



Τεχνολογικό Εκπαιδευτικό ίδρυμα Κρήτης

*ΔΜΠΣ: Προηγμένα συστήματα παραγωγής,
αυτοματισμού και ρομποτικής*

Μεταπτυχιακή εργασία

*Ανάπτυξη συστήματος ελέγχου
βραχίονα 3 DOF*

Αλεξιάδης Δημήτριος

Επιβλέπων καθηγητής: Δρ. Καβουσσάνος Εμμ.

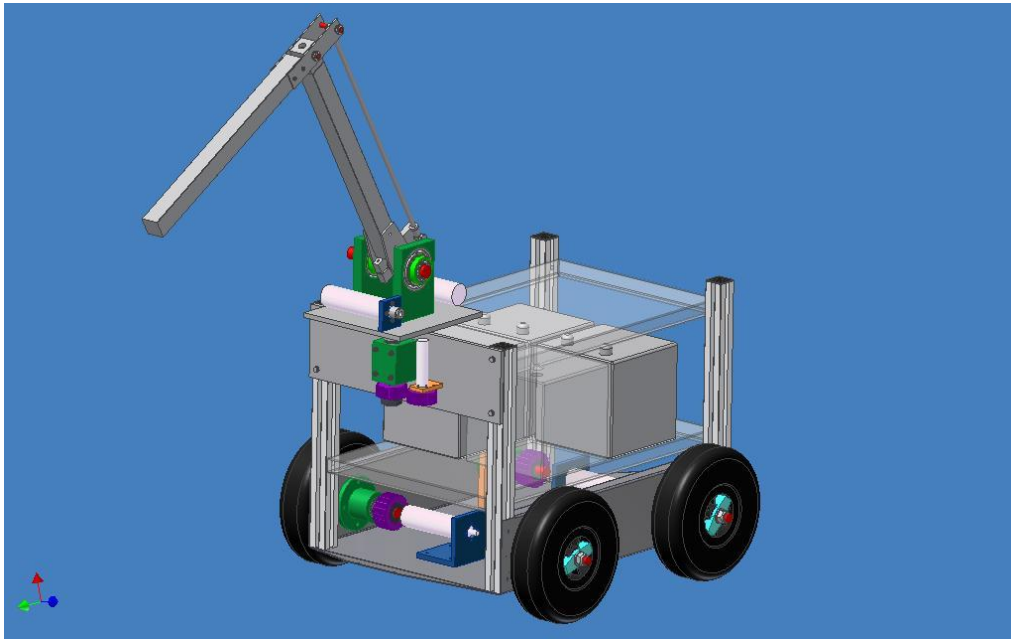
Ευχαριστίες

Η παρούσα μεταπτυχιακή εργασία εκπονήθηκε από τον Αλεξιάδη Δημήτριο απόφοιτο της σχολής μηχανολογίας και μεταπτυχιακό του τμήματος *Προηγμένα συστήματα παραγωγής, αυτοματισμού και ρομποτικής* του ΤΕΙ Κρήτης, κατά το έτος 2016.

Θα ήθελα να ευχαριστήσω θερμά τον επιβλέποντα καθηγητή μου Δρ. Εμμανουήλ Καβουσσάνο για την εμπιστοσύνη και τη βοήθεια που μου πρόσφερε τόσο σε προπτυχιακό όσο και σε μεταπτυχιακό επίπεδο καθώς και στην επαγγελματική αποκατάστασή μου. Η στήριξή του ήταν ευεργετική για την ολοκλήρωση των σπουδών μου στο ΤΕΙ Κρήτης και την έναρξη της επαγγελματικής μου σταδιοδρομίας.

Επίσης, θα ήθελα να ευχαριστήσω τη σύζυγο μου κα Τσουρακάκη Μαρία για την υπομονή της και την απaráμιλλη βοήθεια της καθ'όλη τη διάρκεια της μεταπτυχιακής μου φοίτησης.

Δεν θα μπορούσα να παραλείψω την ευγνωμοσύνη μου στους γονείς μου, που χάρη σε αυτούς μου δόθηκε η ευκαιρία να σπουδάσω στην Κρήτη και εν τέλει να σταδιοδρομήσω εδώ...



Σχήμα 1a τροχήλατο όχημα θερμοκηπίου



Σχήμα1b τροχήλατο όχημα θερμοκηπίου

Εισαγωγή

Σκοπός εργασίας

Σκοπός της παρούσας εργασίας είναι η ανάπτυξη συστήματος ελέγχου των τριών βαθμών ελευθερίας ρομποτικού βραχίονα. Ο συγκεκριμένος ρομποτικός βραχίονας είναι μέρος ενός τροχήλατου ρομποτικού οχήματος για τις ανάγκες περισυλλογής καρπών από θερμοκήπιο. Ο έλεγχος του συστήματος είναι κλειστού βρόγχου και για την πραγματοποίηση χρησιμοποιήθηκαν :ελεγκτές καθώς και το λογισμικό πρόγραμμα matlab.

Θα αναλυθούν τα μηχανολογικά μέρη του βραχίονα καθώς και τα ηλεκτρολογικά και ηλεκτρονικά, τα οποία χρησιμοποιήθηκαν για την επίτευξη της επικοινωνίας μεταξύ των μερών αφενός και για την μετακίνηση των συνδέσμων στην επιθυμητή θέση αφετέρου.

Στη συνέχεια περιγράφεται το κινηματικό και το αντίστροφο κινηματικό πρόβλημα καθώς και ο κινηματικός μετασχηματισμός που απαιτείται για την κίνηση του βραχίονα. Θα αναλυθεί το σύστημα ελέγχου της κίνησης και ο τρόπος που γράφηκε ο κώδικας για την επίτευξη του ελέγχου.

Τέλος θα γίνει περιγραφή του πειραματικού μέρους που πραγματοποιήθηκε και τα τελικά συμπεράσματα της όλης λειτουργίας του συστήματος.

Περιεχόμενα

Κεφάλαιο 1^ο Εισαγωγικά στοιχεία

1.1 Γενική ιδέα	1
1.2 Εισαγωγή.....	3
1.2.1. Γεωργική τεχνολογική εξέλιξη.....	3
1.2.2. Προηγούμενη έρευνα.....	4

κεφάλαιο 2^ο Μηχανολογικός σχεδιασμός

2.1 Δομικά εξαρτήματα.....	13
2.2 Ηλεκτρολογικά εξαρτήματα.....	16
2.3 Το κινηματικό πρόβλημα	18
2.4 Ευθύ κινηματικό πρόβλημα.....	19
2.5 Αντίστροφο κινηματικό πρόβλημα.....	21
2.6 Κινηματικός μετασχηματισμός.....	22

Κεφάλαιο 3^ο Το σύστημα ελέγχου

3.1 Γενική περιγραφή	27
3.1.1 Ο ρόλος των προγραμμάτων.....	27
3.1.2 Επικοινωνία μεταξύ των προγραμμάτων.....	28
3.1.3 Το συνολικό σύστημα ελέγχου.....	29
3.2 Συνδεσμολογία.....	30
3.3 Μοτέρ κίνησης	30
3.4 Ελεγκτές.....	34
3.5 Τροφοδοσία	37
3.6 Το περιβάλλον δοκιμών της Roboteq.....	39
3.6.1 Λογισμικό Roboteq γενικά.....	39
3.6.2 Λειτουργίες ελεγκτή.....	40
3.7 Περιγραφή του συστήματος ελέγχου.....	42

Κεφάλαιο 4^ο Ρύθμιση των παραμέτρων λειτουργίας των επί μέρους κινητήρων

4.1 Απόκριση συστήματος.....	44
4.1.1 Εύρεση PID.....	44
4.1.2 Προφίλ κίνησης	50
4.2 Λειτουργία συστήματος.....	52

Κεφάλαιο 5^ο Λογισμικό ελέγχου

5.1 Δομή λογισμικού ελέγχου.....	54
5.2 Αρχικοποίηση συστήματος ελέγχου.....	55
5.3 Κώδικας που αναπτύχθηκε στο MatLab.....	61
5.3.1 Γενικά.....	61
5.3.2 Αλγόριθμος μενού.....	61
5.3.3 Αλγόριθμος αρχικοποίησης.....	63
5.3.4 Κίνηση από σημείο σε σημείο – point to point.....	65
5.3.5 Αλγόριθμος επανεκκίνησης	67
5.3.6 Κίνηση επί ευθυγράμμου τμήματος.....	67
5.4 Αλγόριθμοι Microbasic.....	72
5.4.1 Γενικά	72
5.4.2 Αρχικοποίηση	72
5.4.3 Αλγόριθμος επανεκκίνησης.....	76
5.4.4 Κυρίως πρόγραμμα	76

Κεφάλαιο 6^ο Συμπεράσματα

Συμπεράσματα.....	80
Βιβλιογραφικές αναφορές.....	82

Κεφάλαιο

1

1. Εισαγωγικά στοιχεία

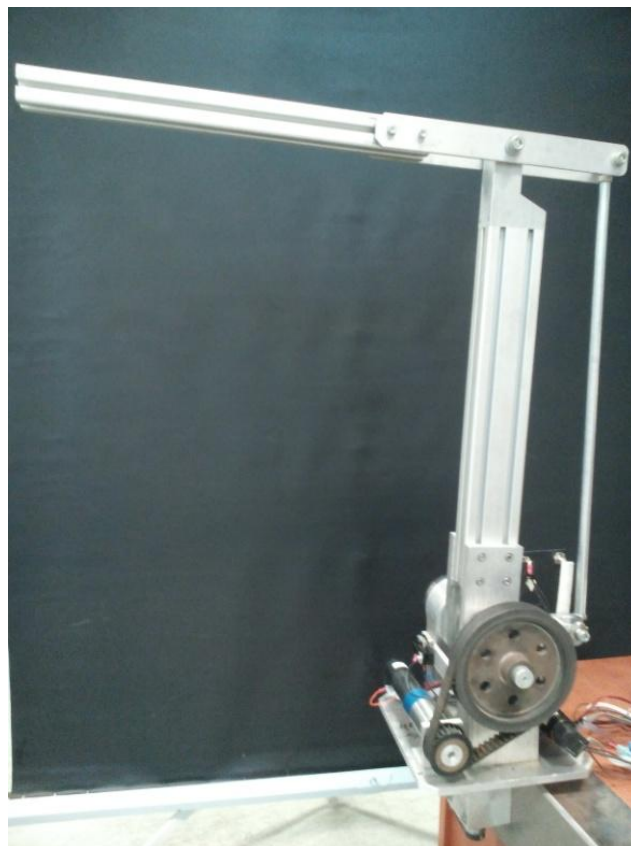
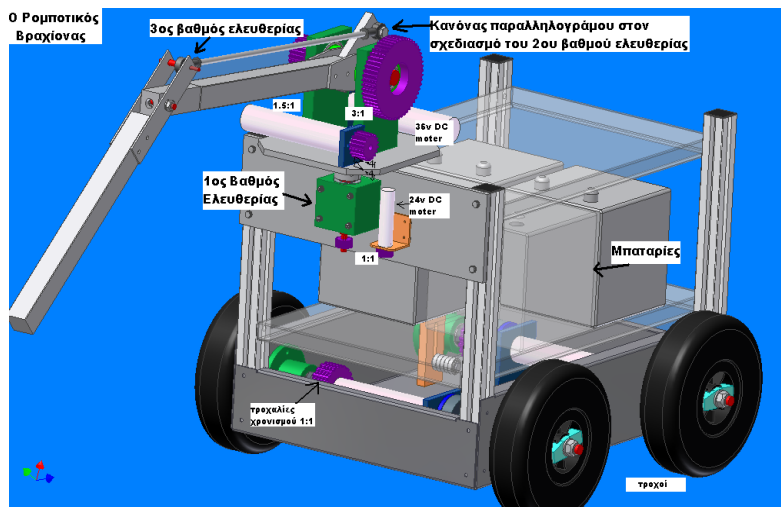
1.1 Γενική ιδέα

Το συνολικό project πέραν από τη συγκεκριμένη εργασία προβλέπει ένα ρομποτικό σύστημα ικανό να οδηγείται αυτόνομα εντός του χώρου ενός θερμοκηπίου, όπου θα εντοπίζει και θα συλλέγει τους καρπούς των φυτών. Η τεχνολογία της οδήγησης και της εύρεσης των καρπών θα γίνεται μέσω τεχνητής όρασης.

Στην παρούσα εργασία περιγράφεται η κίνηση και ο έλεγχος του βραχίονα που δημιουργήθηκε από φοιτητές του ΤΕΙ Κρήτης στα πλαίσια πτυχιακών εργασιών και με την καθοδήγηση του Δρ. Καβουσσανού Εμ.

Ο βραχίονας είναι τριών βαθμών ελευθερίας αποτελούμενος από 2 συνδέσμους και 3 περιστροφικές αρθρώσεις. Η κίνηση τους γίνεται με ηλεκτρικά μοτέρ και ο έλεγχος τους από ελεγκτές.

Γίνεται επικοινωνία των ελεγκτών με το πρόγραμμα matlab σε PC μέσω θύρας usb, το οποίο αναλαμβάνει να λύνει το αντίστροφο κινηματικό πρόβλημα και να αποστέλλει τα κατάλληλα δεδομένα της επιθυμητής θέσης στους ελεγκτές. Συνεπώς επιτυγχάνεται να βρίσκονται οι μοίρες της μετατόπισης για την κάθε άρθρωση και για την εκάστοτε επιθυμητή θέση. Το matlab με την σειρά του επικοινωνεί με τους ελεγκτές αποστέλλοντας τους παλμούς που απαιτούνται και αυτοί με την σειρά τους πραγματοποιούν την κίνηση ελέγχοντας πάντα την μετατόπιση με τα encoders που διαθέτουν τα μοτέρ. Σε όλη την διαδικασία υπάρχει επικοινωνία για το σωστό συγχρονισμό των κινήσεων.



Εικόνα 2 α Σχέδιο cad, b πραγματική κατασκευή

1.2 Εισαγωγή

1.2.1 Γεωργική τεχνολογική εξέλιξη

Σύμφωνα με τις μέχρι τώρα γνώσεις μας η γεωργία πρωτοεμφανίστηκε στη νεολιθική εποχή (10.000 με 3.000 π.Χ). Ο άνθρωπος καλλιεργούσε τη γη με απλά εργαλεία ξύλινα με ή χωρίς λίθινη άκρη. Όλες οι εργασίες γίνονταν με τη χρήση της μυϊκής του δύναμης.

Με την εμφάνιση της βιομηχανίας και την αύξηση των πληθυσμών των πόλεων άρχισε να αυξάνεται η ζήτηση γεωργικών προϊόντων. Εμφανίζεται το σύστημα της αμειψισποράς. Στα μέσα του 19ου αιώνα γίνονται οι πρώτες προσπάθειες χρήσης ατμοκίνητου κινητήρα για το όργωμα. Στις αρχές του 20ού αιώνα άρχισαν να χρησιμοποιούνται τρακτέρ με κινητήρες εσωτερικής καύσης.

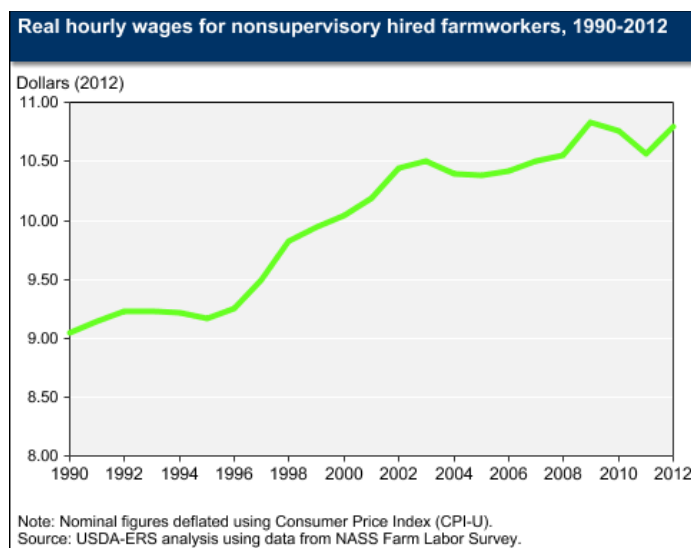
Με την ανάπτυξη της χημικής βιομηχανίας αρχίζει να διαδίδεται και η χρήση χημικών λιπασμάτων και φαρμάκων για την καταπολέμηση των γεωργικών παρασίτων, των ασθενειών των φυτών και άλλων αναγκών (όπως η "συγκράτηση" των καρπών στα δένδρα μέχρι τη συγκομιδή).

Η γεωργία θεωρείται ότι ήταν το κλειδί για την αύξηση του ανθρώπινου πληθυσμού και την ανάπτυξη του πολιτισμού, αφού η γεωργία και η κτηνοτροφία δημιούργησαν πλεονάσματα τροφίμων που επέτρεψαν και την αύξηση του πληθυσμού και την εξέλιξη του πολιτισμού.

Αυτή η ανάπτυξη όμως, έφερε συγχρόνως και περισσότερη ζήτηση στα γεωργικά προϊόντα. Παράλληλα, η ανάπτυξη του πολιτισμού και της ποιότητας ζωής έφερε την ανάγκη ποιοτικότερων καρπών και τη βελτίωση ή ακόμα και τη δημιουργία νέων ειδών γεωργικών προϊόντων.

Από οικονομικής άποψης, στη γεωργία βασίζονται οι περισσότερες χώρες καθώς αποτελεί ένα σημαντικό παράγοντα ανάπτυξης. Μέσω της παγκοσμιοποίησης κάποιες χώρες επωφελήθηκαν και κάποιες όχι. Ο ανταγωνισμός αυξήθηκε δραματικά με τις εισαγωγές και εξαγωγές των προϊόντων, με άμεση απόρροια την αύξηση της ζήτησης σε εποχικό αγροτικό δυναμικό.

Αυτή η συνεχόμενη ζήτηση ανθρώπινου δυναμικού έφερε και την αύξηση του κόστους των προϊόντων, καθώς ακόμα και τώρα σχεδόν το 100% των καρπών συλλέγονται από εργάτες με χειρωνακτική εργασία.



Σχήμα 3 Κόστος ανθρώπινου δυναμικού 1990-2012

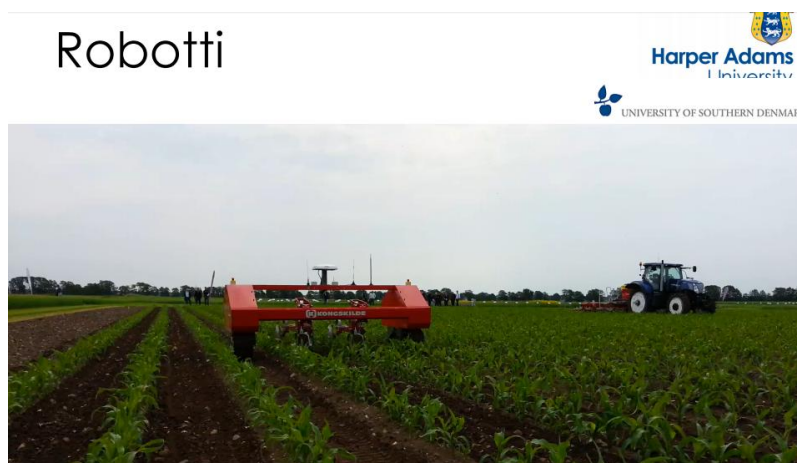
1.2.2 Προηγούμενη έρευνα

Παρόλη την τεχνολογική εξέλιξη στα γεωργικά μηχανήματα, τα γεωργικά προϊόντα, ως επί τον πλείστον, συλλέγονται ακόμα με χειρωνακτικό τρόπο καθιστώντας επιτακτική την ανάγκη να στραφεί ένα κομμάτι της έρευνας στον τομέα της ρομποτικής και την εφαρμογή της στην γεωργία.

Η υλοποίηση αυτής της ανάγκης για ανάπτυξη ρομποτικών συστημάτων εστίασε στις καθημερινές ανάγκες και εργασίες της γεωργικής καλλιέργειας, ώστε να παραχθούν αυτόνομα συστήματα, αντικαθιστώντας με αυτό τον τρόπο την «απαραίτητη» ανθρώπινη παρουσία. Ανάμεσα σε αυτές τις ανάγκες είναι και τα ρομπότ περισυλλογής καρπών σε θερμοκήπια.

Τα ρομπότ θερμοκηπίου δεν είναι καινούρια ιδέα καθώς έχει απασχολήσει κατά καιρούς διάφορα πανεπιστήμια και ερευνητές. Πανεπιστήμια όπως το Harper Adams της Αγγλίας και το Πανεπιστήμιο Hokkaido της Ιαπωνίας ξοδεύουν εκατομμύρια χρημάτων στην έρευνα και την ανάπτυξη τέτοιων συστημάτων. Ήδη έχουν εισχωρήσει στην αγορά ρομποτικά συστήματα για συγκεκριμένες εργασίες όπως : το όργωμα, την τοποθέτηση βλαστών και τη συγκομιδή.

Το Harper Adams university της Αγγλίας σε συνεργασία με άλλα πανεπιστήμια της Ευρώπης ξεκίνησε έρευνες για ρομποτικά οχήματα, τα οποία δρουν αυτόνομα.



Εικόνα 4 Αυτόνομο μηχάνημα ραντίσματος



Σχήμα 5 Όχημα περισυλλογής HarperAdamsUniversity

Ρομποτικές τεχνολογίες έχουν ενσωματωθεί στα υπάρχοντα γεωργικά μηχανήματα, εξελίσσοντας τα παραδοσιακά μηχανήματα χωρίς ωστόσο να έχουν ανεξαρτητοποιηθεί από την ανθρώπινη παρέμβαση. Τέτοιου είδους μηχανήματα υπάρχουν στην αγορά εδώ και μερικά χρόνια.



Εικόνα 6 αυτόνομο τρακτέρ με ενσωματωμένο πλοηγό

Εκτός από πανεπιστήμια, μεγάλες εταιρίες έχουν δημιουργηθεί με στελέχωση από ερευνητές του είδους και σπόνσορες με αφορμή προγράμματα έρευνας που έχουν δημιουργηθεί και στοχεύουν στην ανάπτυξη νέων τεχνολογιών. Ένα από αυτά είναι το ευρωπαϊκό πρόγραμμα «Horizon 2020» που ανάμεσα σε άλλα, επιδοτεί πανεπιστήμια και ερευνητές για έρευνα σε νέα τεχνολογικά συστήματα για γεωργικά μηχανήματα.

Μια ερευνητική ομάδα από χώρες όπως : η Ολλανδία, το Βέλγιο, η Σουηδία και το Ισραήλ συναθροίστηκε μέσα από αυτό το ευρωπαϊκό πρόγραμμα και δουλεύει πάνω σε ένα αυτόνομο ρομπότ συγκομιδής. Σκοπός του είναι η συλλογή πιπεριών και άλλων καρπών μέσα σε ελεγχόμενο περιβάλλον θερμοκηπίου. Αποτελείται από ένα βραχίονα ο οποίος δημιουργεί κενό αέρος στο άκρο του και εγκλωβίζει με αυτόν τον τρόπο τον καρπό και έπειτα τον κόβει με ειδικό εργαλείο.



Εικόνα 7 sweeper ρομπότ συγκομιδής

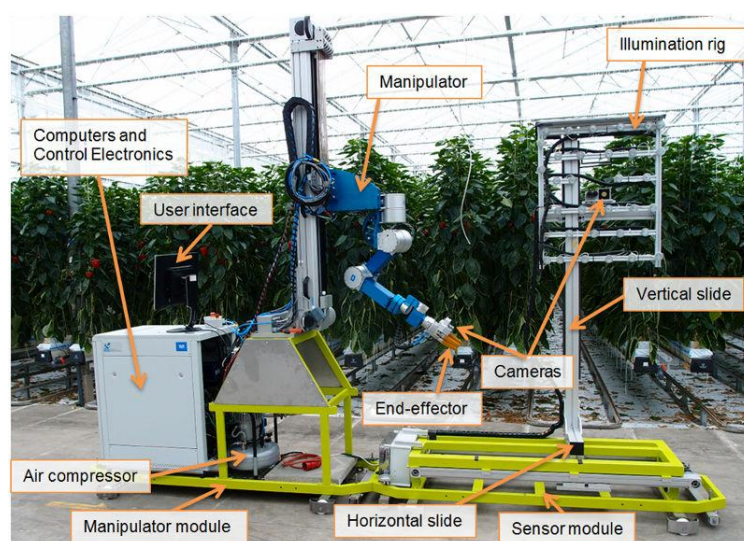
Μια άλλη ερευνητική ομάδα στο ίδιο πρόγραμμα δουλεύει πάνω σε ένα σχέδιο, το οποίο δεν επιφέρει όχημα για τη μετακίνηση του βραχίονα αλλά κινείται πάνω σε ράγες. Το Πανεπιστήμιο Wageningen στην Ολλανδία σε συνεργασία με άλλες χώρες όπως Σουηδία και Βέλγιο αλλά και ερευνητικά κέντρα κατασκεύασαν αυτό το ρομποτικό σύστημα φέρει δυο συστήματα καμερών. Το πρώτο που βρίσκεται πάνω στο όχημα χρησιμοποιείται για την ανίχνευση των καρπών, τον προσδιορισμό της ωριμότητας και την ποιότητα των φρούτων και ένα χαμηλής ανάλυσης εντοπισμό των καρπών για μια πρώτη σχεδίαση της τροχιάς του βραχίονα. Η δεύτερη κάμερα είναι τοποθετημένη στην κορυφή του άκρου και χρησιμοποιείται για υψηλής ευκρίνειας στερεοφωνική απεικόνιση της γύρω περιοχής του αγγουριού κατά τη διάρκεια της προσέγγισης. Το αποτέλεσμα των πειραμάτων σε περιβάλλον θερμοκηπίου είναι η σωστή ανίχνευση ώριμων αγγουριών κατά 95% από τα οποία το 75% συλλέχθηκαν επιτυχώς.



Εικόνα 8 project του Πανεπιστημίου Wageningen

Ο προάγγελος του «Horizon 2020» με σκοπό την ανάπτυξη αυτοματισμών και ρομποτικών συστημάτων για τη διαχείριση καλλιεργειών και δασοκομίας ήταν το «7th Frame work programme» κατά τα έτη 2007-2013. Αναδειχτήκαν αρκετά σχέδια ρομποτικών συστημάτων για περισυλλογή με πολλές αξιώσεις.

Ένα από αυτά είναι το «WP5 sweet pepper» που αφορά την περισυλλογή πιπεριών σε θερμοκήπιο για την χώρα της Ολλανδίας. Αποτελείται από μια πλατφόρμα μεταφοράς όπου φέρει ένα ρομποτικό βραχίονα με άκρο, το οποίο λειτουργεί με πνευματικό κύκλωμα και περισυλλέγει τον καρπό. Κάμερες υψηλής ανάλυσης εντοπίζουν τον καρπό και τελικά ο βραχίονας κινείται προς το μέρος του.



Εικόνα 9 sweet-pepper ρομποτικός βραχίονας θερμοκηπίου



Εικόνα 10 εικόνες από την επεξεργασία εντοπισμού καρπού WP5 sweet pepper

Ένα από τα δύσκολα ερευνητικά κομμάτια στην συγκομιδή με έξυπνα ρομποτικά συστήματα είναι η υλοποίηση του τελικού άκρου. Σε αυτό ανατίθεται η εργασία να καταφέρει να πιάσει τον καρπό και να τον κόψει από το υπόλοιπο φυτό χωρίς να προκαλέσει κάποιο χτύπημα ή φθορά στο καρπό. Στα πλαίσια της Ευρωπαϊκής κομισιόν δημιουργήθηκε το ερευνητικό σχέδιο «crop solution». Συμμετέχοντες είναι η εταιρία Festo, το τεχνολογικό πανεπιστήμιο του Μονάχου, το πανεπιστήμιο Wageningen, το ινστιτούτο έρευνας CSIC, και η εταιρεία FORCE-A. Ένα από τα αποτελέσματα αυτής της έρευνας είναι το τελικό στοιχείο του «WP5 sweet pepper». Ευλύγιστα δάκτυλα πιάνουν τον καρπό και ένα προσαρμοσμένο μαχαίρι τον ξεχωρίζει από το υπόλοιπο φυτό. Τα δάκτυλα έχουν δοκιμαστεί και σε άλλο δημιούργημα της Festo, στο «Festo Bionic Handling Assistant»



Εικόνα 11 Τελικό στοιχείο WP5 sweet pepper

Μεμονωμένοι ερευνητές όπως ο Dr. Medhat Moussa του πανεπιστημίου Guelph του Καναδά επιδοτείται από εταιρίες όπως οι: Fanuc Robotics, Canadian Microsystems Corp, the Ontario Centres of Excellence κ.α. για τη δημιουργία ενός βραχίονα, ο οποίος θα προσαρτηθεί σε ένα ρομποτικό όχημα για περισυλλογή ντοματών μέσα σε θερμοκήπιο. Υπάρχουν πλέον πολλοί αλγόριθμοι, οι οποίοι μπορούν να ξεχωρίσουν τον ώριμο καρπό από τον άγουρο. Όλοι χρησιμοποιούν την τεχνητή όραση και πολύπλοκα προγράμματα τόσο για να πλησιάσουν τον καρπό όσο και να τον ξεχωρίσουν.



Εικόνα 12 Δρ. Medhat Moussa, ρομποτικός βραχίονας περισυλλογής

Στην Γαλλία μεγάλο μέρος της αγροτικής παραγωγής είναι οι αμπελώνες. Ένας μεμονωμένος ερευνητής ο Christophe Millot δημιούργησε ένα ρομποτικό όχημα που ονομάζεται Wall-Ye. Είναι 50cm σε ύψος και ζυγίζει 20 κιλά. Διαθέτει 6 κάμερες που το καθοδηγούν ανάμεσα στα αμπέλια. Κάποιες από τις κάμερες και με την βοήθεια από GPS το καθιστά ικανό να αποθηκεύσει την ακριβή θέση και το σχήμα του κάθε αμπελώνα. Οι εργασίες που μπορεί να κάνει είναι το κλάδεμα και η συλλογή δεδομένων που βοηθούν τον παραγωγό. Παρόμοια ρομποτικά οχήματα αναπτύσσουν και στην Καλιφόρνια η εταιρία Vision Robotics και στο πανεπιστήμιο Canterbury της Αυστραλίας.



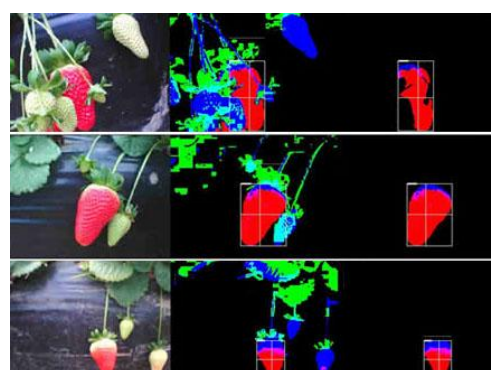
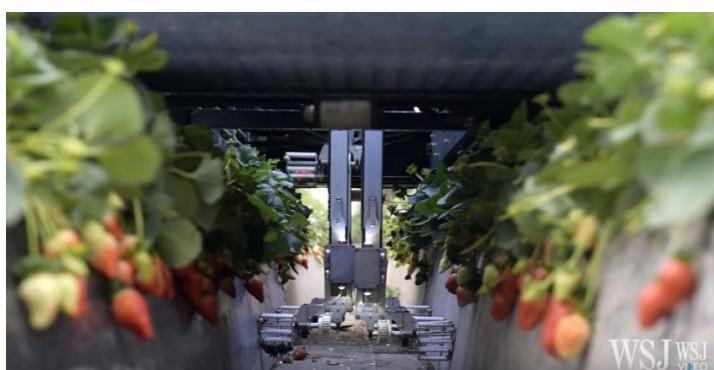
Εικόνα 13 Ρομποτικό όχημα Wall-Ye

Στην Ισπανία μια εταιρία εν' ονόματι agrobot, κατασκεύασε ένα αυτόματο ρομποτικό σύστημα για περισυλλογή σε θερμοκήπιο φραουλών. Ονομάζεται SW6010, και διαθέτει το κυρίως όχημα το οποίο κινείται πάνω από τα φυτά κατά μήκος της σοδιάς. Έχει ένα σύστημα πλοήγησης που επιτρέπει την αυτόματη καθοδήγησή του. Στο κάτω μέρος του διαθέτει καρτεσιανούς βραχίονες οι οποίοι περισυλλέγουν τις φράουλες.



Εικόνα 14 Ρομποτικό όχημα SW6010

Διαθέτει ένα σύστημα τεχνητής όρασης το οποίο αναγνωρίζει τον ώριμο καρπό από τον άγουρο με την χρήση της μορφής και του χρώματος. Εντοπίζει τον προς περισυλλογή καρπό και ένας από τους βραχίονες θα κατευθυνθεί προς το μέρος του. Όταν γίνει η περισυλλογή, ένας ιμάντας κατευθύνει τις φράουλες στο πάνω μέρος του μηχανήματος όπου εκεί θα συσκευαστούν σε μικρές ποσότητες προς πώληση. Η τελευταία εργασία απαιτεί όμως την ανθρώπινη παρέμβαση.



Εικόνα 5 Ρομποτικό όχημα SW6010 Βραχίονες περισυλλογής και λογισμικό τεχνητής όρασης

Αρκετές εταιρίες, πανεπιστήμια και ερευνητικά ιδρύματα αναπτύσσουν ρομποτικά συστήματα για χρήση σε γεωργικές εφαρμογές. Τα αγροτικά ρομπότ είναι ηλεκτροκίνητα οχήματα και συχνά διαθέτουν πνευματικά κυκλώματα για τη λειτουργία του άκρου. Για την οδήγησή τους είναι εφοδιασμένα με συστήματα πλοήγησης, ασύρματης δικτύωσης και απομακρυσμένης επίβλεψης. Τα περισσότερα είναι ακόμα σε πρώιμο στάδιο και λειτουργούν υπό συγκεκριμένες καιρικές συνθήκες και σε ελεγχόμενο περιβάλλον.

Καθώς αυτοί οι περιορισμοί είναι αρκετοί, πολλά ερευνητικά ινστιτούτα προσπαθούν να δημιουργήσουν ελεγχόμενο περιβάλλον, όπως σε ένα θερμοκήπιο και μέσα σε αυτό να εργάζονται μόνο αυτόνομα ρομπότ χωρίς την ανθρώπινη παρέμβαση.



Εικόνα 13 Αυτοματοποιημένο αγρόκτημα εταιρίας Spread

Η ιαπωνική εταιρεία Spread αναπτύσσει ένα εντελώς αυτοματοποιημένο αγρόκτημα, στο οποίο ρομποτικοί βραχίονες θα δρουν αυτόνομα κάνοντας όλες τις απαραίτητες εργασίες, από το πότισμα μέχρι τη συγκομιδή.

Μέχρι τα μέσα του 2017, η κλειστή εγκατάσταση θα αρχίσει να παράγει 30.000 μαρούλια ημερησίως και φιλοδοξεί μέσα σε πέντε χρόνια να καλλιεργεί μισό εκατομμύριο καθημερινά. Το αγρόκτημα, επιφάνειας περίπου 4.400 τ.μ., θα είναι εξοπλισμένο με ράφια από το πάτωμα μέχρι το ταβάνι στα οποία θα καλλιεργούνται τα μαρούλια.

2. Μηχανολογικός σχεδιασμός

2.1 Δομικά εξαρτήματα

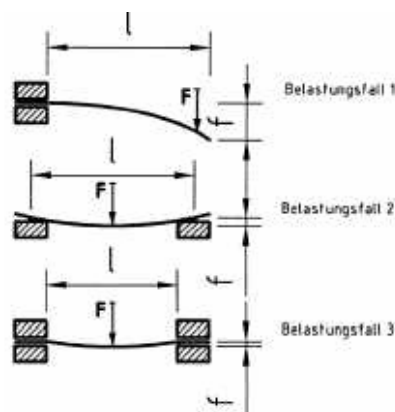
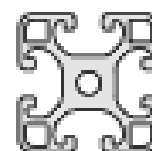
Ο βραχίονας είναι κατασκευασμένος από προφίλ αλουμινίου. Στον πρώτο σύνδεσμο, καθώς καταπονείται περισσότερο από φορτίσεις, τοποθετήθηκε προφίλ με διαστάσεις 80 * 40mm, ενώ στο δεύτερο σύνδεσμο οι διαστάσεις του προφίλ είναι 40 * 40mm. Για μήκος 680mm και 610mm αντίστοιχα και βάρος 20N ανακτήθηκαν από τον κατασκευαστή (item), τα στοιχεία για την παραμόρφωση και την τάση των προφίλ.

Για το προφίλ του συνδέσμου 2: Profile 8 40x40 light, natural

Παραμόρφωση f_1 : 0,29mm Τάση σ_1 : 2,71N/mm²

Παραμόρφωση f_2 : 0,02mm Τάση σ_2 : 0,68N/mm²

Παραμόρφωση f_3 : 0,00mm Τάση σ_3 : 0,34N/mm²

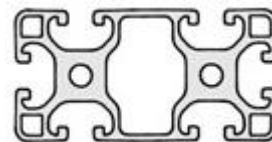


Για το προφίλ του συνδέσμου 1: Profile 8 80x40 light, natural

Παραμόρφωση f_1 : 0,06mm Τάση σ_1 : 0,78 N/mm²

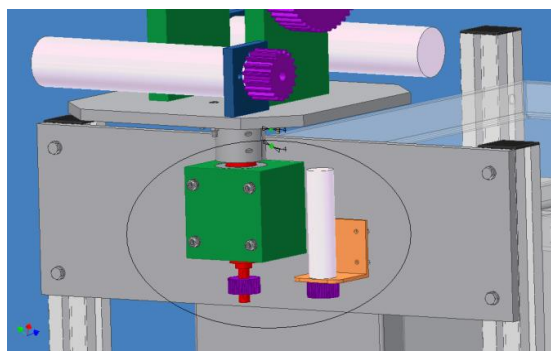
Παραμόρφωση f_2 : 0,00 mm Τάση σ_2 : 0,2N/mm²

Παραμόρφωση f_3 : 0,00 mm Τάση σ_3 : 0,1N/mm²



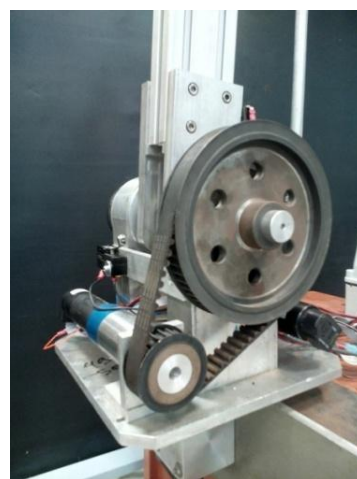
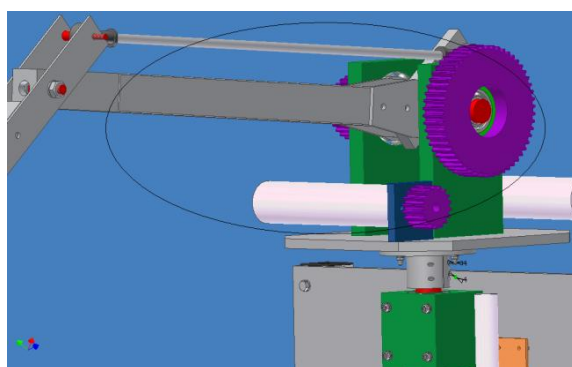
Οι τρεις αρθρώσεις που απαιτούνται για τους βαθμούς ελευθερίας βρίσκονται στη βάση του βραχίονα. Είναι και οι τρεις περιστροφικές αρθρώσεις και η κίνηση τους γίνεται με dc motor.

Η μετάδοση της κίνησης πραγματοποιείται με ιμάντες χρονισμού και τροχαλίες για τη δημιουργία κατάλληλης σχέσης μετάδοσης, ώστε το σύστημα να ανταποκρίνεται στις ροπές που χρειάζονται. Οι μέγιστες πιμές αυτές αναφορικά ανακτήθηκαν από μελέτη που πραγματοποιήθηκε στα πλαίσια πτυχιακής εργασίας και είναι : 3,5Nm για τον πρώτο βαθμό ελευθερίας, 58Nm για το δεύτερο και 29Nm για τον τρίτο βαθμό. Οι σχέσεις μετάδοσης είναι : 1:1, 3:1 και 1,5:1 αντίστοιχα.



Εικόνα 14 άρθρωση 1 Σχέδιο cad και πραγματική άρθρωση

Στη δεύτερη άρθρωση υπάρχει η ανάγκη για τη μέγιστη ικανή ροπή που μπορεί να προσφέρει ο κινητήρας μειώνοντας τις στροφές του. Υπάρχει μείωση 3:1 υπερδιπλασιάζοντας την προσφερόμενη ροπή. Οι υπολογισμοί αυτοί έγιναν στο «*design accelerator*» στα πλαίσια πτυχιακής εργασίας.

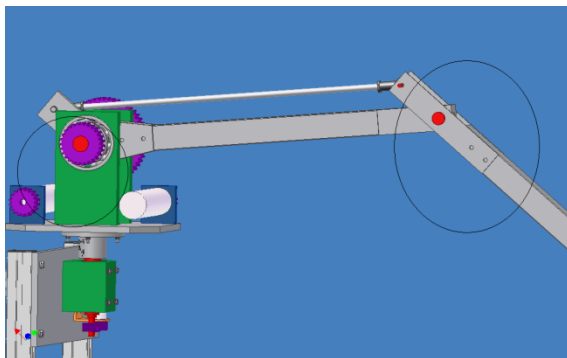


Εικόνα 15 άρθρωση 2 Σχέδιο cad και πραγματική άρθρωση

Ονομασία	Τροχαλία 1	Τροχαλία 2
Αριθμός Δοντιών (N)	22	66
Σχέση Μετάδοσης (i)	1	3
Ταχύτητα (n)	20 rpm	6,6667rpm
Ροπή (T)	20Nm	58,2Nm
Διάμετρος Τροχαλίας (dw)	56,02 mm	168,07 mm

Βασικοί παράμετροι μείωσης 2^{ης} άρθρωσης

Στον τρίτο βαθμό ελευθερίας, έχει τοποθετηθεί το μοτέρ στη βάση του βραχίονα και με προέκταση μετακινείται η άρθρωση, ακολουθώντας τον κανόνα παραλληλογράμμου. Με αυτόν τον τρόπο έχει μειωθεί το βάρος στον πρώτο σύνδεσμο, σε σχέση με την περίπτωση που θα ήταν κοντά στην άρθρωση, καθώς και αποφεύχθηκαν και οι κραδασμοί που θα υπήρχαν όταν ο κινητήρας δούλευε.



Εικόνα 16 άρθρωση 3 Σχέδιο cad και πραγματική άρθρωση

Όνομασία	Τροχαλία 1	Τροχαλία 2
Αριθμός Δοντιών (N)	22	33
Σχέση Μετάδοσης (i)		1,5
Ταχύτητα (n)	20 rpm	13,3333 rpm
Ροπή (T)	20 Nm	29,1 Nm
Διάμετρος Τροχαλίας (dw)	56,02 mm	84,03 mm

Βασικοί παράμετροι μείωσης 3^{ης} άρθρωσης

2.2 Ηλεκτρολογικά εξαρτήματα

Για να μην υπερβεί κάποια άρθρωση τον επιτρεπτό χώρο εργασίας της, δηλαδή να μετακινηθεί σε μοίρες που μπορεί να προκαλέσει σύγκρουση των συνδέσμων, έχουν τοποθετηθεί 3 τερματικοί διακόπτες. Οι διακόπτες που χρησιμοποιήθηκαν διαθέτουν 1 είσοδο και 2 εξόδους. Στην μία έξοδο υπάρχει τάση όταν ενεργοποιείτε ο διακόπτης και στην άλλη όταν δεν υπάρχει επαφή. Στην παρούσα εφαρμογή χρησιμοποιείτε σαν διακόπτης, δηλαδή επιστέφει σήμα όταν ενεργοποιηθεί.



Εικόνα 17 Τερματικός διακόπτης

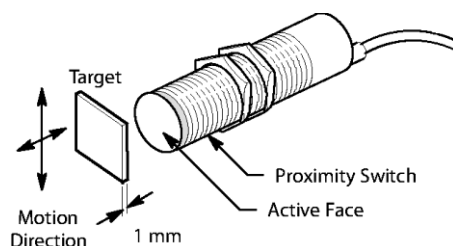
Τα encoder που χρησιμοποιούνται περιέχουν 500 εγκοπές ανά περιστροφή και εξάγουν 2 παλμοσειρές με διαφορά φάσης 90°. Χρησιμοποιώντας και τις 4 διακριτές καταστάσεις, έχουν ανάλυση γωνιακής θέσης 1/4 της μοίρας στον άξονα του μοτέρ. Αν υπάρχει εσωτερική μείωση στον κινητήρα, όπως στην περίπτωση της παρούσας χρήσης, η ανάλυση πολλαπλασιάζεται. Τα encoder είναι τύπου Incremental, αυτό σημαίνει ότι μόλις χαθεί το ρεύμα από τους ελεγκτές δεν θα έχουν κρατήσει τον αριθμό των παλμών για τις συγκεκριμένες θέσεις των συνδέσμων. Για να λειτουργήσει το σύστημα χρειάζεται ένα σημείο αναφοράς στην εκκίνηση ώστε να γίνεται γνωστή η θέση του κάθε συνδέσμου. Για την αρχικοποίηση του συστήματος χρησιμοποιήθηκαν αισθητήρες proximity και τοποθετήθηκαν σε καίρια σημεία του βραχίονα έτσι ώστε οι σύνδεσμοι να είναι κάθετοι μεταξύ τους. Πρόκειται για επαγωγικούς αισθητήρες που ανιχνεύουν την ύπαρξη μετάλλου σε απόσταση χιλιοστών. Η τροφοδοσία τους είναι 24 volt και το σήμα που εξάγουν είναι ψηφιακό (0/1) 24volt. Στον connector του ελεγκτή αυτή η τάση σαν είσοδος είναι απαγορευτική, καθώς ο κατασκευαστής ορίζει ως σήματα εισόδου-εξόδου τάση που δεν ξεπερνά τα 5 volt.

Για να λυθεί το πρόβλημα αυτό, χρησιμοποιήθηκαν σε κάθε ένα proximity ένα ρελέ, το οποίο ενεργοποιείται από το σήμα του αισθητήρα και κλείνει το κύκλωμα από το θετικό πόλο των εξόδων του ελεγκτή, που είναι 5 volt, με την είσοδό που έχει οριστεί.

Σε κάποιες περιπτώσεις ένας τερματικός διακόπτης ή ένας αισθητήρας θα πρέπει να δώσει σήμα και στους δύο ελεγκτές ή να πάρει ρεύμα από έναν ελεγκτή και να επιστρέψει σήμα στον άλλον. Για το λόγο αυτό χρησιμοποιήθηκαν και δίοδοι σε σημεία που χρειαζόταν.

Εξαιτίας των πολλών εισόδων που χρειάζεται η υλοποίηση των παραπάνω, αν προστεθούν και τα σήματα των encoders από κάθε μοτέρ, οι προσφερόμενες εισοδοί δεν επαρκούν. Για το λόγο αυτό τα σήματα των τερματικών ή των proximity σε κάποιες περιπτώσεις ενώθηκαν σε σειρά.

Με τον τρόπο αυτό, αντιμετωπίστηκε η έλλειψη εισόδων ,αλλά παράλληλα δημιουργήθηκαν περισσότερες καλωδιώσεις και χρήση των διόδων και των ρελέ.



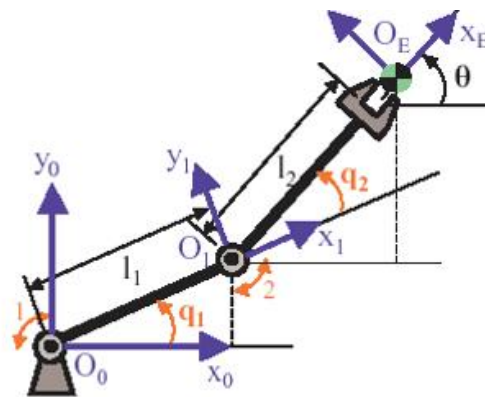
Εικόνα 18 Αισθητήρας proximity

2.3 Το κινηματικό πρόβλημα

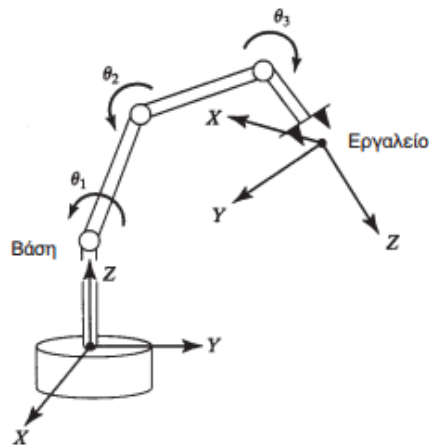
Κινηματική ορίζεται ο κλάδος της μηχανικής που μελετά την κίνηση των στερεών σωμάτων χωρίς να λαμβάνει υπόψη τις δυνάμεις που την προκαλούν. Αντίστοιχα, η κινηματική ανάλυση των ρομπότ ασχολείται με τη γεωμετρία της κίνησης ως προς ένα σύστημα συντεταγμένων. Η ρομποτική κινηματική δηλαδή αναφέρεται στη μορφή ρομπότ μέσα στο χώρο και στις σχέσεις μεταξύ των μεταβλητών μοντελοποίησης τους. Όταν οι σύνδεσμοι ενός βραχίονα είναι συνδεδεμένοι μεταξύ τους μέσω αρθρώσεων, τότε απαρτίζουν μια κινηματική αλυσίδα. Η ρομποτική κινηματική διακρίνεται στην ορθή κινηματική, που ασχολείται με το ευθύ κινηματικό πρόβλημα, την εύρεση δηλαδή των συντεταγμένων του άκρου αν είναι γνωστές οι γωνίες των αρθρώσεων και στην αντίστροφη κινηματική, που ασχολείται με το αντίστροφο κινηματικό πρόβλημα όπου με γνωστές τις συντεταγμένες του άκρου ορίζει τις γωνίες των αρθρώσεων.

2.4 Ευθύ κινηματικό πρόβλημα

Το ευθύ κινηματικό πρόβλημα συνίσταται στον προσδιορισμό της θέσης και του άκρου του βραχίονα ως προς το πλαίσιο βάσης, όταν είναι γνωστές οι μεταβλητές των αρθρώσεων.



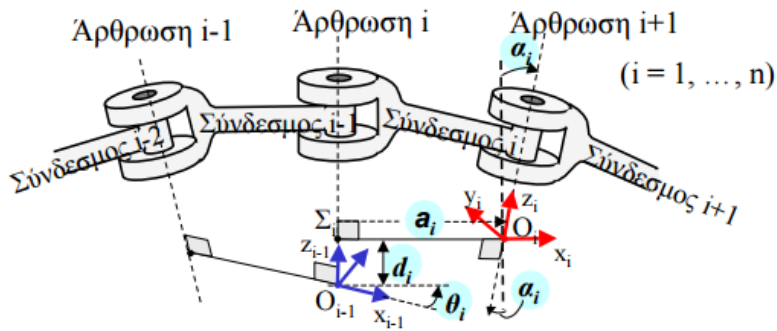
Σχήμα 19 τοποθέτηση πλαισίων



Σχήμα 20 τοποθέτηση πλαισίων βάσης και άκρου

Οι αρθρώσεις επιτρέπουν στα γειτονικά μέλη να κινούνται σχετικά το ένα ως προς το άλλο. Οι αρθρώσεις που κινούνται περιστροφικά καλούνται περιστροφικές.

Η κίνηση μιας περιστροφικής άρθρωσης μετρείται με βάση τη γωνία που σχηματίζουν τα δυο κινούμενα μέλη. Η γωνία αυτή καλείται γωνία άρθρωσης. Γενικά περιγράφουμε τη θέση του βραχίονα, περιγράφοντας τη σχετική θέση δυο συγκεκριμένων πλαισίων : του πλαισίου του άκρου και του πλαισίου του ακίνητου τμήματος του βραχίονα, που καλείται πλαίσιο βάσης.



Σχήμα 21 Κινηματική αλυσίδα

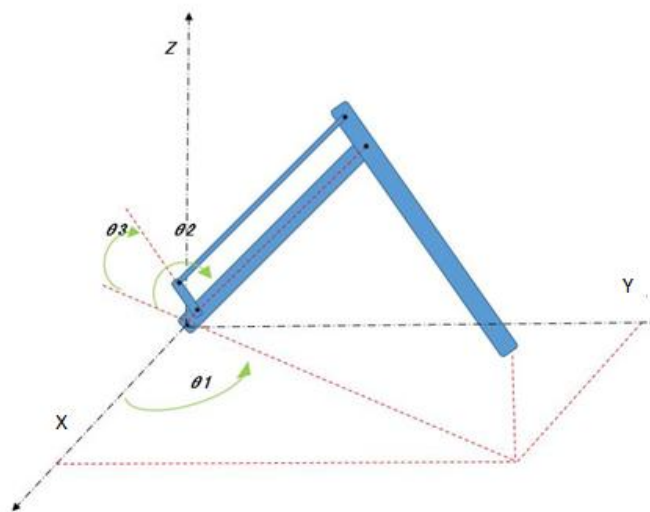
Ένας βραχίονας αποτελείται από σειριακούς συνδέσμους οι οποίοι τοποθετούνται σε κάθε άρθρωση από τη βάση μέχρι το εργαλείο τελικής δράσης. Η μεθοδολογία Denavit - Hartenberg που χρησιμοποιεί τέσσερις παραμέτρους είναι η πιο κοινή μέθοδος για την περιγραφή του κινηματικού προβλήματος ενός ρομποτικού βραχίονα. Ένα σύστημα συντεταγμένων επισυνάπτεται σε κάθε άρθρωση και καθορίζει τις D-H παραμέτρους. Βάσει της μεθόδου αυτής επιλέγονται με συγκεκριμένο τρόπο τα συστήματα συντεταγμένων που είναι προσαρμοσμένα σε κάθε σύνδεσμο του ρομποτικού βραχίονα εκτελώντας τα βήματα του αλγορίθμου. Έτσι προσδιορίζεται η θέση και ο προσανατολισμός του άκρου εργασίας ως προς το ακίνητο σύστημα συντεταγμένων.

Ένας άλλος τρόπος είναι η λύση του κινηματικού με γεωμετρικό τρόπο. Και οι δυο τρόποι καταλήγουν στις εξισώσεις κινηματικού μετασχηματισμού. Στο συγκεκριμένο βραχίονα επιλέχτηκε η γεωμετρική προσέγγιση.

2.5 Αντίστροφο κινηματικό πρόβλημα

Το αντίστροφο κινηματικό πρόβλημα λύνεται ώστε να βρεθούν οι γωνίες που πρέπει να σχηματιστούν για να παράγουν την επιθυμητή θέση του άκρου. Απαραίτητη προϋπόθεση καθίσταται η γνώση των συντεταγμένων στο χώρο και ο προσανατολισμός του άκρου. Η λύση του προβλήματος είναι πιο δύσκολη από το ευθύ κινηματικό, καθώς οι εξισώσεις που προκύπτουν είναι μη γραμμικές. Επιπλέον, οι λύσεις που ενδέχεται να προκύψουν μπορεί να είναι περισσότερες από μία ή και καμία, στην περίπτωση που είναι εκτός του χώρου εργασίας του βραχίονα. Για την αριθμητική επίλυση μη γραμμικών συστημάτων εξισώσεων υπάρχουν πολλές αριθμητικές μέθοδοι που συγκλίνουν προς τη λύση.

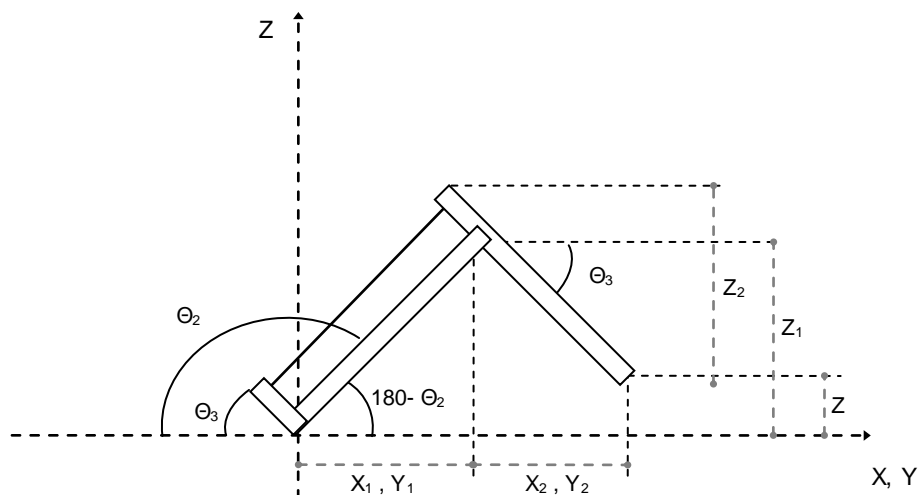
Οι άξονες έχουν τοποθετηθεί όπως φαίνεται στο σχήμα και βάση αυτού γίνεται ο κινηματικός μετασχηματισμός



Σχήμα 22 αρθρώσεις και πλαίσιο βάσης του βραχίονα

2.6 Κινηματικός μετασχηματισμός

Για το συγκεκριμένο βραχίονα που αποτελείται από τρεις περιστροφικές αρθρώσεις και δυο συνδέσμους ο κινηματικός μετασχηματισμός για το ευθύ κινηματικό πρόβλημα περιγράφεται ως εξής:



Σχήμα 23 Απεικόνιση βραχίονα σε δύο διαστάσεις

Παρατηρώντας το σχήμα του βραχίονα σε δυο διαστάσεις εξάγονται οι εξισώσεις των σημείων P_y και P_z .

$$P_y = Y_1 + Y_2$$

$$\begin{aligned} Y_1 &= L_1 \cos(180 - \theta_2) \\ &= -L_1 \cos(\theta_2) \end{aligned}$$

$$Y_2 = L_2 \cos \theta_3$$

$$P_y = -L_1 \cos(\theta_2) + L_2 \cos \theta_3$$

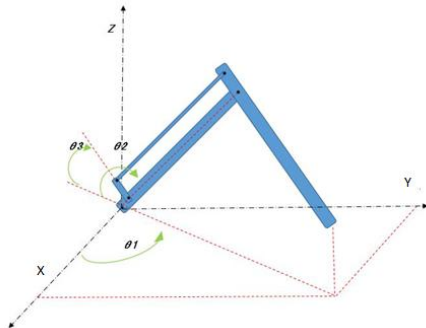
$$P_z = Z_1 - Z_2$$

$$Z_1 = L_1 \sin(180 - \theta_2) = \sin \theta_2$$

$$Z_2 = L_2 \sin \theta_3$$

$$P_z = \sin \theta_2 - L_2 \sin \theta_3$$

Εισάγοντας και την Τρίτη διάσταση καταλήγουμε στις εξισώσεις του τελικού άκρου. Όπου l_1, l_2 οι σύνδεσμοι του βραχίονα μετρημένοι σε cm και $\theta_1, \theta_2, \theta_3$ οι γωνίες των αρθρώσεων σε rad



$$P_x = (-l_1 \cos \theta_2 + l_2 \cos \theta_3) \cos \theta_1$$

$$P_y = (-l_1 \cos \theta_2 + l_2 \cos \theta_3) \sin \theta_1$$

$$P_z = l_1 \sin \theta_2 - l_2 \sin \theta_3$$

. Ακολουθεί η γεωμετρική λύση του αντίστροφου κινηματικού.

Για την γωνία θ_1 :

$$\sin \theta_1 = \frac{y}{r} \Rightarrow y = \sin \theta_1 r$$

$$\cos \theta_1 = \frac{x}{r} \Rightarrow x = \cos \theta_1 r$$

όπου r η υποτείνουσα μεταξύ x, y

εξ ορισμού $\tan \theta_1 = \frac{\sin \theta_1 r}{\cos \theta_1 r} \Rightarrow \theta = \arctan \frac{y}{x} \Rightarrow$

$$\theta = \arctan \frac{\sin \theta_1 r}{\cos \theta_1 r} \Rightarrow$$

$$\theta_1 = \text{atan2}(y, x)$$

Για τις γωνίες θ_2 και θ_3 :

Από την εξίσωση του ευθύ κινηματικού :

$$Y = (-l_1 \cos \theta_2 + l_2 \cos \theta_3) \sin \theta_1$$

$$(l_1 \cos \theta_2 - l_2 \cos \theta_3)^2 = \left(-\frac{y}{\sin \theta_1}\right)^2$$

υψώνονται και τα 2 μέλη στο τετράγωνο

$$l_1^2 \cos^2 \theta_2 - 2l_1 l_2 \cos \theta_2 \cos \theta_3 + l_2^2 \cos^2 \theta_3 = \left(-\frac{y}{\sin \theta_1}\right)^2 \quad (1)$$

$$(l_1 \sin \theta_2 - l_2 \sin \theta_3)^2 = (Z)^2$$

$$l_1^2 \sin^2 \theta_2 - 2l_1 l_2 \sin \theta_2 \sin \theta_3 + l_2^2 \sin^2 \theta_3 = (Z)^2 \quad (2)$$

$$(1) + (2) \Rightarrow -2l_1 l_2 (\cos \theta_2 \cos \theta_3 + \sin \theta_2 \sin \theta_3) = \left(-\frac{y}{\sin \theta_1}\right)^2 + (Z)^2$$

$$\Rightarrow -2l_1 l_2 \cos (\theta_2 - \theta_3) = \left(-\frac{y}{\sin \theta_1}\right)^2 + (Z)^2$$

$$-\frac{y^2}{\sin^2 \theta_1} + (Z)^2 + l_1^2 + l_2^2$$

$$\cos (\theta_2 - \theta_3) = \frac{\text{-----}}{2l_1 l_2}$$

$$2l_1 l_2$$

Θέτω $K = (\theta_2 - \theta_3)$ ή $\theta_2 = \theta_3 + K$

η εξίσωση του Z γίνεται:

$$l_1 \sin(\theta_3 + K) - l_2 \sin \theta_3 = Z$$

$$l_1 (\sin \theta_3 \cos K + \cos \theta_3 \sin K) - l_2 \sin \theta_3 = Z$$

$$l_1 \sin \theta_3 \cos K + l_1 \cos \theta_3 \sin K - l_2 \sin \theta_3 = Z \quad (3)$$

θέτω $K_1 = l_1 \cos K - l_2$ ή $l_1 \cos K = K_1 + l_2$

$$K_2 = l_1 \sin K$$

$$(3) = (K_1 + l_2) \sin \theta_3 + K_2 \cos \theta_3 - l_2 \sin \theta_3 = Z$$

$$K_1 \sin \theta_3 + l_2 \sin \theta_3 + K_2 \cos \theta_3 - l_2 \sin \theta_3 = Z$$

$$K_1 \sin \theta_3 + K_2 \cos \theta_3 = Z \quad (4)$$

Από τις ταυτότητες διπλής γωνίας καταλήγουμε στους τύπους:

$$\sin \Theta = \frac{2 \tan \left(\frac{\theta}{2} \right)}{1 + \tan^2 \left(\frac{\theta}{2} \right)} \quad \cos \Theta = \frac{1 - \tan^2 \left(\frac{\theta}{2} \right)}{1 + \tan^2 \left(\frac{\theta}{2} \right)}$$

$$(4) \Rightarrow K_1 \frac{2 \tan \left(\frac{\theta}{2} \right)}{1 + \tan^2 \left(\frac{\theta}{2} \right)} + K_2 \frac{1 - \tan^2 \left(\frac{\theta}{2} \right)}{1 + \tan^2 \left(\frac{\theta}{2} \right)} = Z$$

θέτω $\mu = \tan \frac{\theta_3}{2}$

$$K_1 \frac{2\mu}{1+\mu^2} + K_2 \frac{1-\mu}{1+\mu^2} = Z$$

Πολλαπλασιασμός και στα 2 μέρη με $(1+\mu^2)$

$$K_1 2\mu + K_2 - K_2 \mu^2 = Z + Z\mu^2$$

$$0 = -K_1 2\mu - K_2 + K_2 \mu^2 + Z + Z\mu^2$$

$$(K_2 + Z)\mu^2 + (-2K_1)\mu + (K_2 + Z) = 0$$

Που είναι της μορφής:

$$aX^2 + \beta X + \gamma$$

οπότε και εφαρμόζεται διακρίνουσα και ρίζες

$$\Delta = (-2K_1)^2 - 4(K_2 + Z)(Z + K_2)$$

$$= 4(-Z^2 + K_1^2 + K_2^2)$$

$$\mu = \frac{(-2K_1) + \sqrt{\Delta}}{2(K_2 + Z)}$$

$$\mu = \tan \frac{\theta_3}{2} \Rightarrow \theta_3 = 2 \tan^{-1} \mu \Rightarrow \theta_3 = 2 \operatorname{atan} \mu$$

$$\theta_3 = 2 \operatorname{atan} \frac{(-2K_1) + \sqrt{\Delta}}{2(K_2 + Z)}$$

$$\theta_2 = \cos(\theta_2 - \theta_3) + \theta_3$$

3. Το σύστημα ελέγχου

3.1 Γενική περιγραφή

3.1.1 Ο ρόλος των προγραμμάτων

Όπως προαναφέρθηκε, το κάθε μέρος του συστήματος (matlab, ελεγκτές) περιέχει και τα δικά του προγράμματα. Το λογισμικό matlab το οποίο είναι εγκατεστημένο σε υπολογιστή, λειτουργεί και κατευθύνει τα υπόλοιπα προγράμματα των ελεγκτών. Εφόσον συνδεθούν μέσω usb οι ελεγκτές με τον υπολογιστή, τον πρωτεύοντα ρόλο έχει το matlab. Από το βασικό μενού που έχει δημιουργηθεί, μπορεί ο χρήστης να χειριστεί όλες τις διεργασίες που προσφέρονται. Στην κίνηση του βραχίονα σε επιθυμητή θέση, το matlab στέλνει τα κατάλληλα δεδομένα στους ελεγκτές εφόσον έχει λύσει το αντίστροφο κινηματικό της νέας θέσης και μετασχηματίζοντας τις μοίρες της κάθε άρθρωσης σε παλμούς encoder.

Οι ελεγκτές από την μεριά τους τηρούν θέση αναμονής μέχρι να λάβουν δεδομένα με μορφή εντολής. Όταν αυτό πραγματοποιηθεί, ο κάθε ένας καθορίζει την ταχύτητα που πρέπει να έχει ο κάθε κινητήρας των αρθρώσεων, ανάλογα με το πόσο πρέπει να κινηθεί η κάθε άρθρωση. Αυτό γίνεται για να υπάρχει συντονισμός και η κίνηση να πραγματοποιείται παράλληλα σε όλες τις αρθρώσεις. Οι ελεγκτές επίσης, ελέγχουν την κίνηση μέσω PID ώστε ο κάθε σύνδεσμος να κινείται ομαλά στην νέα θέση. Τα υπόλοιπα υποσυστήματα: τερματικοί διακόπτες, αισθητήρες, encoder και ηλεκτρομαγνητικά φρένα των μοτέρ, ελέγχονται εξ ολοκλήρου από τους ελεγκτές.

3.1.2 Επικοινωνία μεταξύ των προγραμμάτων

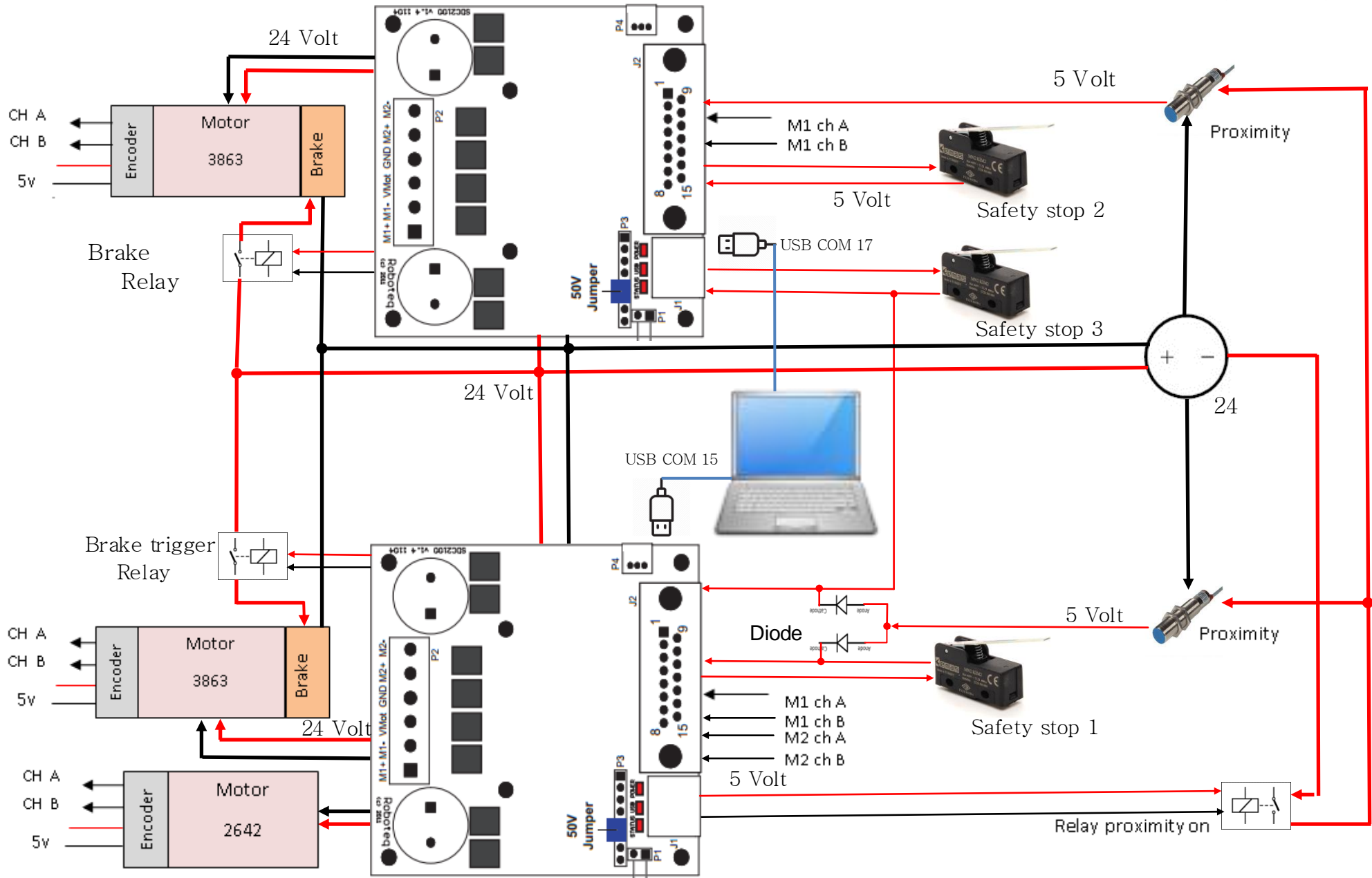
Το κάθε πρόγραμμα μπορεί να ειπωθεί πως χωρίζεται σε δύο τμήματα: την αρχικοποίηση και την κίνηση σε νέα θέση. Το κάθε ένα από τα τρία προγράμματα είναι μεν ανεξάρτητο αλλά εξαιτίας του ότι ανταλλάζουν πληροφορίες μεταξύ τους πρέπει να υπάρχει μια σχέση master-slave. Σαν master ορίστηκε το matlab και σαν slave οι ελεγκτές. Αυτή η σχέση απαιτείται επειδή η ροή του προγράμματος του κάθε ένα από τα μέρη του συστήματος έχουν διαφορετική ταχύτητα ως προς την υλοποίηση της κάθε εντολής. συνεπώς, για να είναι συγχρονισμένα και όποτε το ένα στέλνει δεδομένα και το άλλο να είναι σε θέση να τα δεχτεί έγινε αφενός η προ αναφερθείσα σχέση και αφετέρου έχουν εισαχθεί σαν εντολές καθυστερήσεις μερικών msec.

Οι δυο ελεγκτές δεν επικοινωνούν μεταξύ τους απευθείας, αλλά μόνο μέσω του matlab. Η ανάγκη να ανταλλάξουν πληροφορίες μεταξύ τους υπάρχει μόνο στην αρχικοποίηση του συστήματος και έπειτα ο κάθε ελεγκτής επικοινωνεί μόνο με το matlab.

Όταν ολοκληρωθεί ένας κύκλος προγράμματος, δηλαδή μια μετακίνηση σε νέα θέση και η επανεκκίνηση της όλης διαδικασίας, οι ελεγκτές παραμένουν σε αναμονή μέχρι να λάβουν από το matlab νέες εντολές.

Ο χρόνος του κάθε κύκλου δεν έχει πρωταρχικό ρόλο σε αυτό το στάδιο της υλοποίησης για το λόγο ότι μέσα σε αυτό τον χρόνο είναι και η μετακίνηση των μηχανικών μερών του βραχίονα. ο χρόνος δηλαδή της μετακίνησης είναι κατά πολύ μεγαλύτερος και όποια διαφοροποίηση του κώδικα για τη μικρότερη διάρκεια κύκλου θα έχει μηδαμινά αποτελέσματα.

3.1.3 Το συνολικό σύστημα ελέγχου



3.2 Συνδεσμολογία

Ένα από τα μεγάλα προβλήματα που προέκυψαν ήταν η έλλειψη πολλών εισόδων στον connector των ελεγκτών. Για την επίτευξη αυτού, ενώθηκαν τα σήματα των τερματικών με τους αισθητήρες.

Το πρόβλημα βρίσκεται στον ελεγκτή που ελέγχει τα δύο μοτέρ, τις αρθρώσεις θ1 και θ2. Οι εισοδοί δεν επαρκούσαν για να καλυφθούν οι δυο τερματικοί, οι αισθητήρες και τα encoder. Για να μπορέσουν να τοποθετηθούν όλα, συνδέθηκαν με τον εξής τρόπο: ο κάθε ένας τερματικός έχει μια είσοδο για το σήμα τους, ο αισθητήρας όμως στέλνει το σήμα του σε αυτές τις δυο εισόδους ταυτόχρονα. Έτσι, με κατάλληλες εντολές στο πρόγραμμα του ελεγκτή ξεχωρίζουν τα σήματα. Για τη σύνδεση χρειάστηκαν να τοποθετηθούν δίοδοι στο καλώδιο του αισθητήρα, για την μη έλευση ρεύματος προς τις δυο εισόδους αν ενεργοποιηθεί μόνο ο ένας τερματικός.

Οι υπόλοιπες συνδέσεις είναι πιο απλές καθώς δεν αντιμετώπιζαν κάποιο άλλο πρόβλημα, εκτός βέβαια τη χρησιμοποίηση ρελέ για τον μετασχηματισμό του ρεύματος από 24v σε 5v.

3.3 Μοτέρ κίνησης

Τα μοτέρ που έχουν εγκατασταθεί στο βραχίονα είναι της εταιρίας FAULHABER . Συγκεκριμένα δύο μοτέρ 3863 036C, τα οποία διαθέτουν ηλεκτρομαγνητικό φρένο, ενσωματωμένο μειωτήρα (gearbox) 111:1 δίνοντας 20Nm μέγιστη ροπή και ταχύτητα μετά της μείωσης 57,6rpm και ένα μοτέρ 2642 024CR για τον πρώτο βαθμό ελευθερίας, το οποίο διαθέτει ενσωματωμένη μείωση 134:1 δίνοντας 2Nm ροπή και ταχύτητα μετά της μείωσης 29,8rpm.



Εικόνα 24 μοτέρ 3863 036 Καθρώσεις Θ_1 και Θ_2

Series 3863 ... CR

Values at 22°C and nominal voltage	3863 H	012 CR	018 CR	024 CR	036 CR	048 CR	
1 Nominal voltage	U_N	12	18	24	36	48	V
2 Terminal resistance	R	0,16	0,36	0,64	1,55	2,58	Ω
3 Output power	P_{2nom}	205	211	214	201	217	W
4 Efficiency, max.	η_{max}	83	84	85	86	86	%
5 No-load speed	n_0	5 600	5 900	5 800	5 800	5 800	min^{-1}
6 No-load current, typ. (with shaft \varnothing 6 mm)	I_0	0,335	0,232	0,168	0,112	0,084	A
7 Stall torque	M_H	1 424	1 394	1 455	1 363	1 461	mNm
8 Friction torque	M_f	6,5	6,5	6,5	6,5	6,5	mNm
9 Speed constant	k_n	480	332	240	160	120	min^{-1}/V
10 Back-EMF constant	k_e	2,08	3,01	4,17	6,25	8,33	$\text{mV}/\text{min}^{-1}$
11 Torque constant	k_M	19,9	28,8	39,8	59,8	79,7	mNm/A
12 Current constant	k_i	0,05	0,035	0,025	0,017	0,013	A/mNm
13 Slope of n-M curve	$\Delta n/\Delta M$	3,9	4,1	3,9	4,1	3,9	$\text{min}^{-1}/\text{mNm}$
14 Rotor inductance	L	45	90	180	400	700	μH
15 Mechanical time constant	τ_m	4,8	4,8	4,8	4,8	4,7	ms
16 Rotor inertia	J	120	110	120	110	115	gcm^2
17 Angular acceleration	α_{max}	119	127	121	124	127	$\cdot 10^3 \text{rad}/\text{s}^2$
18 Thermal resistance	R_{th1} / R_{th2}	2,5 / 6					K/W
19 Thermal time constant	τ_{th1} / τ_{th2}	50 / 900					s
20 Operating temperature range:							
- motor		-30 ... +125					°C
- winding, max. permissible		+155					°C
21 Shaft bearings		ball bearings, preloaded					
22 Shaft load max.:							
- with shaft diameter		6					mm
- radial at 3 000 min^{-1} (3 mm from bearing)		60					N
- axial at 3 000 min^{-1}		6					N
- axial at standstill		50					N
23 Shaft play:							
- radial	\leq	0,015					mm
- axial	$=$	0					mm
24 Housing material		steel, black coated					
25 Mass		390					g
26 Direction of rotation		clockwise, viewed from the front face					
27 Speed up to	n_{max}	7 000					min^{-1}
28 Number of pole pairs		1					
29 Magnet material		NdFeB					



Εικόνα 25 μωτέρ 2642 024CR άρθρωση Θ₁

Series 2642 ... CXR

Values at 22°C and nominal voltage	2642 W	012 CXR	024 CXR	048 CXR	
1 Nominal voltage	U_N	12	24	48	V
2 Terminal resistance	R	1,46	5,84	24,06	Ω
3 Output power	$P_{2\omega m}$	22,1	23,1	22,9	W
4 Efficiency, max.	η_{max}	76	78	79	%
5 No-load speed	n_0	5 800	5 900	5 900	min^{-1}
6 No-load current, typ. (with shaft ϕ 4 mm)	I_0	0,092	0,045	0,022	A
7 Stall torque	M_M	144,6	150,5	149	mNm
8 Friction torque	M_f	1,7	1,7	1,7	mNm
9 Speed constant	k_n	514	252	125	min^{-1}/V
10 Back-EMF constant	k_e	1,945	3,962	7,994	$\text{mV}/\text{min}^{-1}$
11 Torque constant	k_M	18,57	37,83	76,34	mNm/A
12 Current constant	k_i	0,054	0,026	0,013	A/mNm
13 Slope of n-M curve	$\Delta n/\Delta M$	40,4	39	39,4	$\text{min}^{-1}/\text{mNm}$
14 Rotor inductance	L	135	560	2 280	μH
15 Mechanical time constant	τ_m	5,1	4,9	5	ms
16 Rotor inertia	J	12	12	12	gcm^2
17 Angular acceleration	α_{max}	121	125	124	$\cdot 10^3 \text{rad}/\text{s}^2$
18 Thermal resistance	R_{th1} / R_{th2}	4,7 / 15,2			K/W
19 Thermal time constant	τ_{th1} / τ_{th2}	20 / 720			s
20 Operating temperature range:					
- motor		-30 ... +100			°C
- winding, max. permissible		+125			°C
21 Shaft bearings		sintered bearings	ball bearings, preloaded		
22 Shaft load max.:		(standard)	(optional version)		
- with shaft diameter		4	4		mm
- radial at 3 000 min^{-1} (3 mm from bearing)		10	20		N
- axial at 3 000 min^{-1}		2	2		N
- axial at standstill		50	20		N
23 Shaft play:					
- radial	\leq	0,03	0,015		mm
- axial	\leq	0,2	0		mm
24 Housing material		steel, zinc galvanized and passivated			
25 Mass		114			g
26 Direction of rotation		clockwise, viewed from the front face			
27 Speed up to	n_{max}	7 000			min^{-1}
28 Number of pole pairs		1			
29 Magnet material		NdFeB			

Στο πίσω μέρος και των τριών μοτέρ υπάρχουν encoders HEDS5540 με 500 παλμούς ανά περιστροφή του μοτέρ. Μετά την ενσωματωμένη μείωση γίνονται οι ακόλουθοι υπολογισμοί για να βρεθεί η σχέση παλμών και μοιρών στον άξονα της εξόδου του μοτέρ.

Για τη γωνία θ1 με το μικρότερο μοτέρ, η τροχαλία που μεταφέρει την κίνηση με τις τροχαλίες έχουν σχέση 1:1, η εσωτερική μείωση του μοτέρ είναι 134:1 και οι παλμοί του encoder είναι 500 ανά περιστροφή του άξονα του, χρησιμοποιώντας όμως 4x μέτωπο δηλαδή θα είναι

$500 \cdot 4 \cdot 134 = 268000$ παλμοί ανά περιστροφή της τροχαλίας ή αλλιώς $268000 / 360^\circ = 744,4444$ παλμοί ανά μοίρα

Στην γωνία θ2 το encoder είναι το ίδιο αλλά η εσωτερική μείωση του μοτέρ είναι 111:1 και οι τροχαλίες έχουν σχέση 3:1 οπότε

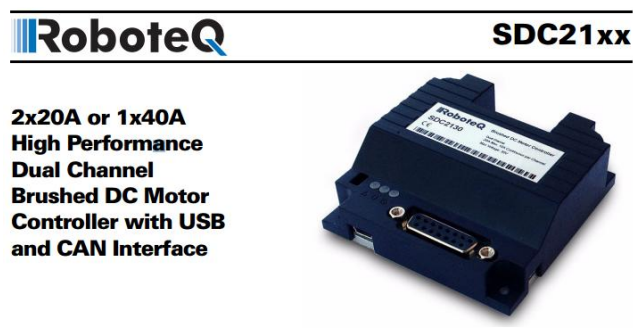
$500 \cdot 4 \cdot 111 \cdot 3 = 666000$ παλμοί ανά περιστροφή της μεγάλης τροχαλίας. Αλλιώς 1850 παλμοί ανά μοίρα

Στη γωνία θ3 το μοτέρ είναι το ίδιο και το μόνο που αλλάζει είναι η σχέση στις τροχαλίες που είναι 1,5:1.

$500 \cdot 4 \cdot 111 \cdot 1,5 = 333000$ παλμοί ανά περιστροφή και 925 παλμοί ανά μοίρα της τροχαλίας

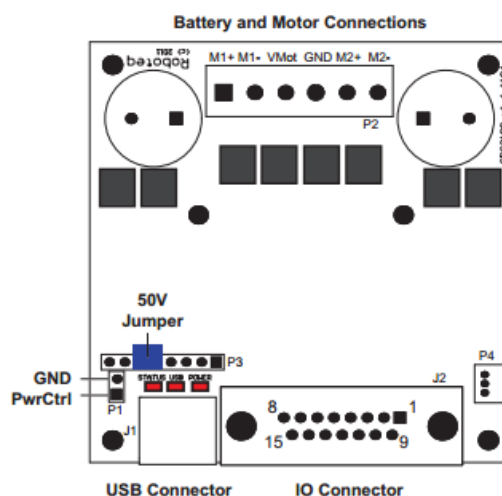
3.4 Ελεγκτές

Για την καθοδήγηση των μοτέρ απαιτούνται ελεγκτές και ένα κύκλωμα ισχύος. Αυτά βρέθηκαν ενσωματωμένα στο SDC 2130 της εταιρίας ROBOTEQ.



Εικόνα 26 Ελεγκτής sdc2130 της Roboteq

Ο ελεγκτής sdc 2130 της Roboteq έχει διαστάσεις 70 * 70 χιλιοστά και έχει σχεδιαστεί ώστε να μετατρέπει τα σήματα που δέχεται από τηλεχειρισμό, από αναλογικό joystick, από ασύρματο δίκτυο ή από υπολογιστή (θύρα RS 232 ή USB) , σε υψηλή τάση και ρεύμα για την οδήγηση ενός ή δύο κινητήρων συνεχόμενου ρεύματος.



Σχήμα 27 Ελεγκτής sdc2130

Ο ελεγκτής διαθέτει έναν 32bit μικροεπεξεργαστή και εισόδους για encoder ώστε να γίνεται έλεγχος μέσω αλγορίθμων στις κινήσεις είτε σε ανοικτό βρόγχο είτε σε κλειστό. Διαθέτει εισόδους αναλογικές, παλμικές και ψηφιακές οι οποίες μπορούν να χρησιμοποιηθούν για εντολές, ανάδραση, τερματικούς διακόπτες και άλλα.

Σε ρομποτικές εφαρμογές ο ελεγκτής μπορεί να χειριστεί δύο μοτέρ είτε ανεξάρτητα το ένα από το άλλο ή σε σύνδεση ώστε να εγκαθιδρύσει τη διεύθυνση και την περιστροφή του οχήματος με συντονισμό της κίνησης του κάθε μοτέρ. Έχουν ενσωματωθεί χαρακτηριστικά ασφαλείας ώστε να εξασφαλιστεί η αξιόπιστη και ασφαλής λειτουργία του ελεγκτή. Η λειτουργία του ελεγκτή μπορεί να αυτοματοποιηθεί και να προσαρμοστεί χρησιμοποιώντας γλώσσα προγραμματισμού.

Μειονέκτημα στον προγραμματισμό του αποτελεί η έλλειψη των μαθηματικών πράξεων μεταξύ ημιτόνων και συνημίτονων καθώς και οι λιγότες καθολικές μεταβλητές που μπορούν να διαχειριστούν μέσω επικοινωνίας με άλλες συσκευές. Κυρίως αυτή η έλλειψη στα ημίτονα και συνημίτονα οδήγησε στην χρησιμοποίηση ενός λογισμικού ικανό να λύσει το αντίστροφο κινηματικό πρόβλημα και να διοχετεύσει τα αποτελέσματα στους ελεγκτές.

Τα χαρακτηριστικά του ελεγκτή είναι τα εξής:

- USB, RS232, 0-5V αναλογικό σήμα ή παλμικό (RC radio) λειτουργία εντολών
- Αυτόματη εναλλαγή στους τρόπους επικοινωνίας που ορίζεται από τον χρήστη
- Είσοδος για άμεση σύνδεση με Spektrum
- Ενσωματωμένους οδηγούς ενέργειας υψηλής ισχύος για δύο DC κινητήρες, μέχρι και 20A έξοδο ανά κανάλι
- Έλεγχος σε δεξιόστροφη και αριστερόστροφη κίνηση των μοτέρ
- Ενσωματωμένη γλώσσα προγραμματισμού για αυτοματοποίηση και παραμετροποίηση
- Λειτουργία από μια πηγή ενέργειας
- Προγραμματιζόμενο όριο ρεύματος για κάθε κανάλι μέχρι 2x20A ή 1x40A για την προστασία του ελεγκτή, των κινητήρων, τη καλωδίωση και τη μπαταρία
- 4 αναλογικές εισόδους για χρήση με εντολές ή / και ανάδραση
- 5 Pulse Length, κύκλος συχνότητας για να ληφθούν νέες εντολές ή ανάδραση

- 6 ψηφιακές εισόδους για χρήση ως «Dead man Switch», τερματικών διακοπών, διακόπτη έκτακτης ανάγκης ή είσοδοι που ορίζει ο χρήστης
- 2 κανάλια εισόδου encoder με 32bit μετρητή
- Επιλέξιμο ελάχιστο, μέγιστο, κέντρο και νεκρή ζώνη στην είσοδο παλμών και στην αναλογική είσοδο
- Δράση σκανδάλης αν αναλογικό, παλμού ή encoderσήμα εντοπιστεί να είναι έξω από τα όρια που όρισε ο χρήστης
- Κλειστός και ανοικτός βρόγχος ελέγχου στην ταχύτητα του κινητήρα
- Κλειστός βρόγχος count position με αναλογικό ή σήμα παλμού για ανάδραση
- Ακριβής ταχύτητα και έλεγχος θέσης όταν χρησιμοποιηθεί ανατροφοδότηση με encoder
- PID έλεγχος με διαφορετικά κέρδη σε κάθε κανάλι
- Παραμετρική Καταγραφή δεδομένων των λειτουργικών παραμέτρων μέσω του RS232 για ανάλυση
- Ενσωματωμένο όργανο μέτρησης τάσης μπαταρίας και αισθητήρες θερμοκρασίας
- Ενεργοποίηση ή απενεργοποίηση του ελεγκτή από εξωτερικές πηγές
- Ρυθμιζόμενη έξοδος 5v για την τροφοδοσία encoder ή μικροϋπολογιστή
- Ξεχωριστές προγραμματιζόμενες επιταχύνσεις και επιβραδύνσεις για κάθε μοτέρ
- Ξεχωριστές προγραμματιζόμενες δεξιόστροφες ή αριστερόστροφες ταχύτητες για κάθε μοτέρ
- Απευθείας σύνδεση σε πολυκάναλο Spectrum
- 10k Ohm αντίσταση MOSFETs
- Ανίχνευση καθυστέρησης και επιλέξιμη δράση αν τα Amps είναι εκτός του εύρους που επιλέξει ο χρήστης
- Προστασία υπέρτασης και υπότασης
- Προγραμματιζόμενη Watchdog για αυτόματο κλείσιμο κινητήρα σε περίπτωση απώλειας εντολής
- Προστασία υπερθέρμανσης
- Διαγνωστικά led
- Μείωση θερμότητας με χρήση αγωγιμότητας της κάτω πλάκας. Λειτουργεί χωρίς ανεμιστήρα στις περισσότερες εφαρμογές
- Λειτουργικό περιβάλλον :-40o με + 85°C
- Δυνατότητα αναβάθμισης λογισμικού

Reference	Number of Channels	Amps/Channel	Volts	CAN	USB
SDC2130	2	20	30	No	Yes
SDC2160	2	20	60	No	Yes
SDC2130S	1	40	30	No	Yes
SDC2160S	1	40	60	No	Yes
SDC2160N	2	20	60	Yes	No
SDC2160SN	1	40	60	Yes	No

Σχήμα 28 κατάλογος σειράς sdc21xx

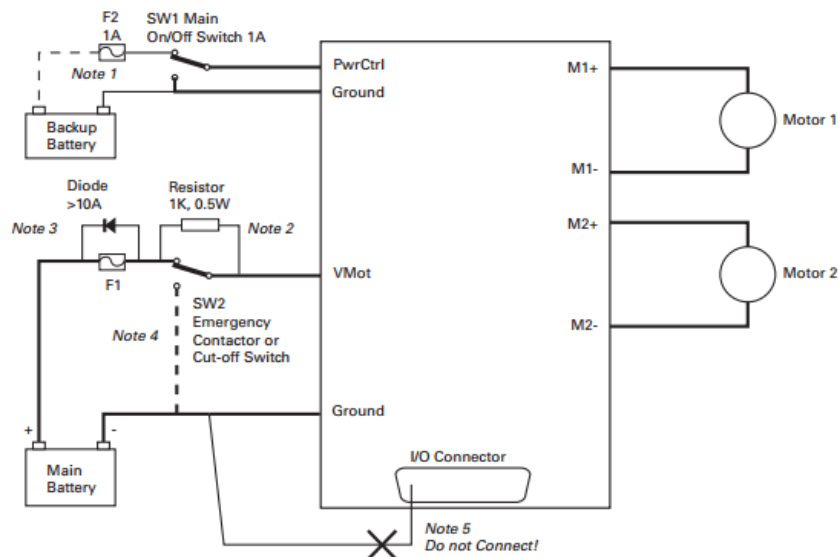
Στον connector συνδέονται τα σήματα από τερματικούς διακόπτες, proximity και οι encoders από τα μοτέρ. Για τη σωστή διαχείριση των σημάτων εισόδου-εξόδου αλλά και των μοτέρ η εταιρία προμηθεύει μαζί με τους ελεγκτές και λογισμικό για την παραμετροποίηση το οποίο περιγράφεται στο κεφάλαιο 3.6

Ο συγκεκριμένος ελεγκτής μπορεί να διαχειριστεί μέχρι δύο DC μοτέρ. Διαθέτει 15 pins σε connector, τα οποία μπορούν να χρησιμοποιηθούν για τη σύνδεση σημάτων εισόδου και εξόδου που είναι απαραίτητα για την ανατροφοδότηση του κλειστού συστήματος ελέγχου. Μπορεί να υπάρξει ζωντανή σύνδεση με υπολογιστή ή άλλον ελεγκτή μέσω usb ή και επικοινωνία μέσω matlab. Το λογισμικό Roborun δίνει τη δυνατότητα δημιουργίας προγράμματος για τον έλεγχο των σημάτων οδήγησης των μοτέρ και έλεγχο κλειστού βρόγχου.

3.5 Τροφοδοσία

Η τροφοδοσία των μοτέρ είναι 24v, όπως και των ελεγκτών. Ο connector του ελεγκτή μετασχηματίζει τις εξόδους σε 5 v και απαιτεί στις εισόδους ίδια τάση. Οι αισθητήρες proximity χρειάζονται 24 v και το σήμα που επιστρέφουν είναι και αυτό 24v. Για να μετασχηματιστεί σε 5 v ώστε να μπορεί να συνδεθεί στον ελεγκτή, τοποθετήθηκε ρελέ που οπλίζει από το σήμα του αισθητήρα και την κύρια τροφοδοσία ώστε να ενώνει την έξοδο των 5 v του connector με την είσοδο που έχει οριστεί. Όπως φαίνεται και στο σχήμα 29, λόγω της ύπαρξης μοτέρ, είναι αναγκαίο να τοποθετηθούν στο κύκλωμα της κεντρικής τροφοδοσίας μπαταρίες, ώστε να εκτονώνεται εκεί το ρεύμα της

επιστροφής από τα μοτέρ. Σε αντίθετη περίπτωση υπάρχει μεγάλος κίνδυνος να καταστραφεί ο ελεγκτής.



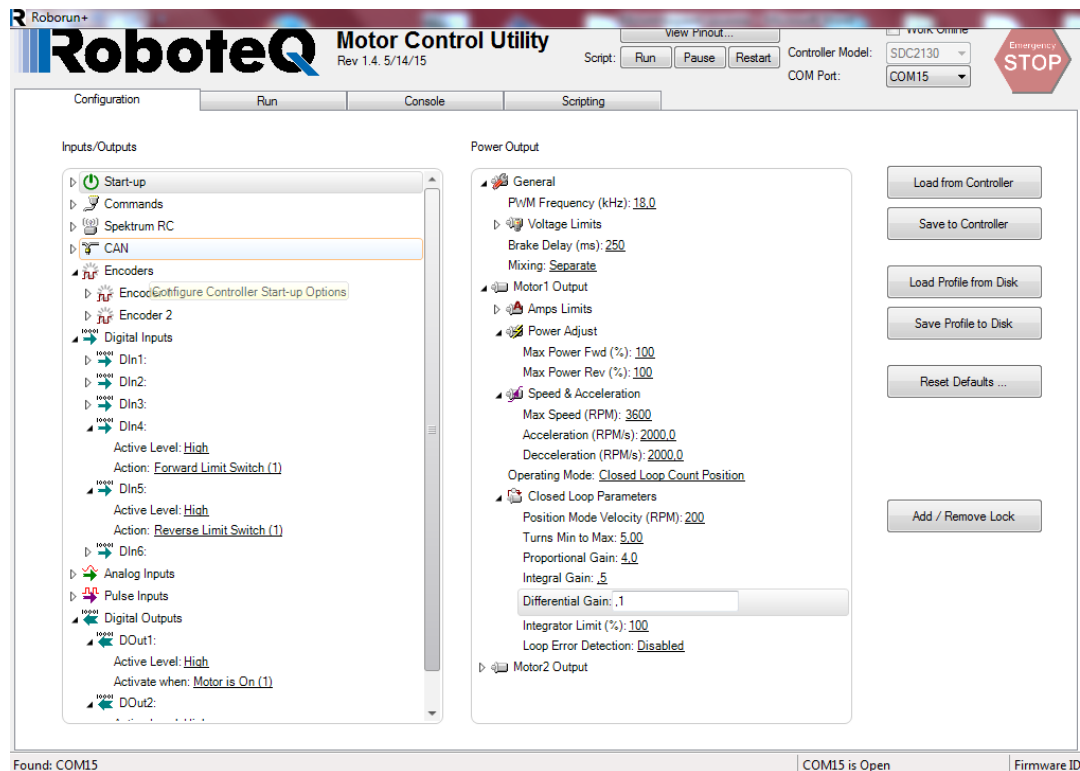
Σχήμα 29 Συνδεσμολογία τροφοδοσίας ρεύματος sdc2130

Όσον αφορά τους τερματικούς διακόπτες που έχουν χρησιμοποιηθεί, έχουν συνδεθεί με τάση από τον connector 5 v και επιστρέφει το σήμα σε αυτόν. Στην περίπτωση της δεύτερης και τρίτης άρθρωσης, οι τερματικοί διακόπτες σε κάποιες περιπτώσεις είναι συνδεδεμένοι και με τους δυο ελεγκτές καθώς χρησιμοποιούνται και για τους δυο συνδέσμους του βραχίονα.

3.6 Περιβάλλον δοκιμών της Roboteq

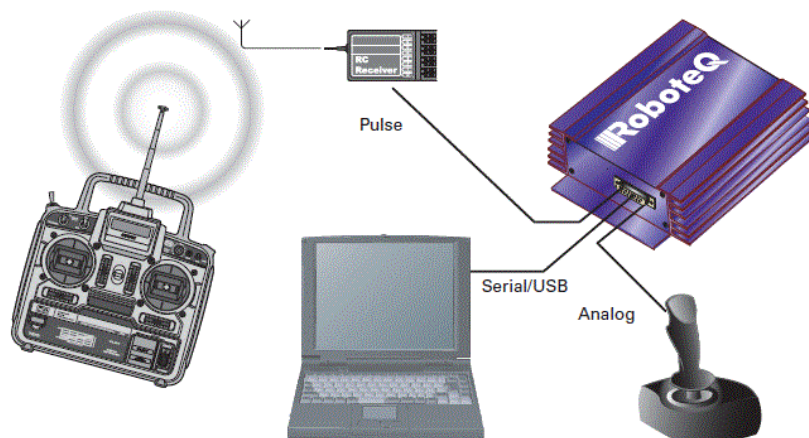
3.6.1 Λογισμικό Roboteq γενικά

Η εταιρία Roboteq προμηθεύει μαζί με τους ελεγκτές ένα λογισμικό ελέγχου που ονομάζεται roborun. Μέσω του roborun δίνεται η δυνατότητα ελέγχου των ελεγκτών, αλλά και η παραμετροποίηση τους. Οι παράμετροι λοιπόν, που απαιτούνται για τη λειτουργία των μοτέρ έχουν να κάνουν με τη μέγιστη ταχύτητα τους, τη συχνότητα που λειτουργούν, τον encoder που έχουν, την επιτάχυνση και την επιβράδυνση που είναι επιθυμητή. Τα μοτέρ ωστόσο μπορούν να ελεγχθούν είτε σε ανοικτό είτε σε κλειστό βρόγχο. Για την ανάδραση του συστήματος χρησιμοποιούνται τα encoders αλλά δίνεται και η δυνατότητα να χρησιμοποιηθούν άλλοι μέθοδοι όπως ο αισθητήρας hall. Φυσικά στον έλεγχο κλειστού βρόγχο υπάρχει και ο PID, ο οποίος ρυθμίζεται πειραματικά.



Εικόνα 30 roborun software της Roboteq

Η επικοινωνία του ελεγκτή με τον υπολογιστή μπορεί να γίνει με τρεις διαφορετικούς τρόπους. Σύνδεση με σειριακή θύρα ή usb, σύνδεση ασύρματα με τηλεχειρισμό (rc receiver) και με αναλογική σύνδεση (joystick)



Σχήμα 31 Commands modes

3.6.2. Λειτουργίες ελεγκτή

Το λογισμικό παρέχει τη δυνατότητα να επιλεγεί ο τρόπος που θα ελεγχθούν τα μοτέρ μέσα από έξι διαφορετικές λειτουργίες.

- Λειτουργία ανοιχτού βρόγχου (open loop).

Ο ελεγκτής διοχετεύει μια ποσότητα ρεύματος που ο χρήστης επιλέγει και κινείται το μοτέρ. Η ταχύτητα του δεν μετριέται, οπότε αν τοποθετηθεί ένα φορτίο στον κινητήρα τότε αυτός θα επιβραδυνθεί.

- Λειτουργία κλειστού βρόγχου με έλεγχο ταχύτητα (speed control)

Σε αυτήν τη λειτουργία απαιτείται ανάδραση από ένα ταχύμετρο ή ένα encoder για τη μέτρηση της πραγματικής ταχύτητας του κινητήρα. Αν κατά τη διάρκεια της λειτουργίας αλλάξει το μέγεθος του φορτίου, τότε ο ελεγκτής θα αυξομειώσει ανάλογα το ποσό του ρεύματος έτσι ώστε να διατηρηθεί η ταχύτητα που έχει ορίσει ο χρήστης.

- Λειτουργία σχετικής θέσης και λειτουργία εντοπισμού θέσης (relative position and tracking position)

Όπως και στη λειτουργία ελέγχου ταχύτητας υπάρχει ανάδραση, έτσι και εδώ πραγματοποιείται αλλά από αισθητήρα θέσης όπως ένα ποτενσιόμετρο, το οποίο χρησιμοποιείται για να συγκριθεί με την επιθυμητή θέση και να μετατρέψει το σήμα σε εντολή-1000/1000. Το μοτέρ θα κινηθεί με μια προκαθορισμένη ταχύτητα και επιτάχυνση προς την επιθυμητή θέση, διαγράφοντας μια ομαλή κίνηση. Παρόμοια, στη λειτουργία εντοπισμού (tracking) ισχύει η ίδια αρχή με τη διαφορά ότι στο σήμα της ανάδρασης προστίθεται και ο έλεγχος από PID για γρηγορότερη απόκριση του συστήματος.

- Λειτουργία παλμών θέσης(count position)

Πρόκειται για λειτουργία κλειστού βρόγχου, όπου η ανάδραση γίνεται μέσω encoder. Το μοτέρ θα κινηθεί για προκαθορισμένο αριθμό παλμών και με προκαθορισμένη ταχύτητα, επιτάχυνση και επιβράδυνση

- Λειτουργία ροπής (torque mode)

Είναι μια ειδική λειτουργία που σαν ανάδραση χρησιμοποιείται το ρεύμα που ρέει στο μοτέρ και ο χρήστης εισάγει την επιθυμητή ροπή που επιθυμεί χωρίς να λαμβάνεται υπόψη η ταχύτητα που θα έχει ο κινητήρας.

Στην παρούσα εργασία χρησιμοποιήθηκε η λειτουργία close loop count position καθώς η μετακίνηση σε επιθυμητή θέση βασίζεται σε αριθμό παλμών. Η ανάδραση γίνεται μέσω των encoders των μοτέρ και χρησιμοποιείται PID έλεγχος για την εξομάλυνση της κίνησης και τον έλεγχο της ταχύτητας. Ο PID ρυθμίστηκε σύμφωνα με την μέθοδο Ziegler–Nichols που περιγράφεται στο κεφάλαιο 4.1

3.7 Περιγραφή συστήματος ελέγχου

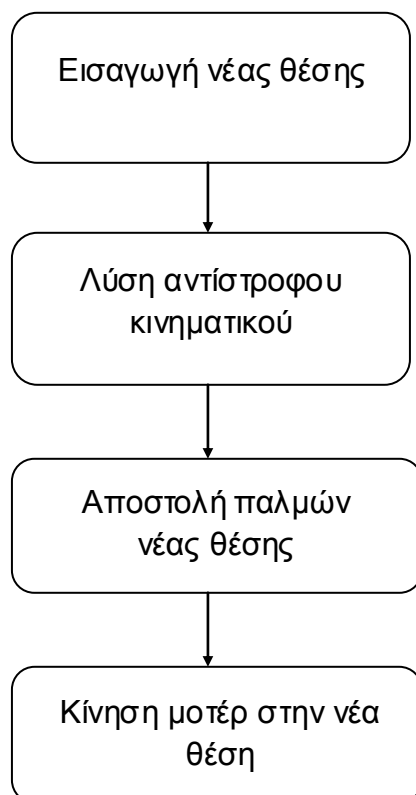
Για να πραγματοποιηθεί η κίνηση σε ένα σημείο στο χώρο, πρέπει να είναι γνωστή η παρούσα θέση του βραχίονα και η επιθυμητή θέση με μορφή συντεταγμένων στο χώρο. Τις συντεταγμένες της επιθυμητής θέσης τις εισάγει ο χρήστης. Παράλληλα ο προσδιορισμός των απαιτούμενων γωνιών στροφής των αρθρώσεων λύνεται από το αντίστροφο κινηματικό πρόβλημα. Έπειτα από τη λύση του αντίστροφου κινηματικού προβλήματος, παράγονται οι αριθμοί των παλμών του encoder που απαιτούνται για να φτάσει η άρθρωση στην επιθυμητή γωνία.

Εφόσον δημιουργηθούν οι παλμοί της νέας θέσης στο χώρο γίνεται η επικοινωνία ανάμεσα στο matlab και τους ελεγκτές μέσω της θύρας usb. Ο εκάστοτε ελεγκτής εκείνη την στιγμή βρίσκεται σε φάση αναμονής για να μπορέσει να δεχτεί δεδομένα. Όταν ολοκληρώνεται η λήψη στέλνει πίσω ένα δεδομένο επιβεβαίωσης της λήψης. Η μεταφορά των δεδομένων γίνεται με δεκαεξαδικούς αριθμούς αφού αυτό έχει οριστεί σαν κοινό σημείο επικοινωνίας από τον κατασκευαστή των ελεγκτών (roboteq). Δεδομένα στην επικοινωνία δεν χάνονται καθώς χρησιμοποιείται buffer τόσο από το matlab όσο και από τους ελεγκτές.

Ο κάθε ελεγκτής έχοντας λάβει από το matlab τον αριθμό των παλμών του encoder που απαιτούνται για να μεταφερθεί το άκρο του βραχίονα στην νέα θέση, ενεργοποιεί το εκάστοτε μοτέρ από τη θέση που είναι στην επιθυμητή. Ο έλεγχος της θέσης γίνεται με τα encoders που διαθέτουν τα μοτέρ στέλνοντας τα σαν ανατροφοδότηση στο ενσωματωμένο PID που έχουν οιελεγκτές. Η παραμετροποίηση των P,I και D έγιναν με πειραματικό τρόπο.

Ολοκληρώνοντας την κίνηση στη νέα θέση ο κάθε ελεγκτής αναλαμβάνει να στείλει στο matlab δεδομένο επιβεβαίωσης, ώστε να ολοκληρωθεί ο κύκλος και να ξαναμπει όλο το σύστημα στην αρχική κατάσταση αναμονής.

Ένα σημαντικό στοιχείο για την πραγματοποίηση αυτής της κίνησης είναι να γνωρίζει το σύστημα ποια είναι η αρχική του κατάσταση και πού βρίσκεται αρχικά ο βραχίονας, αφού όταν κλείσει η τροφοδοσία του ρεύματος τότε χάνονται και οι τιμές των παλμών από τα encoders. Για το λόγο αυτό στην εκκίνηση του συστήματος πρέπει να γίνει μια αρχικοποίηση ώστε να υπάρχει η αρχική σχετική θέση του κάθε συνδέσμου.



Σχήμα 32 Διάγραμμα κίνησης σε νέα θέση

Κεφάλαιο

4

4. Ρύθμιση των παραμέτρων λειτουργίας των επί μέρους κινητήρων

4.1 Απόκριση συστήματος

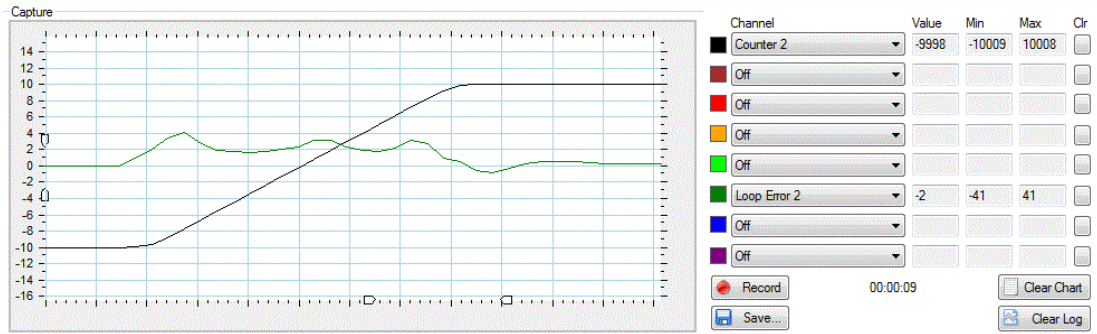
4.1.1 Εύρεση κερδών PID

Όπως προαναφέρθηκε η ρύθμιση του PID έγινε με την δεύτερη μέθοδο Ziegler – Nichols (Z-N). Σύμφωνα με αυτή τη μέθοδο τα κέρδη I και D μένουν 0 και αυξάνεται μόνο η αναλογική βαθμίδα P μέχρι την οριακή κρίσιμη τιμή K_u (ultimate K_u) όπου η έξοδος του ελέγχου είναι μια συνεχής ταλάντωση με σταθερό πλάτος (oscillation period T_u). Στη συνέχεια υπολογίζεται η σταθερά ολοκληρωτικού ελέγχου K_i και το κέρδος του διαφορικού ελεγκτή K_d χρησιμοποιώντας τις σχέσεις βαθμονόμησης των Ziegler - Nichols (Z-N).

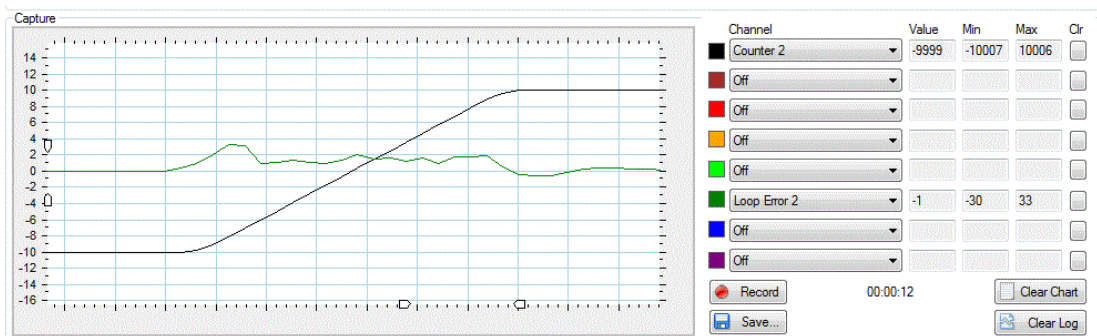
Από την απόκριση σε κλειστό βρόγχο χρησιμοποιώντας μόνο το κέρδος P, προέκυψαν τα διαγράμματα που ακολουθούν. Να σημειωθεί ότι η ρύθμιση των PID για τις 3 αρθρώσεις έγινε στον συναρμολογημένο βραχίονα έχοντας το φορτίο που αναλογεί σε κάθε κινητήρα.

Για την πρώτη άρθρωση οι σύνδεσμοι ήταν κάθετα μεταξύ τους, όπως είναι και κατά την αρχικοποίηση του βραχίονα. Για την δεύτερη άρθρωση η θέση του συνδέσμου του άκρου ξεκινάει κάθετα σε σχέση με τον πρώτο σύνδεσμο. Καθώς κινείται η άρθρωση, ο σύνδεσμος του άκρου παραμένει σε επίπεδο παράλληλο προς το έδαφος λόγω της μεταφοράς της τρίτης άρθρωσης στην βάση του βραχίονα. Τέλος στην τρίτη άρθρωση ο πρώτος σύνδεσμος παραμένει σε κάθετη θέση σε σχέση με το έδαφος.

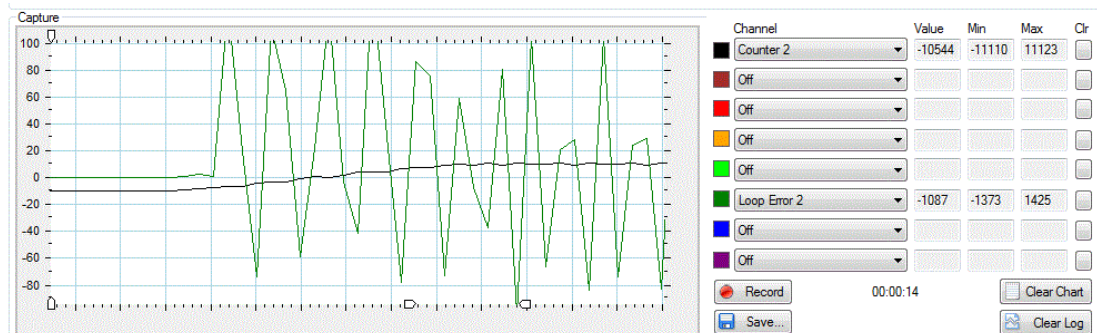
Για την άρθρωση Θ_1 :



Εικόνα 33 Εύρεση κερδών PID για Θ_1 : $P=4, I=0, D=0$

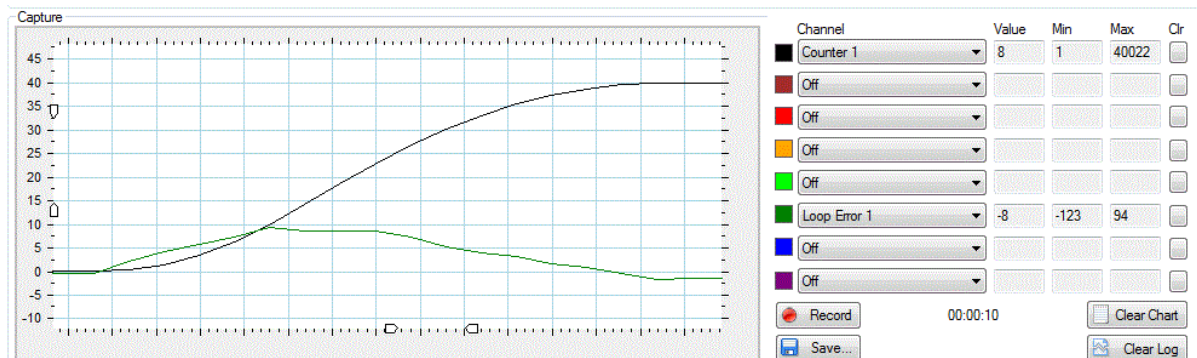


Εικόνα 34 Εύρεση κερδών PID για Θ_1 : $P=6, I=0, D=0$

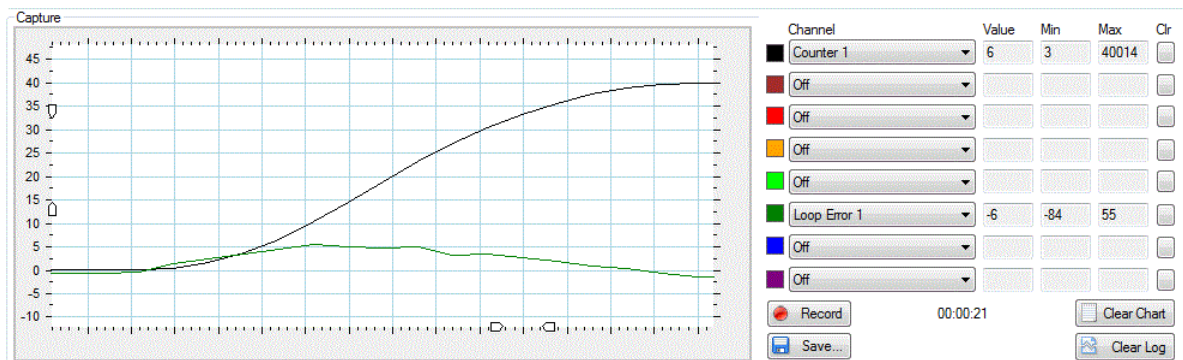


Εικόνα 35 Εύρεση κερδών PID για Θ_1 : $P7, I=0, D=0$ critical

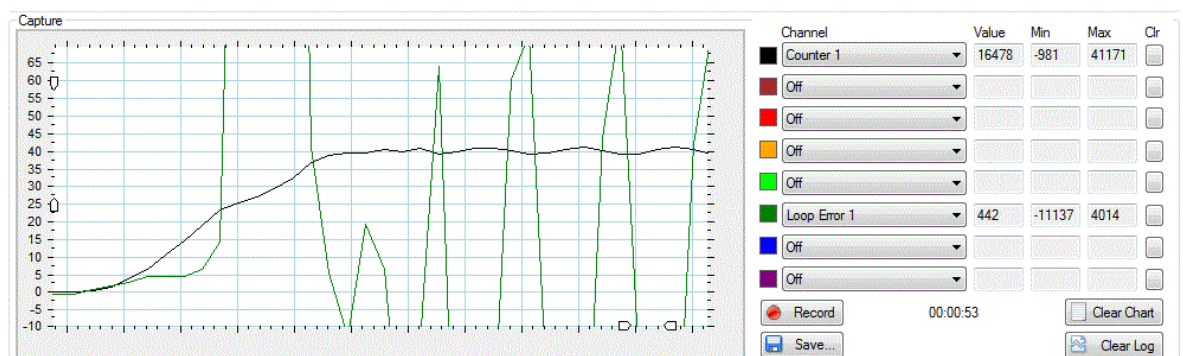
Για την άρθρωση Θ_2 :



Εικόνα 39 Εύρεση κερδών PID για Θ_2 : P=4,I=0, D=0

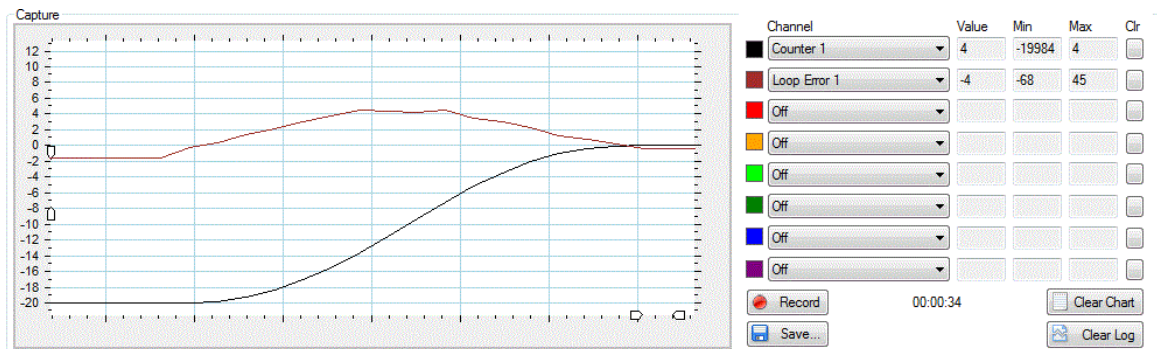


Εικόνα 40 Εύρεση κερδών PID για Θ_2 : P=6,I=0, D=0

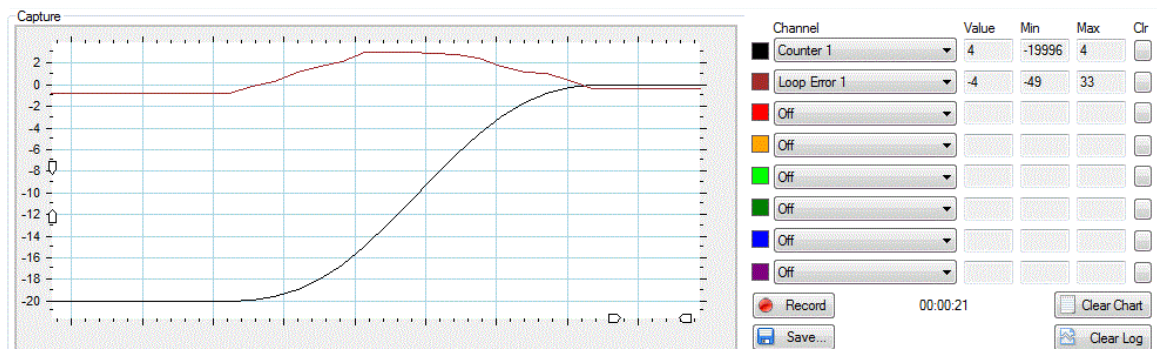


Εικόνα 41 Εύρεση κερδών PID για Θ_2 : P8,I=0, D=0 critical

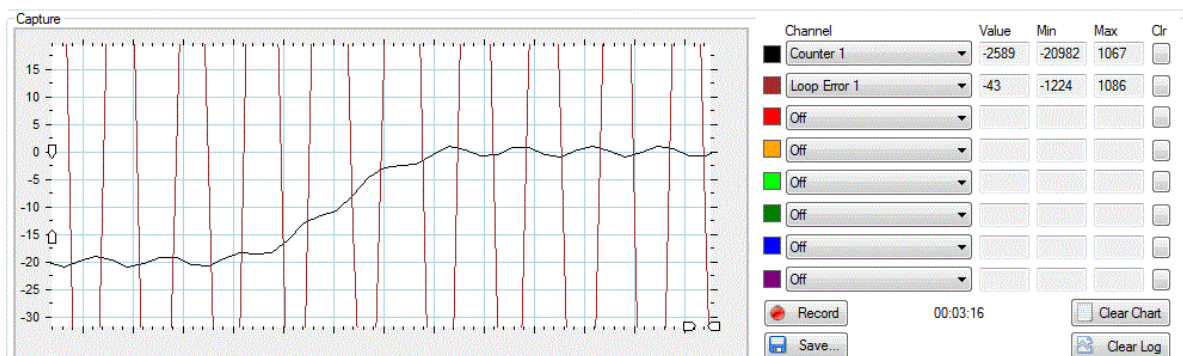
Αντίστοιχα για την άρθρωση της Θ_3



Εικόνα 42 Εύρεση κερδών PID για Θ_3 : P=4,I=0, D=0



Εικόνα 43 Εύρεση κερδών PID για Θ_3 : P=6,I=0, D=0



Εικόνα 44 Εύρεση κερδών PID για Θ_3 : P=8,I=0, D=critical

Μετά την εφαρμογή της μεθόδου και βρίσκοντας την οριακή τιμή της αναλογικής βαθμίδας ίση με 8 για Θ_2 , Θ_3 και ίση με 7 για Θ_1 . Η συχνότητα T_u ίση με 0,8sec για την Θ_2 , 0,7 για Θ_3 και 0,2sec για την Θ_1 . Χρησιμοποιείται ο πίνακας Z-N ώστε να υπολογιστούν τα κέρδη I και D. Και ο PID διαμορφώνεται ως εξής:

P: $0.6K_u = 4,8$

I: $P_u/2 = 0,4$ για την Θ_2

D: $P_u/8 = 0,1$

P: $0.6K_u = 4,8$

I: $P_u/2 = 0,35$ για την Θ_3

D: $P_u/8 = 0,087$

P: $0.6K_u = 4,2$

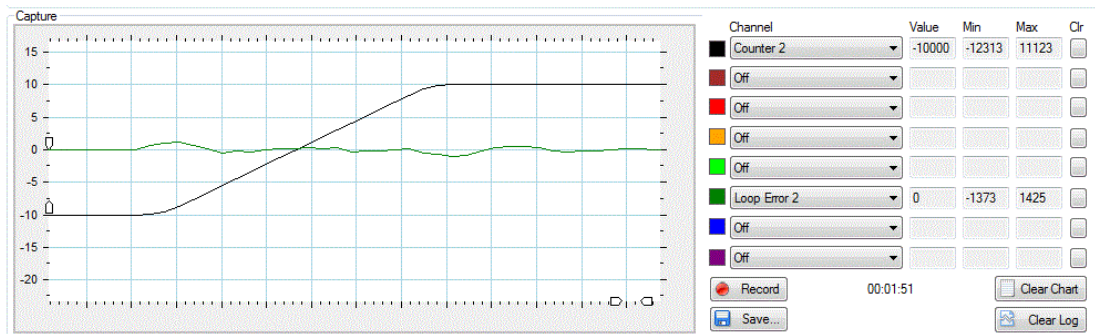
I: $P_u/2 = 0,1$ για την Θ_1

D: $P_u/8 = 0,025$

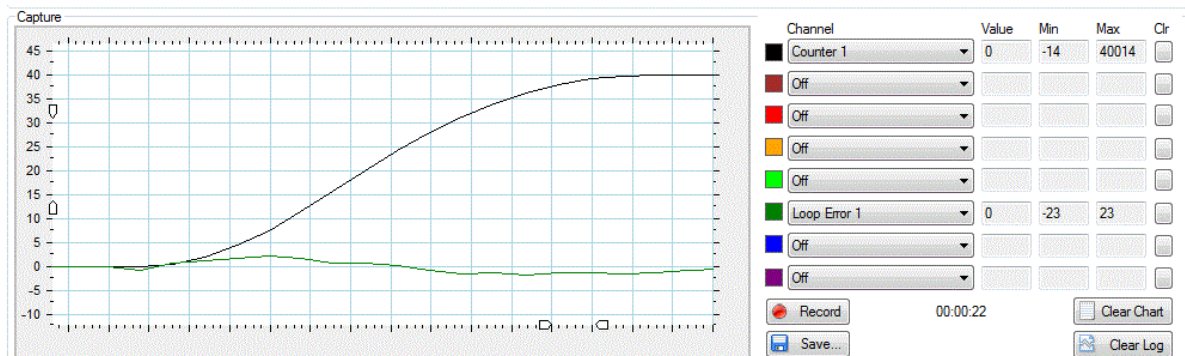
Ziegler–Nichols method ^[1]			
Control Type	K_p	T_i	T_d
<i>P</i>	$0.5K_u$	-	-
<i>PI</i>	$0.45K_u$	$T_u/1.2$	-
<i>PD</i>	$0.8K_u$	-	$T_u/8$
<i>classic PID</i> ^[2]	$0.60K_u$	$T_u/2$	$T_u/8$

Σχήμα 45 Πίνακας Z-N

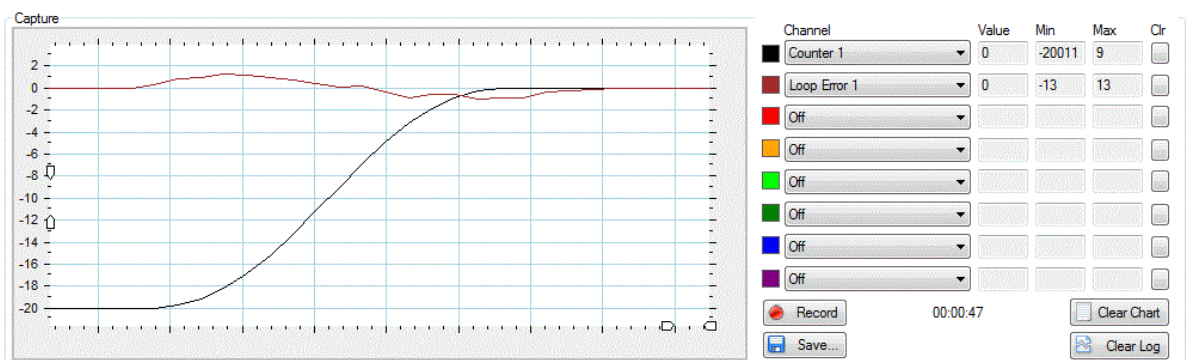
Κεφάλαιο 4- Ρύθμιση των παραμέτρων λειτουργίας των επί μέρους κινητήρων



Εικόνα 46 Εύρεση κερδών PID για Θ_1 . PID έλεγχος



Εικόνα 47 Εύρεση κερδών PID για Θ_2 . PID έλεγχος

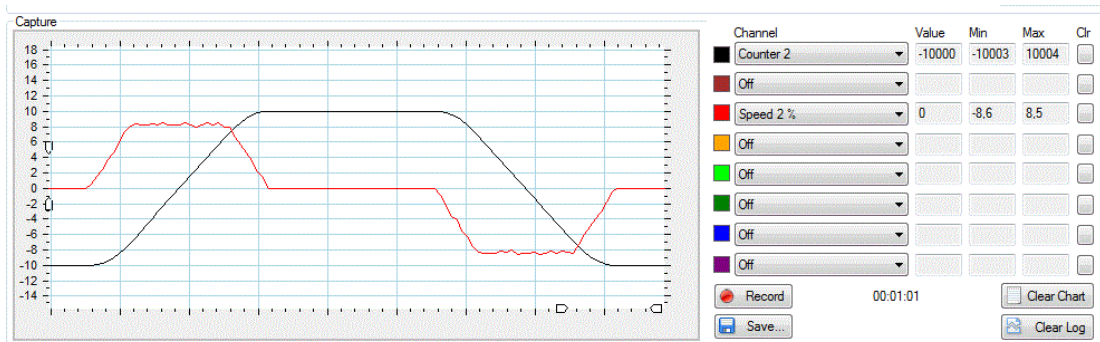


Εικόνα 48 Εύρεση κερδών PID για Θ_3 . PID έλεγχος

4.1.2 Προφίλ κίνησης

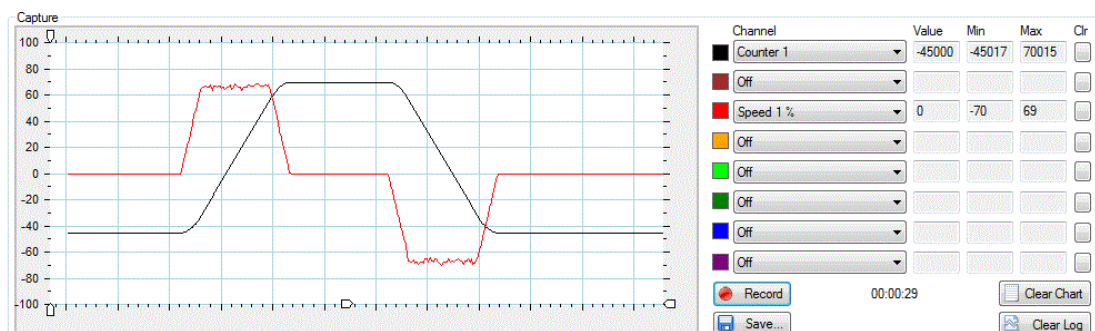
Στη διάρκεια της κίνησης από ένα σημείο σε ένα νέο, έγιναν τα παρακάτω διαγράμματα όπου φαίνονται στον άξονα Υ'Υ η θέση σε παλμούς και το ποσοστό % της μέγιστης ταχύτητας και στον άξονα Χ'Χ ο χρόνος.

Για την Θ_1 : ταχύτητα 400rpm, επιτάχυνση και επιβράδυνση 500 rpm/sec



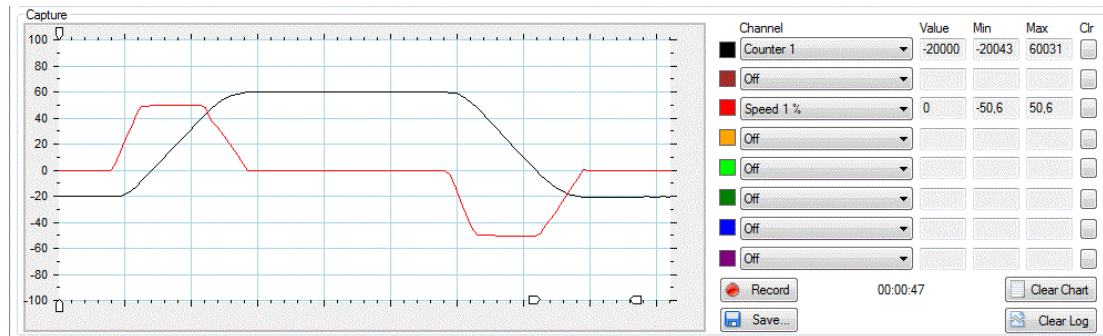
Εικόνα 49 Θ_1 ταχύτητα 400 rpm

Για την Θ_2 : ταχύτητα 200, 400 και 450rpm/sec. Επιτάχυνση 200, 600 και 1000rpm/sec και επιβράδυνση 200,600, 600rpm/sec αντίστοιχα.

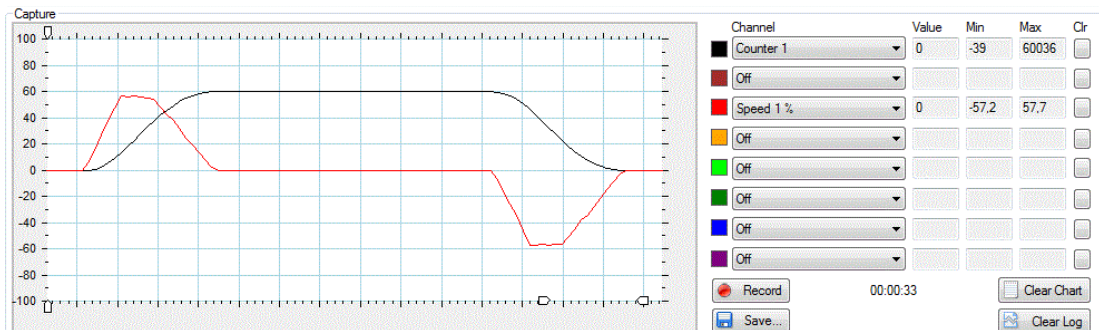


Εικόνα 50 Θ_2 ταχύτητα 200 rpm

Κεφάλαιο 4- Ρύθμιση των παραμέτρων λειτουργίας των επί μέρους κινητήρων

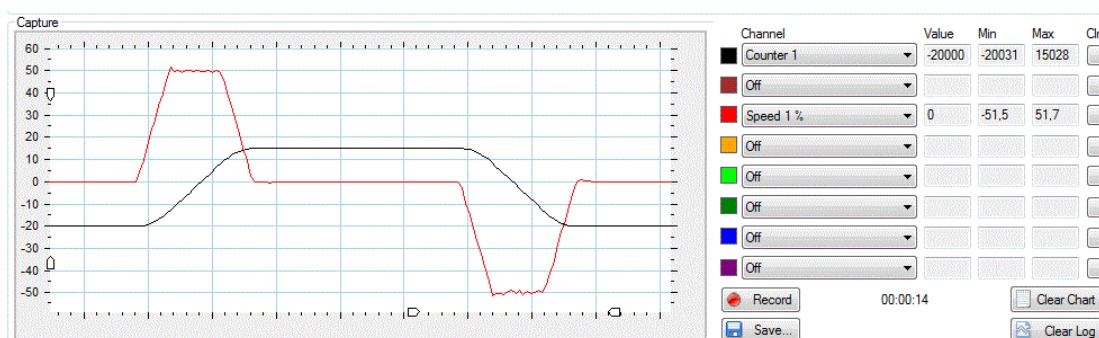


Εικόνα 51 Θ_2 ταχύτητα 400 rpm

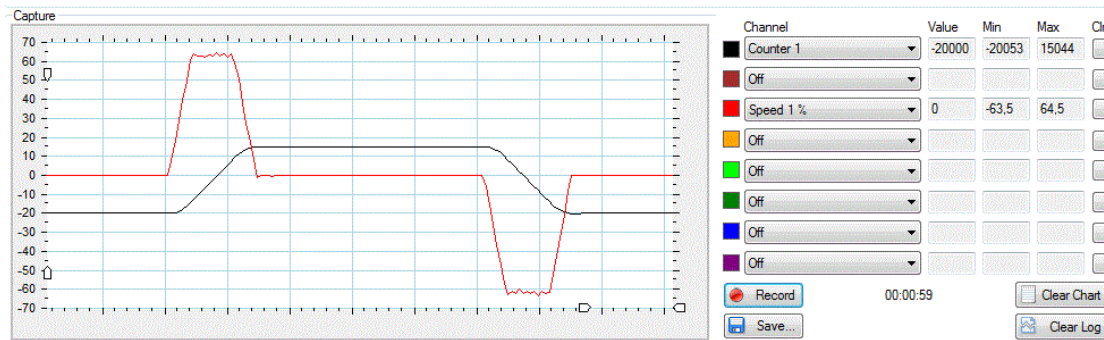


Εικόνα 52 Θ_2 ταχύτητα 450 rpm

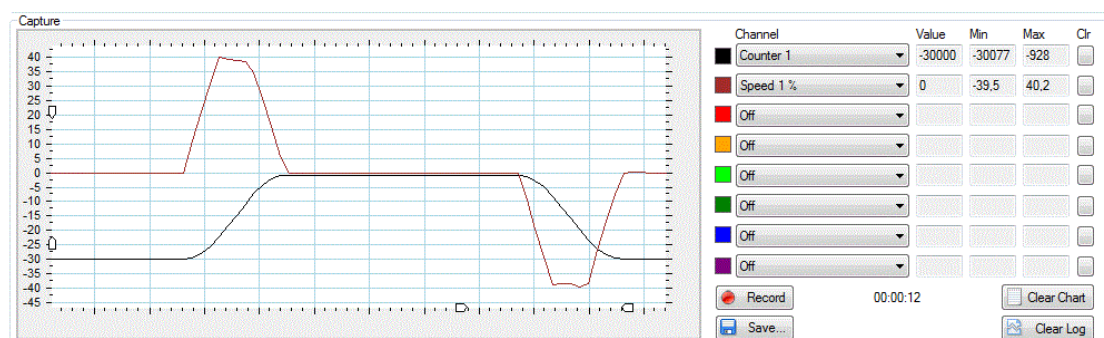
Αντίστοιχα για την γωνία Θ_3 : ταχύτητα 200,250 και 350 rpm με επιτάχυνση 400, 700, 1100 rpm/sec και επιβράδυνση 400,700,700rpm/sec



Εικόνα 53 Θ_3 : ταχύτητα 200 rpm



Εικόνα 54 Θ₃ : ταχύτητα 250rpm



Εικόνα 55 Θ₃ : ταχύτητα 350 rpm

4.2 Λειτουργία συστήματος

Όταν δοθεί τάση στους ελεγκτές, θα χρειαστεί να γίνει η αρχικοποίηση που προαναφέρθηκε μέσω του αντίστοιχου προγράμματος στο matlab. Εφόσον εκτελεστεί το πρόγραμμα, γίνεται επιστροφή στο αρχικό μενού του matlab και ο χρήστης μπορεί να επιλέξει την κίνηση του βραχίονα. Όπως φαίνεται και στην εικόνα 56 εισάγονται οι συντεταγμένες των x,y,z της νέας επιθυμητής θέσης.

Μέσα στο πρόγραμμα του matlab, εφόσον λυθεί το αντίστροφο κινηματικό πρόβλημα, μετασχηματίζονται οι μοίρες των γωνιών σε παλμούς. Όταν ολοκληρωθεί η κίνηση στη νέα θέση γίνεται επιστροφή στο κεντρικό μενού.

```
Command Window
eisagogi x,y,z
x?0
x =
    0
y?60
y =
    60
z?60
z =
    60
```

Εικόνα 56 Εισαγωγή νέας επιθυμητής θέσης

```
b3 =
190646.0898
b3 =
67000
b3 =
-12073.045
```

Εικόνα 57 Παλμοί encoder για την επιθυμητή θέση

Κεφάλαιο

5

5. Λογισμικό ελέγχου

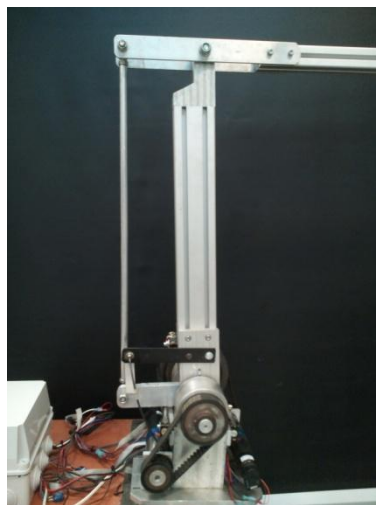
5.1 Δομή λογισμικού ελέγχου

Σε αυτό το κεφάλαιο περιγράφεται ο τρόπος που έχουν γραφτεί τα προγράμματα των ελεγκτών και του Matlab. Όταν μπου σε λειτουργία οι ελεγκτές ξεκινούν αυτόματα την ροή του προγράμματος που έχουν αποθηκευμένο στη μνήμη τους. Στην αρχή και των δύο προγραμμάτων υπάρχει το κομμάτι της αρχικοποίησης. Αυτό σημαίνει ότι δεν υπάρχει η δυνατότητα να γίνει κίνηση σε επιθυμητή θέση εάν πρώτα δεν ολοκληρωθεί αυτή η διεργασία. Εφόσον πραγματοποιηθεί, μπαίνουν σε ετοιμότητα να λάβουν τις συντεταγμένες για την νέα θέση. Τους χρόνους που δεν έχουν πάρει εντολή για εκκίνηση από το Matlab, είτε για την αρχικοποίηση είτε για την κίνηση των κινητήρων, μένουν σε κατάσταση αναμονής.

Στον Matlab από την άλλη, δεν υπάρχει μόνο μία ροή του προγράμματος. Έχουν δημιουργηθεί πέντε διαφορετικά προγράμματα που το καθένα από αυτά πραγματοποιεί μια διεργασία. Υπάρχει το κυρίως μενού όπου από εκεί ο χρήστης επιλέγει το πρόγραμμα που θα τρέξει. Το κάθε ένα από αυτά στην αρχή του έχει την παραμετροποίηση της επικοινωνίας και την δημιουργία της σύνδεσης με τους ελεγκτές. Όταν αυτό πραγματοποιηθεί γίνεται η ανταλλαγή πληροφοριών ανάλογα με το πρόγραμμα που τρέχει. Για την αρχικοποίηση που απαιτείται να γίνει πρώτα στους ελεγκτές, υπάρχει έλεγχος στην επιλογή του χρήστη ώστε να μην επιτραπεί να γίνει κίνηση εάν πρώτα δεν γίνει η αρχικοποίηση. Όταν ένα πρόγραμμα ολοκληρωθεί κλείνει τον δίαυλο επικοινωνίας ώστε να μείνει ελεύθερος για τα υπόλοιπα προγράμματα. Οι εντολές που κάνει αποστολή το Matlab είναι είτε αλλαγή σε κάποια μεταβλητή έτσι ώστε να βγουν από κατάσταση αναμονής οι ελεγκτές, είτε εντολή κίνησης των μοτέρ σε επιθυμητή θέση με αριθμό παλμών encoder. Ο έλεγχος της θέσης και της ταχύτητας των μοτέρ γίνεται σε κάθε ελεγκτή. Τα υπόλοιπα υποσυστήματα όπως το ηλεκτρομαγνητικό φρένο και proximity ελέγχονται επίσης από τους ελεγκτές.

5.2 Αρχικοποίηση συστήματος ελέγχου

Όπως ήδη έχει αναφερθεί, το αντίστροφο κινηματικό πρόβλημα χρησιμοποιείται για την εύρεση της επιθυμητής θέσης, αλλά για να εκτελεστεί η κίνηση απαιτείται να είναι γνωστή η προηγούμενη παρούσα θέση. Στις επαναλαμβανόμενες κινήσεις αυτό δε δημιουργεί πρόβλημα αλλά υφίσταται στην πρώτη κίνηση καθώς με την διακοπή της τάσης το σύστημα χάνει τον αριθμό των παλμών του encoder. Για το λόγο αυτό σε κάθε εκκίνηση του συστήματος, ο κάθε σύνδεσμος του βραχίονα μετακινείται ώστε να βρει την αρχική του θέση. Αυτό πραγματοποιείται με τη βοήθεια των proximity και των τερματικών διακοπών.



Εικόνα 58 Θέση αρχικοποίησης

Η υλοποίηση της όλης διαδικασίας ξεκινάει με εντολή που θα εισάγει ο χρήστης στο Matlab και έπειτα θα διαβιβαστούν τα κατάλληλα δεδομένα στους ελεγκτές ώστε να ξεκινήσουν τη διαδικασία. Έπειτα από αυτή την εντολή, το matlab παραμένει σε κατάσταση αναμονής για την επιβεβαίωση της διεκπεραίωσης. Οι ελεγκτές πλέον αναλαμβάνουν να ολοκληρώσουν την διεργασία καθώς όλες οι υπόλοιπες εντολές που χρειάζονται για την υλοποίηση της αρχικοποίησης βρίσκονται στο λογισμικό τους πρόγραμμα.

Κατά την διάρκεια αυτής της διεργασίας ο ένας από τους δυο ελεγκτής ελέγχει το ρεύμα των proximity μέσω ρελέ.

Όταν ξεκινήσει η αρχικοποίηση, ενεργοποιείται και χρησιμοποιεί πρώτα αυτός το δικό του proximity για να βρει την αρχική του θέση και έπειτα επικοινωνεί έμμεσα, μέσω του matlab με τον άλλον ελεγκτή ώστε να ξεκινήσει και αυτός με την σειρά του την εύρεση θέσης. Όταν τελειώσει ενημερώνει και αυτός τον ελεγκτή ώστε να διακοπεί το ρεύμα στα proximity.

```
Command Window
>> menu
for home press 1, for restart press 2,for move press 3 , for scripts restart press 4, for stop pree 5
HOME 1

k1 =
+

k1 =
+

k1 =
1

done
homing 2
done2
homing com 17 coplete
homing 3 start
```

Εικόνα 59 αρχικοποίηση μέσω matlab

```
Command Window

k1 =
^ECHOFF 1

k1 =
+

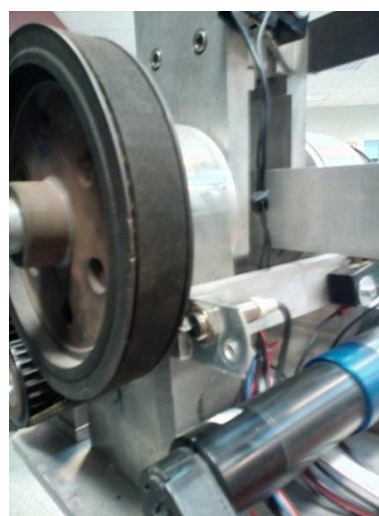
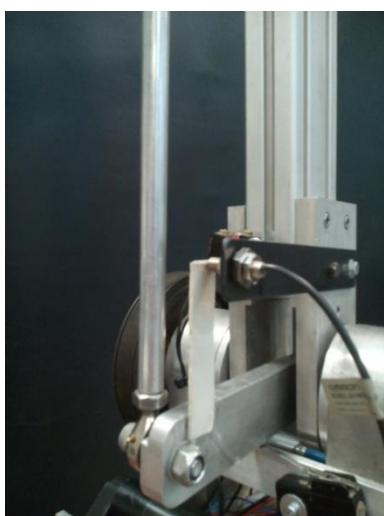
k1 =
+

k1 =
1

done3
homing com 17 + com 15 coplete
found home.
ready for main program
fx for home press 1, for restart press 2,for move press 3 , for scripts restart press 4, for stop pree 5
```

Εικόνα 60 ολοκλήρωση αρχικοποίησης

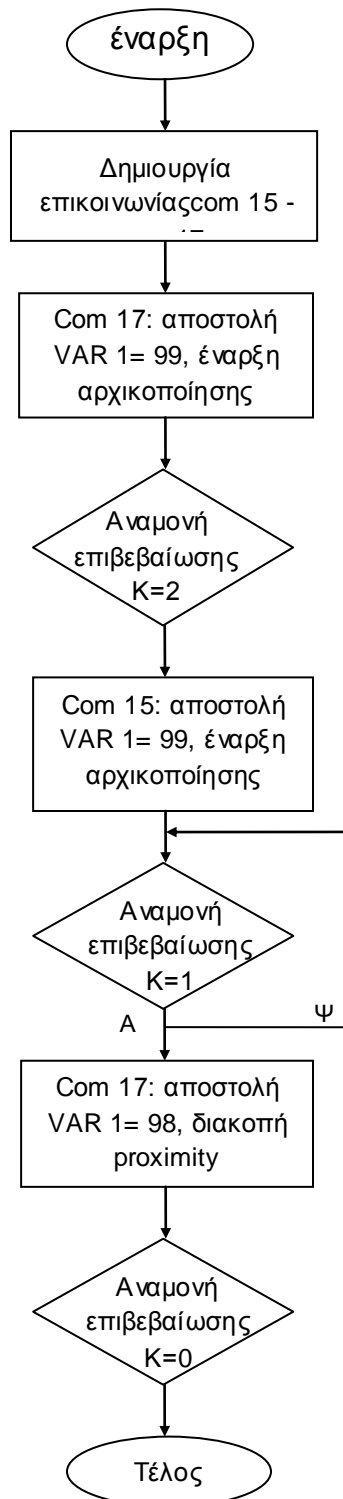
Το κάθε μοτέρ με τη σειρά του εκκινείται σε ανοιχτό βρόγχο μέχρις ότου ακουμπήσει τερματικό διακόπτη ή κλείσει κύκλωμα το proximity. Στην περίπτωση που φτάσει σε τερματικό διακόπτη τότε ξεκινάει κίνηση αντίστροφα. Όταν ο αισθητήρας ενεργοποιηθεί και δώσει σήμα εξόδου, τότε σημαίνει ότι ο συγκεκριμένος σύνδεσμος βρίσκεται στην αρχική του θέση. Το ίδιο πραγματοποιείται και στα υπόλοιπα μοτέρ. Όταν ολοκληρωθεί αυτή η διαδικασία μηδενίζονται οι μετρητές των παλμών encoder και ενημερώνεται το matlab για την ετοιμότητα των ελεγκτών να λάβουν τις νέες επιθυμητές θέσεις.



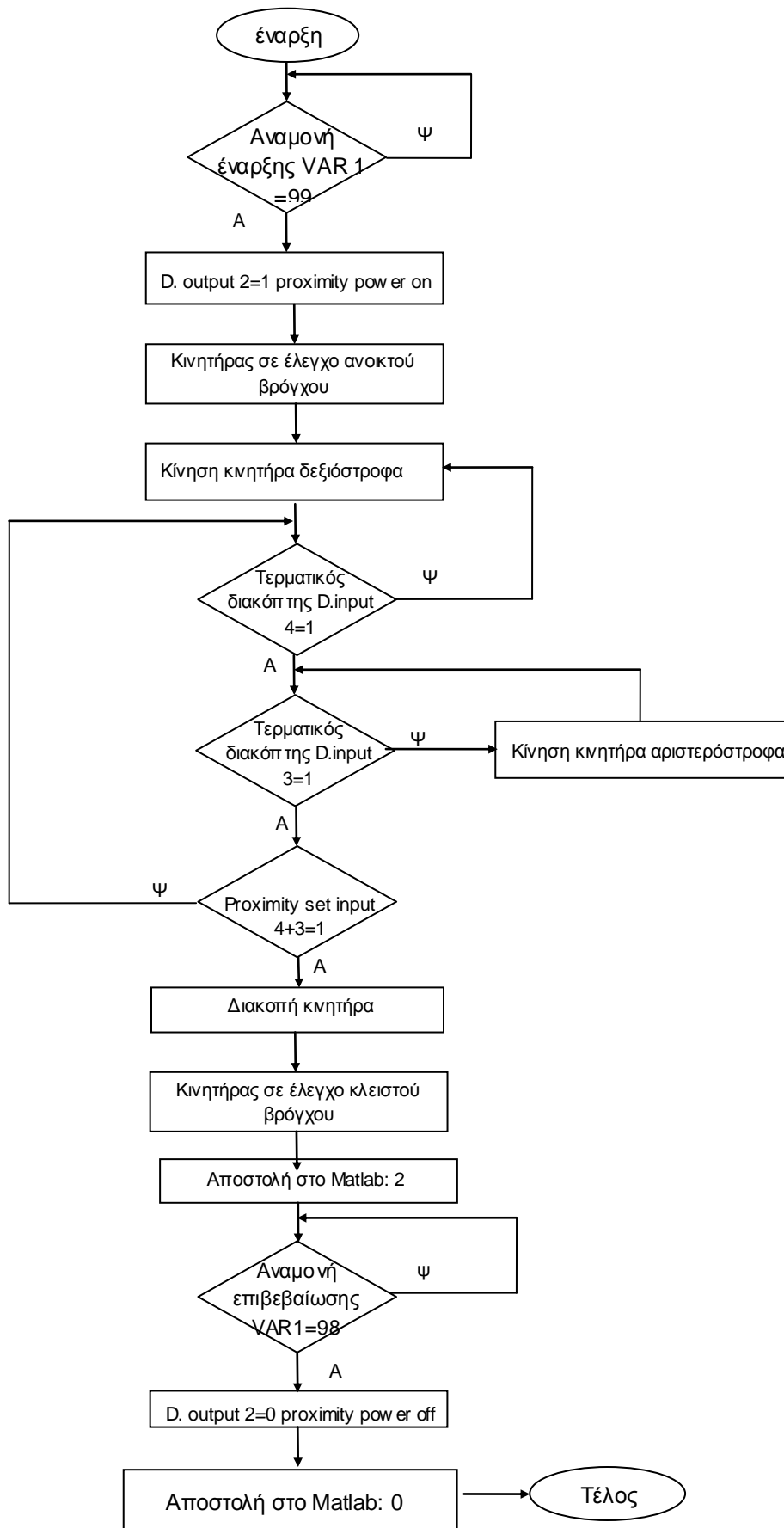
Εικόνα 61 Αισθητήρες proximity της άρθρωσης Θ_3 και Θ_2

Το σήμα που επιστρέφει το proximity όταν ενεργοποιηθεί έχει οριστεί σαν σήμα safety stop από την παραμετροποίηση των σημάτων του ελεγκτή. Αυτό έχει γίνει ώστε να σταματάει άμεσα το μοτέρ στη θέση όπου βρίσκεται το proximity. Για γίνει άρση του safety stop και να μπορεί να κινηθεί εκ νέου το μοτέρ, πρέπει να σταματήσει να υφίσταται το σήμα του proximity. Για να επιτευχθεί αυτό φυσικά πρέπει να κινηθεί το μοτέρ. Όπως είναι φανερό, δημιουργείται ένας φαύλος κύκλος που η λύση του είναι η διακοπή ρεύματος προς τα αισθητήρια. Με αυτόν τον τρόπο διακόπτεται το σήμα και μπορεί να κινηθεί το μοτέρ. Αυτή η διαδικασία γίνεται σε όλες τις αρθρώσεις και τελικά όταν ολοκληρωθεί η αρχικοποίηση το ρεύμα στα αισθητήρια μένει κλειστό.

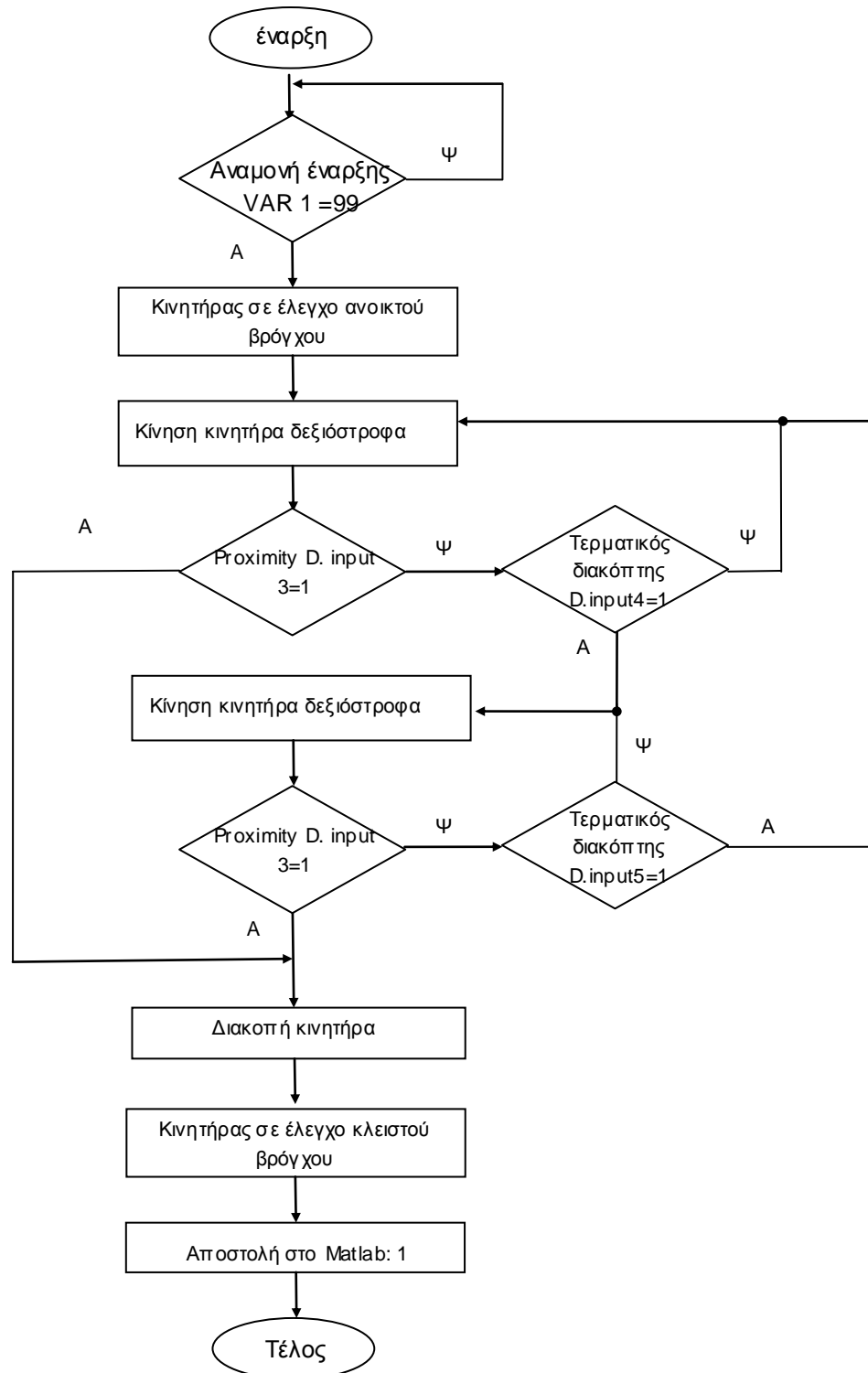
Λογικό διάγραμμα αλγόριθμου αρχικοποίησης του Matlab



Λογικό διάγραμμα αλγόριθμου αρχικοποίησης του ελεγκτή com 17



Λογικό διάγραμμα αλγόριθμου αρχικοποίησης του ελεγκτή com 15



5.3 Κώδικας που αναπτύχθηκε στο Matlab

5.3.1 Γενικά

Για το λειτουργικό matlab έχουν γραφεί πέντε προγράμματα με τη μορφή script. Τα προγράμματα αυτά καλούν όποτε χρειάζεται το ένα το άλλο και στέλνουν τα δεδομένα ή τις εντολές στους ελεγκτές μέσω θύρας usb. Για τη σωστή μετάδοση και λήψη τα δεδομένα μετασχηματίζονται από δεκαεξαδικό σε δεκαδικό σύστημα.

5.3.2 Αλγόριθμος μενού

Το πρώτο πρόγραμμα είναι το κύριο μενού, που περιέχει ερώτηση προς το χρήστη για τις λειτουργίες που μπορούν να γίνουν. Ανάλογα με την απάντηση καλείται και το αντίστοιχο πρόγραμμα. Έχει προστεθεί μια βιβλιοθήκη που χρωματίζει κάποιες λέξεις στα μηνύματα που εμφανίζει το matlab

```
1 | %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%main menu%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
2 | - ap=0;
3 | - while (ap ~=5)
4 | -     cprintf('text', 'for home press ');
5 | -     cprintf('red', '1\n');
6 | -     cprintf('text', 'for locomotion press ');
7 | -     cprintf('red', '2\n');
8 | -     cprintf('text', 'for move press ');
9 | -     cprintf('red', '3\n');
10 | -     cprintf('text', 'for scripts restart press ');
11 | -     cprintf('red', '4\n');
12 | -     cprintf('text', 'for stop press ');
13 | -     cprintf('red', '5\n');
14 | -     message = '';
15 | -     ap = input(message);
16 | -     while ((ap~=1) && (ap~=2) && (ap~=3) && (ap~=4) && (ap~=5))
17 | -         disp('incorrect choice input 1 to 5')
18 | -         ap=input(message);
19 | -     end
20 |
```

Εικόνα 62a Αλγόριθμος μενού

```
21 -     if ap==1
22 -         HOME ()
23 -         pause(1)
24 -         clc;
25 -         cprintf('Green', 'ready for main program\n') ;
26 -     end
27 -     if ap==2
28 -         locomotion()
29 -     end
30 -     if ap==3
31 -         MAIN()
32 -     end
33 -     if ap==4
34 -         REPEAT()
35 -         pause(1)
36 -         clc;
37 -         cprintf('Green', 'scripts restartet\n') ;
38 -     end
39 - end
```

Εικόνα 62b Αλγόριθμος μενού

```
Command Window
>> menu
for home press 1
for locomotion press 2
for move press 3
for scripts restart press 4
for stop press 5
fx |
```

Εικόνα 63 menu matlab

5.3.3 Αλγόριθμος αρχικοποίησης

Το δεύτερο πρόγραμμα είναι για την αρχικοποίηση του συστήματος. Οι πρώτες σειρές του κώδικα αφορούν την επίτευξη της επικοινωνίας με τους ελεγκτές. Στον υπολογιστή που χρησιμοποιήθηκε για την εργασία οι δυο ελεγκτές επιτυγχάνουν την επικοινωνία μέσω των θυρών com 15 και com 17. Όπως φαίνεται και στην εικόνα 64a γίνεται η παραμετροποίηση της ταχύτητας της επικοινωνίας και ανοίγουν οι θύρες με τους ελεγκτές.

```

1  function HOME
2  %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
3
4  %% serials start%%
5  s = serial('COM17');    % COM 17
6  set(s, 'BaudRate', 115200)           %set epikoinonias
7  set(s, 'Terminator', 'CR', 'Timeout', 10000)
8  fopen(s);
9
10 s2 = serial('COM15');    %COM 15
11 set(s2, 'BaudRate', 115200)           %set epikoinonias
12 set(s2, 'Terminator', 'CR', 'Timeout', 10000)
13 fopen(s2);
14

```

Εικόνα 64a Αλγόριθμος αρχικοποίησης

Στις αμέσως επόμενες εντολές το matlab στέλνει την εντολή εκκίνησης της αρχικοποίησης. Αυτό γίνεται γράφοντας 99 στην μεταβλητή var 1 του προγράμματος του ελεγκτή. Εφόσον σταλεί η εντολή ενεργοποίησης του μοτέρ, το matlab περιμένει την επιβεβαίωση που θα στείλει ο ελεγκτής. Υπάρχει μια εντολή επανάληψης, γιατί όταν ο ελεγκτής λαμβάνει μια εντολή μέσω του Buffer, στέλνει κάποια σύμβολα σαν επιβεβαίωση και πρέπει να εκκαθαριστούν. Οπότε και υπάρχει ο βρόγχος επανάληψης που λαμβάνει τα δεδομένα μέχρι να του αποσταλεί η λέξη done ως επιβεβαίωση της διεκπεραίωσης της κίνησης. Όταν αυτή πραγματοποιηθεί τότε εμφανίζει το matlab σχετικό μήνυμα στον χρήστη.

```

15  %% home 1 + 2 %%
16  disp ('home 1+2')
17  fwrite(s, strcat('!VAR 1 =99', hex2dec('0D'))) %ekinisi home mesaio
18  k1=fscanf(s); %serial scan
19  pin=size (k1); %% eisagogi se metabliti to megethos tou pin.
20  flag=0;
21  while pin(2)~= 5 || flag==0
22      k1=fscanf(s); %perimenei na parei epibebaiosoi home1+2
23      pin=size(k1);
24      if pin(2)==5
25          if k1(1:4)=='done'
26              flag=1;
27          end
28      end
29  end
30  disp ('done1+2')
31  disp('homing com 17 coplete')
32

```

Εικόνα 64b Αλγόριθμος αρχικοποίησης

Αμέσως μετά γίνεται η εκκίνηση στο δεύτερο ελεγκτή και το Matlab περιμένει την επιβεβαίωση της πραγματοποίησης με τον ίδιο τρόπο λαμβάνοντας τη λέξη done2.

```

33  %% homing 3 com 15
34  disp ('homing 3 start')
35  fwrite(s2, strcat('^ECHO 1', hex2dec('0D')))
36  fwrite(s2, strcat('!VAR 1 =99', hex2dec('0D'))) %ekinisi home akro
37  k1=fscanf(s2);
38  pin2=size (k1); % i epistrofi epalitheusis exei 6 stoixeia
39  flag2=0;
40  while pin2(2)~= 6 || flag2==0
41      k1=fscanf(s2); %perimenei na parei epibebaiosoi home1+2
42      pin2=size(k1);
43      if pin2(2)==6
44          if k1(1:5)=='done2'
45              flag2=1;
46          end
47      end
48  end
49  disp('homing com 17 coplete')
50

```

Εικόνα 64c Αλγόριθμος αρχικοποίησης

Η εντολή !VAR 1 =98 εισάγει το νούμερο 98 στην μεταβλητή var 1 του προγράμματος του ελεγκτή. Είναι εντολή που περιμένει ο ελεγκτής του com 17 να λάβει, για να κλείσει το ρεύμα στα αισθητήρια καθώς σε αυτόν είναι συνδεδεμένη η έξοδος που ανοιγοκλείνει το ρελέ του ρεύματος στα proximity. Και αμέσως μετά σταματά η επικοινωνία κλείνοντας τις σειριακές πύλες.

```

51 %%
52 - fwrite(s, strcat('!VAR 1 =98', hex2dec('0D'))) %sentoli gia power off proximit;
53 - disp ('found home.')
54
55 - fclose(s);
56 - fclose(s2);
57 - end
58

```

Εικόνα 64d Αλγόριθμος αρχικοποίησης

5.3.4 Κίνηση από σημείο σε σημείο – point to point

Το τρίτο πρόγραμμα του matlab αφορά την κυρίως λειτουργία του βραχίονα. Εφόσον έχει ολοκληρωθεί η αρχικοποίηση μπορεί ο βραχίονας να κινηθεί σε επιθυμητές θέσεις. Στην αρχή του προγράμματος ο χρήστης πληκτρολογεί την νέα επιθυμητή θέση με τη μορφή συντεταγμένων στον χώρο σε εκατοστά. Έπειτα, ρυθμίζεται και ξεκινά η επικοινωνία μεταξύ του matlab και των ελεγκτών.

```

1 function MAIN
2 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% main program %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
3 - L1=68; L2=61; %mikos braxiona
4 - prompt = 'eisagogi x,y,z'; %eigagogi desire position
5 - disp (prompt)
6 - prompt1 = 'x?';
7 - prompt2 = 'y?';
8 - prompt3 = 'z?';
9 - x = input(prompt1)
10 - y = input(prompt2)
11 - z = input(prompt3)
12
13 %set epikoinonias
14 - s = serial('COM17'); % COM 17
15 - set(s, 'BaudRate', 115200) %set epikoinonias
16 - set(s, 'Terminator', 'CR', 'Timeout', 10000)
17 - fopen(s);
18
19 - s2 = serial('COM15'); %COM 15
20 - set(s2, 'BaudRate', 115200) %set epikoinonias
21 - set(s2, 'Terminator', 'CR', 'Timeout', 10000)
22 - fopen(s2);

```

Εικόνα 65a Κίνηση από σημείο σε σημείο

Αμέσως μετά λύνονται οι εξισώσεις του αντίστροφου κινηματικού προβλήματος για την νέα επιθυμητή θέση και τέλος μετατρέπονται τα παραδοσεί μοίρες.

```

24      %antistrofo kinimatiko |
25 -    th1=atan2(y,x);
26 -    th1m=rad2deg(th1);
27 -    th2mth3=acos((-y/sin(th1))^2-z^2+L1^2+L2^2)/(2*L1*L2);
28 -    k1=L1*cos(th2mth3)-L2;
29 -    k2=L1*sin(th2mth3);
30 -    D=4*(-z^2+k2^2+k1^2);
31 -    th3=2*atan((2*k1+sqrt(D))/(2*z+2*k2));
32 -    th3m=rad2deg(th3);
33 -    th2=th2mth3+th3;
34 -    th2m=rad2deg(th2);
35

```

Εικόνα 65b Κίνηση από σημείο σε σημείο

Στη συνέχεια γίνεται ο μετασχηματισμός των μοιρών σε παλμούς των encoder, σύμφωνα με τον τρόπο που προαναφέρθηκε στο κεφάλαιο 3.3 και η αποστολή τους στους ελεγκτές. Τέλος, κλείνει η επικοινωνία με τη σειριακή θύρα.

```

36      %com 17 thita 2
37 -    b1= '!VAR';
38 -    b2=' 1 =';
39 -    b3=(th2m*616.666666)*3;    %encoder pulses 500*(X4) *motor me meiosi /imanta
40 -    b3 = num2str(b3)
41 -    fwrite(s, strcat(b1,b2,b3, hex2dec('0D')))    %apostoli sto buffer
42
43      % com 17 mikro moter THITA 1
44 -    b1= '!VAR';
45 -    b2=' 2 =';
46 -    b3=(th1m*744.444444);    %ENCODER PULSE 500 * (4X METOPA) * 134 MEIOSI MOTER
47 -    b3 = num2str(b3)
48 -    fwrite(s, strcat(b1,b2,b3, hex2dec('0D')))
49 -    fwrite(s, strcat('!VAR 9 =5', hex2dec('0D')))    %DINEI TO OK gia start
50
51      %com 15 thita 3
52 -    b1= '!VAR';
53 -    b2=' 1 =';
54 -    b3=(th3m*616.666666)*1.5;
55 -    b3 = num2str(b3)
56 -    fwrite(s2, strcat(b1,b2,b3, hex2dec('0D')))
57 -    fwrite(s2, strcat('!VAR 9 =5', hex2dec('0D')))    %DINEI TO OK gia start
58
59 -    fclose(s);
60 -    fclose(s2);
61 -    end

```

Εικόνα 65c Κίνηση από σημείο σε σημείο

5.3.5 Αλγόριθμος επανεκκίνησης

Αυτή η διαδικασία αφορά την επανεκκίνηση των προγραμμάτων στους ελεγκτές. Μπορεί να γίνει οποιαδήποτε στιγμή και αφορά κυρίως την επιλογή να επαναληφθεί η διαδικασία του homing.

```
1 function REPEAT
2 %% restart the scripts %%
3
4 s = serial('COM17');           % COM 17
5 set(s, 'BaudRate', 115200)     %set epikoinonias
6 set(s, 'Terminator', 'CR', 'Timeout', 10)
7 fopen(s);
8
9 s2 = serial('COM15');          %COM 15
10 set(s2, 'BaudRate', 115200)   %set epikoinonias
11 set(s2, 'Terminator', 'CR', 'Timeout', 10)
12 fopen(s2);
13
14 fwrite(s, strcat('!VAR 4 = 1', hex2dec('0D'))) %serial entoli gia repeat script
15 fwrite(s2, strcat('!VAR 4 = 1', hex2dec('0D')))
16
17 fclose(s);
18 fclose(s2);
19 end
20
```

Εικόνα 66 Αλγόριθμος επανεκκίνησης

5.3.6 Κίνηση επί ευθυγράμμου τμήματος

Αυτό το πρόγραμμα δημιουργήθηκε για τη μετατόπιση από ένα σημείο σε ένα επιθυμητό σημείο, σε ευθύγραμμο τμήμα. Απόρροια αυτού είναι ο έλεγχος του άκρου του βραχίονα και ο σχεδιασμός μονοπατιού «αναγκάζοντας» το να περάσει από ενδιάμεσα σημεία. Για να επιτευχθεί αυτό, στέλνεται νέα επιθυμητή θέση ανά συγκεκριμένο χρονικό διάστημα, έτσι ώστε πριν ο βραχίονας φτάσει σε ένα σημείο να του δίνεται εντολή να κινηθεί στο επόμενο. Έτσι, διαγράφει σταθερή ευθεία και όχι τυχαία τροχιά μέχρι το τελικό επιθυμητό σημείο. Το συγκεκριμένο πρόγραμμα γράφτηκε για κίνηση κατά τον άξονα ΥΥ. (παράλληλα με το έδαφος).

Το πρόγραμμα ξεκινάει καθορίζοντας το σετ επικοινωνίας. Θέτονται οι ταχύτητες και οι επιταχύνσεις που έχουν καθοριστεί καθώς και το πλήθος των ενδιάμεσων σημείων, ο χρόνος αποστολής του κάθε νέου σημείου και η απόσταση του ενός από το άλλο σημείο.

```

1  function locomotion
2  %set episkoinonias
3  s2 = serial('COM15');           %COM 15
4  set(s2,'BaudRate',115200)       %set epikoinonias
5  set(s2, 'Terminator','CR','Timeout',10000)
6
7  s = serial('COM17');           % COM 17
8  set(s,'BaudRate',115200)       %set epikoinonias
9  set(s, 'Terminator','CR','Timeout',10000)
10 fopen(s);
11 fopen(s2);
12
13
14 % set speed, acc, dec
15 fwrite(s2, strcat('!AC 1 2500', hex2dec('0D'))) %SET ACC TO 250RPM
16 fwrite(s2, strcat('!DC 1 2500', hex2dec('0D'))) %SET DEC TO 250RPM com15
17 fwrite(s2, strcat('!S 1 50', hex2dec('0D'))) %VELOCITY 50 RPM
18
19 fwrite(s, strcat('!AC 1 4000', hex2dec('0D'))) %SET ACC TO 400RPM
20 fwrite(s, strcat('!DC 1 4000', hex2dec('0D'))) %SET DEC TO 400RPM com17
21 fwrite(s, strcat('!S 1 200', hex2dec('0D'))) %VELOCITY 200RPM
22
23 points=130; %%%%%%%%%%%%%%%%%%%%%%%%%%checkpoints%%%%%%%%%%%%%%%%%%%%%%%%%
24 time=0.045; %%%%%%%%%%%%%%%%%%%%%%%%%%send new point after time (sec)%%%%%%%%%
25 metat=0.3; %%%%%%%%%%%%%%%%%%%%%%%%%%next point location (cm)%%%%%%%%%%%%%
26 repeat=1;

```

Εικόνα 67a Κίνηση επί ευθυγράμμου τμήματος

Για τη γρηγορότερη εκτέλεση του συγκεκριμένου script δημιουργούνται άδειοι πίνακες για τις θέσεις των σημείων. Λύνεται το αντίστροφο κινηματικό για την πρώτη θέση, που θα εισάγει ο χρήστης και έπειτα γεμίζουν οι πίνακες με τα προκαθορισμένα πλέον σημεία.

```

28 - i=1;
29 - th3m=zeros(points,1); %dimiourgia pinakon
30 - th2m=zeros(points,1);
31 - y=zeros(points,1);
32
33 - L1=68; L2=61; %mikos braxiona
34 - prompt = 'eisagogi x,y,z'; %eigagoki arxikis thesis
35 - disp(prompt)
36 - prompt1='x?';
37 - prompt2='y?';
38 - prompt3='z?';
39 - x = input(prompt1);
40 - y(1) = input(prompt2);
41 - z = input(prompt3);
42
43 %antistrofo kinimatiko
44 - th1=atan2(y(1),x);
45 - th1m=rad2deg(th1);
46 - th2mth3=acos((-y(1)/sin(th1))^2-z^2+L1^2+L2^2)/(2*L1*L2);
47 - k1=L1*cos(th2mth3)-L2;
48 - k2=L1*sin(th2mth3);
49 - D=4*(-z^2+k2^2+k1^2);
50 - th3=2*atan((2*k1+sqrt(D))/(2*z+2*k2));
51 - th3m(1)=rad2deg(th3);
52 - th2=th2mth3+th3;
53 - th2m(1)=rad2deg(th2);

```

Εικόνα 67b Κίνηση επί ευθυγράμμου τμήματος

```

54
55
56 - for i=2:points
57 -     y(i)=y(i-1)+metat;
58 -     %antistrofo kinimatiko
59 -     th1=atan2(y(i),x);
60 -     th1m=rad2deg(th1);
61 -     th2mth3=acos((-y(i)/sin(th1))^2-z^2+L1^2+L2^2)/(2*L1*L2);
62 -     k1=L1*cos(th2mth3)-L2;
63 -     k2=L1*sin(th2mth3);
64 -     D=4*(-z^2+k2^2+k1^2);
65 -     th3=2*atan((2*k1+sqrt(D))/(2*z+2*k2));
66 -     th3m(i)=rad2deg(th3);
67 -     th2=th2mth3+th3;
68 -     th2m(i)=rad2deg(th2);
69 - end

```

Εικόνα 67c Κίνηση επί ευθυγράμμου τμήματος

Εφόσον δημιουργούνται όλα τα σημεία που θα αποσταλούν, μειώνεται ο χρόνος αποστολής νέων σημείων. Στο πειραματικό μέρος της παρούσας εργασίας έχει επιλεγεί να γίνεται η αποστολή σημείων ανά 50 millisecond, αλλά υπάρχει η δυνατότητα αποστολής ανά 25 millisecond. Ακολουθεί η αποστολή των θέσεων στους ελεγκτές.

```

70      %% apostoli points sta controllers
71      aa=1;
72      for aa=1:repeat
73          tic;
74          b1= '!P!';
75          b2=' 1 =';
76          b3=(th2m(1)*616.666666)*3;    %encoder pulses 500*(X4) *motor me meiosi /
77          b3 = num2str(b3);
78          fwrite(s, strcat(b1,b2,b3, hex2dec('0D'))); %apostoli sto buffer gia 21
79
80          b1= '!P!';
81          b2=' 1 =';
82          b3=(th3m(1)*616.666666)*1.5;
83          b3 = num2str(b3);
84          fwrite(s2, strcat(b1,b2,b3, hex2dec('0D')))
85      while toc<time;
86          pause(0.001);
87      end

```

Εικόνα 67d Κίνηση επί ευθυγράμμου τμήματος

```

89      for i= 2:points
90          tic;
91          b1= '!P!';
92          b2=' 1 =';
93          b3=(th2m(i)*616.666666)*3;    %encoder pulses 500*(X4) *motor me meio
94          b3 = num2str(b3);
95          fwrite(s, strcat(b1,b2,b3, hex2dec('0D'))); %apostoli sto buffer gi
96
97          b1= '!P!';
98          b2=' 1 =';
99          b3=(th3m(i)*616.666666)*1.5;
100         b3 = num2str(b3);
101         fwrite(s2, strcat(b1,b2,b3, hex2dec('0D')))
102
103     while toc<time;
104         pause(0.001);
105     end
106     toc
107 end
108 pause(4);    %pause stin teliki thesi 4 sec

```

Εικόνα 67e Κίνηση επί ευθυγράμμου τμήματος

Πραγματοποιείται μια καθυστέρηση στο τελικό σημείο 4 δευτερόλεπτα και αμέσως μετά η επιστροφή του βραχίονα στο σημείο εκκίνησης με τον ίδιο τρόπο. Τέλος, κλείνει το πρόγραμμα ελευθερώνοντας τις συριακές θύρες.

```
109     %% epistrofi stin arxiki thesi
110     tic;
111     j=points-1;
112     for i= j:-1 : 1
113         tic;
114         b1= '!P';
115         b2=' 1 =';
116         b3=(th2m(i)*616.666666)*3;    %encoder pulses 500*(X4) *motor me meik
117         b3 = num2str(b3);
118         fwrite(s, strcat(b1,b2,b3, hex2dec('0D'))); %apostoli sto buffer gi
119
120         b1= '!P';
121         b2=' 1 =';
122         b3=(th3m(i)*616.666666)*1.5;
123         b3 = num2str(b3);
124         fwrite(s2, strcat(b1,b2,b3, hex2dec('0D')))
125
126         while toc<time;
127             pause(0.001);
128         end
129         toc
130     end
131     pause(4)
132 end
133
134 fclose(s);
135 fclose(s2);
136 end
```

Εικόνα 67f Κίνηση επί ευθυγράμμου τμήματος

5.4 Microbasic Αλγόριθμοι

5.4.1 Γενικά

Έχει δημιουργηθεί ένα πρόγραμμα για κάθε έναν ελεγκτή μέσω του λογισμικού που τους συνοδεύει. Ο τρόπος γραφής του κώδικα έχει πολλά κοινά με τη γλώσσα προγραμματισμού c. Στον ελεγκτή που οδηγεί τα δυο μοτέρ και έχει οριστεί η θύρα com 17 για την επικοινωνία του, το πρόγραμμα ξεκινάει αρχικοποιώντας με το 0 τις μεταβλητές που θα χρησιμοποιηθούν.

5.4.2 Αρχικοποίηση

```

1  | '*****METAPTUXIAKI ERGASIA ALEXIADIS DIMITRIOS*****'
2  | '*****COM 17 version 4.0*****'
3  | '*****SCRIPT GIA CONTROLLER ARTHROSIS THITA 1 KAI THITA 2*****'
4  |
5  | ' FIRST STEP IS TO FIND ZERO POINT (PROXIMITY HOMING)
6  | ' Apply 50% forward or reverse in open loop power until home switch(proximity) is activated
7  | ' then change to Closed Loop Count Position mode
8  |
9  | home:
10 | i=0
11 | G=0
12 | setcommand (_VAR, 1 , 0)
13 | setcommand (_VAR, 4 , 0)
14 |
15 | while (i <> 99 )
16 |   i=getvalue (_var,1 )
17 |   WAIT(10)
18 | end while
19 |
20 | setcommand(_DSET, 2)           ' sets digital output 2 on      'POWER TO  proximity
21 | wait (500)
22 | setconfig (_MMOD, 1, 0)       ' Configure channel 1 in open loop mode
23 | setconfig (_MMOD, 2, 0)       ' Configure channel 2 in open loop mode
24 |
25 | topagain:
26 | ' Loop until limit switch is set
27 | topia:
28 | if (getvalue(_DI, 3) = 0)      'IF LIMIT SWITCH NOT SET MOVE
29 |   ' switch not hit
30 |   setcommand(_G, 1, 100)      'OPEN LOOP SPEED SET 100 RPM
31 |   goto topia
32 | end if
33 |

```

Εικόνα 68a Αρχικοποίηση com 17

Η μεταβλητή var 1 είναι αυτή που ξεκινάει τη διαδικασία της αρχικοποίησης και με τη δημιουργία ενός βρόγχου επανάληψης αναμένει την εγγραφή σε αυτή το νούμερο 99. Όπως παρατηρείται και στην εικόνα 68a, τα τμήματα του προγράμματος μπορούν να ονομαστούν. Αυτό βοηθάει όταν γίνει χρήση της εντολής goto. Μετά την εγγραφή της μεταβλητής από το matlab ακολουθούνται εντολές για την αρχικοποίηση.

Πρώτα, ενεργοποιείται η έξοδος 2 που δίνει τάση στα αισθητήρια και έπειτα παραμετροποιούνται τα μοτέρ με έλεγχο ανοικτού βρόγχου. Αμέσως μετά, δίνεται η εντολή να κινηθεί το πρώτο μοτέρ της γωνίας θ_2 , ορίζοντας την ταχύτητα και ελέγχοντας αν ενεργοποιηθεί η είσοδος 3 που έχει συνδεθεί ο τερματικός διακόπτης προς την μεριά όπου κινείται το μοτέρ. Στην περίπτωση που ενεργοποιηθεί η είσοδος του τερματικού τότε κινείται το μοτέρ με την αντίθετη κατεύθυνση.

```

34 top1b:
35 if (getvalue(_DI, 4) = 0)                                'IF LIMIT SWITCH NOT SET MOVE
36     setcommand(_G, 1, -200)                             'OPEN LOOP SPEED SET REVERCE 200 RPM
37     goto top1b
38 end if
39
40 while (getvalue(_DI, 3) = 0 or getvalue(_DI, 4) = 0)    'IF INPUT 3 AND 4 NOT SET (NO PROXIMITY SET) REPEAT PROCESS
41     goto topagain
42 end while
43
44
45 wait(2000)
46 print( "done\r")                                       ''SERIAL TO MATLAB
47 WAIT(100)                                             ''WAIT UNTIL MOTOR STOP
48 setcommand (_C , 1, 166500)                          ''ZERO COUNTS, HOME IS 90 DECGEES 166500 COUNTS
49 setconfig(_MMOD, 1, 3)                                ' Switch to closed loop encoder count mode
50
51
52     *****'home 2'*****
53
54 setcommand (_C , 2, 67000)                             ''midenismos counter 67000=90 moires dil na koitaei mprosta
55 setconfig(_MMOD, 2, 3)                                ' Switch to closed loop encoder count mode
56 wait(500)
57
58 while (i <> 98 )                                       ''WAIT UNTIL MATLAB SEND ORDER TO POWER OFF PROXIMITY
59 i=getvalue (_var,1 )
60 WAIT(100)
61 end while
62 setcommand(_DRES, 2)                                  ' sets digital output 2 off      POWER OFF PROXIMITY
63 print ("0\r")                                         'all set... SERIAL TO MATLAB

```

Εικόνα 68b Αρχικοποίηση com 17

Ομοίως, γίνεται έλεγχος για τον αντίστοιχο τερματικό διακόπτη που συνδέθηκε στην είσοδο 4. Η διαφορά στην ταχύτητα που υπάρχει εξηγείται από το γεγονός ότι όταν ο βραχίονας κινείται ώστε να σηκώσει το σύνδεσμο σε κάθετη θέση, χρειάζεται περισσότερη ροπή. Στον αντίποδα, όταν κινείται αντίθετα η ροπή είναι αρνητική.

Ο αισθητήρας όταν ενεργοποιηθεί στέλνει σήμα και στις 2 εισόδους των τερματικών. Όταν συμβεί αυτό, ορίζεται εκείνο το σημείο σαν αρχή ή μηδέν και εισάγεται σαν νούμερο παλμών το 166500 που αντιστοιχεί στην ορθή θέση του συνδέσμου. Αμέσως μετά αλλάζει ο τρόπος ελέγχου του μοτέρ σε κλειστού βρόγχου με ανάδραση των encoders (close loop count position) και ενημερώνεται το matlab για την διεκπεραίωση της λειτουργίας με τη λέξη done.

Το επόμενο βήμα είναι η αρχικοποίηση του μοτέρ της γωνίας θ_1 και έπειτα υπάρχει βρόγχος επανάληψης για το πάγωμα του προγράμματος μέχρι να γίνει αρχικοποίηση στον ελεγκτή του com 15. Όταν αυτό πραγματοποιηθεί το matlab θα αποστέλλει εντολή, όπου η μεταβλητή VAR 1 πάρει τιμή 98 και ο ελεγκτής θα κλείσει την έξοδο 2 που είναι το ρεύμα των proximity.

Ομοίως, η αρχικοποίηση στον ελεγκτή της θύρας com 15 αρχίζει δηλώνοντας τις μεταβλητές του προγράμματος ίσες με το μηδέν και ακολουθεί μια επανάληψη για όσο είναι σε αναμονή. Και αυτό γιατί όση ώρα κινείται ο σύνδεσμος του ελεγκτή com 17, υπάρχει περίπτωση ο παρών σύνδεσμος να τερματίσει και να προκληθεί ζημιά. Για το λόγο αυτό σε περίπτωση που ενεργοποιηθεί τερματικός διακόπτης στις εισόδους 4 ή 5 να κινηθεί αντίθετα κατά 500 παλμούς. Υπάρχει διπλός έλεγχος της κατάστασης των εισόδων 4 και 5 για την αποφυγή θορύβου.

```

9  home:
10
11  setcommand (_VAR, 4 , 0)
12  i=0
13  setcommand (_VAR, 1 , 0)
14
15  while (i <> 99 )           ''ENTOLI APO MATLAB GIA EKKINISI HOMING
16  if (getvalue(_DI, 4) = 1)   ''DIAKOPTES ASFALIAS''
17      WAIT(20)
18      if (getvalue(_DI, 4) = 1)
19          setcommand(_S, 1, 900)           '' SET VELOCITY 900RPM
20          SETCOMMAND(_AC, 1, 8000)         ''SET ACC TO 800RPM
21          SETCOMMAND(_DC, 1, 6000)         ''SET DEC TO 600RPM
22          setcommand(_PR, 1, 500)         ''MOVE 500 PULSES
23          wait(50)
24      END IF
25  END IF
26  if (getvalue(_DI, 5) = 1)
27      WAIT(20)
28      IF (getvalue(_DI, 5) = 1)           '' SET VELOCITY 900RPM
29          setcommand(_S, 1, 900)         ''SET ACC TO 800RPM
30          SETCOMMAND(_AC, 1, 8000)         ''SET ACC TO 800RPM
31          SETCOMMAND(_DC, 1, 6000)         ''SET DEC TO 600RPM
32          setcommand(_PR, 1, -500)       ''MOVE 500 PULSES REVERCE
33          wait(50)
34      END IF
35  END IF
36  i=getvalue (_var,1 )
37  WAIT (10)
38  end while
39  wait (2000)

```

Εικόνα 69a αρχικοποίηση com 15

Όταν δοθεί εντολή από το matlab για την εκκίνηση της αρχικοποίησης βάζοντας 99 στην μεταβλητή VAR 1, τότε ο έλεγχος περνάει σε ανοικτό βρόγχο κινώντας το σύνδεσμο προς τις δυο κατευθύνσεις.

```
--
42 topagain:
43 setconfig (_MMOD, 1, 0)           ' Configure channel 1 in open loop mode
44 ' Loop until limit switch is set
45 top1a:
46 if (getvalue(_DI, 5) = 0) AND (getvalue(_DI, 3) = 0)   ' switch not hit 5 and home not hit 3
47     setcommand(_G, 1, 150)                               ''SET SPEED OPEN LOOP 150 RPM
48     goto top1a
49 end if
50
51 top1b:
52 if (getvalue(_DI, 4) = 0) AND (getvalue(_DI, 3) = 0)   ' switch not hit 4 and home not hit 3
53     setcommand(_G, 1, -250)                              ''SET SPEED OPEN LOOP REVERCE 250 RPM
54     goto top1b
55 end if
56
57 while (getvalue(_DI, 3) = 0 )
58     goto topagain                                       ''IF PROXIMITY NOT ACTIVE YET REPEAT PROCESS
59 end while
60
61
62 setcommand(_G, 1, 0)                                   'moter stop
63 wait(50)                                              'WAIT UNTIL MOTOR STOP
64 setcommand (_C , 1, 0)                                'proximity set....home count=0
65 setconfig(_MMOD, 1, 3)                               'Switch to closed loop encoder count mode
66 wait(2000)
67 '
68 print( "done2\r")                                     'SERIAL SEND TO MATLAB
```

Εικόνα 69b Αρχικοποίηση com 15

Όταν ενεργοποιηθεί η είσοδος 3 που είναι το σήμα του proximity, τότε ο έλεγχος του μοτέρ γίνεται κλειστού βρόγχου (close loop count position) και ενημερώνεται το matlab για τη συνέχεια.

5.4.3 Αλγόριθμος επανεκκίνησης

Όπως προαναφέρθηκε, ανάμεσα στα προγράμματα του matlab υπάρχει ένα το οποίο ονομάζεται `repeat` και αφορά την επανεκκίνηση των προγραμμάτων των ελεγκτών. Η δομή στα προγράμματα των ελεγκτών παρουσιάζεται στην εικόνα 70.

```
topmain:  
  
restart=getvalue (_var,4 )  
  
if (restart=1) then  
goto home  
  
end if
```

Εικόνα 70 Αλγόριθμος επανεκκίνησης

5.4.4 Κυρίως πρόγραμμα

Εφόσον έχει πραγματοποιηθεί η αρχικοποίηση και στους δυο ελεγκτές τα προγράμματα μένουν αδρανή, μέχρις ότου το matlab δώσει εντολή για μετακίνηση σε νέα θέση. Για να γίνει αυτό πρέπει να γραφεί στη μεταβλητή `VAR 9=5`. Όταν συμβεί αυτό ο κάθε ελεγκτής θα λάβει και τον αριθμό των παλμών που πρέπει να κινηθεί στην μεταβλητή `VAR 1`.

```

101 k=getvalue (_var,9 )           'MENU CHOICE TO MOVE TO POSITION
102 if (k=5) then
103   setcommand(_VAR,9,1)
104   i=getvalue (_var,1 )         'DESIRE POSITION COUNTS
105   C=GETVALUE (_C, 1)          'ACTUAL COUNTS FOR SPEED SET
106   C=i-C
107   if ((C<20000) AND (C>0))     'SPEED SET. IF NEW POSITION IS TO FAR THEN SET SPEED
108     setcommand(_S, 1, 100)     ' SET VELOCITY 100RPM
109     SETCOMMAND(_AC, 1, 1500)   'SET ACC TO 150RPMpsec
110     SETCOMMAND(_DC, 1, 1500)  'SET DEC TO 150RPMpsec
111   end if
112   if ((C>=20000) and (C<36000))
113     setcommand(_S, 1, 150)     ' SET VELOCITY 150RPM
114     SETCOMMAND(_AC, 1, 4000)   'SET ACC TO 400RPMpsec
115     SETCOMMAND(_DC, 1, 4000)  'SET DEC TO 400RPMpsec
116   end if
117   if (C>=36000)
118     setcommand(_S, 1, 250)     ' SET VELOCITY 250RPM
119     SETCOMMAND(_AC, 1, 7000)   'SET ACC TO 700RPMpsec
120     SETCOMMAND(_DC, 1, 7000)  'SET DEC TO 700RPMpsec
121   end if
122   if ((C>-20000)AND (C<0))
123     setcommand(_S, 1, 100)     ' SET VELOCITY 100RPM
124     SETCOMMAND(_AC, 1, 1500)   'SET ACC TO 150RPMpsec
125     SETCOMMAND(_DC, 1, 1500)  'SET DEC TO 150RPMpsec
126   end if
127   if ((C<=-20000) and (C>-36000))
128     setcommand(_S, 1, 150)     ' SET VELOCITY 150RPM
129     SETCOMMAND(_AC, 1, 4000)   'SET ACC TO 4000RPMpsec
130     SETCOMMAND(_DC, 1, 4000)  'SET DEC TO 4000RPMpsec
131   end if

```

Εικόνα 71a Κυρίως πρόγραμμα Com 15

```

132 if (C<=-36000)
133   setcommand(_S, 1, 250)       ' SET VELOCITY 250RPM
134   SETCOMMAND(_AC, 1, 7000)    'SET ACC TO 700RPMpsec
135   SETCOMMAND(_DC, 1, 7000)    'SET DEC TO 700RPMpsec
136   end if
137   WAIT(50)
138   setcommand(_P, 1, i)         'MOVE DESIRE POS. COUNTS
139   end if
140   WAIT(50)
141   goto topmain
142

```

Εικόνα 71b Κυρίως πρόγραμμα Com 15

Εφόσον έχουν ληφθεί οι επιθυμητοί παλμοί του encoder, τότε ανάλογα από το πόσο απέχουν από την παρούσα θέση ρυθμίζεται και η ταχύτητα που θα κινηθεί ο βραχίονας όπως και η επιτάχυνση και επιβράδυνση του.

```

67 topmain:
68 restart=getvalue (_var,4 )           ''CHOICE FROM MENU TO RESTART
69 if (restart=1) then
70     goto home
71 end if
72 k=getvalue (_var,9 )                 ''CHOICE FROM MENU TO MOVE TO DESIRE POSITION
73
74 if (k=5) then
75     setcommand(_VAR,9,1)
76     i=getvalue (_var,1 )             ''DESIRE POSITION COUNTS MOTOR 1
77     C=GETVALUE (_C, 1)              ''ACTUAL COUNTS FOR SPEED SET
78     C=i-C
79
80     if ((C<30000)AND (C>0))          '' SET SPEED. IF DESIRE POS. IS TO FAR SPEED IS MAX
81         setcommand(_S, 1, 200)      '' SET VELOCITY 200RPM
82         SETCOMMAND(_AC, 1, 2000)    ''SET ACC TO 200RPMpsec
83         SETCOMMAND(_DC, 1, 2000)    ''SET DEC TO 200RPMpsec
84     end if
85     if ((C>=30000) and (C<80000))
86         setcommand(_S, 1, 400)      '' SET VELOCITY 400RPM
87         SETCOMMAND(_AC, 1, 10000)   ''SET ACC TO 1000RPMpsec
88         SETCOMMAND(_DC, 1, 6000)    ''SET DEC TO 600RPMpsec
89     end if
90     if (C>=80000)
91         setcommand(_S, 1, 450)      '' SET VELOCITY 450RPM
92         SETCOMMAND(_AC, 1, 10000)   ''SET ACC TO 1000RPMpsec
93         SETCOMMAND(_DC, 1, 6000)    ''SET DEC TO 600RPMpsec
94     end if
95     if ((C>-30000)AND (C<0))
96         setcommand(_S, 1, 200)      '' SET VELOCITY 200RPM
97         SETCOMMAND(_AC, 1, 2000)    ''SET ACC TO 200RPMpsec
98         SETCOMMAND(_DC, 1, 2000)    ''SET DEC TO 200RPMpsec
99     end if

```

Εικόνα 72a Κυρίως πρόγραμμα com 17

```

99     end if
100    if ((C<=-30000) and (C>-80000))
101        setcommand(_S, 1, 400)      '' SET VELOCITY 400RPM
102        SETCOMMAND(_AC, 1, 10000)   ''SET ACC TO 1000RPMpsec
103        SETCOMMAND(_DC, 1, 6000)    ''SET DEC TO 600RPMpsec
104    end if
105    if (C<=-80000)
106        setcommand(_S, 1, 450)      '' SET VELOCITY 450RPM
107        SETCOMMAND(_AC, 1, 10000)   ''SET ACC TO 1000RPMpsec
108        SETCOMMAND(_DC, 1, 6000)    ''SET DEC TO 600RPMpsec
109    end if
110    setcommand(_P, 1, i)              ''MOVE TO DESIRE COUNTS MOTOR 1 THITA 2
111    wait(10)
112    G=getvalue (_var,2 )              ''DESIRE POSITION COUNTS MOTOR 2
113    CO=GETVALUE (_C, 2)              ''ACTUAL COUNTS
114    CO=G-CO
115    WAIT(10)
116    setcommand(_P, 2, G)              ''MOVE TO DESIRE COUNTS MOTOR 2 THITA 1
117    wait(10)
118 end if
119
120 FL=GETVALUE (_C, 1)                 ''ACTUAL COUNTS
121 if ((getvalue(_DI, 4) = 1) AND (FL<169000)) ''SAFETY SWITCH ACTIVE IF THITA 2 IS
122     WAIT(20)
123 if (getvalue(_DI, 4) = 1)           ''SAFETY MOVE
124     setcommand(_S, 1, 200)          '' SET VELOCITY 200RPM
125     SETCOMMAND(_AC, 1, 1200)        ''SET ACC TO 120RPM
126     SETCOMMAND(_DC, 1, 1200)        ''SET DEC TO 120RPM
127     setcommand(_PR, 1, 500)          ''MOVE 500 PULSES
128     wait(200)
129 END IF

```

Εικόνα 72b Κυρίως πρόγραμμα com 17

```
129:   END IF
130: END IF
131:
132: if (getvalue(_DI, 3) = 1)
133:   WAIT(20)
134:   if (getvalue(_DI, 3) = 1)
135:     setcommand(_S, 1, 200)
136:     SETCOMMAND(_AC, 1, 1200)
137:     SETCOMMAND(_DC, 1, 1200)
138:     setcommand(_PR, 1, -500)
139:     wait(200)
140:   END IF
141: END IF
142:
143: goto topmain
```

Εικόνα 72c Κυρίως πρόγραμμα com 17

Τέλος, κινείται ο βραχίονας προς την επιθυμητή θέση και έπειτα με εντολή goto επιστρέφει στην αρχή του κυρίως προγράμματος. Όπως παρατηρείται στην εικόνα 72b,c στον ελεγκτή που ελέγχει το μεσαίο σύνδεσμο (com 17) έχει ληφθεί υπόψη και η περίπτωση όπως και στο δεύτερο ελεγκτή να ενεργοποιήσει κάποιον τερματικό κατά την κίνηση του. Και εδώ γίνεται η ίδια μετατόπιση κατά 500 παλμούς προς την αντίθετη κατεύθυνση.

Κεφάλαιο

6

6. Συμπεράσματα

Η διαδικασία του προγραμματισμού και η επικοινωνία μεταξύ των ελεγκτών και matlab πραγματοποιήθηκε σε αρκετά ικανοποιητικό βαθμό. Στο πειραματικό μέρος της εργασίας συνδέθηκαν όλα τα μέρη του συστήματος και ελέγχθηκε η λειτουργία του. Η ακρίβεια του συστήματος εξαρτάται σε μεγάλο βαθμό από τα μηχανικά μέρη όπως οι ιμάντες και οι τροχαλίες. Έχει παρατηρηθεί ότι στα μοτέρ, λόγω ότι έχουν εσωτερικό μειωτήρα, δημιουργούν κάποιο τζόγο στην τελική κίνηση των συνδέσμων. Αυτό αναμένεται να αυξηθεί από τη συνεχή λειτουργία του και την κόπωση των μοτέρ.

Η αρχικοποίηση του βραχίονα για τις άρθρωσης Θ_2 και Θ_3 έγινε ικανοποιητικά, για την άρθρωση Θ_1 δεν τοποθετήθηκε το proximity καθώς πρέπει να τοποθετηθεί στο κάτω μέρος του βραχίονα που θα βρίσκεται το τροχήλατο όχημα. Η αρχικοποίηση της γωνίας αυτής γίνεται στο σημείο που θα βρίσκεται όταν δοθεί τάση στους ελεγκτές.

Ο χρόνος του κύκλου της κάθε κίνησης μπορεί να μειωθεί με διορθώσεις στον κώδικα των ελεγκτών αν αυτό κριθεί απαραίτητο και με αύξηση υπολογιστικής ισχύος από τη μεριά του υπολογιστή που περιέχει το Matlab. Το τελευταίο κρίνεται περισσότερο αναγκαίο εφόσον σε αυτόν προστεθεί και η πλοήγηση του οχήματος με τεχνική όραση.

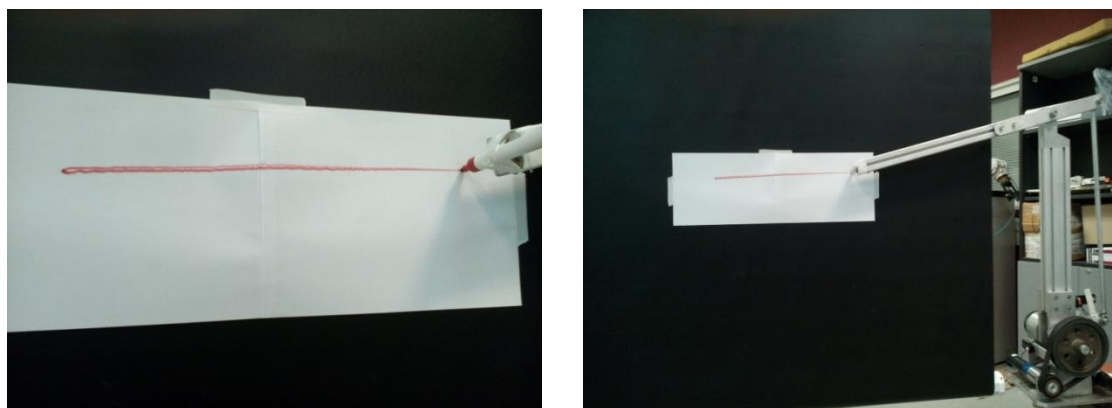
Στο πειραματικό μέρος πραγματοποιήθηκαν κινήσεις από σημείο σε σημείο με αρκετά καλό βαθμό ακρίβειας. Τοποθετήθηκε μαρκαδόρος στο τέλος του τρίτου συνδέσμου και δόθηκε εντολή να κινηθεί σε νέα επιθυμητή θέση του βραχίονα 10 εκατοστά χαμηλότερα από την θέση που βρίσκετε κατά μήκος του άξονα Z.



Εικόνα 73 Μετακίνηση 10cm ως προς τον άξονα Z'Z

Πραγματοποιήθηκε επίσης κίνηση επί ευθυγράμμου τμήματος κατά μήκος του άξονα Y. Τα αποτελέσματα δεν ήταν πολύ ικανοποιητικά καθώς υπάρχει μεν ευθεία μετατόπιση αλλά υπάρχει επίσης μια ελαφριά μετατόπιση και κατά άξονα Z. αυτό οφείλετε κυρίως στην αρχικοποίηση καθώς για να επιτευχθεί τέλεια παράλληλη ως προς τον άξονα Y κίνηση, πρέπει οι σύνδεσμοι να βρίσκονται εντελώς κάθετα μεταξύ τους και ο δεύτερος σύνδεσμος να είναι παράλληλος με το έδαφος. Αυτά μπορούν να διορθωθούν τοποθετώντας τον βραχίονα στην πραγματική βάση του που είναι το τροχήλατο όχημα.

Όσο αναφορά την επαναληψιμότητα, τόσο σε κίνηση επί ευθύγραμμου τμήματος όσο και στην κίνηση από σημείο σε σημείο δεν παρατηρήθηκαν αποκλίσεις από την προηγούμενη θέση. Στην περίπτωση όμως που θα γίνει μετατόπιση και ο πρώτος σύνδεσμος περάσει από την κάθετη προς το έδαφος θέση τότε λόγω του τζόγου του μοτέρ θα υπάρχει απόκλιση στο άκρο του βραχίονα της τάξεως 1-2 εκατοστά.



Εικόνα 74 Κίνηση επί ευθύγραμμου τμήματος κατά τον άξονα Y

Βιβλιογραφικές αναφορές

Σημειώσεις και διδακτικά συγγράμματα

- [1] John J. Craig, *Έισαγωγή στη Ρομποτική, Μηχανική & Έλεγχος*, εκδόσεις Τζιώλα
- [2] Mark W. Spong, M. Vidyasagar, John Wiley & Sons *Robot Dynamics and Control*,
- [3] Δρακούλης Βασίλειος, Πτυχιακή εργασία *Ρομποτικός βραχίονας τριών βαθμών ελευθερίας ΤΕΙ Κρήτης*
- [4] Πολυγερινός Π. Πτυχιακή εργασία *Σχεδιασμός ρομποτικής πλατφόρμας κατάλληλης για εργασίες θερμοκηπίου ΤΕΙ Κρήτης*

Άρθρα

- [1] Prof Simon Blackmore, *Farming with robots 2050* ,Harper Adams University
- [2] Ziegler, J.G and Nichols, N. B. (1942). *"Optimum settings for automatic controllers"*

Ιστοσελίδες

- [1] manual της Roboteq, ανακτήθηκε 10 Ιανουαρίου 2015 από <http://www.roboteq.com/index.php/docman/motor-controllers-documents-and-files/documentation/user-manual/7-nextgen-controllers-user-manual/file>
- [2] manual της Roboteq microbasicscript, ανακτήθηκε 10 Ιανουαρίου 2015 από <http://www.roboteq.com/index.php/support/microbasic-reference>
- [3] <http://www.sweeper-robot.eu/>
- [4] <http://www.aginnovationontario.ca/en/robots-could-soon-be-harvesting-ontario-greenhouse-veggies/>
- [5] <http://www.crops-robots.eu/>
- [6] <http://www.greenhousecanada.com/>
- [7] <http://www.ffrobotics.com/>
- [8] <http://cdn.intechopen.com/pdfs-wm/379.pdf>
- [9] <https://deepblue.lib.umich.edu/bitstream>
- [10] <http://www1.cs.columbia.edu/~allen/F15/NOTES/invkin.pdf>
- [11] <http://product.item24.gr/>

[12] http://www.ie.itcr.ac.cr/einteriano/control/clase/Zomorodi_Shahrokhi_PID_Tunning_Comparison.pdf

[13] https://www.wageningenur.nl/upload_mm/5/5/7/c221711e-98da-4865-805a-8fc8531aa624_flyer_cucumber%20harvesting_robot_uk.pdf