχίας επίσης πηγάζουν και από εσφαλμένες μετρήσεις των αισθητήρων. Τα σφάλματα των αισθητήρων προέρχονται από φυσικούς παράγοντες όπως χρώμα υλικού και είδος επιφάνειας υλικού (ρούχο ή κάποια γυαλιστερή επιφάνεια).

4. Extra

4.1 Πορεία εργασίας (φωτογραφικό υλικό)



Εικ. 5.1 α) Φύλλο αλουμινίου



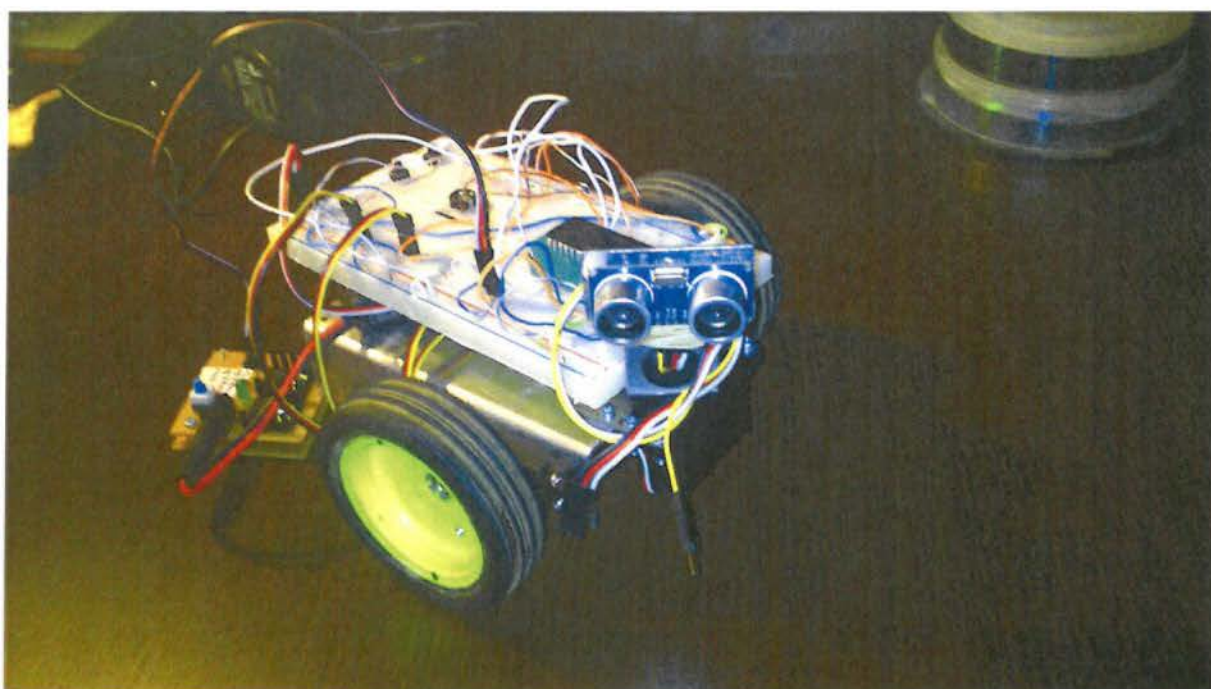
Εικ. 5.1 β,γ) Φύλλο αλουμινίου στρατζαρισμένο ως προς τα σχέδια έτοιμο για τροποποίηση



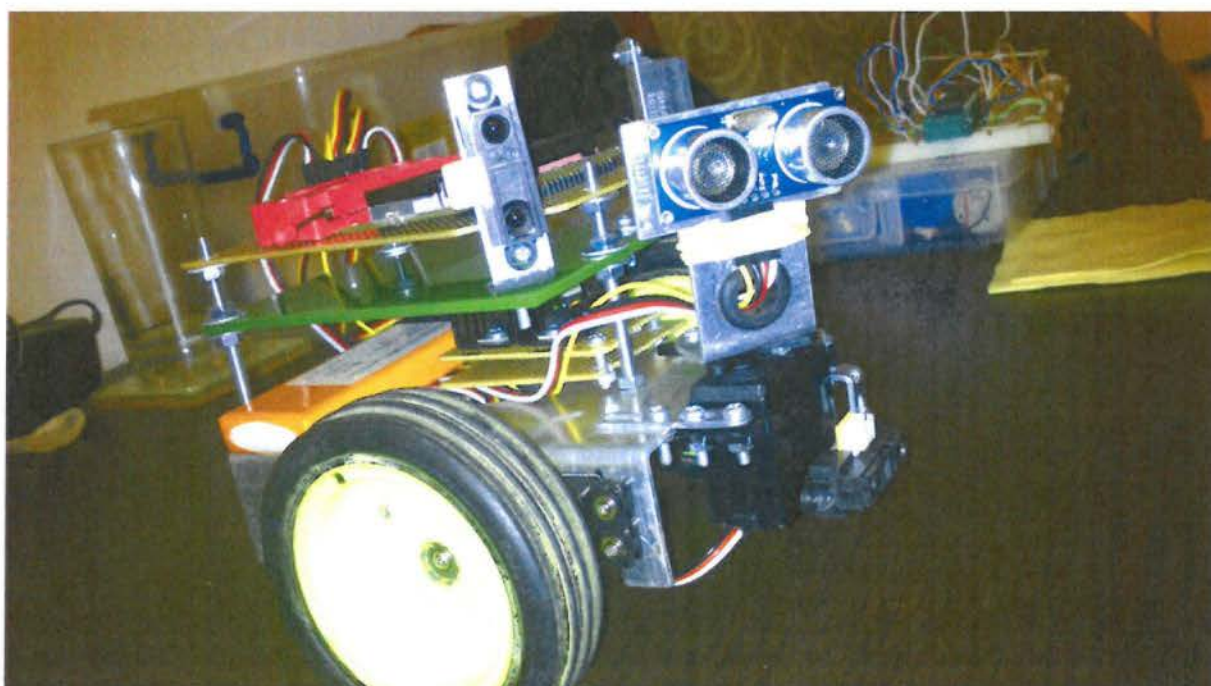
Εικ. 5.1 δ) Μετά τις τροποποιήσεις



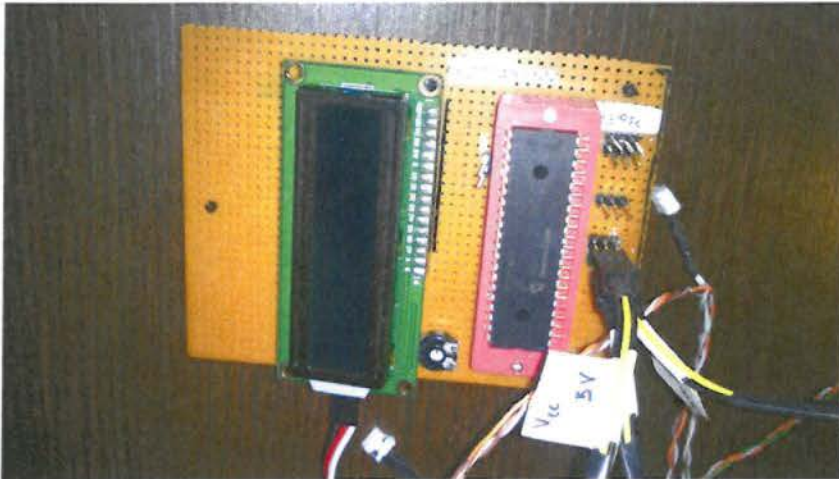
Εικ. 5.1 ε,ζ) Τοποθέτηση κινητήρων , τροχών και βοηθητικής μπίλιας



Εικ. 5.1 η) Δοκιμαστικό στάδιο κατασκευής



Εικ. 5.1 θ) Τοποθέτηση υπέρυθρων αισθητήρων και πλακέτας κύκλώματος



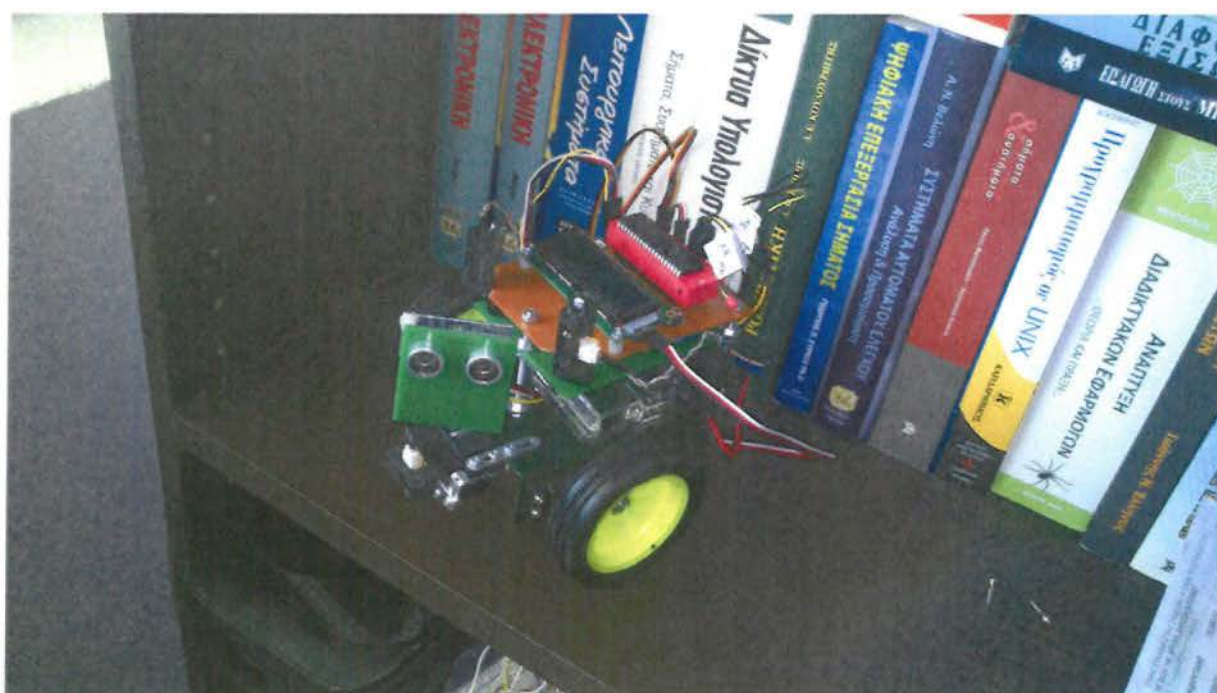
Εικ. 5.1 ι) Πλακέτα



Εικ. 5.1 κ) Προετοιμασία για βάψιμο



Εικ. 5.1 λ) Μετά το βάψιμο



Εικ. 5.1 μ) Τελικό στάδιο

5. Βιβλιογραφία

John Iovine - "PIC Robotics"

Mcgraw Hil - "PIC Robotics"

Δρ. Σταμάτης Αλατσαθιανός - "Εισαγωγή στη ΜΗΧΑΤΡΟΝΙΚΗ και στα ενσωματωμένα συστήματα"

Paulo E. Merloti - "How to Interface a Microchip PIC MCU with a hobby R/C Servo"

Warren Geek - "Motion Control Made Simple"

Χρήσιμες ιστοσελίδες :

https://docs.google.com/document/d/1Y-yZnNhMYy7rwhAgyL_pfa39RsB-x2qR4vP8saG73rE/edit

[http://www.electfreaks.com/store/download/product/Sensor/HC-SR04/HC-SR04 Ultrasonic Module User Guide.pdf](http://www.electfreaks.com/store/download/product/Sensor/HC-SR04/HC-SR04%20Ultrasonic%20Module%20User%20Guide.pdf)

<https://gist.github.com/flakas/3294829>

<http://www.parallax.com/sites/default/files/downloads/28125-Robotics-With-The-Boe-Bot-v3.0.pdf>

<http://en.wikipedia.org/wiki/Boe-Bot>

<http://www.acroname.com/articles/sharp.html>

<http://www.engineersgarage.com/articles/servo-motor>

