ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΑΤΡΩΝ - ΠΟΛΥΤΕΧΝΙΚΗ ΣΧΟΛΗ
ΤΜΗΜΑ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ
ΚΑΙ ΤΕΧΝΟΛΟΓΙΑΣ ΥΠΟΛΟΓΙΣΤΩΝ



ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΑΤΡΩΝ
UNIVERSITY OF PATRAS

ΤΟΜΕΑΣ: Συστηματα Αυτοματου Ελεγχου
ΕΡΓΑΣΤΗΡΙΟ: Εργαστηριο Αυτοματισμου και Ρομποτικης

Διπλωματικη Εργασια

του φοιτητή του Τμήματος Ηλεκτρολόγων Μηχανικών και Τεχνολογίας Υπολογιστών της Πολυτεχνικής Σχολής του Πανεπιστημίου Πατρών

Αργυριος Ματσουκας

ΑΡΙΘΜΟΣ ΜΗΤΡΩΟΥ: 227839

Θέμα

Κατασκευή Delta μικρο-Ρομποτικού Βραχίονα

Επιβλέπων

Καθηγητής Κούσουλας Νικόλαος

Αριθμός Διπλωματικής Εργασίας:

Πάτρα, Μάιος 2017

ΠΙΣΤΟΠΟΙΗΣΗ

Πιστοποιείται ότι η διπλωματική εργασία με θέμα

Κατασκευή Delta μικρο-Ρομποτικού Βραχίονα

του φοιτητή του Τμήματος Ηλεκτρολόγων Μηχανικών και Τεχνολογίας Υπολογιστών

Αργύριου Ματσούκα

(Α.Μ.: 227839)

παρουσιάτηκε δημόσια και εξετάστηκε στο τμήμα Ηλεκτρολόγων Μηχανικών και Τεχνολογίας Υπολογιστών στις

__/__/___

Ο Επιβλέπων

Κούσουλας Νικόλαος
Καθηγητής

Ο Διευθυντής του Τομέα

Κούσουλας Νικόλαος
Καθηγητής

Στοιχεία διπλωματικής εργασίας

Θέμα: Κατασκευή Delta μικρο-Ρομποτικού Βραχίονα

Φοιτητής: Αργύριος Ματσούκας

Εργαστήρια
Εργαστήριο Αυτοματισμού και Ρομποτικής

Περίοδος εκπόνησης της εργασίας:
Μάιος 2015 - Ιούνιος 2017

Η εργασία αυτή γράφτηκε στο XeLaTeX και χρησιμοποιήθηκε η γραμματοσειρά GFS Didot του Greek Font Society.

Περίληψη

Η διπλωματική εργασία σχετίζεται με τον σχεδιασμό και κατασκευή ενός παράλληλου μικρο-ρομπότ με τέσσερις βαθμούς ελευθερίας και βάρος μικρότερο από 0,5Kgr. Η δομή του ρομπότ είναι όμοια με αυτή ενός τριών βαθμών ελευθερίας Delta-robot με ένα επιπλέον περιστροφικό βαθμό στη βάση του. Το ρομπότ σχεδιάστηκε και κατασκευάστηκε με μεθόδους ταχείας προτυποποίησης. Ο χρήστης προσδιορίζει την επιθυμητή θέση της κινούμενης βάσης και την περιστροφή της γύρω από τον z-άξονα και μέσω αντίστροφης κινηματικής υπολογίζονται οι γωνίες περιστροφής των σερβοκινητήρων. Σχεδιάστηκε και υλοποιήθηκε ένας απλός ελεγκτής οπτικής ανατροφοδότησης όπου λαμβάνει μετρήσεις προσανατολισμού και θέσης από μία κάμερα με την χρήση fiducial markers επικολλημένων στη κινούμενη βάση του ρομπότ. Το σύστημα ελέγχου υλοποιήθηκε σε περιβάλλον Robot Operating System (ROS) και έγινε χρήση λογισμικού OpenCV και ARuCo.

Abstract

The diploma thesis is concerned with the design and implementation of one parallel microrobot with four degrees of freedom (DoF) and weight less than 0.5 Kgr. The robot's structure imitates that of a three DoF Delta-robot with an additional rotational degree at its stationary base. The robot was designed and manufactured using 3D-rapid prototyping techniques. Given the user's desired position and orientation of the moving base, the rotation angles of the utilized servomotors are computed. A simple controller was implemented relying on visual servoing techniques and feedback from fiducial marker attached at the robot's moving base. The control system relies on the Robot Operating System and the OpenCV and ARuCo libraries.

Ευχαριστίες

Πολλες φορές εργασίες σαν κι αυτή εκτιμώνται σαν ατομική δουλεία, όμως είναι αποτέλεσμα συνεισφοράς πολλών ανθρώπων πέρα από τον ίδιο το συγγραφέα. Σε αυτό το πλαίσιο, αρχικά, θα ήθελα να ευχαριστήσω τον καθηγητή κ. Αντώνη Τζε για το ιδιαίτερα ενδιαφέρον θέμα που μου πρότεινε να ασχοληθώ και για την πολύτιμη βοήθειά του. Ωστόσο, χωρίς τις συμβουλές του διδακτορικού Ευαγγελίου Νικόλαου, καθώς και των συναδέλφων του εργαστηρίου δε θα είχε φτάσει σε πέρας η διπλωματική εργασία αυτή. Τέλος, άξιοι ευχαριστιών χρήζουν και άνθρωποι που δε συνεισέφεραν γνωστικά στην εκπόνηση της εργασίας αυτής, όπως η οικογένεια και οι φίλοι μου.

ΠΕΡΙΕΧΟΜΕΝΑ

Κατάλογος σχημάτων xiii

Κατάλογος πινάκων xvii

1 Θεωρητική Ανάλυση 7
1.1 Κινηματική Ανάλυση 7
1.1.1 Ευθεία Κινηματική 7
1.1.2 Αντίστροφη Κινηματική 12
1.2 Χώρος Εργασίας - Workspace 15
1.3 Σύνοψη Κεφαλαίου 18

2 Λογισμικά Εργασίας 19
2.1 SolidWorks 19
2.2 Robot Operating System (ROS) 25
2.3 Υπόλοιπα Λογισμικά 27
2.3.1 OpenCV 27
2.3.2 Έλεγχος και ArUco 29
2.3.3 Πακέτα σχετικά με Phidgets 31
2.4 Πακέτα ROS 31
2.5 Σύνοψη Κεφαλαίου 32

3 Πειραματική Διάταξη 33
3.1 Μηχανολογικά Μέρη 33
3.1.1 Κατασκευή και 3D εκτύπωση 33
3.1.2 Μηχανουργείο 38
3.2 Κινητήρες 39
3.3 Phidgets Device 41

3.4 Camera 42
3.5 Εντοπισμός ArUco 46
3.6 Σχεδιασμός και Υλοποίηση Ελέγχου 49
3.7 Πείραμα 53
3.7.1 Γραμμική Τροχιά 53
3.7.2 Κυκλική Τροχιά 56
3.8 Σύνοψη Κεφαλαίου 58

4 Συμπεράσματα 59
4.1 Σύνοψη 59
4.2 Προτεινόμενες αλλαγές 59
4.3 Μελλοντική Έρευνα 60

Α΄ Λογισμικά Εργασίας 61

Βιβλιογραφία 63

ΚΑΤΑΛΟΓΟΣ ΣΧΗΜΑΤΩΝ

1 Stewart Platform 2
2 Εξάποδο Ρομπότ - Προσομοιωτής Πτήσης (Lufthansa) . . . 2
3 Εξάποδο Ρομπότ - Βοηθός Χειρούργου 3
4 Reymond Clavel (αριστερά) - Εφευρέτης του Delta Robot. Στο σχήμα διακρίνεται και η κατασκευή του 4
5 Delta Robot της ABB 4
6 Novint Falcon Haptic Device [7] 5
7 Συνολική Παρουσίαση της Εργασίας 5

1.1 Ορθή Κινηματική 8
1.2 Αναπαράσταση τριών σφαιρών 8
1.3 Ορισμός Παραμέτρων 9
1.4 Κάτοψη Ρομπότ για Μαθηματική Ανάλυση 10
1.5 Μετασχηματισμός Denavit Hartenberg 11
1.6 Αντίστροφη Κινηματική 13
1.7 Μαθηματική Ανάλυση Αντίστροφης Κινηματικής 13
1.8 Εύρεση υπόλοιπων γωνιών μέσω συμμετρίας 14
1.9 Επιφάνεια Χώρου Εργασίας του Delta Robot του Clavel [11] 16
1.10 Πλάγια όψη Χώρου Εργασίας 16
1.11 Κάτοψη Χώρου Εργασίας 16
1.12 Τρισδιάστατη Απεικόνιση Χώρου Εργασίας 17
1.13 Δισδιάστατη Απεικόνιση Χώρου Εργασίας 17

2.1 3D Απεικόνιση Άνω Βάσης 20
2.2 Πλάγια Όψη Άνω Βάσης 20
2.3 Κάτοψη Άνω Βάσης 21
2.4 Κάτοψη Κύριας Βάσης 21

2.5 3D Απεικόνιση Κύριας Βάσης 21
2.6 3D Απεικόνιση της Κινούμενη Πλατφόρμας 22
2.7 Κάτοψη Κινούμενης Πλατφόρμας 22
2.8 Πλάγια Όψη Βραχίονα 22
2.9 3D Απεικόνιση Βραχίονα 23
2.10 Σύνδεσμος 1 23
2.11 Σύνδεσμος 2 23
2.12 Πλάγια Όψη Κάτω Βραχίονα 24
2.13 3D Απεικόνιση Κάτω Βραχίονα 24
2.14 Σύνδεση Άνω Βραχίονα - Κινητήρα 24
2.15 Σύνδεση Περιστροφικού Κινητήρα - Κύριας Βάσης 25
2.16 Σύνδεση Κινούμενη Πλατφόρμας - Κάτω Βραχίονα 25
2.17 3D Απεικόνιση του Ρομπότ 26
2.18 ArUco Marker 30

3.1 Εκτυπωτής Replicator 2x 34
3.2 Άνω Βάση 35
3.3 Κάτοψη κύριας Βάσης 35
3.4 Άνοψη κύριας Βάσης 36
3.5 Κινούμενη Πλατφόρμα - End Effector 36
3.6 Βραχίονας 37
3.7 Κάτω Βραχίονας σε ουδέτερη και υπό γωνία θέση 37
3.8 Κοντινή Λήψη Κάτω Βραχίονα 38
3.9 PWM Διαμόρφωση των Servo 39
3.10 HS-422 Deluxe Standard Servo της Hitec 40
3.11 Πλάγια όψη Phidgets 41
3.12 Κάτοψη Phidgets 41
3.13 Camera 43
3.14 Μηχανισμός Στήριξης Κάμερας 43
3.15 Σειρά Κινήσεων Σκακιέρας 45
3.16 Λήψη από Κάμερα και Εικόνα (Μετρήσεις) από το Terminal των Ubuntu 46
3.17 Λήψη από Κάμερα και Εικόνα (Μετρήσεις) από το Terminal των Ubuntu 47
3.18 Μέτρηση Από Κάμερα 48
3.19 P Ελεγκτής 49
3.20 Επίδραση Όρου Kp 50
3.21 3D Απεικόνιση Τροχιάς Ελέγχου 52
3.22 Δισδιάστατη Απεικόνιση Τροχιάς Ελέγχου 52
3.23 3D Απεικόνιση Γραμμικής Τροχιάς $\theta_4 = 0°$ 54
3.24 2D Απεικόνιση Γραμμικής Τροχιάς $\theta_4 = 0°$ 54
3.25 3D Απεικόνιση Γραμμικής Τροχιάς $\theta_4 \neq$ σταθ 55
3.26 2D Απεικόνιση Γραμμικής Τροχιάς $\theta_4 \neq$ σταθ 55
3.27 3D Απεικόνιση Κυκλικής Τροχιάς $\theta_4 = 0°$ 56
3.28 2D Απεικόνιση Κυκλικής Τροχιάς $\theta_4 = 0°$ 56

3.29 3D Απεικόνιση Κυκλικής Τροχιάς $\theta_4 \neq$ σταθ 57
3.30 2D Απεικόνιση Κυκλικής Τροχιάς $\theta_4 \neq$ σταθ 58

ΚΑΤΑΛΟΓΟΣ ΠΙΝΑΚΩΝ

3.1 Printing Specifications του εκτυπωτή 34
3.2 Perfomance Specifications 40
3.3 Physical Specifications 40
3.4 Servo Controller [15] 42
3.5 Electrical Properties [15] 42
3.6 Physical Properties [15] 42
3.7 Σφάλματα Τροχιάς 51
3.8 Σφάλματα Γραμμικής Τροχιάς 53
3.9 Σφάλματα Κυκλικής Τροχιάς 57

ΕΙΣΑΓΩΓΗ

Η παρούσα εργασία βρίσκεται στην τομή δύο επιστημονικών πεδίων της ρομποτικής, των παράλληλων ρομπότ και της υπολογιστικής όρασης. Πρώτα, γίνεται μία προσπάθεια να δοθεί ένας ορισμός και μία σύντομη ιστορική αναδρομή των παράλληλων ρομπότ.

Διάφορα προβλήματα των σειριακών ρομποτικών συστημάτων οδήγησαν στην εφεύρεση των παράλληλων ρομπότ. Τα παράλληλα ρομπότ αποτελούνται από κλειστές κινηματικές αλυσίδες κατά τις οποίες οι αρθρωτοί σύνδεσμοι ενώνονται παράλληλα συνήθως μεταξύ μιας σταθερής βάσης και μιας κινούμενης πλατφόρμας [1]. Οι επενεργητές πολλές φορές συνδέονται αμφότεροι στη βάση και στην κινούμενη πλατφόρμα. Η λέξη "παράλληλο (ρομπότ)" προέρχεται από την ταυτόχρονη (παράλληλη) λειτουργία των ανεξάρτητων επενεργητών.

Τα παράλληλα ρομπότ προσφέρουν πληθώρα πλεονεκτημάτων. Κατά πρώτον, κύριο χαρακτηριστικό τους είναι η στιβαρότητα. Επίσης, τα σφάλματα θέσης είναι μικρότερα διότι δε συσσωρεύονται όπως συμβαίνει με τα σειριακά ρομπότ, καθιστώντας τα παράλληλα πιο ακριβή. Ακόμα ένα πλεονέκτημα τους είναι ο καλός λόγος βάρους προς φορτίου. Αυτό συμβαίνει διότι λόγω κατασκευής το κέντρο βάρους βρίσκεται σε τέτοιο σημείο ώστε να μπορούν να κινήσουν μεγαλύτερο φορτίο. Τέλος, δύναται να κινούνται σε υψηλές ταχύτητες.

Υπάρχουν πολλές παραλλαγές στο μηχανολογικό σχεδιασμό τους, ο οποίος εξαρτάται άμεσα από τους βαθμούς ελευθερίας του παράλληλου ρομπότ. Ως αποτέλεσμα ένα παράλληλο ρομπότ με 6 βαθμούς ελευθερίας έχει πολλά διαφορετικά χαρακτηριστικά σε σχέση με ένα ρομπότ λιγότερων βαθμών. Στη συνέχεια, θα εξηγηθούν τα δύο βασικά είδη παράλληλων ρομποτικών μηχανισμών.

Εξάποδο Ρομπότ (Hexapod - Stewart Platform)

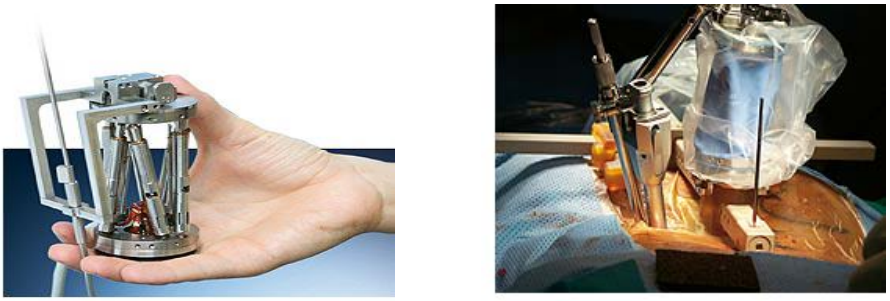
Το εξάποδο ρομπότ είναι ένα παράλληλο ρομπότ που έχει 6 βαθμούς ελευθερίας [2]. Χρησιμοποείται ευρέως σε εφαρμογές όπως προσομοιωτές πτήσης, υποθαλάσσια έρευνα, ορθοπεδική χειρουργική[4] κ.ά. Έχει την ικανότητα να επιτρέπει μεγάλου εύρους κινήσεις για μεγάλα φορτία με ακρίβεια και με δυνατότητα επαναληψιμότητας. Η κατασκευή του διακρίνεται στην εικόνα 1, όπου φαίνεται η συνδεσμολόγια κατά ζευγάρια των 6 επενεργητών. Στις εικόνες 2 και 3 δείχνονται μερικές από τις εφαρμογές τους.



Σχήμα 1: Stewart Platform



Σχήμα 2: Εξάποδο Ρομπότ - Προσομοιωτής Πτήσης (Lufthansa)



Σχήμα 3: Εξάποδο Ρομπότ - Βοηθός Χειρούργου

Ρομπότ Δέλτα (Delta Robot)

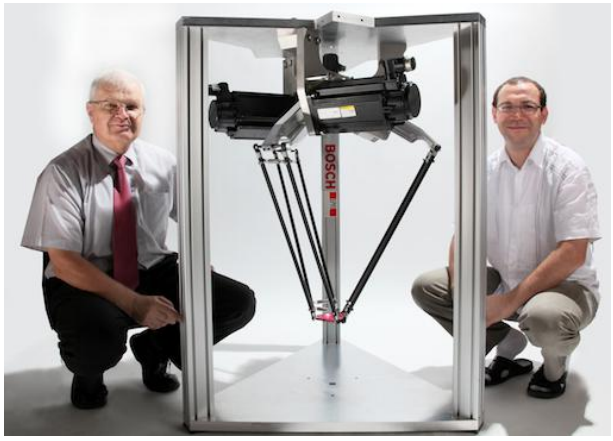
Αρχές της δεκαετίας του 1990 προτάθηκε η ιδέα ενός παράλληλου ρομποτικού μηχανισμού σε σχήμα δέλτα (Delta Robot[5]) προκειμένου να καλύψει βιομηχανικές ανάγκες γρήγορης μεταφοράς αντικειμένων σε χώρο τάξης λίγων εκατοστών (pick 'n' place operations). Το βασικό χαρακτηριστικό του σχεδιασμού του είναι η χρήση των παραλληλογράμμων στους βραχίονες, το οποίο διατηρεί τον προσανατολισμό της κινούμενη πλατφόρμας (end effector) [3].

Τα Delta Robots χρησιμοποιούνται κυρίως στη βιομηχανία για τη συσκευασία προϊόντων και σε ιατρικές εφαρμογές. Συνήθως ο σχεδιασμός του συγκεκριμένου ρομπότ είναι με την αρπάγη προς τα κάτω έτσι ώστε να έχει πρόσβαση στο χώρο εργασίας. Αποτελείται από 2 πλατφόρμες, μία μεγάλη και μια μικρότερη, πάντα σε παραλληλία μεταξύ τους και αυτές συνδεόνται με βραχίονες. Κάθε ένας τέτοιος βραχίονας κινείται από έναν επενεργητή (κινητήρα). Αυτό επιτρέπει στους βραχίονες να κινούνται με μεγάλη ευελιξία υπό την προϋπόθεση ότι κινούνται αρμονικά και συνδυαστικά μεταξύ τους. Χαρακτηριστικά παράδειγματα είναι το ρομπότ του εφευρέτη του Delta Robot Reymond Clavel στην εικόνα 4 και το Flexpicker της ABB, το οποίο διακρίνεται στην εικόνα 5.

Ακόμα ένα παράδειγμα της παραπάνω κατηγορίας ρομπότ είναι το Novint Falcon (εικόνα 6) το οποίο υπάρχει στο εργαστήριο όπου εκπονήθηκε η εργασία αυτή και βοήθησε στην κατασκευή της.

Στη συγκεκριμένη διπλωματική εργασία ζητήθηκε η κατασκευή ενός ρομπότ δέλτα (Delta Robot) και η προσθήκη ενός ακόμα βαθμού ελευθερίας που θα επέτρεπε κυρίως τον καθορισμό του προσανατολισμού του εργαλείου τελικής δράσης (end-effector). Εκτός αυτού, τέθηκε και ο περιορισμός που αφορούσε στο συνολικό βάρος της κατασκευής. Ο περιορισμός αυτός θα αφαιρούσε σημαντικό κομμάτι του χώρου εργασίας, όμως θα μας έδινε τη δυνατότητα μιας πιο εμπορικής χρήσης, όπως η ενσωμάτωσή του σε κάποιο εναέριο ρομποτικό όχημα (πχ hexacopter).

Επίσης, προκειμένου να επιβεβαιωθεί η θέση και ο προσανατολισμός του end-effector τα οποία προκύπτουν από την κινηματική ανά-



Σχήμα 4: Reymond Clavel (αριστερά) - Εφευρέτης του Delta Robot. Στο σχήμα διακρίνεται και η κατασκευή του



Σχήμα 5: Delta Robot της ABB

λυση, προσκολλήθηκε πάνω του ένας στόχος Επαυξημένης Πραγματικότητας (ArUco Marker) και μία κάμερα για τον εντοπισμό του.

Στις ενότητες που ακολουθούν, αναλύεται σε θεωρητικό επίπεδο η εργασία αυτή και έπειτα εξηγείται η διάταξη, ο έλεγχος που εφαρμόστηκε και τα περιβάλλοντα (λογισμικά) στα οποία πραγματοποιήθηκε.



Σχήμα 6: Novint Falcon Haptic Device [7]



Σχήμα 7: Συνολική Παρουσίαση της Εργασίας

ΚΕΦΑΛΑΙΟ

1

ΘΕΩΡΗΤΙΚΗ ΑΝΑΛΥΣΗ

Στόχος του παρόντος κεφαλαίου είναι να αναλυθεί κινηματικά ο συγκεκριμένος τύπος ρομπότ και να παρουσιαστούν οι παράμετροι σχεδίασης, εκ των οποίων μία από αυτές είναι ο χώρος εργασίας (workspace) και οι οποίες οδήγησαν στα τελικά χαρακτηριστικά του.

1.1 Κινηματική Ανάλυση

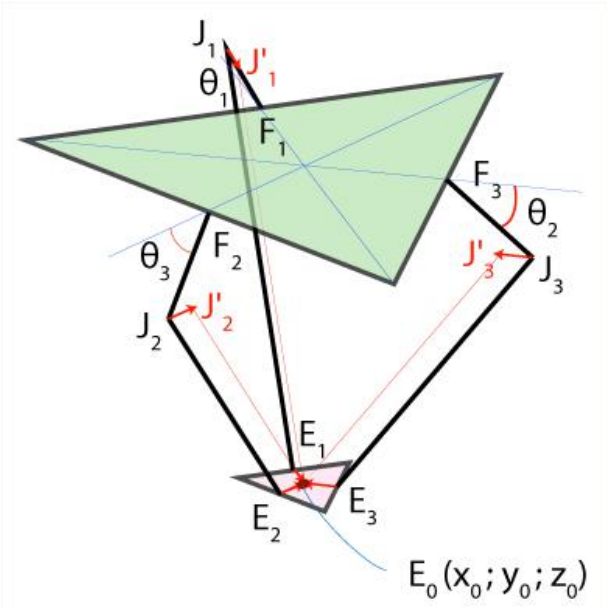
1.1.1 Ευθεία Κινηματική

Η κίνηση ενός ρομποτικού βραχίονα διέπεται από ορισμένες εξισώσεις που αποτελούν την ευθεία/ορθή και την αντίστροφη κινηματική του. Αυτές συνδέουν τις γωνίες των κινητήρων του ρομπότ με τη θέση του εργαλείου τελικής δράσης (end effector). Η αντίστροφη κινηματική υπολογίζει τις γωνίες των κινητήρων για δεδομένη θέση του end effector και η ορθή το αντίστροφο.

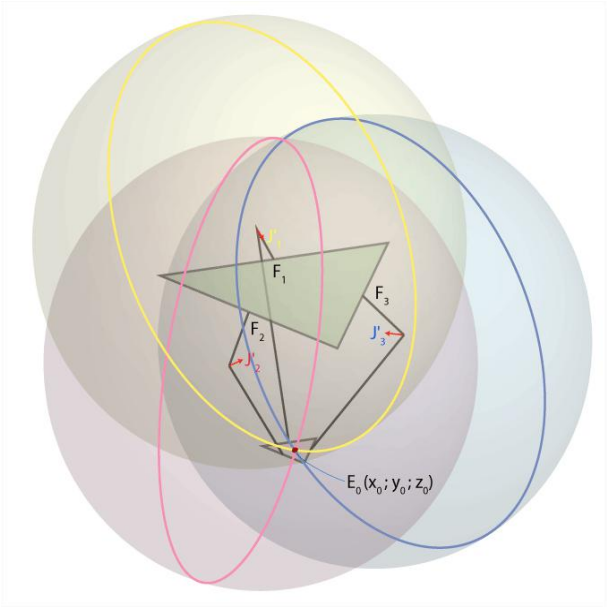
Θα ξεκινήσουμε την ανάλυση των κινηματικών εξισώσεων του Delta Robot τριών βαθμών ελευθερίας όπως προτάθηκε από τον R. Clavel το 1991 [5] (χωρίς τον περιστροφικό βαθμό) και θα καταλήξουμε σταδιακά στο παρόν ρομπότ.

Για την επίλυση της ευθείας κινηματικής η ιδέα είναι η εξής [6]. Θεωρούμε τρεις σφαίρες με κέντρα τα σημεία των αρθρώσεων J_1, J_2, J_3 μετακινημένα κατά μικρή απόσταση στις θέσεις J_1', J_2', J_3' και ακτίνα r_e , τα οποία φαίνονται στο σχήμα 1.1.

Οι τρεις αυτές σφαίρες παρουσιάζουν συνολικά δύο σημεία τομής



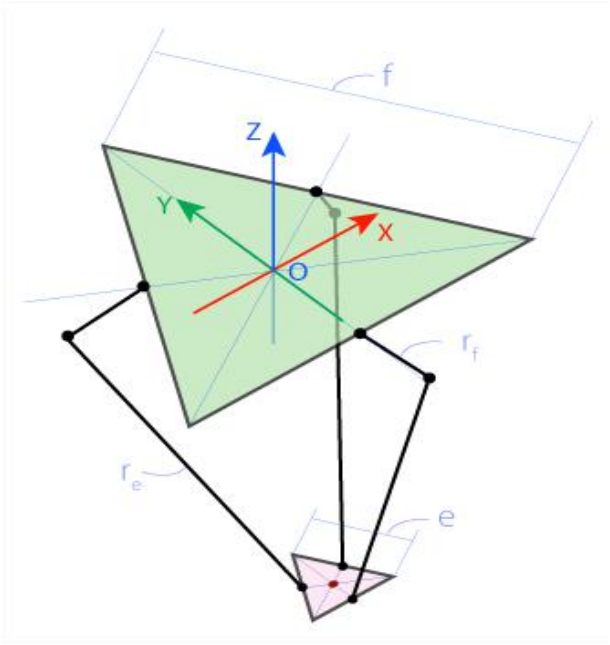
Σχήμα 1.1: Ορθή Κινηματική



Σχήμα 1.2: Αναπαράσταση τριών σφαιρών

(εικόνα 1.2), τα οποία θα υπολογισθούν αμέσως. Πρώτα όμως, χρειάζεται ο ορισμός ορισμένων παραμέτρων του ρομπότ (εικόνα 1.3). Ορίζουμε:

- f πλευρά νοητού τριγώνου βάσης
- e πλευρά νοητού τριγώνου end effector
- r_f μήκος άνω βραχίονα
- r_e μήκος κάτω βραχίονα



Σχήμα 1.3: Ορισμός Παραμέτρων

Επίσης, κάθε σφαίρα περιγράφεται από την εξίσωση

$$(x - x_i)^2 + (y - y_i)^2 + (z - z_i)^2 = r_e^2, \quad i = 1, 2, 3$$

όπου x_i, y_i, z_i οι συντεταγμένες των κέντρων τους, στην προκειμένη περίπτωση τα σημεία J_1', J_2', J_3'.

Για την εύρεση των τομών χρησιμοποιούμε την κάτοψη του Delta Robot (εικόνα 1.4) Οπότε έχουμε:

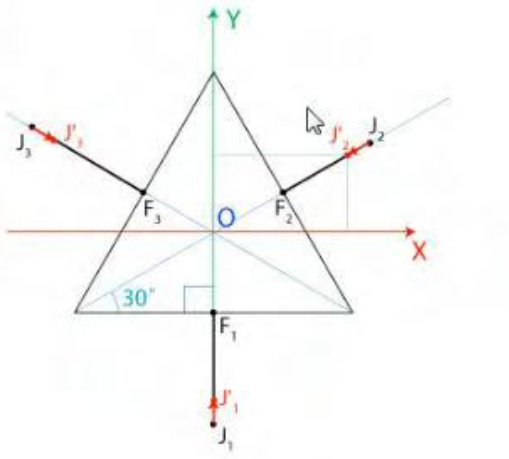
$$OF_1 = 0F_2 = OF_3 = \frac{f}{2}\tan 30° = \frac{f}{2\sqrt{3}}$$

$$J_1J_1' = J_2J_2' = J_3J_3' = \frac{e}{2}\tan 30° = \frac{e}{2\sqrt{3}}$$

$$J_1' \quad (0, -\frac{f-e}{2\sqrt{3}} - r_f\cos\theta_1, -r_f\sin\theta_1)$$

$$J_2' \quad ([\frac{f-e}{2\sqrt{3}} + r_f\cos\theta_2]\cos 30°, [\frac{f-e}{2\sqrt{3}} + r_f\cos\theta_2]\sin 30°, -r_f\sin\theta_2)$$

$$J_3' \quad ([\frac{f-e}{2\sqrt{3}} + r_f\cos\theta_3]\cos 30°, [\frac{f-e}{2\sqrt{3}} + r_f\cos\theta_3]\sin 30°, -r_f\sin\theta_3)$$



Σχήμα 1.4: Κάτοψη Ρομπότ για Μαθηματική Ανάλυση

Επομένως, για τις τρεις σφαίρες προκύπτουν οι παρακάτω εξισώσεις:

$$\left.\begin{aligned} &x^2 + (y - y_1)^2 + (z - z_1)^2 = r_e^2 \\ &(x - x_2)^2 + (y - y_2)^2 + (z - z_2)^2 = r_e^2 \\ &(x - x_3)^2 + (y - y_3)^2 + (z - z_3)^2 = r_e^2 \end{aligned}\right\} \Rightarrow$$

$$\Rightarrow \left\{ \begin{aligned} x^2 + y^2 + z^2 - 2y_1 y - 2z_1 z &= r_e^2 - y_1^2 - z_1^2 \quad (1) \\ x^2 + y^2 + z^2 - 2x_2 x - 2y_2 y - 2z_2 z &= r_e^2 - x_2^2 - y_2^2 - z_2^2 \quad (2) \\ x^2 + y^2 + z^2 - 2x_3 x - 2y_3 y - 2z_3 z &= r_e^2 - x_3^2 - y_3^2 - z_3^2 \quad (3) \end{aligned} \right.$$

Αφαιρούμε τις παραπάνω σχέσεις ανά δύο, θέτουμε $w_i^2 = x_i^2 + y_i^2 + z_i^2$ και προκύπτει:

$$x = a_1 z + b_1 \quad (4)$$

$$y = a_2 z + b_2 \quad (5)$$

όπου

$$a_1 = \frac{1}{d}[(z_2 - z_1)(y_3 - y_1) - (z_3 - z_1)(y_2 - y_1)]$$

$$a_2 = -\frac{1}{d}[(z_2 - z_1)x_3 - (z_3 - z_1)x_2]$$

$$b_1 = -\frac{1}{2d}[(w_2 - w_1)(y_3 - y_1) - (w_3 - w_1)(y_2 - y_1)]$$

$$b_2 = \frac{1}{2d}[(w_2 - w_1)x_3 - (w_3 - w_1)x_2]$$

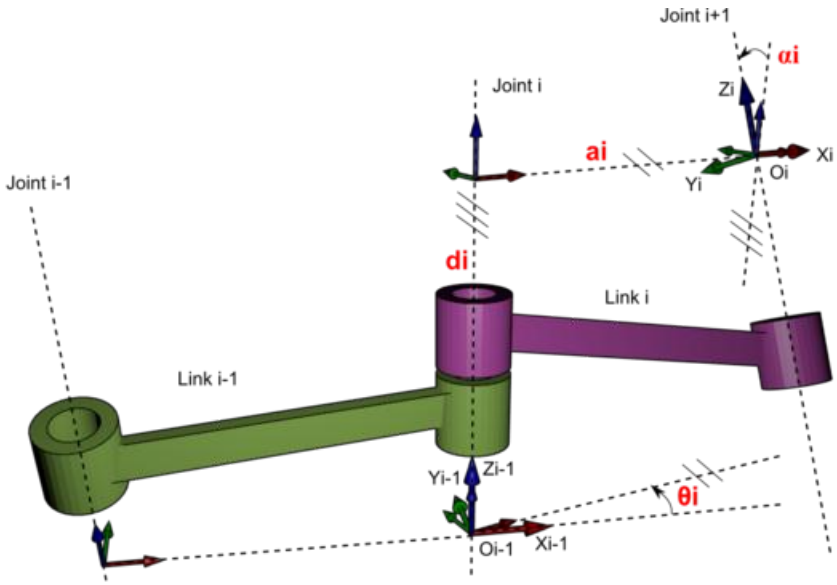
$$d = [(y_2 - y_1)x_3 - (y_3 - y_1)x_2]$$

Εν συνεχεία, αντικαθιστούμε στην (1) τα x,y με βάση τις σχέσεις (4), (5) και προκύπτει εξίσωση με μοναδικό άγνωστο το z, την οποία επιλύουμε και βρίσκουμε το z_e. Τέλος, από τις (4), (5) προκύπτουν τα x_e και y_e.

Οι παραπάνω συντεταγμένες, που αποτελούν το σημείο του εργαλείου τελικής δράσης, υπολογίσθηκαν ως προς το σύστημα συντεταγμένων της κύριας βάσης του Delta Robot. Όμως στην περίπτωσή μας, θέλουμε οι συντεταγμένες να υπολογίζονται με βάση το σημείο περιστροφής του πάνω κινητήρα. Για τον υπολογισμό αυτό θα εφαρμόσουμε μετασχηματισμό Denavit-Hartenberg (D-H).

Πριν εφαρμοστεί ο μετασχηματισμός αυτός, θα επιχειρήσουμε μία σύντομη περιγραφή του. Κάθε άρθρωση στη ρομποτική θεωρείται ότι έχει το δικό της σύστημα συντεταγμένων. Προκειμένου να συσχετίσουμε αυτά τα διαφορετικά συστήματα χρησιμοποιούμε το μετασχηματισμό D-H που αποτελείται από τέσσερις παραμέτρους [10]:

- d, απόσταση του O_{i-1} από το O_i κατά μήκος του άξονα z_{i-1}
- θ, η γωνία του άξονα x_{i-1} από τον x_i με στροφή γύρω από τον z_{i-1}
- a, το μήκος της κοινής καθέτου
- α, γωνία μεταξύ του z_{i-1} και του z_i, με στροφή γύρω από την κοινή κάθετο



Σχήμα 1.5: Μετασχηματισμός Denavit Hartenberg

Για δύο διαχοχικές αρθρώσεις ισχύει:

$$O_i = \begin{bmatrix} \cos\theta_n & -\sin\theta_n\cos\alpha_n & \sin\theta_n\sin\alpha_n & a_n\cos\theta_n \\ \sin\theta_n & \cos\theta_n\cos\alpha_n & -\cos\theta_n\sin\alpha_n & a_n\sin\theta_n \\ 0 & \sin\alpha_n & \cos\alpha_n & d_n \\ 0 & 0 & 0 & 1 \end{bmatrix} O_{i-1}$$

Στην περίπτωσή μας:

$d = 59.74(mm)$

$\theta_n = \theta_4$, δηλαδή η γωνία του σερβοκινητήρα

$\alpha = 0°$

$a = 0$

οπότε προκύπτει ο ομογενής μετασχηματισμός, αφού εκτελεστούν και οι όποιες πράξεις:

$$H_{01} = \begin{bmatrix} \cos\theta_4 & -\sin\theta_4 & 0 & 0 \\ \sin\theta_4 & \cos\theta_4 & 0 & 0 \\ 0 & 0 & 1 & 59.74 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

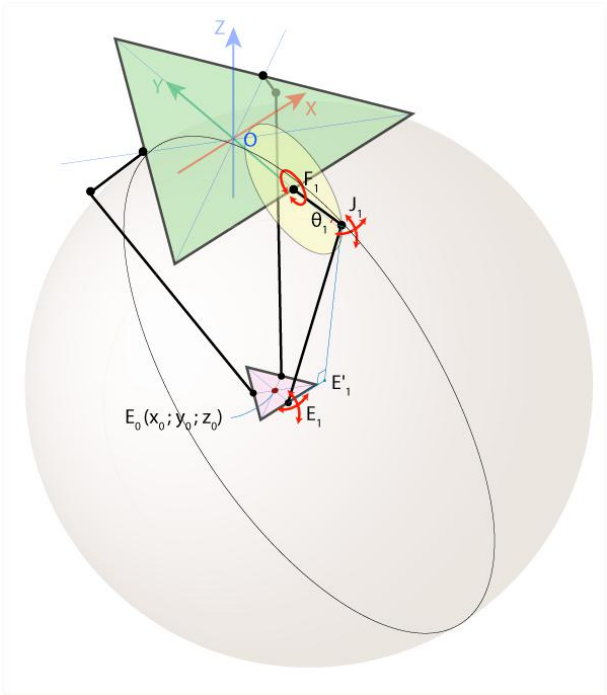
Οπότε, η θέση του end effector δίνεται τελικά από τη σχέση:

$$\begin{bmatrix} x'_e \\ y'_e \\ z'_e \\ 1 \end{bmatrix} = \begin{bmatrix} \cos\theta_4 & -\sin\theta_4 & 0 & 0 \\ \sin\theta_4 & \cos\theta_4 & 0 & 0 \\ 0 & 0 & 1 & 59.74 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_e \\ y_e \\ z_e \\ 1 \end{bmatrix}$$

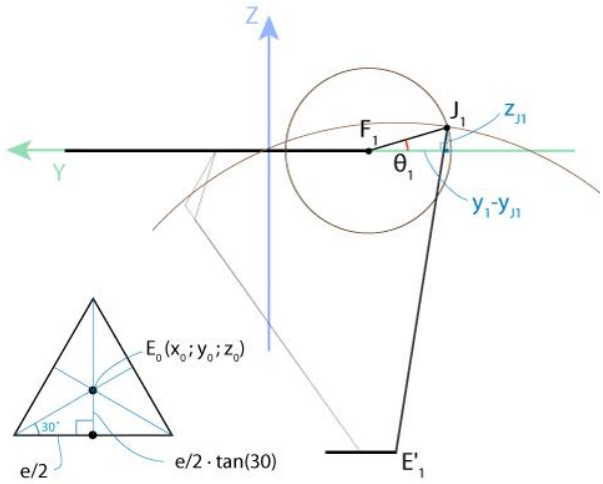
1.1.2 Αντίστροφη Κινηματική

Σε πρακτικές εφαρμογές, αυτό που μας ενδιαφέρει κυρίως είναι για δεδομένη θέση στην οποία θέλουμε να βρίσκεται ο end effector ποιες πρέπει να είναι οι γωνίες των κινητήρων. Το παραπάνω πρόβλημα αποτελεί την αντίστροφη κινηματική. Όπως και στην ευθεία κινηματική, θα εξάγουμε πρώτα το κινηματικό μοντέλο για το Delta Robot τριών βαθμών ελευθερίας και έπειτα το μοντέλο του ρομπότ της παρούσας εργασίας. Παρατηρούμε ότι ο σύνδεσμος F_1J_1 (εικόνα 1.6) δύναται να κινηθεί μόνο στο επίπεδο YZ και να σχηματίσει κύκλο με κέντρο το F_1 και ακτίνα r_f. Απεναντίας, ο σύνδεσμος J_1E_1 μπορεί να κινηθεί σε όλο το χώρο και έτσι με κέντρο το σημείο E_1 σχηματίζει σφαίρα με ακτίνα r_e.

Η τομή της σφαίρας αυτής και του XY επιπέδου είναι ένας κύκλος με κέντρο το E'_1 και ακτίνα E'_1J_1, όπου E'_1 η προβολή του σημείου E_1 στο XY επίπεδο. Τελικά, το σημείο J_1 υπολογίζεται ως η τομή των κύκλων με κέντρα το E'_1 και το F_1. Αφού γνωρίζουμε τις συντεταγμένες του J_1, εύκολα προκύπτει η γωνία θ_1, όπως θα εξηγηθεί μέσω της εικόνας 1.7.



Σχήμα 1.6: Αντίστροφη Κινηματική



Σχήμα 1.7: Μαθηματική Ανάλυση Αντίστροφης Κινηματικής

Γνωρίζουμε ότι $E_0E_1 = \frac{e}{2\sqrt{3}}$ και αφού $E_1 \equiv (x_0, y_0 - \frac{e}{2\sqrt{3}}, z_0)$ τότε $\Rightarrow E_1' \equiv (0, y_0 \frac{e}{2\sqrt{3}}, z_0)$
Προφανώς, $E_1E_1' = x_0$, άρα $E_1'J_1 = \sqrt{(E_1J_1^2 - E_1E_1'^2)} = \sqrt{(r_e^2 - x_0^2)} \Rightarrow (y_{J_1} - y_{E_1'})^2 + (z_{J_1} - z_{E_1'})^2 = r_e^2 - x_0^2 \quad (1)$

Επίσης, γνωρίζουμε και το σημείο $F_1 \equiv (0, -\frac{f}{2\sqrt{3}}, 0)$, οπότε $J_1F_1 = r_f \Rightarrow$

$$(y_{J_1} - y_{F_1})^2 + (z_{J_1} - z_{F_1})^2 = r_f^2 \quad (2)$$

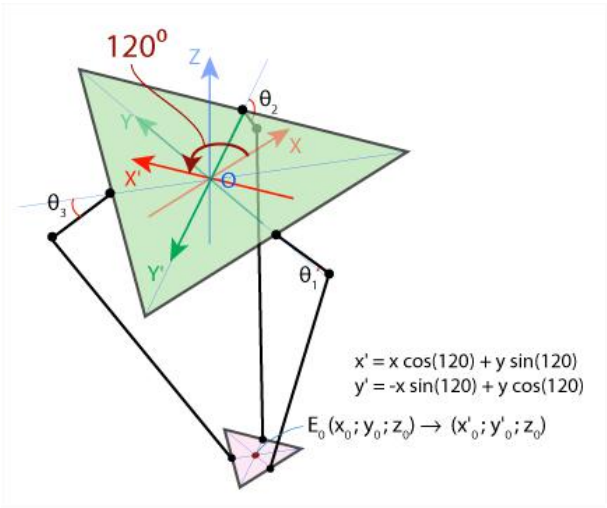
Από τις (1) και (2) αντικαθιστούμε τις γνωστές μεταβλητές:
$(y_{J_1} + \frac{f}{2\sqrt{3}})^2 + {z_{J_1}}^2 = r_f^2$
$(y_{J_1} - y_0 + \frac{e}{2\sqrt{3}})^2 + (z_{J-1} - z_0)^2 = r_e^2 - x_0^2$

Και από το παραπάνω σύστημα βρίσκουμε τις συντεταγμένες του σημείου $J_1(0, y_{J_1}, z_{J_1})$

Επομένως, προκύπτει και η γωνία $\theta_1 = \arctan\left[\dfrac{z_{J_1}}{y_{F_1} - y_{J_1}}\right]$

Η απλότητα των υπολογισμών έγκειται στη βολική επιλογή του συστήματος συντεταγμένων αναφοράς. Με αυτό τον τρόπο εξαλείφθηκε ο άξονας x.

Προκειμένου να υπολογίσουμε και τις εναπομείνουσες γωνίες θ_2, θ_3, χρησιμοποιούμε τη συμμετρία του ρομπότ. Αρκεί να στρέψουμε το σύστημα συντεταγμένων κατά 120° γύρω από τον άξονα Z (εικόνα 1.8).



Σχήμα 1.8: Εύρεση υπόλοιπων γωνιών μέσω συμμετρίας

Τέλος, αφού υπολογίσαμε την αντίστροφη κινηματική του Delta Robot, αρκεί να αντιστρέψουμε τον ομογενή μετασχηματισμό που υπολογίσαμε στην ευθεία κινηματική. Στη γενική του μορφή ο D-H μετασχηματισμός είναι:

$$\left[\begin{array}{ccc|c} & R & & T \\ \hline 0 & 0 & 0 & 1 \end{array}\right]$$

Προκύπτει ότι ο αντίστροφος μετασχηματισμός έχει τον τύπο [8]:

$$\left[\begin{array}{ccc|c} & R^T & & -R^T * T \\ \hline 0 & 0 & 0 & 1 \end{array}\right]$$

Στο σημείο, αξίζει να γίνει μία σημαντική παρατήρηση. Τα σειριακά ρομπότ από τη φύση της κατασκευής τους έχουν την εξής ιδιότητα. Η ευθεία κινηματική τους χαρακτηρίζεται από μία μοναδική λύση, σε αντίθεση με την αντίστροφη, η οποία επιδέχεται απειρία λύσεων.

Το ακριβώς αντίθετο διέπει τους παράλληλους ρομποτικούς μηχανισμούς. Ενώ η ευθεία κινηματική παρουσιάζει μία πληθώρα λύσεων (την οποία θα προσπελάσουμε με τη μέθοδο που παρουσιάστηκε στο κεφάλαιο 1.1.1), η αντίστροφη κινηματική έχει μία και μόνο λύση. Ωστόσο, όσον αφορά στην παρούσα εργασία όπου δεν είναι αντικείμενο ενδιαφέροντος ο προσανατολισμός του end effector, προσθέτοντας τον περιστροφικό βαθμό, αίρουμε αυτή τη μοναδικότητα της λύσης (στην αντίστροφη κινηματική). Για το λόγο αυτό, λοιπόν, σε οποιαδήποτε εφαρμογή της εργασίας αυτής, θεωρείται δεδομένος ο προσανατολισμός του end effector, ο οποίος εισάγεται από το χρήστη ως γωνία του 4ου κινητήρα και με τον τρόπο αυτό επαναφέρεται η μοναδικότητα της λύσης.

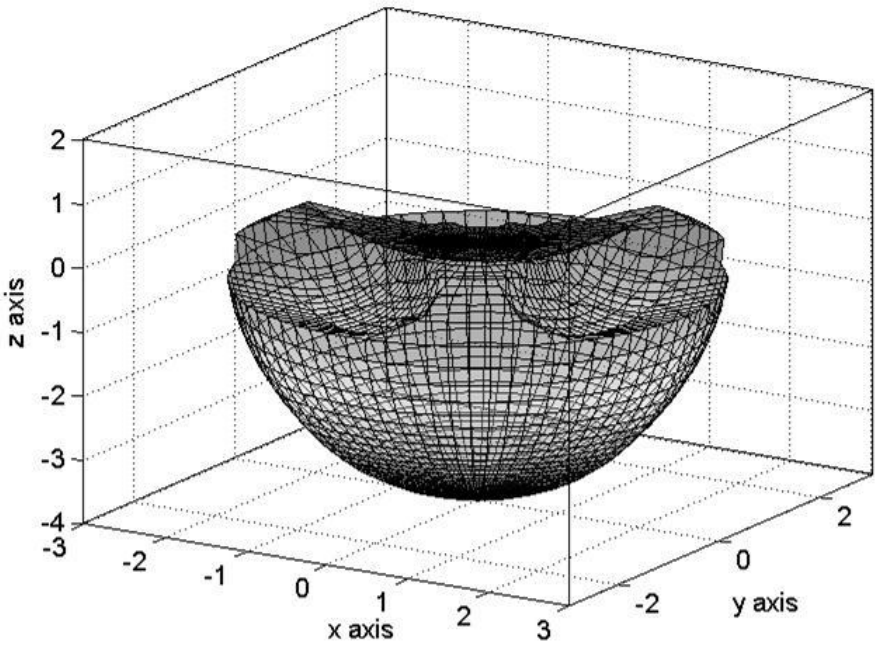
1.2 Χώρος Εργασίας - Workspace

Για τη διαστασιοποίηση του ρομπότ πριν την κατασκευή, όπως αναφέρθηκε και νωρίτερα, υπήρχε η ανάγκη απαίτησης δύο προδιαγραφών. Ειδικότερα, προείχε το συνολικό βάρος της κατασκευής να μη ξεπερνά τα 500gr και ο ελάχιστος χώρος εργασίας να κυμαίνοταν στα $100 - 120cm^3$.

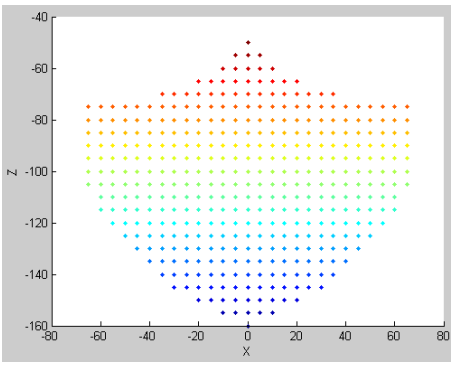
Για την εύρεση του χώρου εργασίας χρησιμοποιήθηκε η μέθοδος η οποία ακολουθήθηκε και στην κινηματική ανάλυση. Πρώτα, θα εξηγηθεί ο χώρος εργασίας του Delta Robot του Clavel, που έχει επιλυθεί αρκετές φορές [11] [12] και μέσω αυτού θα εξαχθεί για το ρομπότ της παρούσας εργασίας. Θυμίζεται εδώ ότι η εργασία αυτή περιλαμβάνει έναν έξτρα περιστροφικό βαθμό ελευθερίας, γεγονός που διαφοροποιεί το χώρο εργασίας.

Για το Delta Robot τριών βαθμών ελευθερίας, έχουν καταγραφεί διάφορες προσπάθειες επίλυσης. Όλες υποδεικνύουν τον παρακάτω χώρο εργασίας (εικόνες 1.9, 1.10, 1.11):

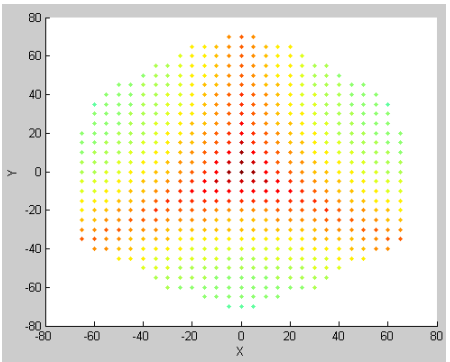
Παρατηρούμε ότι κατά την περιστροφή γύρω από τον κατακόρυφο άξονα z εμφανίζεται συμμετρία ανά 120°.



Σχήμα 1.9: Επιφάνεια Χώρου Εργασίας του Delta Robot του Clavel [11]



Σχήμα 1.10: Πλάγια όψη Χώρου Εργασίας

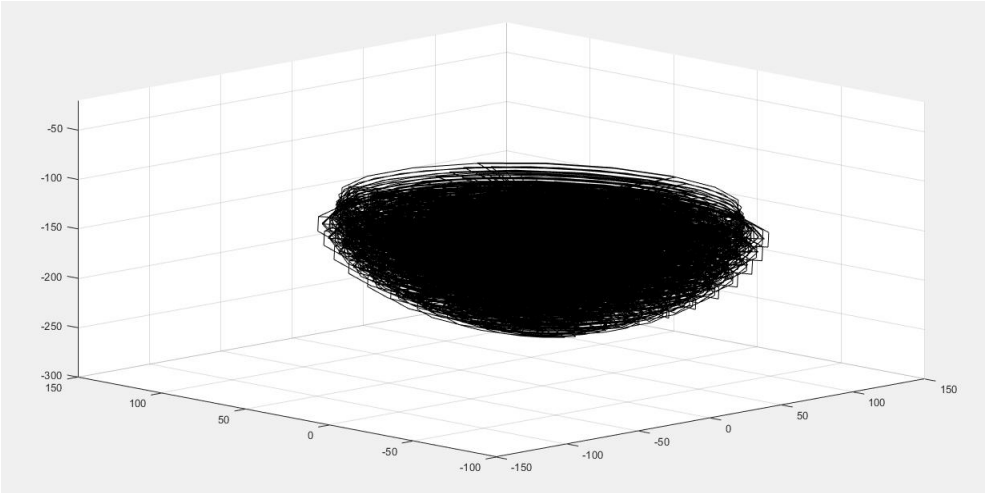


Σχήμα 1.11: Κάτοψη Χώρου Εργασίας

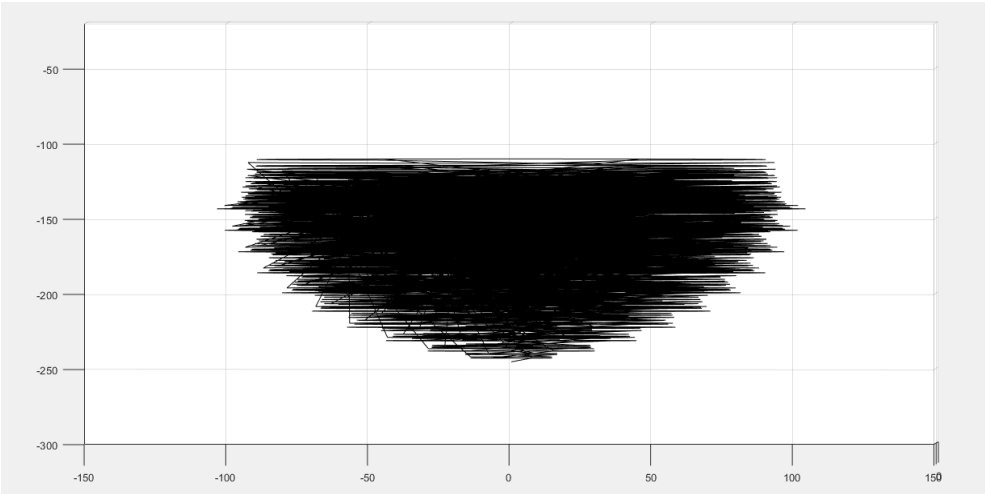
Δεδομένου του επιπλέον περιστροφικού βαθμού ελευθερίας, προκύπτει ότι ο χώρος εργασίας του ρομπότ θα έχει τη μορφή ωοειδούς σχήματος όπως φαίνεται στις εικόνες 1.12, 1.13.

Λόγω του τέταρτου κινητήρα έχουμε συμμετρία ως προς τον άξονα Z. Τα αποτελέσματα προέκυψαν από κώδικα γραμμένο σε MATLAB.

Το workspace των εικόνων 1.12, 1.13 προέκυψε δια της δοκιμής. Σε κάθε συνδυασμό διαστάσεων του ρομπότ, υπολογιζόταν το βάρος και ο χώρος εργασίας. Όποτε δεν ικανοποιούταν κάποια προδιαγραφή, δοκιμαζότανε και νέος συνδυασμός διαστάσεων.



Σχήμα 1.12: Τρισδιάστατη Απεικόνιση Χώρου Εργασίας



Σχήμα 1.13: Δισδιάστατη Απεικόνιση Χώρου Εργασίας

Αφού ολοκληρώθηκε η παραπάνω διαδικασία και αργότερα η κατασκευή και συναρμολόγηση του ρομπότ, καταστάθηκε αναγκαίο να βρεθεί εάν ο χώρος εργασίας στην πράξη ήτανε σύμφωνος με τη θεωρία. Επομένως, πραγματοποιήθηκε πείραμα όπου για κάθε εκατοστό κατά τον κατακόρυφο άξονα z στην κίνηση του end effector βρέθηκαν οι μέγιστες τιμές που μπορεί αυτός να κινηθεί. Το ελάχιστο, λοιπόν, workspace που προέκυψε είναι 10cm x 3.5cm x 3.5cm. Να επαναληφθεί στο σημείο αυτό ότι ο αρχικός στόχος ήταν λίγο πάνω από $100cm^3$ ο οποίος και επετεύχθη. Όσον αφορά στο βάρος της κατασκευής, πραγματοποιείται ανάλυση σε ύστερο κεφάλαιο.

1.3 Σύνοψη Κεφαλαίου

Στον παρόν κεφάλαιο πραγματοποιήθηκε η κινηματική ανάλυση του ρομπότ της εργασίας, αμφότερες η ευθεία και η αντίστροφη. Έπειτα, θέτοντας τις προδιαγραφές για το χώρο εργασίας και το βάρος, εξήχθησαν οι διαστάσεις που θα είχε το ρομπότ, προκειμένου σε επόμενο κεφάλαιο να σχεδιαστεί και να κατασκευαστεί.

ΚΕΦΑΛΑΙΟ

2

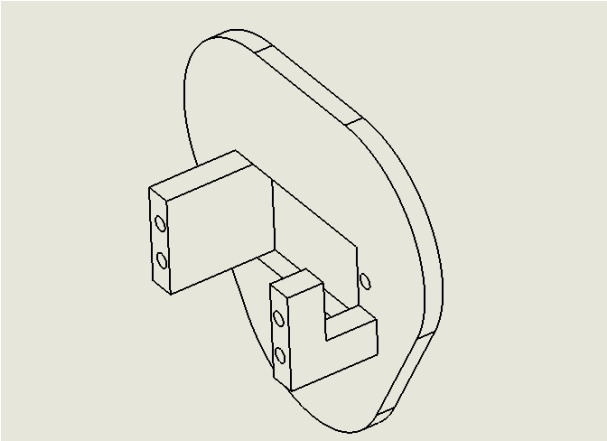
ΛΟΓΙΣΜΙΚΑ ΕΡΓΑΣΙΑΣ

Στόχος του παρόντος κεφαλαίου είναι να εξηγηθούν τα λογισμικά που χρησιμοποιήθηκαν για την εργασία αυτή. Δίδονται ορισμοί για τα περιβάλλοντα εργασίας ώστε να γίνει κατανοητό από τον αναγνώστη ο σκοπός και η λειτουργία που εξυπηρετούν.

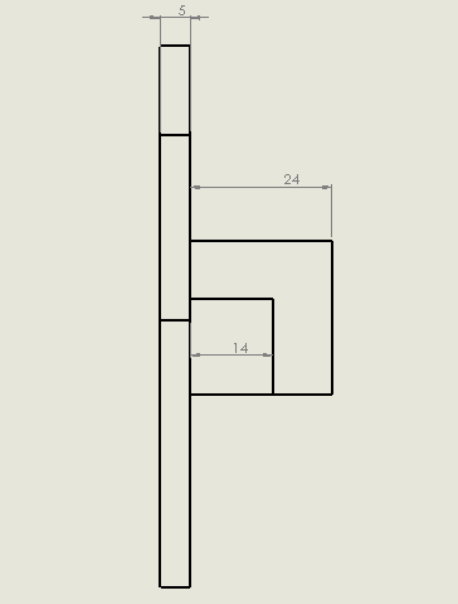
2.1 SolidWorks

Αρχικά, για τη σχεδίαση του Delta Robot χρησιμοποιήθηκε το λογισμικό SolidWorks. Εκτός από τη σχεδίαση χρησιμοποιήθηκε και για την προσομοίωση της κίνησής του. Πρόκειται για Λογισμικό Σχεδίασης Τριών Διαστάσεων μέσω Ηλεκτρονικού Υπολογιστή (3D CAD - Computer-Aided Design). Παρακάτω, παρατίθενται όλα τα σχέδια που χρησιμοποιήθηκαν για την κατασκευή. Όλα τα αναγραφόμενα μεγέθη είναι σε χιλιοστά (mm).

Στα πρώτα σχέδια δείχνεται η πάνω βάση (εικόνες 2.1, 2.2 και 2.3). Πρόκειται για το σημείο επαφής του ρομποτικού μηχανισμού με το σημείο στήριξής του. Να σημειωθεί ότι έχει σχεδιασθεί έτσι ώστε να στηρίζεται σε κατάλληλο πλαίσιο εργασίας (frame) που βρίσκεται στο εργαστήριο. Στην πάνω βάση τοποθετείται ένας περιστροφικός σερβοκινητήρας που ευθύνεται για τον προσανατολισμό και εν μέρει και για τη θέση του εργαλείου τελικής δράσης.

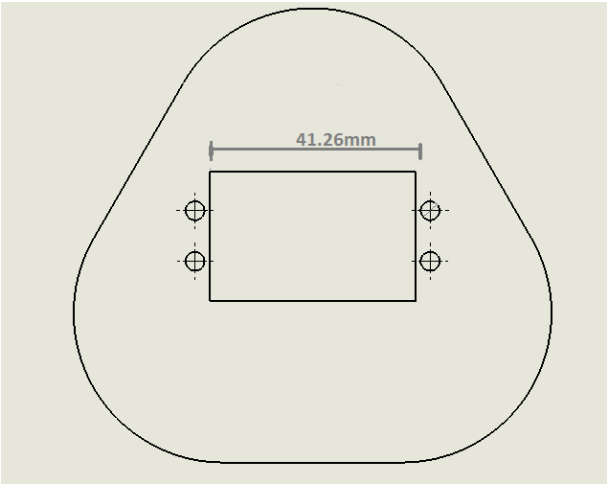


Σχήμα 2.1: 3D Απεικόνιση Άνω Βάσης

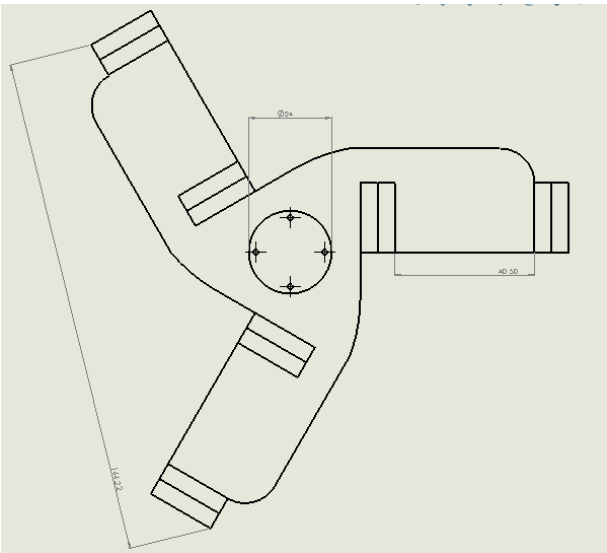


Σχήμα 2.2: Πλάγια Όψη Άνω Βάσης

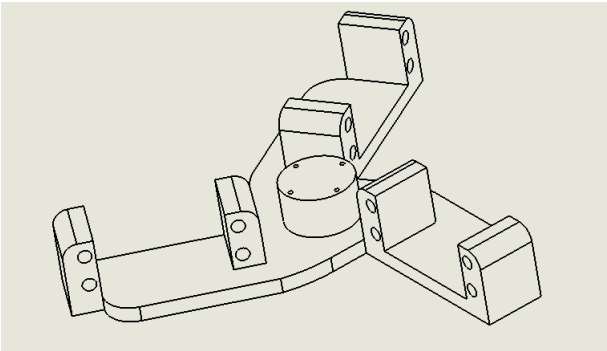
Στις εικόνες 2.4, 2.5 , δείχνεται η κύρια βάση. Αυτή στηρίζει τους τρεις κινητήρες και συνδέεται με τον 4ο περιστροφικό κινητήρα. Λόγω της τελευταίας σύνδεσης, αποκτά περιστροφική κίνηση.



Σχήμα 2.3: Κάτοψη Άνω Βάσης

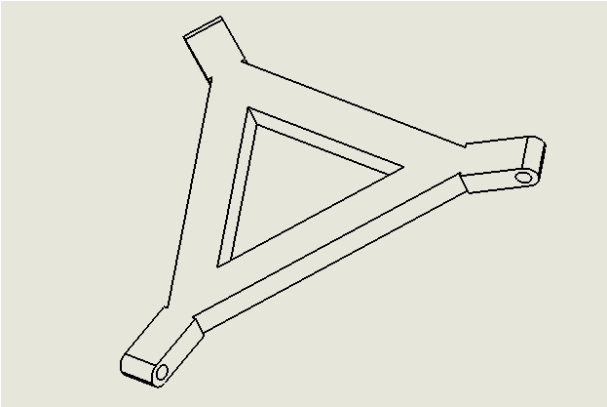


Σχήμα 2.4: Κάτοψη Κύριας Βάσης

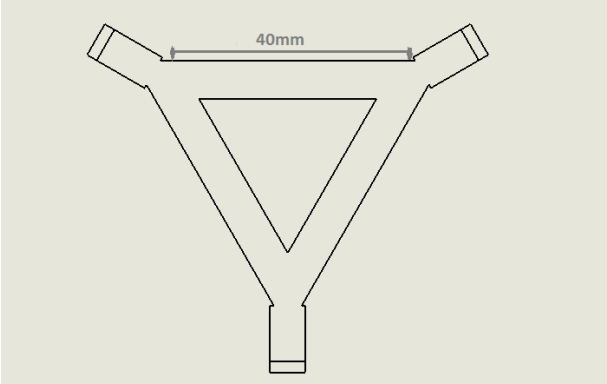


Σχήμα 2.5: 3D Απεικόνιση Κύριας Βάσης

Ως εργαλείο τελικής δράσης χρησιμοποιήθηκε μία απλοϊκή βάση τριγωνικού σχήματος (εικόνες 2.6, 2.7).

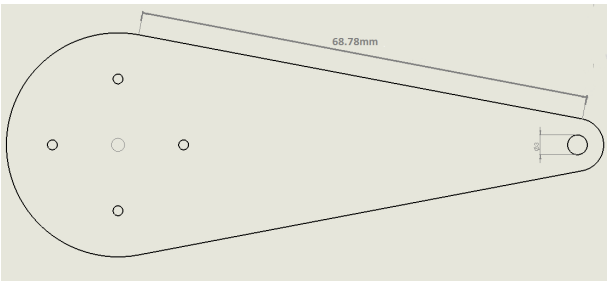


Σχήμα 2.6: 3D Απεικόνιση της Κινούμενη Πλατφόρμας

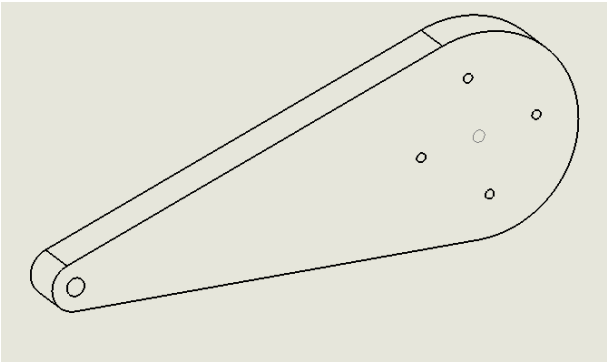


Σχήμα 2.7: Κάτοψη Κινούμενης Πλατφόρμας

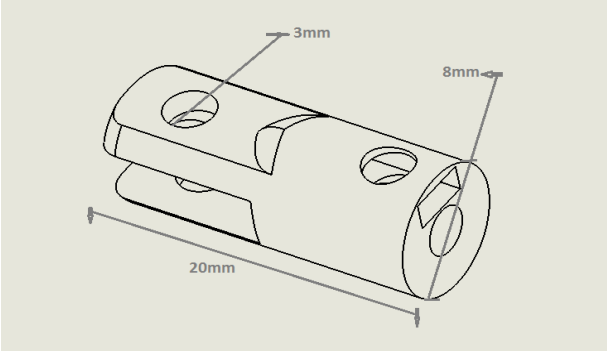
Για τη σύνδεση του end effector με την κύρια βάση αλλά και για τη δυνατότητα ελεύθερης κίνησης, χρησιμοποιήθηκαν τα εξαρτήματα που φαίνονται στις εικόνες 2.8, 2.9, 2.10, 2.11.



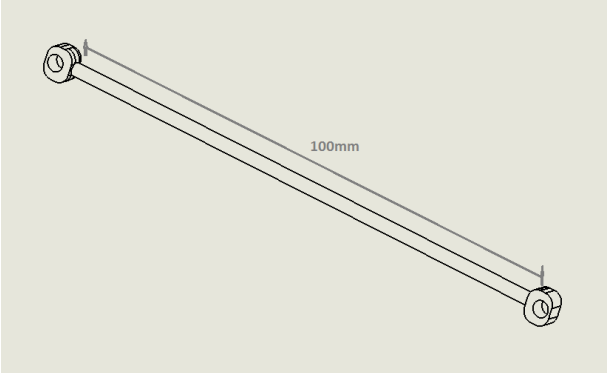
Σχήμα 2.8: Πλάγια Όψη Βραχίονα



Σχήμα 2.9: 3D Απεικόνιση Βραχίονα

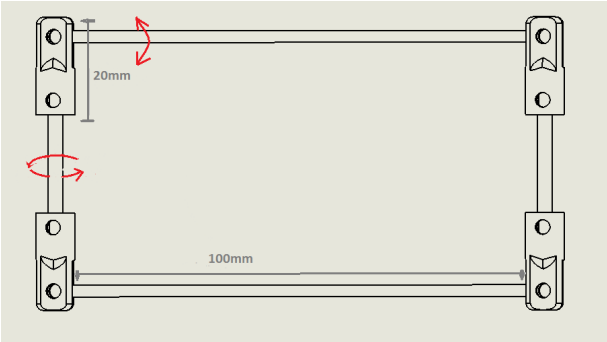


Σχήμα 2.10: Σύνδεσμος 1

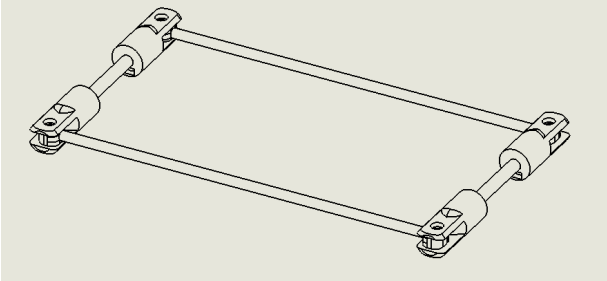


Σχήμα 2.11: Σύνδεσμος 2

Οι σύνδεσμοι 1 και 2 συναρμολογούνται όπως δείχνεται στις εικόνες 2.12, 2.13 και δίνουν τη δυνατότητα περιστροφής ως προς 2 άξονες.

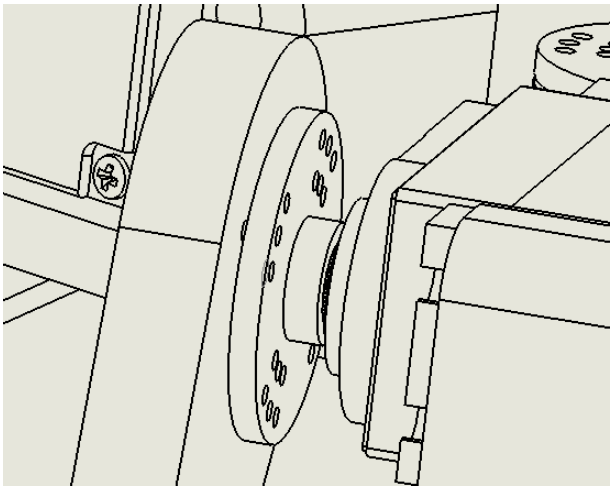


Σχήμα 2.12: Πλάγια Όψη Κάτω Βραχίονα

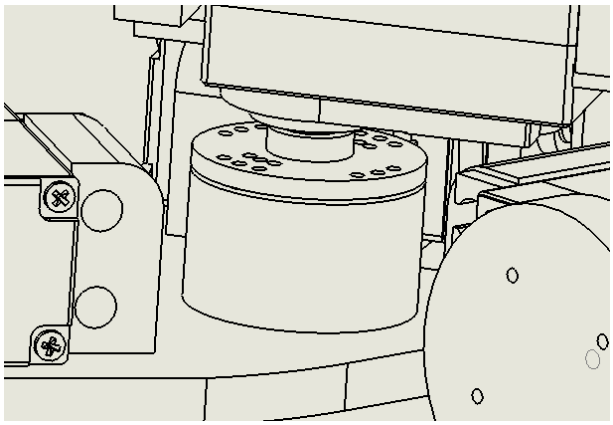


Σχήμα 2.13: 3D Απεικόνιση Κάτω Βραχίονα

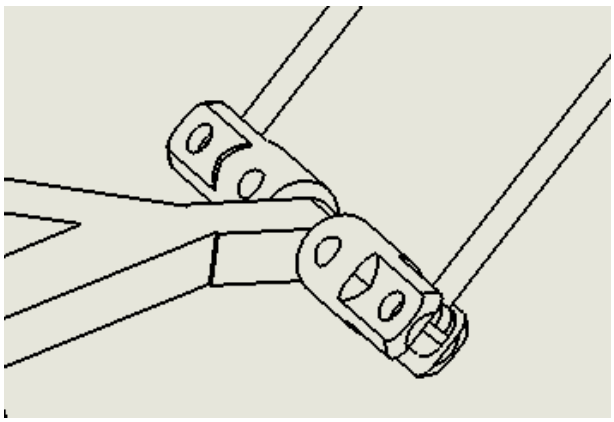
Πριν επιδειχθεί ολόκληρο το προσχέδιο του ρομπότ, θα δοθούν λεπτομερή σχέδια ορισμένων συνδεσμολογιών του ρομπότ (εικόνες 2.14, 2.15 και 2.16).



Σχήμα 2.14: Σύνδεση Άνω Βραχίονα - Κινητήρα



Σχήμα 2.15: Σύνδεση Περιστροφικού Κινητήρα - Κύριας Βάσης



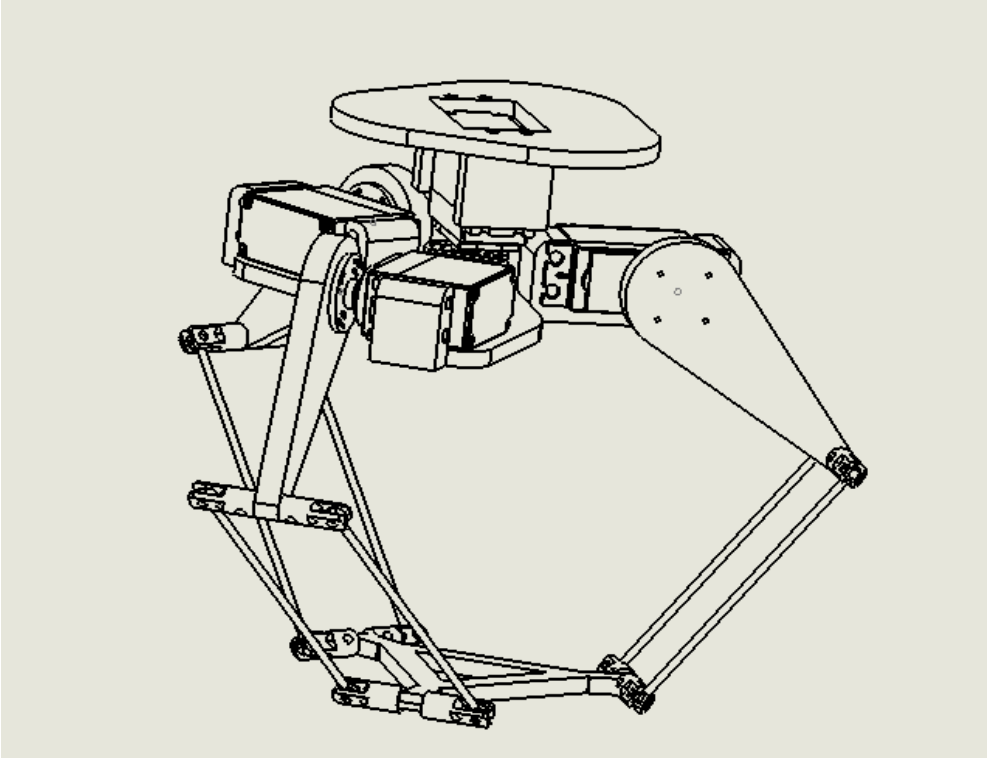
Σχήμα 2.16: Σύνδεση Κινούμενη Πλατφόρμας - Κάτω Βραχίονα

Τελικά, όλα τα παραπάνω σχέδια συν των κινητήρων (οι οποίοι προϋπήρχαν σχεδιασμένοι) απαρτίζουν το ρομπότ που δείχνεται στην εικόνα 2.17.

2.2 Robot Operating System (ROS)

Όσον αφορά στη συγκεκριμένη εργασία, χρησιμοποιήθηκε η έκδοση Kinetic του ROS σε λειτουργικό σύστημα Ubuntu 16.04 LTS. Παρακάτω, αφού εξηγηθούν ορισμένες έννοιες ώστε να βοηθηθεί ο αναγνώστης στην κατανόηση, θα αναφερθούν επιγραμματικά τα πακέτα που αξιοποιήθηκαν.

Το ROS είναι ένα μετα-λειτουργικό σύστημα (ενδιάμεσο λογισμικό) ανοιχτού κώδικα [14]. Είναι, στην ουσία, μια πλατφόρμα ανάπτυξης λογισμικού για ρομπότ. Αποτελείται από μία συλλογή εργαλείων, βιβλιοθηκών και συμβάσεων με στόχο την απλοποίηση της δημιουργίας πολύ-



Σχήμα 2.17: 3D Απεικόνιση του Ρομπότ

πλοκων λογισμικών για ρομποτικές πλατφόρμες. Αυτή η απλοποίηση επιτυγχάνεται μέσω του σχεδιασμού κάθε διαδικασίας ανεξάρτητα από τις υπόλοιπες και μέσω της εύκολης σύνθεσής τους. Δίνει δηλαδή τη δυνατότητα να διαμοιραστεί ο κώδικας και να αναπτυχθεί ξεχωριστά. Ένα από τα πλεονεκτήματα αυτού είναι ότι δε χρειάζεται να κανείς να ασχοληθεί με τετριμμένα προβλήματα της ρομποτικής τα οποία έχουν επιλυθεί στο παρελθόν.

Το ROS παρέχει υπηρεσίες παρόμοιες με αυτές ενός λειτουργικού συστήματος, όπως αφαίρεση υλικού, έλεγχο συσκευών χαμηλού επιπέδου, υλοποίηση συνήθους λειτουργικότητας, ανταλλαγή μηνυμάτων ανάμεσα σε διεργασίες και διαχείριση πακέτων.

Η λογική είναι η εξής: ο κόμβος Master είναι υπεύθυνος για το συγχρονισμό και την επικοινωνία μεταξύ των υπολοίπων κόμβων οι οποίοι εκτελούν διαφορετικές διεργασίες. Οι κόμβοι επικοινωνούν μεταξύ τους μέσω μηνυμάτων (messages) τα οποία δημοσιοποιούν (publish) σε ένα θέμα (topic). Προκειμένου να παραλαμβάνουν αυτά τα μηνύματα κάποιοι άλλοι κόμβοι πρέπει να εγγραφούν (subscribe) στο παραπάνω θέμα. Μία τελευταία έννοια που θα πρέπει να αναφερθεί είναι τα πακέτα (packages). Πακέτο είναι η βασική μονάδα για την οργάνωση λογισμικού στο ROS. Ένα πακέτο δύναται να περιέχει μία διαδικασία χρόνου εκτέλεσης (κόμβος), μία βιβλιοθήκη εξαρτημένη από το ROS, σύνολο δεδομένων ή αρχεία

διαμόρφωσης. Τα πακέτα είναι τα πιο θεμελιώδη στοιχεία στην κατασκευή αντικειμένων στο ROS.

Ο χώρος εργασίας του ROS ονομάζεται Catkin Build System. Εκεί παράγονται όλες οι βιβλιοθήκες και τα εκτελέσιμα αρχεία. Οι φάκελοι που εμπεριέχονται στο χώρο εργασίας είναι ο src (source) που περιέχει τον πηγαίο κώδικα, ο devel (development), όπου οι στόχοι (targets) αποθηκεύονται πριν εγκατασταθούν και ο build, όπου εφαρμόζεται η CMake εντολή. Ο χρήστης ενδιαφέρεται μόνο για τα περιεχόμενα του src φακέλου. Παρακάτω, εξηγούνται λεπτομερώς τα πακέτα που χρησιμοποιήθηκαν από το συγγραφέα.

Μηνύματα (Messages)

Όπως αναφέρθηκε νωρίτερα, για να μεταφερθούν δεδομένα από έναν κόμβο σε έναν άλλο στο πλαίσιο του ROS δημοσιεύονται μηνύματα σε topics. Ένα μήνυμα είναι μία απλή δομή δεδομένων που περιλαμβάνει πεδία δεδομένων, όπου οι βασικοί τύποι δεδομένων που υποστηρίζονται είναι integer, floating point, boolean, arrays, struct κλπ. Πολύ συχνά περιλαμβάνεται και ένας ιδιαίτερος τύπος μηνύματος, το Header, το οποίο περιλαμβάνει συνήθως metadata που χρησιμοποιούνται στο ROS για χρονισμό και ταυτοποίηση, όπως το timestamp και το frameID. Τα Header αξιοποιούνται από την κλάση Synchronizer του πακέτου message filters για να δημιουργήσουμε subscribers μέχρι και σε 9 topics, οι οποίοι ενεργοποιούν τη συνάρτηση callback όταν έχουμε ταυτόχρονα μηνύματα με το ίδιο ή παρόμοιο timestamp, ανάλογα με τις απαιτήσεις του συστήματος.

Tf

Το πακέτο tf του ROS περιέχει την υποδομή με τη χρήση της οποίας μάς επιτρέπει να παρακολουθούμε πολλαπλά συστήματα συντεταγμένων (frames). Αποθηκεύει τη σχετική θέση διάφορων frames σε μία δεντρική δομή για ένα χρονικό παράθυρο και επιτρέπει στο χρήστη να μετασχηματίζει σημεία, διανύσματα ανάμεσα σε δύο frames σε μία δεδομένη χρονική στιγμή. Για παράδειγμα, γνωρίζοντας το μετασχηματισμό ανάμεσα στα frame του χάρτη και μιας κάμερας, μπορούμε εύκολα να μετασχηματίσουμε ένα συγκεκριμένο σημείο σε σχέση με την κάμερα σε συντεταγμένες με βάση το χάρτη.

2.3 Υπόλοιπα Λογισμικά

2.3.1 OpenCV

Προτού εξηγηθούν οι βιβλιοθήκες OpenCV, θα πρέπει να δοθεί ένας σύντομος ορισμός της υπολογιστικής όρασης. Με τον όρο Υπολογι-

στική Όραση εννοείται το διεπιστημονικό πεδίο όπου οι υπολογιστές χρησιμοποιούνται για την πλήρη κατανοήση μιας ψηφιακής εικόνας. Ειδικότερα, για την παρούσα εργασία, το ενδιαφέρον επικεντρώνεται σε εφαρμογές με ρομπότ (robot tracking). Συνήθως, σε τέτοιες εφαρμογές, το ζήτημα είναι η εξαγωγή γεωμετρικών πληροφοριών, κυρίως οι αποστάσεις αντικειμένων σε σχέση με ορισμένο σύστημα συντεταγμένων.

Η OpenCV (Open Source Computer Vision Library) είναι μία ανοιχτού κώδικα βιβλιοθήκη για υπολογιστική όραση και μηχανική μάθηση. Δημιουργήθηκε για να παρέχει μία κοινή υποδομή για εφαρμογές υπολογιστικής όρασης και για να επιταχύνει τη χρήση της μηχανικής αντίληψης στα εμπορικά προϊόντα. Έχει BSD (Berkeley Software Distribution) άδεια, καθιστώντας εύκολη τη χρησιμοποίηση και τροποποίηση του κώδικα.

Η OpenCV έχει ενσωματωμένους στις βιβλιοθήκες περισσότερους από 2500 βελτιστοποιημένους αλγορίθμους. Περιλαμβάνει δηλαδή, ένα ολοκληρωμένο σύνολο κλασσικών αλλά και state-of-the-art αλγορίθμων υπολογιστικής όρασης. Αναφορικά, κάποιοι από τους αλγορίθμους αυτούς χρησιμοποιούνται σε εφαρμογές που αφορούν:

- ανίχνευση και την αναγνώριση προσώπων,
- εντοπισμό στάσιμων αλλά και κινούμενων αντικειμένων,
- εξαγωγή 3D μοντέλων των αντικειμένων,
- παραγωγή 3D point clouds από στερεοσκοπικά ζεύγη καμερών,
- εύρεση παρόμοιων εικόνων από μία βάση δεδομένων με εικόνες,
- διεπαφή χρήστη-υπολογιστή.

Η OpenCV είναι γραμμένη σε C++ και υποστηρίζει τα λειτουργικά συστήματα Linux, Windows, Android και MAC OS. Όλες οι κλάσεις και οι συναρτήσεις της OpenCV τοποθετούνται στο χώρο cv namespace.

Όσον αφορά στη σύνδεση του ROS με τη βιβλιοθήκη OpenCV, υπάρχουν διάφορα πακέτα (packages) που την επιτυγχάνουν. Το vision opencv αποτελεί μία δομή που περιέχει διάφορα πακέτα για την εύκολη συνεργασία του ROS και του OpenCV. Ειδικότερα, δύο κύρια πακέτα είναι:

- cv bridge το οποίο παρέχει τη δυνατότητα επικοινωνίας μεταξύ των ROS μηνυμάτων και της OpenCV και τη μετατροπή των ROS εικόνων σε εικόνες OpenCV και αντίστροφα,
- image geometry, που περιλαμβάνει μία συλλογή μεθόδων για τη διαχείριση τς γεωμετρίας της εικόνας και των pixels.

Για να μπορέσουν, επομένως, να χρησιμοποιηθούν οι παραπάνω βιβλιοθήκες θα πρέπει είτε να καλούνται με τον cv::προσδιοριστή είτε στην περιοχή που καλούνται να υπάρχει η οδηγία cv namespace;. Στην περίπτωση όπου μια συνάρτηση της OpenCV έχει το ίδιο όνομα με μια συνάρτηση κάποιας άλλης βιβλιοθήκης όπως με την STL (Standard Template Library) τότε πρέπει να χρησιμοποιηθεί αποκλειστικά ο cv::προσδιοριστής. Η πρωταρχική δομή δεδομένων της OpenCV είναι το αντικείμενο Mat το

οποίο αποθηκεύει τις εικόνες και τις συνιστώσες τους. Υπάρχει περιορισμένο σύνολο πρωταρχικών τύπων δεδομένων με τα οποία μπορεί να λειτουργήσει η βιβλιοθήκη. Τα στοιχεία πινάκων μπορεί να είναι:

- 8-bit insigned/signed integer,
- 16-bit unsigned/signed interger,
- 32-bit signed integer/floating point.

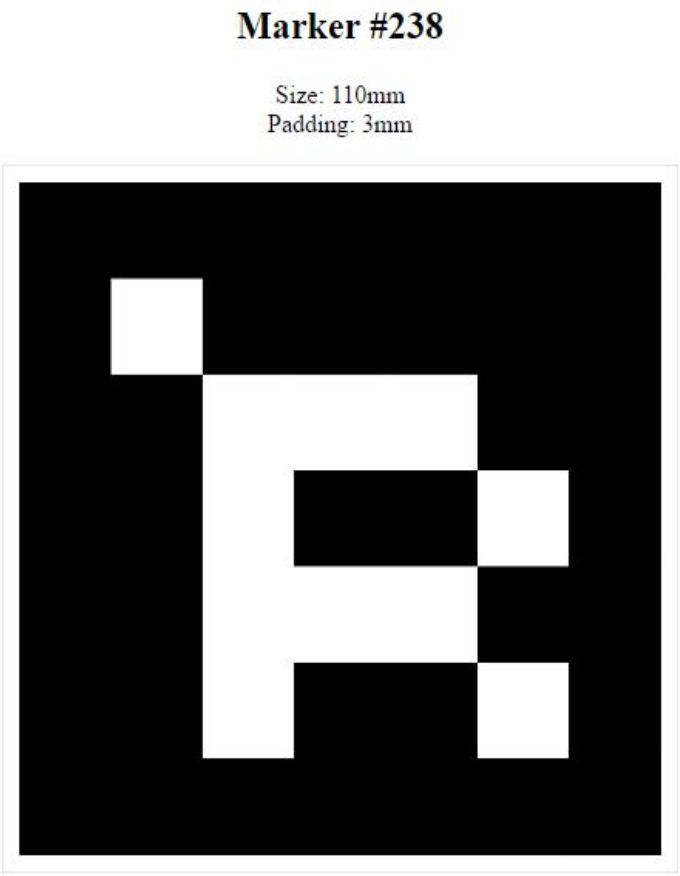
2.3.2 Έλεγχος και ArUco

Προκειμένου να εισαχθεί ένας βρόχος ανάδρασης στο σύστημα, υλοποιήθηκε μία τεχνική ελέγχου μέσω ψηφιακής εικόνας. Τέτοιες τεχνικές αναπτύχθηκαν τη δεκαετία του 1990. Στην ουσία, ο υπολογιστής μέσω της κάμερας λαμβάνει ορισμένα οπτικά δεδομένα για τον έλεγχο της κίνησης του ρομπότ. Στην κατασκευή μας, μία κάμερα είναι τοποθετημένη σε ένα σημείο με γνωστές ως προς τη βάση συντεταγμένες. Με τη βοήθεια ενός marker, ο οποίος βρίσκεται προσκολλημένος στο τελικό σημείο του ρομποτικού μηχανισμού (end effector), θα εντοπισθεί η θέση και ο προσανατολισμός του end effector ως προς την κάμερα. Τέλος, αφού γνωρίζουμε την επιθυμητή αλλά και την πραγματική θέση, μπορούμε να αναμορφώσουμε τις γωνίες των βαθμών ελευθερίας ώστε να μηδενίσουμε το σφάλμα. Το ίδιο ισχύει και για τον προσανατολισμό που εξαρτάται από τη γωνία του περιστροφικού κινητήρα σταθερής βάσης.

Για να καταστεί εφικτός ο ακριβής εντοπισμός της θέσης αποκλειστικά μέσω μιας μόνο κάμερας, ανατρέξαμε στη βοήθεια ενός fiducial marker εν ονόματι ArUco. Η λέξη ArUco προέρχεται από τα αρχικά των λέξεων Augmented Reality University of Cordoba. Με τον όρο Επαυξημένη Πραγματικότητα εννοούμε μία τεχνολογία με την οποία προβάλουμε φανταστικά αντικείμενα πάνω στον πραγματικό κόσμο δίνοντας την ψευδαίσθηση ότι τα αντικείμενα αυτά ανήκουν στον πραγματικό κόσμο. Πιο ειδικά, είναι ένα σύνολο τεχνικών, με τις οποίες βρίσκουμε την θέση της κάμερας σε σχέση με τον πραγματικό κόσμο.

Υπάρχουν δύο είδη επαυξημένης πραγματικότητας, με και χωρίς στόχο (marker). Η συγκεκριμένη εργασία περιλαμβάνει όπως προαναφέραμε ένα στόχο. Αυτός ο στόχος θα μας βοηθήσει να εντοπίσουμε τη θέση του end effector ως προς την κάμερα, της οποίας γνωρίζουμε τη θέση. Ο Marker που θα χρησιμοποιήσουμε διακρίνεται στην εικόνα 2.18. Ο συγκεκριμένος έχει διαστάσεις τετραγώνου 11cm.

Μερικά από τα χαρακτηριστικά του είναι η διχρωμία, η ασυμμετρία, οι υψηλές συχνότητες λόγω των απότομων αλλαγών χρωμάτων και το γεγονός ότι είναι ευδιάκριτοι στο χώρο. Αυτά τα χαρακτηριστικά είναι που δίνουν τη δυνατότητα να εξαχθούν γεωμετρικές πληροφορίες θέσεως. Ειδικότερα, η γνώση των ακριβών διαστάσεών τους επιτρέπει την εύρεση της σχέσης του συστήματος συντεταγμένων τους με την κάμερα. Από τη σχέση αυτή θα προκύψει ο Rotational Matrix, ο οποίος θα εξηγηθεί σε επόμενο



Σχήμα 2.18: ArUco Marker

κεφάλαιο με τη χρήση παραδείγματος. Τέλος, γνωρίζοντας τις διαστάσεις του ArUco καθίσταται εφικτή η εύρεση της απόστασής του από την κάμερα σε κάθε άξονα ξεχωριστά (x,y,z).

Όσον αφορά στη διασύνδεση των βιβλιοθηκών του ArUco με το ROS, έπρεπε να εγκατασταθούν ορισμένα πακέτα στο χώρο εργασίας. Αυτά ήταν:

- το vision opencv που αναλύθηκε σε προηγούμενη παράγραφο,
- το aruco ros που επιτρέπει τον εντοπισμό του στόχου,
- marker pose detection το οποίο υπολογίζει τις συντεταγμένες του στόχου σε σχέση με την κάμερα και τη σχέση των αντίστοιχων συστημάτων συντεταγμένων και
- το pal vision segmentation που επιτρέπει το διαχωρισμό του aruco από τα υπόλοιπα αντικείμενα του χώρου και την πλαισίωσή του.

2.3.3 Πακέτα σχετικά με Phidgets

Με τον όρο Phidgets (Physical Widgets) αναφερόμαστε στη συσκευή που επιτρέπει την αλληλεπίδραση μεταξύ του υπολογιστή και των σερβοκινητήρων (εικόνες 3.11, 3.12). Με άλλα λόγια, αποτελεί τον ελεγκτή των κινητήρων. Εκτενής ανάλυση της συγκεκριμένης συσκευής θα γίνει στο επόμενο κεφάλαιο. Προς τα παρόν, θα εξηγηθούν τα πακέτα που χρειάζονται για τον έλεγχό της.

Για τη χρήση της συσκευής Phidgets, το μόνο που απαιτείται είναι η εγκατάσταση των drivers. Η διαδικασία εγκατάστασης είναι πολύ απλή και υποστηρίζονται τα λειτουργικά συστήματα Windows, OS X, Linux, Phidget SBC, Android, IOS και Windows CE.

Ωστόσο, λόγω των πλεονεκτημάτων του ROS που αναφέρθηκαν νωρίτερα, επιλέχθηκε η σύνδεση του Phidgets με το ROS. Αυτό απαιτούσε ως προϋπόθεση να εγκατασταθούν στο χώρο εργασίας του ROS (Catkin Build System) ορισμένα πακέτα. Σε αυτά συμπεριλαμβάνονται τα πακέτα:

- phidgets, που περιέχει παραδείγματα από κώδικες για τη χρήση της συσκευής
- phidgets drivers, όπου όπως προδίδει και η ονομασία βρίσκονται οι drivers της συσκευής
- cob extern, το οποίο εμπεριέχει τη βιβλιοθήκη libphidget, που είναι υψίστης σημασίας για τους κώδικες που αφορούν τον έλεγχο του Phidget.

Μία διευκόλυνση ακόμα που προέτρεψε στη χρήση του Phidgets ήταν οι έτοιμοι κώδικες-παραδείγματα τους οποίους προσφέρει στο site της η ομώνυμη εταιρία [15]. Οι κώδικες αυτοί είναι γραμμένοι σε πολλές γλώσσες προγραμματισμού. Εξάλλου, παρέχει και έτοιμους κώδικες που δείχνουν τη σύνδεση με το ROS, γεγονός που διευκόλυνε την εκπόνηση της εργασίας αυτής. Οπότε, τα τελικά προγράμματα προέρχονται από τροποποίηση των παραπάνω προγραμμάτων.

2.4 Πακέτα ROS

Στην προκείμενη ενότητα θα γίνει μία ανασκόπηση των πακέτων που χρησιμοποιήθηκαν και θα αναφερθούν και ορισμένα ακόμα και πώς χρησιμοποιήθηκαν. Όλα τα παρακάτω αποτελούν το Catkin Workspace της εργασίας και υπενθυμίζεται ότι είναι αναρτημένα στο λογαριασμό github του συγγραφέα.

- cob extern: Περιέχει τη βιβλιοθήκη libphidgets για τον έλεγχο της συσκευής phidgets.
- phidgets drivers: Πραγματοποιούν τη σύνδεση του ROS με τη συσκευή phidgets.
- argyris: Κώδικας για τον έλεγχο των Servo όπου υλοποείται η αντί-

στροφη κινηματική.

- vision opencv: Αναγκαίο για τη συνδεση του ROS με τις βιβλιοθήκες OpenCV.
- marker pose detection, pal vision segmentation, aruco ros: Ευθύνονται για τον εντοπισμό του ArUco, την πλαισίωσή του, αλλά και την εκτίμηση της θέσης του.
- image pipeline: Περιέχει τον κώδικα για τη βαθμονόμηση της κάμερας, της οποίας η μεθοδολογία αναλύεται στο επόμενο κεφάλαιο.
- usb cam: Επιτρέπει τη σύνδεση κάμερας με τον υπολογιστή μέσω θύρας USB.

2.5 Σύνοψη Κεφαλαίου

Σε αυτό το κεφάλαιο πραγματοποιήθηκε η σχεδίαση του ρομπότ σε λογισμικό και αναλύθηκαν τα λογισμικά πάνω στο οποία πραγματοποιήθηκε ο έλεγχός του. Σε αυτό το σημείο ο αναγνώστης διαθέτει όλα τα εφόδια προκειμένου να κατανοήσει πλήρως την πειραματική διάταξη.

ΚΕΦΑΛΑΙΟ

3

ΠΕΙΡΑΜΑΤΙΚΗ ΔΙΑΤΑΞΗ

Σκοπός του συγκεκριμένου κεφαλαίου είναι να παρουσιαστεί αναλυτικά η διάταξη, αλλά και να δικαιολογηθεί η χρήση του κάθε εξαρτήματος ή διαδικασίας. Με τη σειρά, αναλύονται διεξοδικά τα μηχανολικά μέρη που απαρτίζονται από 3D εκτυπωμένα κομμάτια αλλά και αυτά του μηχανουργείου. Έπειτα, οι κινητήρες, η συσκευή ελέγχου phidgets και η κάμερα. Στο τέλος, δείχνονται ορισμένα πειράματα, τα αποτελέσματά τους και μερικές παρατηρήσεις του συγγραφέα.

3.1 Μηχανολογικά Μέρη

3.1.1 Κατασκευή και 3D εκτύπωση

Για την τρισδιάτητη εκτύπωση των εξαρτημάτων που παρατέθηκαν σε προηγούμενο κεφάλαιο χρησιμοποιήθηκε ο εκτυπωτής Replicator 2x της εταιρίας MakerBot, ο οποίος βρίσκεται στο εργαστήριο (βλ. εικόνα 3.1).

Ο εκτυπωτής, αφού του δοθεί το τρισδιάστατο σχέδιο, το μεταφράζει σε οδηγίες κίνησης. Εφόσον δρομολογηθεί η κίνηση, ζεσταίνει το υλικό (πλαστικό στη συγκεκριμένη περίπτωση) και μέσα από ένα στόμιο εκκρίνει το ζεστό υλικό πάνω σε θερμαινόμενη επιφάνεια ολοκληρώνοντας μία στρώση κάθε φορά. Αυτή η μέθοδος παρασκευής ονομάζεται Fused Filament Fabrication.

Ο συγκεκριμένος εκτυπωτής περιλαμβάνει δύο κεφαλές εκτύπωσης και είναι κατάλληλα σχεδιασμένος ώστε να χρησιμοποιεί ως πρώτη



Σχήμα 3.1: Εκτυπωτής Replicator 2x

ύλη το ABS (είδος ευδιάλυτου και ελαστικού πλαστικού). Οι κεφαλές κινούνται με τη βοήθεια κινητήρων και ιμάντων.

Build Volume (cm)	24.6 L x 15.2 B x 15.5 H
Layer Resolution	0.1 mm
Filament Diameter	1.75 mm

Πίνακας 3.1: Printing Specifications του εκτυπωτή

Μερικά από τα χαρακτηριστικά του εκτυπωτή διακρίνονται στον πίνακα 3.1. Να σημειωθεί ότι για το γέμισμα των εσωτερικών σημείων χρησιμοποιήθηκε ο εξαγωνιαίος τρόπος εκτύπωσης (ορίζεται ως προεπιλογή). Το γέμισα (infill) που επιλέχθηκε ήταν 70-85 επί τοις εκατό για το λόγο ότι οι κατασκευές ήταν αρκετά λεπτές σε πάχος και υπήρχε η πιθανότητα να μην αντέξουν το βάρος των κινητήρων.

Σε αυτό το σημείο παρατίθενται όλα τα μέρη του ρομπότ που τυπώθηκαν. Στην εικόνα 3.2 διακρίνεται η άνω βάση που στηρίζει όλη την κατασκευή όπως και τον κινητήρα ο οποίος είναι υπεύθυνος για τον προσανατολισμό. Στη συνέχεια (εικόνες 3.3 και 3.4) , φαίνεται η κύρια βάση η οποία στηρίζει τους 3 κινητήρες. Όμοια στις εικόνες 3.5 3.6 έχουμε την κινούμενη πλατφόρμα και έναν από τους 3 βραχίονες.

Όλα τα παρακάτω κομμάτια αποτελούν τα πλαστικά εξαρτήματα της κατασκευής. Ωστόσο, λόγω ορισμένων αιτιών που εξηγούνται στην επόμενη παράγραφο, τα υπόλοιπα εξαρτήματα κατασκευάστηκαν από αλουμίνιο.



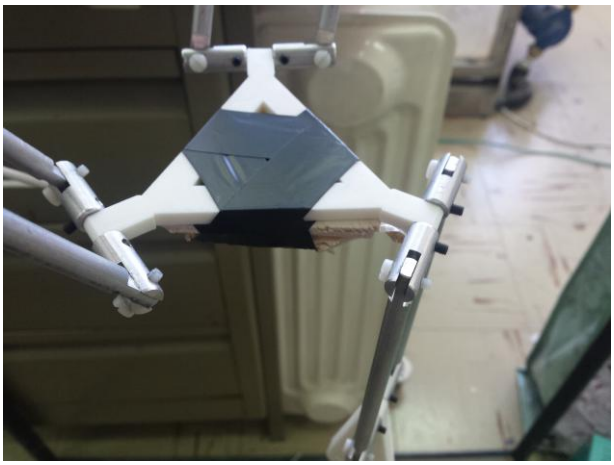
Σχήμα 3.2: Άνω Βάση



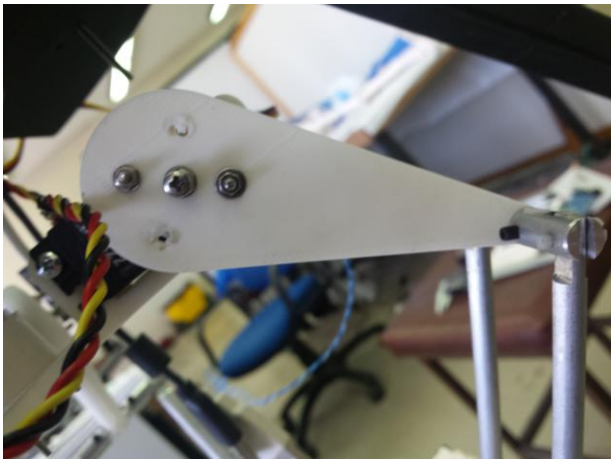
Σχήμα 3.3: Κάτοψη κύριας Βάσης



Σχήμα 3.4: Άνοψη κύριας Βάσης



Σχήμα 3.5: Κινούμενη Πλατφόρμα - End Effector



Σχήμα 3.6: Βραχίονας

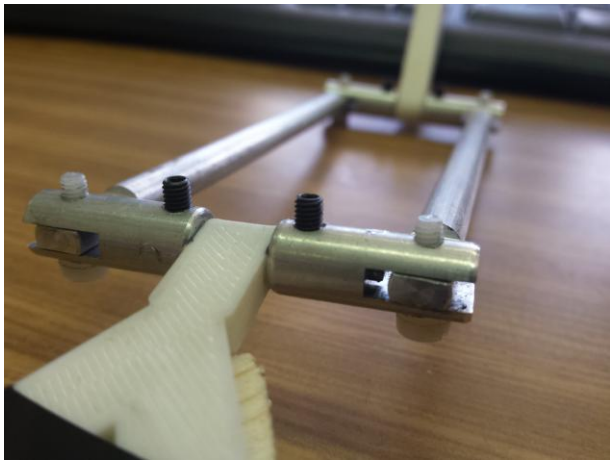


Σχήμα 3.7: Κάτω Βραχίονας σε ουδέτερη και υπό γωνία θέση

3.1.2 Μηχανουργείο

Λόγω της φύσης της κινήσεως θεωρήθηκε ότι κάποια κομμάτια δε θα αποτελούσαν στιβαρή κατασκευή. Το πλαστικό παρότι ελαφρύ και άρα χρήσιμο για τις προδιαγραφές της εργασίας αυτής είναι ευδιάλυτο σε συνθήκες υψηλής θερμοκρασίας. Δεδομένης της τριβής που θα εμφανιζόταν κατά την κίνηση του ρομπότ, θα είχε παρατηρηθεί αλλοίωση μερικών εξαρτημάτων.

Έτσι, αντί για πλαστικό εξαγόμενο από τον 3D εκτυπωτή, αποφασίστηκε να χρησιμοποιηθεί ως πρώτη ύλη κάποια μέταλλο, όπως πχ το αλουμίνιο ή το ανοξείδωτο. Οπότε και ανατέθηκε η κατασκευή στο μηχανουργείο του πανεπιστημίου Πατρών. Στον υπεύθυνο-κατασκευαστή δόθηκε ένα αρχείο σε μορφή STL που προσομοιώνει, εκτός των άλλων, την κίνηση και αναλόγως τη διαθεσιμότητα των υλικών, κατασκεύασε τα κομμάτια που φαίνονται στην εικόνα 3.7. Η κατασκευή έχει τη δυνατότητα στρέψης έως και 43°. Ελάχιστα παρακάτω (εικόνα 3.8), δείχνεται μια λεπτομερής εικόνα του μηχανισμού.



Σχήμα 3.8: Κοντινή Λήψη Κάτω Βραχίονα

Εξαιτίας έλλειψης υλικών και αδυναμίας αγοράς τους, για την κατασκευή των μεταλλικών εξαρτημάτων χρησιμοποιήθηκαν κυλινδρικές ράβδοι αλουμινίου 6mm και 8mm, μεγέθη μεγαλύτερα από τα επιθυμητά που πρόσθεσαν ορισμένα γραμμάρια στη συνολική κατασκευή.

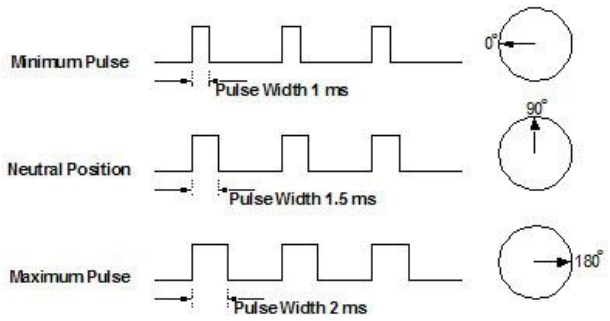
Τελικά, η κατασκευή ύστερα και από την πρόσθεση των κινητήρων και των απαραίτητων εξαρτημάτων συναρμολόγησης (βίδες, παξιμάδια, ροδέλες κλπ) ζυγίστηκε στα 462gr. Δεδομένου του αρχικού στόχου των 500gr, το ρομπότ πληροί τις προδιαγραφές του βάρους.

3.2 Κινητήρες

Η περιστροφική κίνηση των βραχίονων βασίστηκε αποκλειστικά σε σερβοκινητήρες. Ένας σερβοκινητήρας είναι ένας περιστροφικός επενεργητής που επιτρέπει τον έλεγχο της γωνιακής θέσης, της ταχύτητας και της επιτάχυνσης με ακρίβεια. Αποτελείται από ένα μικρό DC κινητήρα, ένα ποτενσιόμετρο και ένα κύκλωμα ελέγχου.

Όταν ο κινητήρας βρίσκεται στην επιθυμητή θέση, τότε διακόπτεται η τροφοδότησή του. Εάν όχι, υλοποιείται ένας P ελεγκτής ανάλογος της διαφοράς πραγματικής και επιθυμητής θέσης. Η υλοποίηση πραγματοποιείται με την αποστολή παλμών ποικίλου πλάτους μέσω του καλωδίου ελέγχου. Τα χαρακτηριστικά του παλμού αυτού είναι η ύπαρξη ενός ελάχιστου (πλάτους) παλμού, ενός μέγιστου και ενός ρυθμού επανάληψης.

Ο σερβοκινητήρας αναμένει ένα παλμό κάθε 20 ms. Η γωνία του εξαρτάται αποκλειστικά από τη διάρκεια του παλμού αυτού, άρα και διαμορφώνεται κατά πλάτος παλμού (Pulse Width Modulation). Συνήθως, το εύρος κίνησης περιορίζεται στις $\pm 90°$ από την ουδέτερη θέση. Ως ουδέτερη θέση ορίζεται εκείνη όπου υπάρχει ίση περιστροφή κατά τη φορά των δεικτών του ρολογιού όση και στην αντίθετη. Ο παλμός της συγκεκριμένης θέσης διαρκεί 1.5 ms σε όλες τις περιπτώσεις. Συνηθίζεται, επίσης, η διάρκεια του ελάχιστου και του μέγιστου παλμού να είναι κοντά στα 1 ms και 2 ms αντίστοιχα. Οι παραπάνω περιπτώσεις διακρίνονται στην εικόνα 3.9.



Σχήμα 3.9: PWM Διαμόρφωση των Servo

Ειδικά, χρησιμοποιήθηκαν οι κινητήρες HS-422 Deluxe Standard Servo της HITEC (εικόνα 3.10). Θεωρήθηκαν ιδανικοί για την εργασία αυτή διότι παρά την έλλειψη ροπής σε σχέση με άλλους κινητήρες είναι μικροί σε βάρος και σε όγκο. Στους πίνακες 3.2 και 3.3 δίδονται τα βασικά χαρακτηριστικά τους. [16]



Σχήμα 3.10: HS-422 Deluxe Standard Servo της Hitec

Operating Voltage (Volts DC)	4.8 V $\sim$ 6 V
Speed (Second @ 60°)	0.21 $\sim$ 0.16
Maximum Torque Range oz./in	46 $\sim$ 57
Maximum Torque Range kg./cm	3.3 $\sim$ 4.1
Current Draw at Idle	8 mA
No Load Operating Current Draw	150 mA
Stall Current Draw	800 mA
Dead Band Width	8 μs

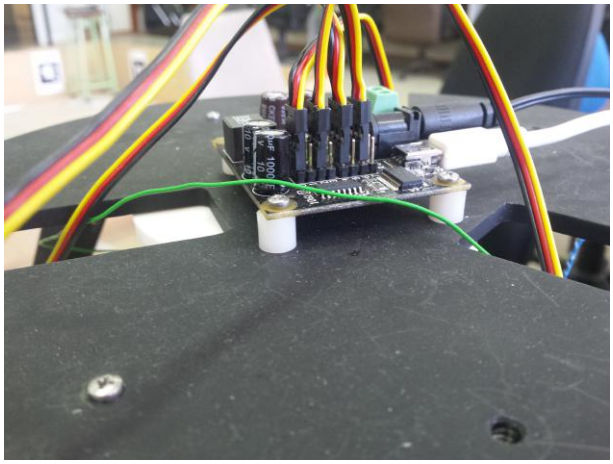
Πίνακας 3.2: Perfomance Specifications

Dimensions (Inches)	1.69 x 0.77 x 1.44
Dimensions (Metric)	40.6 x 19.8 x 36.6
Weight (Ounces)	1.60
Weight (Grams)	45.5
Circuit Type	HT7003 Analog SMT
Motor Type	3 Pole Metal Brush Ferrite
Gear Material	Nylon
Bearing Type	Dual Oilite
Output Shaft (type / ∅mm	Standard 24
Case Material	Plastic
Dust/Water Resistance	N/A
Connector Gauge (AWG)/Strand Count	25 / 40

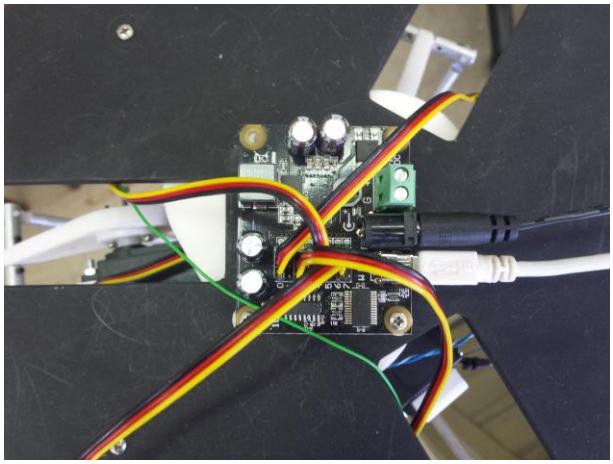
Πίνακας 3.3: Physical Specifications

3.3 Phidgets Device

Σε περασμένο κεφάλαιο αναφέρθηκε ότι οι επενεργητές ελέχθηκαν μέσω μιας συσκευής phidgets. Στις εικόνες 3.11 και 3.12 δείχνεται η συνδεσμολογία της.



Σχήμα 3.11: Πλάγια όψη Phidgets



Σχήμα 3.12: Κάτοψη Phidgets

Το γεγονός ότι το πλαίσιο στήριξης του ρομπότ είναι μεταλλικό προέτρεψε στην τοποθέτηση πλαστικών αποστατών με σκοπό την αποφυγή βραχυκυκλωμάτων.

Η συσκευή phidgets συνδέεται με τον ηλεκτρονικό υπολογιστή μέσω θύρας USB και για την κίνηση των κινητήρων απαιτείται τροφοδότηση από εξωτερική πηγή (μία πρίζα στη συγκεκριμένη εργασία).

Ειδικότερα, χρησιμοποιήθηκε η συσκευή PhidgetAdvancedServo 8-Motor [15], η οποία επιτρέπει τον έλεγχο θέσης, ταχύτητας και επιτάχυνσης

έως και 8 RC σερβοκινητήρων. Στους πίνακες 3.4, 3.5 και 3.6 διακρίνονται ορισμένα βασικά χαρακτηριστικά. Διαθέτει υψηλή ακρίβεια 125 βημάτων ανά μοίρα. Επίσης, μετράει την ενεργειακή κατανάλωση κάθε κινητήρα και χρησιμοποιεί ρυθμιστή μεταγωγής για την προστασία από υπερτάσεις. Τέλος, δύναται να τροφοδοτήσει τους σερβοκινητήρες μέχρι τα 3.4 Amps.

API Object Name	AdvancedServo
Number of Ports	8
Pulse Width Min	83.3 ns
Pulse Width Max	2.7 ms
Pulse Width Resolution	83.3 ns
Pulse Code Period Max	25 ms

Πίνακας 3.4: Servo Controller [15]

Supply Voltage Min	6 V DC
Supply Voltage Max	15 V DC
Current Consumption Max	26 mA
Continuous Motor Current Max	(per motor) 1.6 A
Overcurrent Trigger	(combined) 12 A
Surge Current Max	(per motor) 3 A
Output Impedance (Motor)	600 Ω
Output Motor Voltage	5 V DC
USB Speed	Full Speed

Πίνακας 3.5: Electrical Properties [15]

Power Jack Hole Diameter	5.5 mm
Power Jack Pin Diameter	2.1mm
Power Jack Polarity	Center Positive
Recommended Wire Size	12 - 24 AWG
Object Temperature Min	0°C
Object Temperature Max	70°C

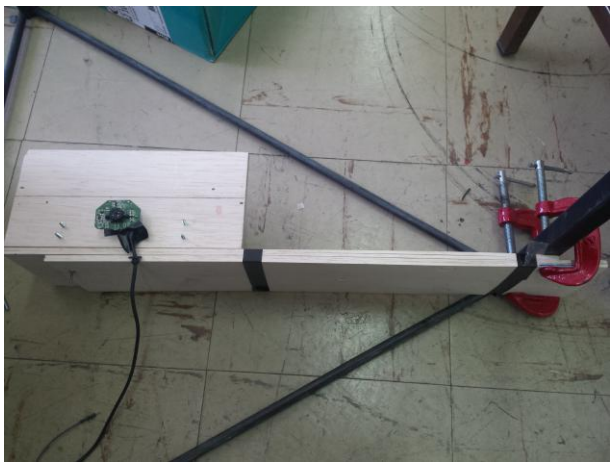
Πίνακας 3.6: Physical Properties [15]

3.4 Camera

Η κάμερα που χρησιμοποιήθηκε για τον έλεγχο του συστήματος είναι το μοντέλο της εικόνας 3.13. Αυτή είναι τοποθετημένη και σταθερή στο χαμηλότερο σημείο της κατασκευή (91.5cm κατά τον άξονα z κάτω από το πλαίσιο στήριξης - frame) στοχεύοντας προς τα πάνω για να εντοπίζει τον marker, όπως φαίνεται στο σχήμα 3.14.



Σχήμα 3.13: Camera



Σχήμα 3.14: Μηχανισμός Στήριξης Κάμερας

Αναλυτικότερα, όσον αφορά τα χαρακτηρηστικά λήψης, η κάμερα προσφέρει ως σήμα εξόδου ψηφιακή εικόνα RGB 32-bit JPEG με ανάλυση εικόνας 640x480, η οποία μεταφέρεται στον υπολογιστή μέσω σύνδεσης USB. Τέλος, επιλέχθηκε αυτή η κάμερα διότι είχε τη δυνατότητα αυτόματης εστίασης σε μικρές αποστάσεις (της τάξης των μερικών δεκάδων cm).

Πριν, ωστόσο χρησιμοποιηθεί η κάμερα, θα έπρεπε πρώτα να βαθμονομηθεί. Αυτή η διαδικασία θα εξηγηθεί παρακάτω, αφού ορισθούν κάποιες έννοιες.

Βαθμονόμηση

Κάθε κάμερα χαρακτηρίζεται από δύο είδη παραμέτρων, τις εγγενείς (εσωτερικές) και τις εξωγενείς (εξωτερικές). Οι εξωγενείς αφορούν στη θέση και τον προσανατολισμό του συστήματος αναφοράς της ως προς το σύστημα αναφοράς του κόσμου. Από την άλλη, οι εγγενείς συνδέουν τις συντεταγμένες σε pixel ενός σημείου της εικόνας με τις αντίστοιχες συντεταγμένες στο σύστημα αναφοράς της κάμερας.

Οι εγγενείς παράμετροι προσδιορίζονται ως ένα σύνολο παραμέτρων το οποίο είναι απαραίτητο για να περιγράψει τα οπτικά, γεωμετρικά και ψηφιακά χαρακτηριστικά της κάμερας. [13]
Οι παράμετροι αυτές είναι:

- f εστιακό μήκος (focal length)
- οι παράμετροι οι οποίες μετασχηματίζουν το σύστημα συντεταγμένων της κάμερας [x,y] στο σύστημα συντεταγμένων της εικόνας σε μονάδες pixel [u,v]. Το κύριο σημείο (principal point) $o = (u_0, v_0)$ θεωρείται το κέντρο της εικόνας και το φυσικό μέγεθος του εικονοστοιχείου (s_x, s_y) στους άξονες x, y είναι σταθερό
- οι γεωμετρικές παραμορφώσεις που προκαλούνται από τους φακούς της κάμερας. Συνήθως, ο φακός εισάγει παραμόρφωση στην εικόνα, η οποία αλλοιώνει κυρίως την περιφέρεια της. Είναι δηλαδή ακτινικές, όσο απομακρυνόμαστε από το κέντρο της κάμερας, τόσο αυξάνονται. Εκτός από ιδιαίτερες περιπτώσεις πχ μία κάμερα fish-eye, αυτός ο τύπος παραμορφώσεων αγνοείται.

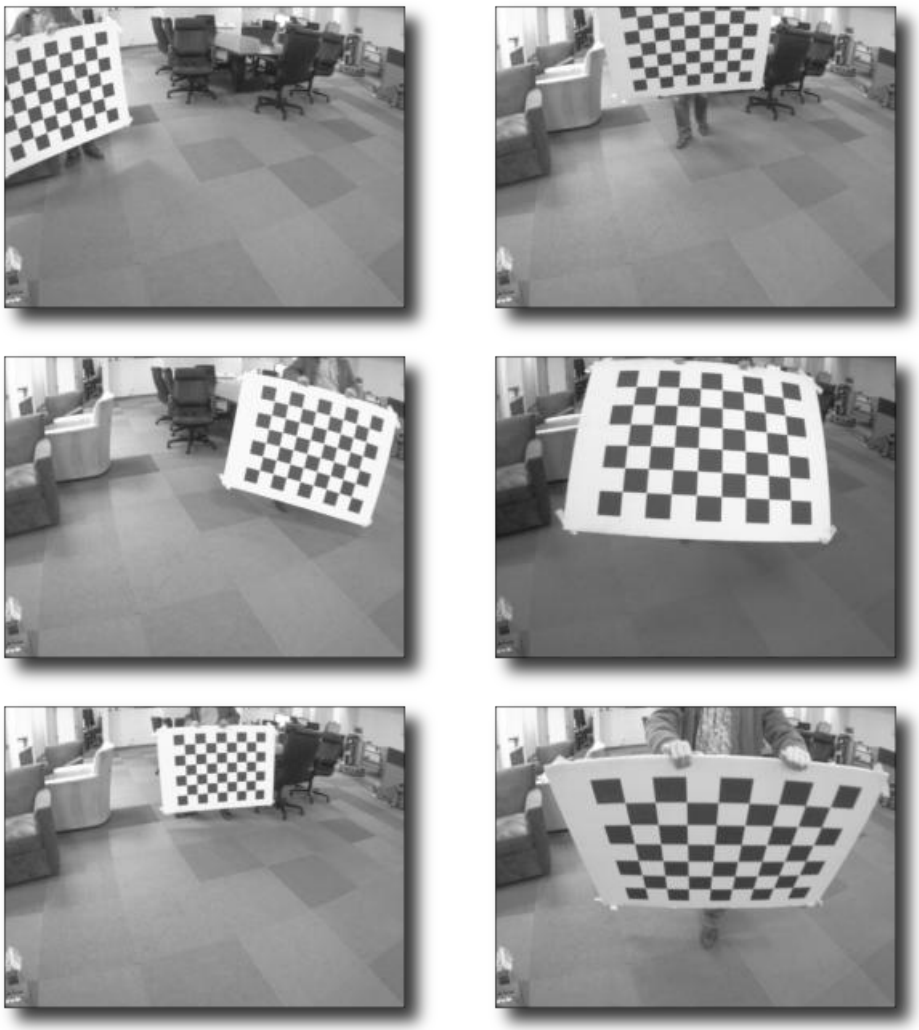
Έχοντας γνώση όλων των παραπάνω, υπολείπεται μία έννοια που θα πρέπει ο αναγνώστης να κατανοήσει, η βαθμονόμηση. Βαθμονόμηση μιας κάμερας είναι η διαδικασία που συνδέει το θεωρητικό μοντέλο ανάλυσής της με την πραγματική συσκευή και υπολογίζει τη θέση και τον προσανατολισμό της σε σχέση με ένα συγκεκριμένο πλαίσιο αναφοράς. Επομένως, είναι η διαδικασία εύρεσης των εγγενών και εξωγενών παραμέτρων της κάμερας. Είναι απαραίτητη η εύρεση εξισώσεων προβολής που συνδέουν τις γνωστές συντεταγμένες ενός συνόλου 3D με τις προβολές του και η επίλυσή τους ως προς τις παραμέτρους της κάμερας. Για να μπορέσουν να βρεθούν οι συντεταγμένες ορισμένων 3D σημείων, οι μέθοδοι βαθμονόμησης βασίζονται σε μία ή περισσότερες εικόνες από ένα μοτίβο βαθμονόμησης με γνωστή γεωμετρία (σκακιέρα) από το οποίο μπορούν να βρεθούν με ακρίβεια τα χαρακτηριστικά της εικόνας.[13]

Η βαθμονόμηση υλοποιείται μέσω της συνάρτησης calibrateCamera() της βιβλιοθήκης OpenCV. Χάρη στα πολλαπλά πλεονεκτήματα του ROS, η διαδικασία εκτελέστηκε μέσω αυτού, απλά τρέχοντας την εντολή:

```
$ rosrun camera_calibration cameracalibrator.py --size 8x6 --square 0.108 image:=/camera/image
_raw camera:=/camera
```

και τη βοήθεια μιας σκακιέρας. Στην παραπάνω εντολή, ο χρήστης, αρ-

χικά, εισάγει το μέγεθος των τετραγώνων της σκακιέρας. Έπειτα, πρέπει να ακολουθήσει μία τυπική διαδικασία που περιλαμβάνει τα εξής βήματα:



Σχήμα 3.15: Σειρά Κινήσεων Σκακιέρας

1. Πρέπει η σκακιέρα να κινηθεί κατά μήκος όλου του X και Y άξονα. Αφού πραγματοποιηθεί σωστά αυτή η διαδικασία, θα πρέπει οι μπάρες X και Y να είναι γεμάτες.
2. Στη συνέχεια, μειώνουμε και αυξάνουμε την απόσταση της σκακιέρας από την κάμερα ώστε να γεμίσει και η μπάρα υπό το όνομα "Size"
3. Τέλος, αρκεί να φέρει ο χρήστης τη σκακιέρα υπό γωνία ώστε να γεμίσει η μπάρα SKEW.

Στο σημείο αυτό ολοκληρώθηκε η βαθμονόμηση. Όπως προαναφέραμε, το αποτέλεσμα αυτής είναι η εύρεση των εγγενών παραμέτρων.

Ύστερα από μερικούς υπολογισμούς, ο υπολογιστής τυπώνει αυτές τις παραμέτρους. Για τη συγκεκριμένη κάμερα προέκυψαν:

$$Distortion = \begin{bmatrix} 0.122063 & -0.163946 & 0.015101 & 0.009662 & 0.000000 \end{bmatrix}$$

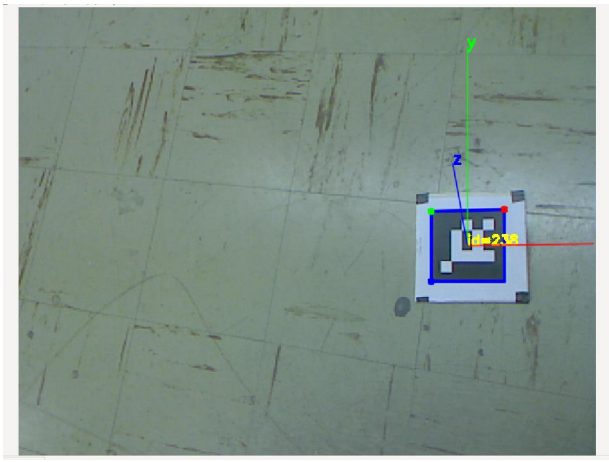
$$Rectification = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$CameraMatrix = \begin{bmatrix} 761.968750 & 0 & 353.514045 & 0 \\ 0 & 765.703979 & 283.294571 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Τέλος, αυτές τις παραμέτρους τις εισάγαμε στο σύστημα ως μόνιμες με την εντολή COMMIT που εμφανίζεται στην οθόνη.

3.5 Εντοπισμός ArUco

Έχοντας βαθμονομήσει την κάμερα, είμαστε σε θέση να τη χρησιμοποιήσουμε για pose estimation. Στην αρχή, πραγματοποιήσαμε ορισμένες δοκιμές. Ο AruCo Marker που χρησιμοποιήθηκε είναι διαστάσεων 11x11cm με περιθώριο 3cm τυπωμένο πάνω σε υλικό με πάχος 8mm.

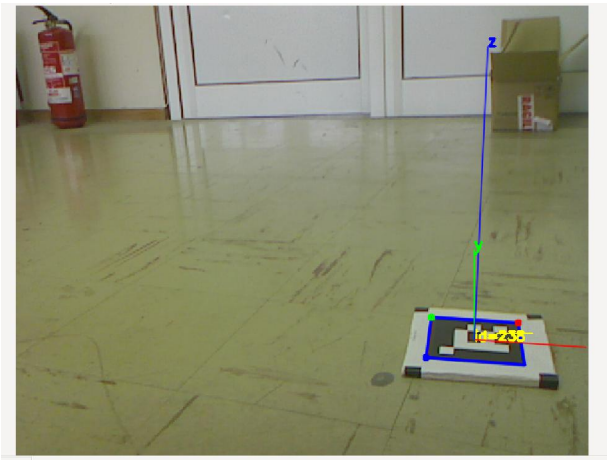


238=(408.248,279.237) (406.116,205.248) (482.854,199.601) (487.13,275.122) Txyz=0.15578 -0.0282654 0.984165 Rxyz=-1.77654 0.226683 0.173942

Σχήμα 3.16: Λήψη από Κάμερα και Εικόνα (Μετρήσεις) από το Terminal των Ubuntu

Στη φωτογραφία (εικόνα 3.16) που τραβήχθηκε από την κάμερα, διακρίνεται ο εντοπισμός και η πλαισίωση του AruCo Marker. Αφού έχει

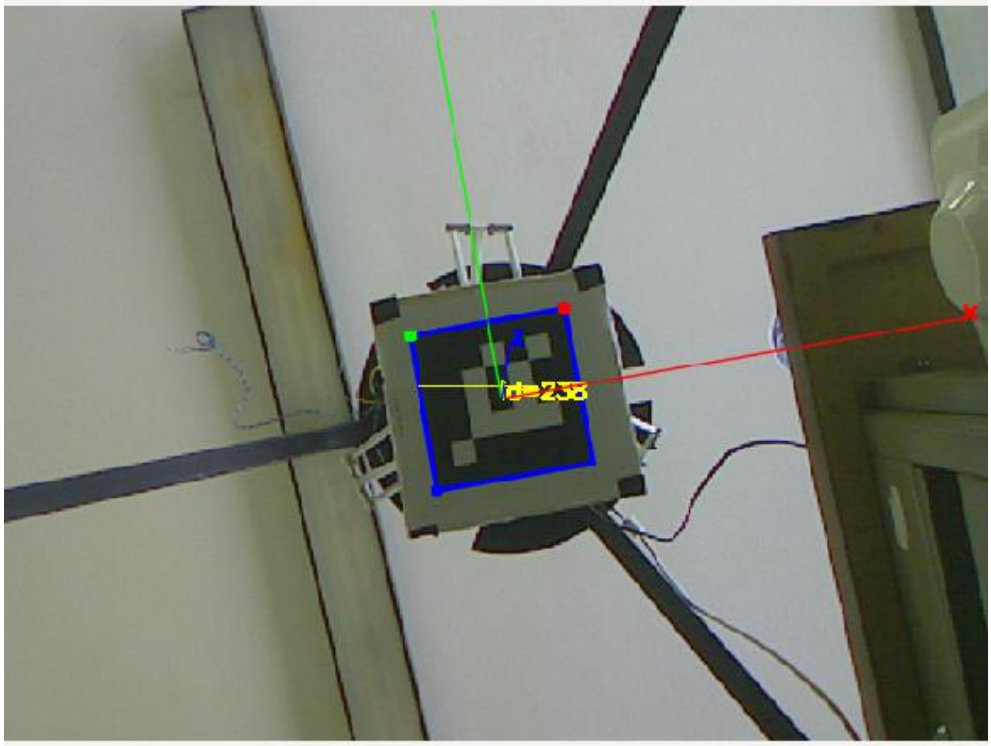
εντοπιστεί, κάθε χρονική στιγμή που ο χρήστης επιλέγει εξάγονται στο terminal ορισμένα στοιχεία. Πρώτα, αναγνωρίζεται το Id του marker. Έπειτα εμφανίζονται τέσσερις δυάδες αριθμών οι οποίες αναπαριστούν τα pixels των γωνιών του marker (ο οποίος είναι πάντα τετράγωνος και άρα έχει τέσσερις γωνίες). Τέλος, εμφανίζονται δύο πίνακες. Ο T_{xyz} και ο R_{xyz} ή ολογράφως Translational και Rotational αντίστοιχα. Από τη μία, ο T_{xyz} στην παρουσία εργασία θα χρησιμοποιηθεί για την εύρεση και διόρθωση της θέσης του end effector (μέσω των γωνιών των κινητήρψν $\theta_1, \theta_2, \theta_3$). Από την άλλη, ο R_{xyz} θα χρησιμοποιηθεί για τη διόρθωση του προσανατολισμού (θ_4 - γωνία περιστροφικού κινητήρα βάσης). Στο συγκεκριμένο παράδειγμα (εικόνα 3.16), οι συντεταγμένες του στόχου είναι $x = 0.155m, y = -0.028m, z = 0.984m$ και ο προσανατολισμός είναι $\theta_4 = 0.174rad = 9.96°$, τα οποία δεδομένα επιβεβαιώθηκαν μέσω γεωμετρικών οργάνων.
Εν συνεχεία, δίδεται άλλο ένα παράδειγμα (εικόνα 3.17). Εδώ, επιχείρεται άλλη μία δοκιμή όπου ο στόχος τοποθετείται υπό γωνία σε σχέση με την κάμερα. Όμοια με πριν, επιβεβαιώθηκαν με γεωμετρικά όργανα τα αποτελέσματα που τυπώνονται στο terminal.



238=(413.704,256.222) (419.62,214.985) (512.821,216.377) (518.277,257.359) Txyz=0.145036 -0.0271103 0.786321 Rxyz=-2.65816 0.0359494 0.24404

Σχήμα 3.17: Λήψη από Κάμερα και Εικόνα (Μετρήσεις) από το Terminal των Ubuntu

Μία τελευταία διαδικασία που απομένει πριν τη χρήση της κάμερας είναι η σωστή τοποθέτησή της στο χώρο. Η κάμερα πρέπει να είναι τοποθετημένη στην κατακόρυφη ευθεία του ρομπότ και παράλληλη στην κύρια βάση του. Για να επιτευχθούν τα παραπάνω, για διάφορες θέσεις η κάμερα τοποθετήθηκε έτσι ώστε να συμφωνεί με την κινηματική του ρομπότ. Για παράδειγμα, στην εικόνα 3.18 δόθηκε η εντολή στο ρομπότ να πάει στη θέση (-3mm, 4mm, 150mm) με προσανατολισμό $\theta_4 = 10°$, τα οποία δεδομένα συμφωνούν με αυτά που τυπώνονται στο terminal.




```
238=(280.972,316.121) (263.577,214.766) (364.216,197.394) (382.581,298.
541) Txyz=-0.00281192 -0.00445672 0.734618 Rxyz=-1.62269 0.11775 -0.156
144
```

Σχήμα 3.18: Μέτρηση Από Κάμερα

3.6 Σχεδιασμός και Υλοποίηση Ελέγχου

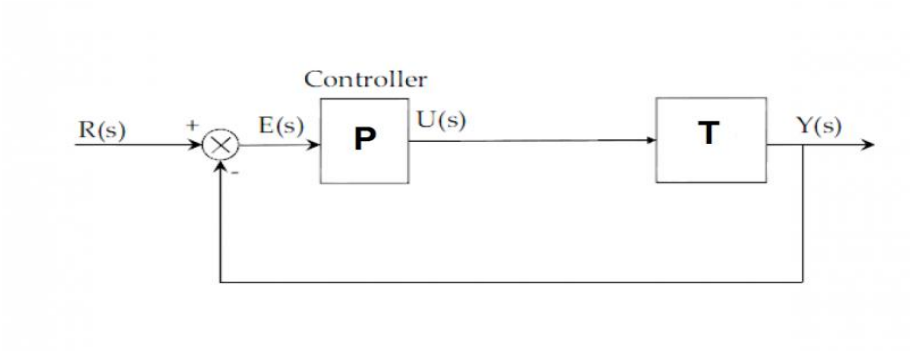
Στο σημείο αυτό, το σύστημα μπορεί να λειτουργήσει ικανοποιητικά, αναλόγως τις απαιτήσεις του χρήστη, και χωρίς ανάδραση, . Η τελική θέση του end effector προκύπτει κατά μέσο όρο με σφάλμα θέσης $1.5cm$ και σφάλμα προσανατολισμού $1°$. Ωστόσο, προκειμένου να ελαχιστοποιηθεί το σφάλμα, σε αυτή την ενότητα εισάγεται έλεγχος ανάδρασης που περιλαμβάνει P ελεγκτή, ο οποίος δέχεται ως είσοδο τις μετρήσεις της κάμερας. Αφού εξηγηθεί συνοπτικά ο P έλεγχος, δείχνεται ο τρόπος με τον οποίο αυτός εφαρμόζεται στην παρούσα εργασία.

Ο P ελεγκτής (Proportional) αποτελεί έναν έλεγχο κλειστού βρόχου και χρησιμοποιείται ευρύτατα. Σκοπός του είναι ο συνεχής υπολογισμός ενός σφάλματος $e(t)$ το οποίο είναι η απόκλιση μεταξύ της επιθυμητής τιμής μιας μεταβλητής και της μετρούμενης τιμής της. Ακόμα, εφαρμόζει μια διόρθωση ανάλογη με το συντελεστή κέρδους έως ότου η παραπάνω απόκλιση να ικανοποιεί προδιαγραφές ορισμένες από το χρήστη. Ο έλεγχος αυτός εκφράζεται μαθηματικά ως:

$$u(t) = K_p e(t)$$

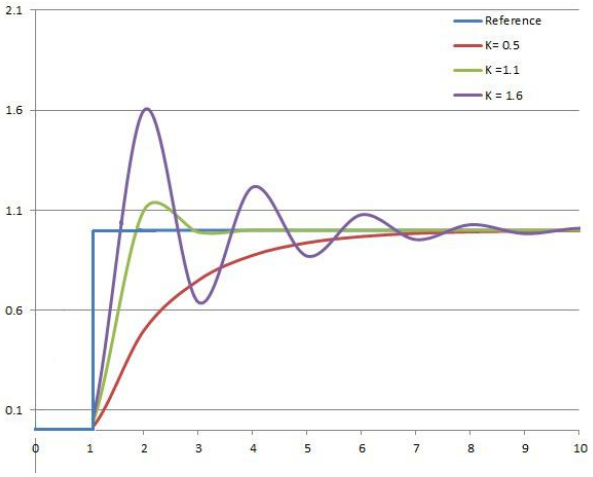
όπου K_p το κέρδος του ελεγκτή.

Στην εικόνα 3.19 δείχνεται το μπλοκ διάγραμμα του συστήματος ύστερα από την πρόσθεση του P ελεγκτή.



Σχήμα 3.19: P Ελεγκτής

Σε αυτό το σημείο θα εξηγηθεί πώς επιδρά ο αναλογικός ελεγτής στο σύστημα [17]. Ο P έλεγχος παράγει μία έξοδο ανάλογη της τιμής του παρόντος σφάλματος. Αυτό έχει ως αποτέλεσμα, μία μεγάλη τιμή του συντελεστή του να οδηγεί σε μεγάλες μεταβολές στο σφάλμα και τελικά το σύστημα να φτάνει στην αστάθεια. Από την άλλη, ένας μικρός συντελεστής ελέγχου προκαλεί μικρή μεταβολή στην απόκριση και μπορεί να μην καταφέρει να αντιμετωπίσει τις ανατάραξεις του συστήματος. Λόγω του γεγονότος ότι απαιτείται η ύπαρξη σφάλματος ώστε να λειτουργήσει ο συγκεκριμένος ελεγκτής, ποτέ δε μηδενίζει πλήρως το σφάλμα (εικόνα 3.20).



Σχήμα 3.20: Επίδραση Όρου Kp

Οπότε από τα παραπάνω συμπεραίνεται ότι πρέπει να βρεθεί η κατάλληλη τιμή του κέρδους K_p ώστε να ελαχιστοποιείται το σφάλμα. Στην παρούσα εργασία, η χρήση αποκλειστικά ενός αναλογικού όρου φαίνεται ιδανική. Αυτό συμβαίνει αφενός διότι δεν απαιτείται ο απόλυτος μηδενισμός του σφάλματος θέσης και γωνίας και αφετέρου χάρη στην απλότητα του υπολογισμού του σε κάθε βήμα.

Πριν εξηγηθεί πώς υλοποιείται ο παραπάνω έλεγχος, γίνονται κάποιες παραδοχές σχετικά με τον ορισμό του σφάλματος στην παρούσα εργασία. Γίνεται η σύμβαση ότι το σφάλμα θέσης ελαχιστοποιείται όταν ικανοποιεί τη συνθήκη:

$$ErrorPose \leq 3mm$$

Η μεταβλητή $ErrorPose$ προκύπτει ως το μέσο σφάλμα ευκλείδιας απόστασης από τον τύπο:

$$ErrorPose = \frac{\sum_{i=1}^{N} \sqrt{({x_d}^i - {x_a}^i)^2 + ({y_d}^i - {y_a}^i)^2 + ({z_d}^i - {z_a}^i)^2}}{N}$$

όπου ${x_d}^i, {y_d}^i, {z_d}^i$ και ${x_a}^i, {y_a}^i, {z_a}^i$ οι συντεταγμένες των επιθυμητών και των πραγματικών σημείων αντίστοιχα και N ο συνολικός αριθμός των παραδειγμάτων που χρησιμοποιήθηκαν.

Όσον αφορά στο σφάλμα προσανατολισμού, γίνεται η παραδοχή ότι ελαχιστοποιείται όταν ικανοποιεί τη συνθήκη:

$$ErrorOrient \leq 0.5°$$

το οποίο προκύπτει ως η διαφορά πραγματικής και επιθυμητής γωνίας θ_4.

Τώρα που έχει ορισθεί και το σφάλμα, εξηγείται η διαδικασία του ελέγχου η οποία έχει ως εξής. Πρώτα, εφαρμόζεται έλεγχος στον προσανατολισμό του end effector. Αυτό συμβαίνει επειδή η γωνία θ_4 (γωνία

περιστροφικού κινητήρα σταθερής βάσης) επηρεάζει και τη θέση και τον προσανατολισμό. Αφενός, ο προσανατολισμός ταυτίζεται με τη γωνία θ_4 και αφετέρου, όπως αποδείχθηκε στο πρώτο κεφάλαιο, ισχύει η παρακάτω σχέση που δείχνει την άμεση επιρροή της γωνίας θ_4 στη θέση του end effector.

$$\begin{bmatrix} x'_e \\ y'_e \\ z'_e \\ 1 \end{bmatrix} = \begin{bmatrix} \cos\theta_4 & -\sin\theta_4 & 0 & 0 \\ \sin\theta_4 & \cos\theta_4 & 0 & 0 \\ 0 & 0 & 1 & 59.74 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_e \\ y_e \\ z_e \\ 1 \end{bmatrix}$$

όπου x'_e, y'_e, z'_e οι συντεταγμένες του τελικού σημείου. Σε περίπτωση που το σφάλμα στον προσανατολισμό δεν ικανοποιεί τη συνθήκη $ErrorOrient \leq 0.5°$ τότε εφαρμόζεται αναλογικός έλεγχος. Η κάμερα λαμβάνει την πραγματική γωνία θ_4 από τον προσανατολισμό και ο Η/Υ, αφού γνωρίζει την επιθυμητή γωνία, εφαρμόζει έλεγχο με συντελεστή K_p. Η διαδικασία επαναλαμβάνεται έως ότου να ικανοποιείται η παραπάνω ανισότητα.

Έπειτα, εφαρμόζεται έλεγχος και στη θέση του end effector με τον παρακάτω τρόπο. Ο έλεγχος θέσης σε αντίθεση με τον προηγούμενο σχετίζεται με τους 3 κινητήρες που σχηματίζουν το Delta Robot και οι οποίοι δεν επηρεάζουν τον προσανατολισμό. Η κάμερα λαμβάνει τα δεδομένα της πραγματικής θέσης. Εάν δεν ικανοποιείται η ανισότητα για το $ErrorPose$, τότε εφαρμόζεται αναλογικός έλεγχος ώστε ο end effector να κινηθεί από τη δεδομένη πραγματική θέση προς την επιθυμητή. Η διαδικασία επαναλαμβάνεται έως ότου να ισχύσει η σχέση για το $ErrorPose$.

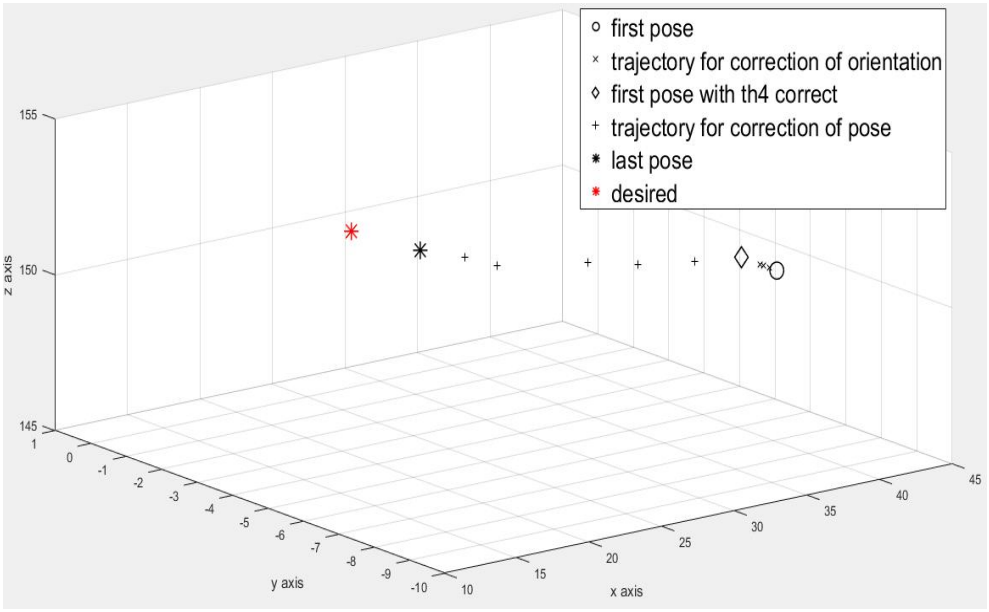
Συνοψίζοντας, εφαρμόζονται δύο ανεξάρτητοι αναλογικοί έλεγχοι. Η σειρά εφαρμογής τους είναι σημαντική διότι ο έλεγχος προσανατολισμού επηρεάζει και τη θέση, ενώ το αντίστροφο δε συμβαίνει. Για όλα τα παραπάνω γίνεται μια προσπάθεια επεξήγησής τους μέσω παραδείγματος.

Ειδικότερα, δίνεται η εντολή στο ρομπότ να πάει στο σημείο (28mm, 0, 150mm) του χώρου εργασίας με γωνία κινητήρα βάσης $\theta_4 = 30°$. Αρχικά, διορθώνεται ο προσανατολισμός του end effector, ο οποίος βρίσκεται στις $31.1°$ και σταδιακά φτάνει τις $30.4°$ μέσω της εφαρμογής ελέγχου. Αυτό έχει ως αποτέλεσμα να επηρεάζεται και η θέση του, όπως φαίνεται στα σχήματα 3.21, 3.22. Έπειτα, αφού διορθωθεί ο προσανατολισμός, εφαρμόζεται έλεγχος θέσης ο οποίος διορθώνοντας τις γωνίες των 3 υπόλοιπων κινητήρων ελαχιστοποιεί το σφάλμα θέσης.

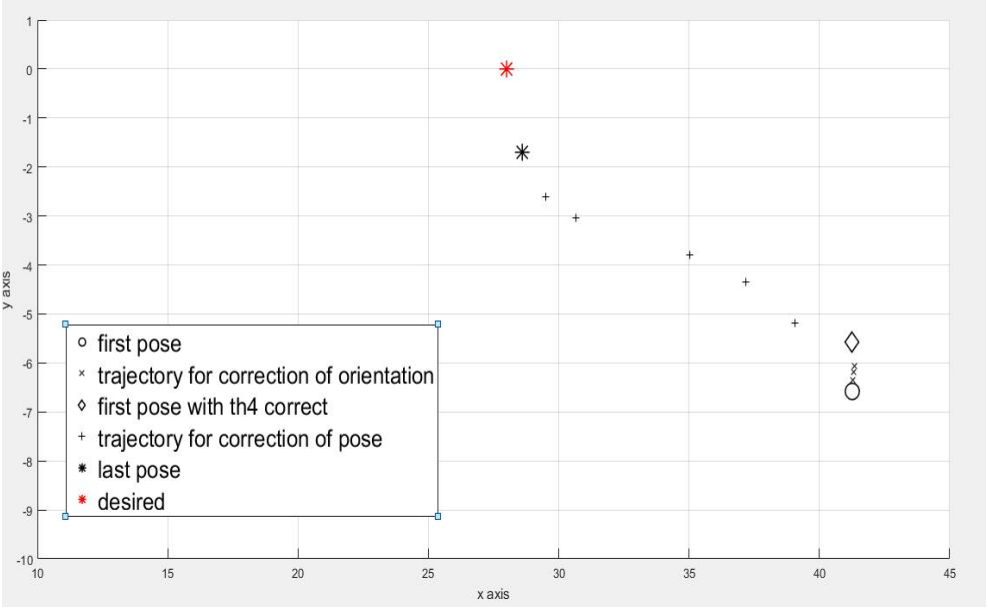
×	Αρχικό	Τελικό
Σφ. Θέσης	$15.2mm$	$1.432mm$
Σφ. Προσανατολισμού	$1.1°$	$0.4°$

Πίνακας 3.7: Σφάλματα Τροχιάς

Στη συγκεκριμένη περίπτωση δείχνεται η τροχιά που ακολούθησε το ρομπότ από το αρχικό σημείο μέχρι το τελικό. Για την παρακάτω τροχιά



Σχήμα 3.21: 3D Απεικόνιση Τροχιάς Ελέγχου



Σχήμα 3.22: Δισδιάστατη Απεικόνιση Τροχιάς Ελέγχου

χρησιμοποιήθηκε αναλογικός ελεγκτής μικρού κέρδους ($K_p = 0.2$) αμφότερα για τον έλεγχο προσανατολισμού και θέσης. Ο πίνακας 3.7 καταγράφει τα αρχικά και τελικά σφάλματα. Η μείωση τους είναι εμφανής.

3.7 Πείραμα

Πραγματοποιήθηκε διαδικασία όμοια με την παραπάνω ώστε να εκτελεστούν ορισμένες τροχιές. Τα πειράματα περιλαμβάνουν δύο γραμμικές τροχιές και δύο κυκλικές. Σε κάθε περίπτωση έχουμε δύο τροχιές, διότι στη μεν πρώτη δείχνεται η κίνηση του ρομπότ χωρίς τη χρήση του τέταρτου κινητήρα ($\theta_4 = 0°$) ενώ στη δεύτερη επιλέγονται τυχαίοι προσανατολισμοί του εργαλείου τελικής δράσης (δηλαδή $\theta_4 \neq$ σταθ).

Όσον αφορά στον έλεγχο, χρησιμοποιήθηκε αναλογικός ελεγκτής P με κέρδος $K_p = 0.5$ με τον τρόπο που εξηγήθηκε στην προηγούμενη ενότητα και για τους δύο ελέγχους. Λόγω της μικρής ταχύτητας και επιτάχυνσης που επιλέχθηκε για το ρομπότ, η μετάβαση από σημείο σε σημείο επιλέχθηκε να διαρκεί $5sec$ ($2sec$ για τον έλεγχο προσανατολισμού και $3sec$ για τον έλεγχο θέσης).

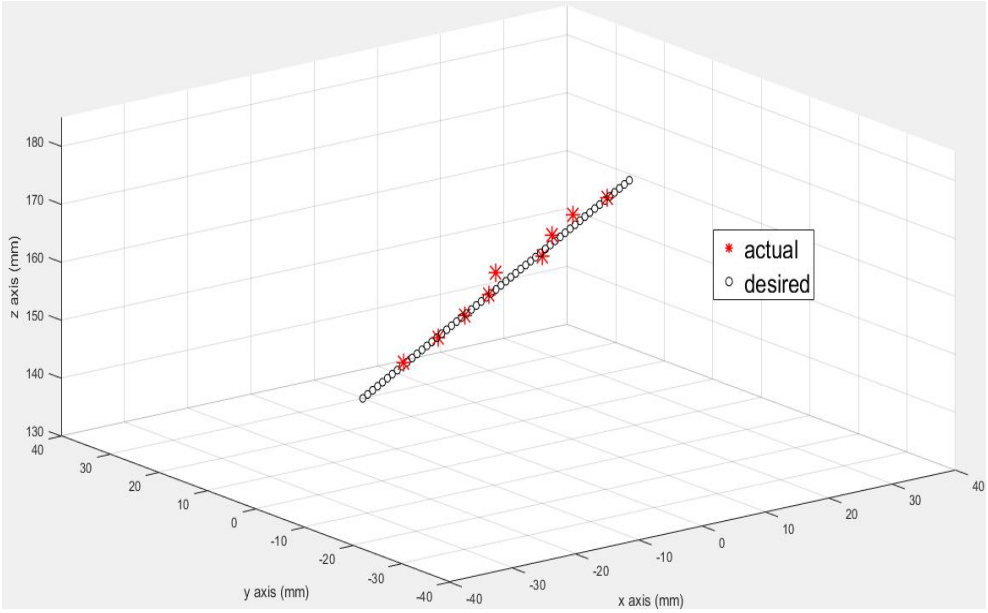
3.7.1 Γραμμική Τροχιά

Η πρώτη τροχιά που ακολούθησε το ρομπότ είναι γραμμική, η οποία μάλιστα βρίσκεται πάνω στην ευθεία $y + z = 160$ και αποτελείται από 9 ισαπέχοντα σημεία στο χώρο. Πραγματοποιήθηκαν δύο δοκιμές, όπου στην πρώτη επιλέχθηκε η γωνία του περιστροφικού κινητήρα να είναι μηδενική, ενώ στη δεύτερη να μεταβάλλεται μεταξύ ορισμένων τιμών. Τα αποτελέσματα των μετρήσεων συγκεντρώνονται στις εικόνες 3.23, 3.24 και 3.25, 3.26. Διευκρινίζεται ότι δίπλα σε κάθε σημείο αναγράφεται η πραγματική γωνία θ_4 (προσανατολισμός) ύστερα από την εφαρμογή ελέγχου.

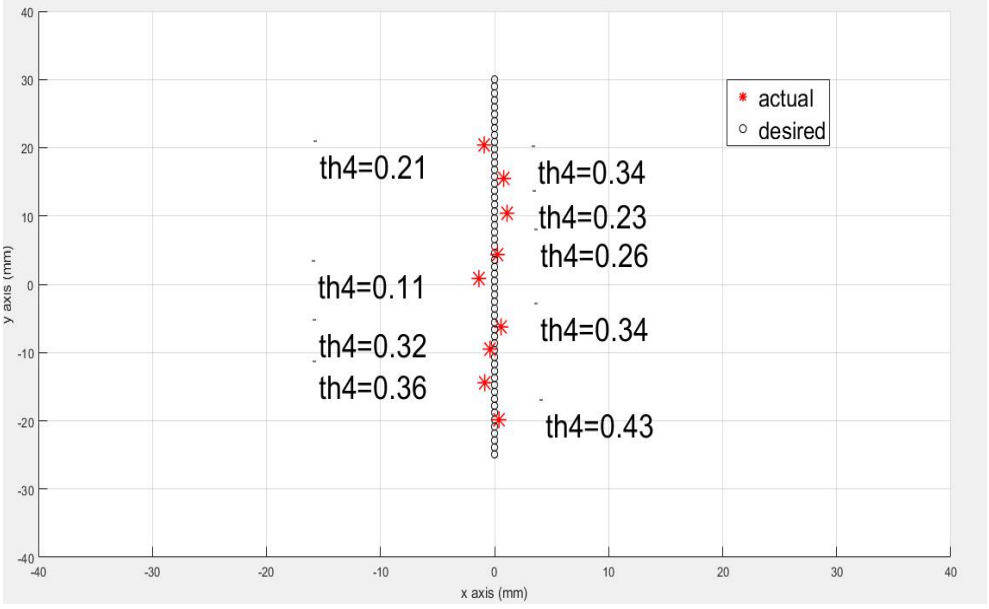
Όπως αναφέρθηκε νωρίτερα, η μετάβαση από σημείο σε σημείο διήρκησε $5sec$. Στον πίνακα 3.8 δείχνονται και τα αντίστοιχα σφάλματα κάθε τροχιάς. Μία μείωση του χρόνου μετάβασης θα είχε επιπτώσεις στα σφάλματα αυξάνοντάς τα εξαιτίας των λιγότερων ανατροφοδοτήσεων που θα πραγματοποιούνταν.

θ_4	ErrorPose	ErrorOrient
0°	1.914	0.29°
Μεταβαλλόμενο (30°, 45°, 135°)	2.211	0.28°

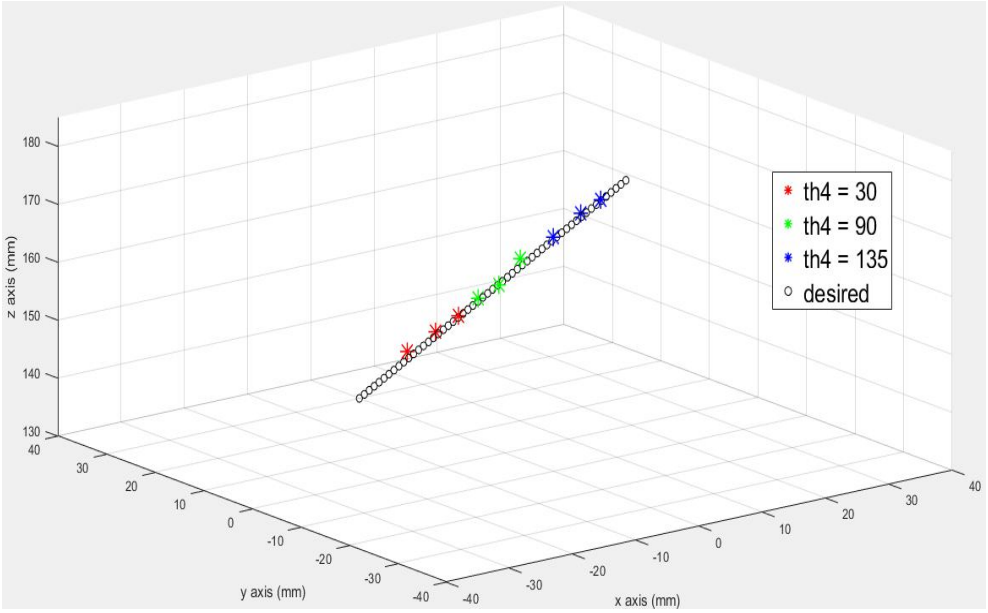
Πίνακας 3.8: Σφάλματα Γραμμικής Τροχιάς



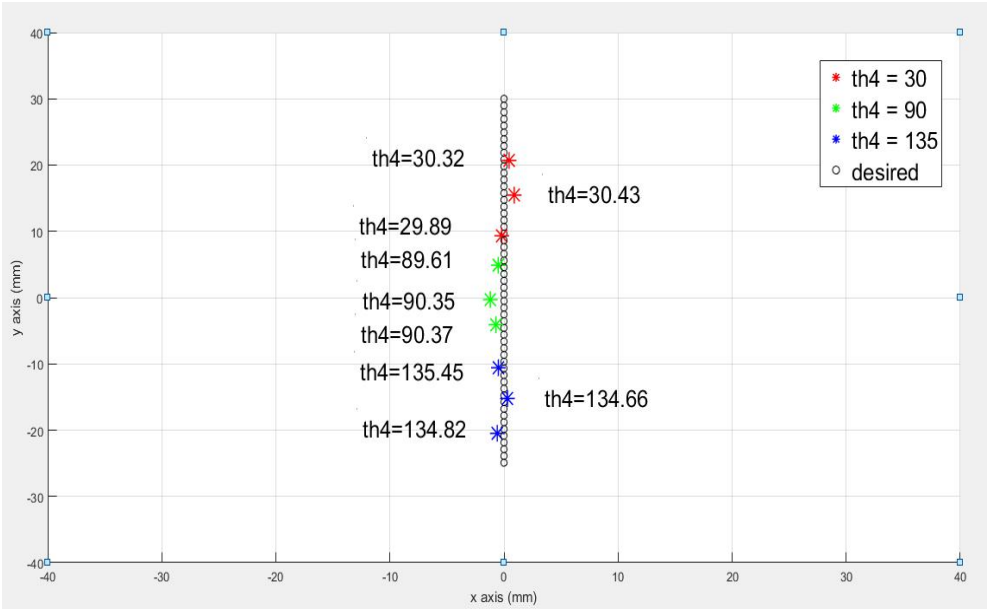
Σχήμα 3.23: 3D Απεικόνιση Γραμμικής Τροχιάς $\theta_4 = 0°$



Σχήμα 3.24: 2D Απεικόνιση Γραμμικής Τροχιάς $\theta_4 = 0°$

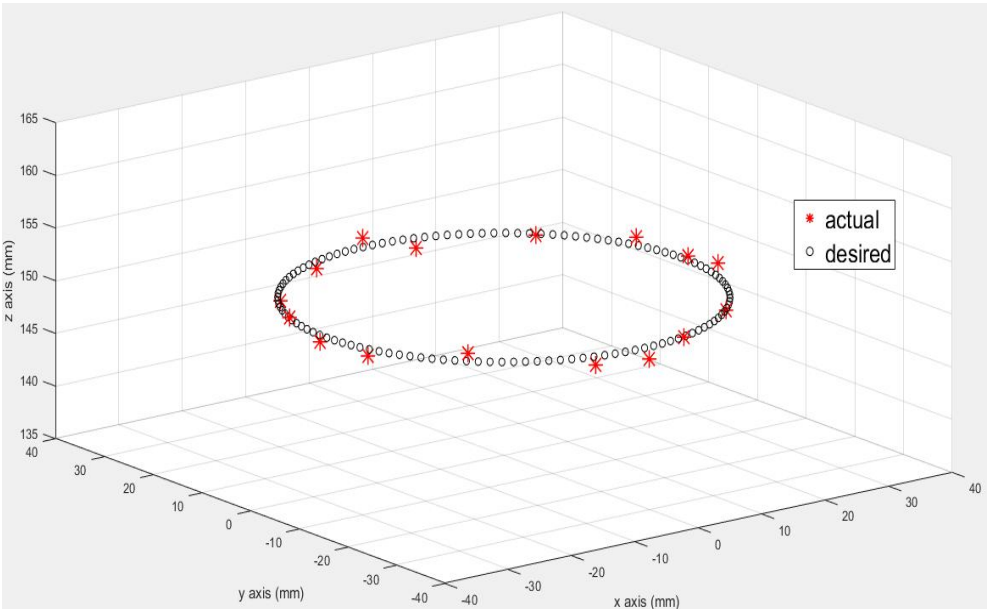


Σχήμα 3.25: 3D Απεικόνιση Γραμμικής Τροχιάς $\theta_4 \neq$ σταθ

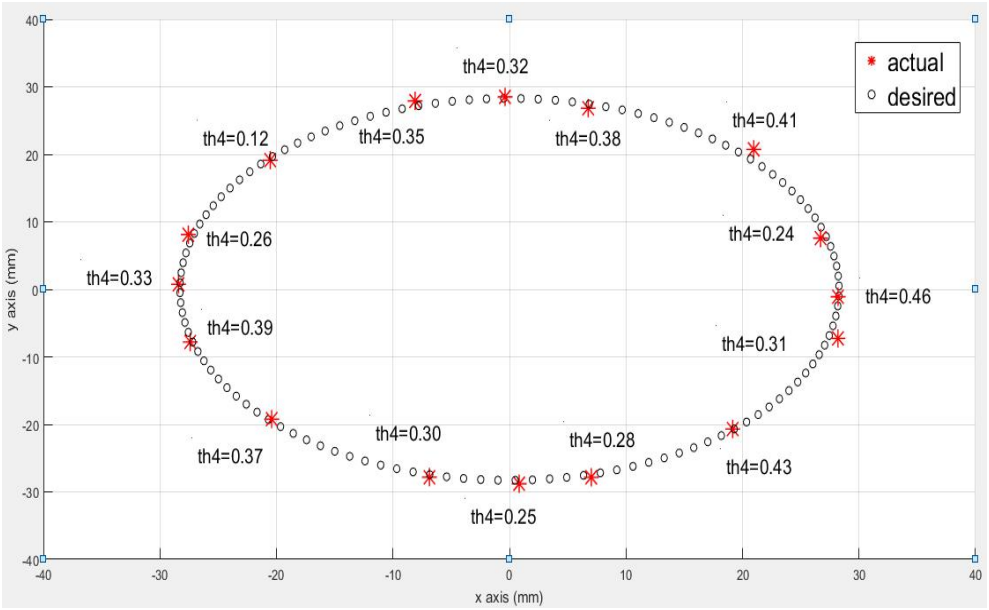


Σχήμα 3.26: 2D Απεικόνιση Γραμμικής Τροχιάς $\theta_4 \neq$ σταθ

3.7.2 Κυκλική Τροχιά



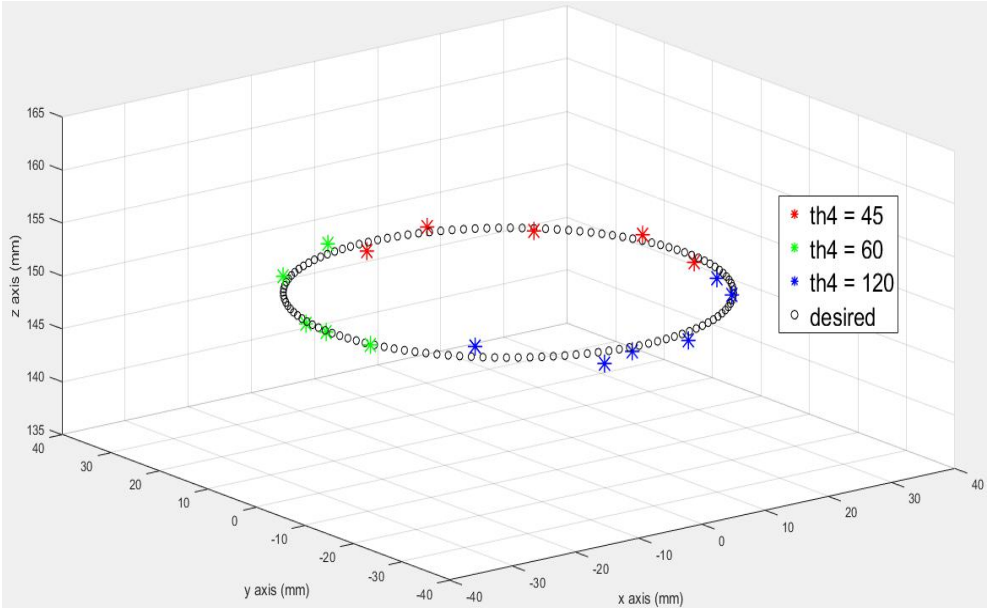
Σχήμα 3.27: 3D Απεικόνιση Κυκλικής Τροχιάς $\theta_4 = 0°$



Σχήμα 3.28: 2D Απεικόνιση Κυκλικής Τροχιάς $\theta_4 = 0°$

Στο πλαίσιο της πειραματικής διαδικασίας πραγματοποιήθηκαν δύο τροχιές ακόμα, αυτή τη φορά κυκλικές. Όπως και νωρίτερα, η μία περιλαμβάνει την εμπλοκή του τέταρτου περιστροφικού κινητήρα ώστε

να δειχθεί κατά ποιο τρόπο επηρεάζει την κίνηση. Ειδικότερα, μετρήσεις πάρθηκαν στα σημεία των $0°, 20°, 45°$ και $70°$ σε κάθε τεταρτοκύκλιο. Στον πίνακα 3.9 παρατίθενται τα σφάλματα σε κάθε περίπτωση και στις εικόνες 3.27, 3.28 και 3.29, 3.30 φαίνονται οι δύο τροχιές που εκτέλεσε το ρομπότ. Διευκρίνιζεται ότι όπως και πριν, δίπλα σε κάθε σημείο αναγράφεται η πραγματική τελική γωνίαθ_4 (προσανατολισμός) ύστερα από την εφαρμογή ελέγχου



Σχήμα 3.29: 3D Απεικόνιση Κυκλικής Τροχιάς $\theta_4 \neq$ σταθ

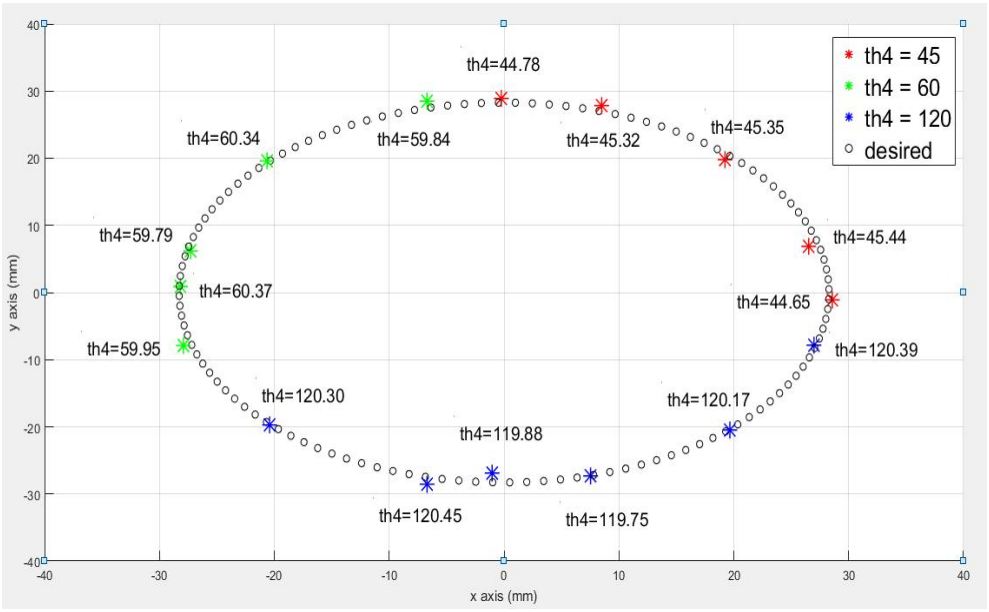
θ_4	ErrorPose	ErrorOrient
$0°$	1.860	$0.325°$
Μεταβαλλόμενο ($30°, 45°, 135°$)	2.351	$0.28°$

Πίνακας 3.9: Σφάλματα Κυκλικής Τροχιάς

Παρατηρήσεις

Στο σημείο αυτό θα γίνουν ορισμένες παρατηρήσεις. Αρχικά, τα σφάλματα οφείλονται σε μία πληθώρα αιτιών. Αυτές είναι:

- Σφάλματα κατά τον υπολογισμό των κινηματικών εξισώσεων. Αποκλίσεις τάξης μεγέθους μικρότερης του ενός χιλιοστού που υπάρχουν στην κατασκευή μπορούν να επιφέρουν αλλοίωση των αποτελεσμάτων της κινηματικής,
- Σφάλματα σε οτιδήποτε αφορά την κάμερα. Εδώ συμπεριλαμβάνεται η πιθανώς λανθασμένη τοποθέτησή της (πχ υπό γωνία και όχι



Σχήμα 3.30: 2D Απεικόνιση Κυκλικής Τροχιάς $\theta_4 \neq$ σταθ

παράλληλα με την κύρια βάση του ρομπότ) καθώς και μία μικρή απόκλιση στον υπολογισμό των εγγενών παραμέτρων της (σφάλμα κατά τη βαθμονόμηση).

- Τέλος, οι σερβοκινητήρες μπορεί να παρουσιάσουν απόκλιση στις γωνίες επιθυμητής και πραγματικής θέσης (μερικά δέκατα μιας μοίρας).

Η σημαντικότερη παρατήρηση ωστόσο έχει να κάνει με τη εισαγωγή μεταβαλλόμενης γωνίας στον περιστροφικό κινητήρα ($\theta_4 \neq$ σταθ). Παρότι τα τελικά αποτελέσματα των παραπάνω πειραμάτων δίνουν ικανοποιητικό σφάλμα σε όλες τις περιστάσεις, μία μεταβολή του θ_4 προκαλούσε μεγαλύτερη μεταβολή στη θέση του κινητήρα σε σχέση με όταν το θ_4 παρέμενε σταθερό. Παρά ταύτα, όμως, ο επαρκής χρόνος μετάβασης επέτρεπε την ελαχιστοποίηση των σφαλμάτων γωνίας και θέσης.

3.8 Σύνοψη Κεφαλαίου

Στο παρόν κεφάλαιο, αρχικά, εξηγήθηκε ο τρόπος κατασκευής όλων των μηχανολογικών εξαρτημάτων και ο ρόλος του καθενός. Εν συνεχεία, παρουσιάστηκαν οι συσκευές κίνησης (σερβοκινητήρες και συσκευή phidgets), η συσκευή ελέγχου (κάμερα) και πώς αυτή έλεγξε το σύστημα. Τέλος, παρουσιάστηκαν ορισμένα πειράματα που στην ουσία πρόκειται για τροχιές κίνησης του ρομπότ και εφαρμόστηκε έλεγχος ανατροφοδότης γωνίας και θέσης.

ΚΕΦΑΛΑΙΟ

4

ΣΥΜΠΕΡΑΣΜΑΤΑ

4.1 Σύνοψη

Στην παρούσα εργασία κατασκευάστηκε ένας παράλληλος ρομποτικός μηχανισμός τεσσάρων βαθμών ελευθερίας τύπου Δέλτα με δυνατότητα ελέγχου του προσανατολισμού του εργαλείου τελικής δράσεως. Το ρομπότ αυτό πληροί ορισμένες προδιαγραφές όπως είναι το βάρος που δεν υπερβαίνει το μισό κιλό και ο χώρος εργασίας που είναι μεγαλύτερος από $120cm^3$. Επίσης, αναπτύχθηκε ένα σχήμα ελέγχου οπτικής ανατροφοδότησης για το μηδενισμό του σφάλματος θέσης. Σε αυτό το πλαίσιο, χρησιμοποιήθηκε ένας στόχος (fiducial marker) και μία κάμερα, ο συνδυασμός των οποίων έδινε χρήσιμες γεωμετρικές πληροφορίες. Τέλος, αξιοποιήθηκαν οι παραπάνω γεωμετρικές πληροφορίες μέσω ενός P ελέγχου.

4.2 Προτεινόμενες αλλαγές

Σίγουρα η συγκεκριμένη εργασία έφτασε εις πέρας και αποτελεί ολοκληρωμένο έργο. Ωστόσο, υπάρχουν τομείς της με τους οποίους θα μπορούσε κάποιος να ασχοληθεί ώστε να τη βελτιώσει.

Το μεγάλο μειονέκτημα της κατασκευής είναι ο μικρός χώρος εργασίας του ρομπότ παρότι ικανοποιεί τους αρχικούς στόχους που ετέθησαν. Μία ιδέα για την αντιμετώπιση αυτού του προβλήματος είναι η επιμήκυνση κατά μερικά εκατοστά των κάτω βραχιόνων, των εξαρτημάτων δηλαδή που κατασκευάστηκαν στο μηχανουργείο. Αυτό θα μεγάλωνε το workspace αν και παράλληλα θα πρόσθετε βάρος στην κατασκευή και

ίσως να επιβάρυνε τους κινητήρες, πράγμα που μας οδηγεί σε μία δεύτερη βελτίωση.

Η τοποθέτηση δυνατότερων κινητήρων, που παράλληλα δε θα πρόσθεταν σημαντικό βάρος, με την ταυτόχρονη επικήκυνση των κάτω βραχιόνων θα είχαν θετική επίδραση στην αύξηση του χώρου εργασίας χωρίς να επηρεάζουν αρνητικά την κατασκευή.

Εξάλλου, παρατηρήθηκε ότι η κινούμενη πλατφόρμα (end effector) τρεμόπαιζε κατά την κίνηση. Για το λόγο προτείνεται η εκ νέου σχεδίαση και κατασκευή των κάτω βραχιόνων ώστε να αποφευχθεί αυτό το πρόβλημα.

Τέλος, η κάμερα παρατηρήθηκε πολλές φορές ότι δυσκολευόταν να εντοπίσει το στόχο. Αυτό συνέβαινε όταν είτε ο στόχος βρισκόταν πολύ κοντά είτε δεν υπήρχε κατάλληλος φωτισμός (πολύ έντονος ή το αντίθετο) στο χώρο. Οπότε μία αναβάθμιση της εργασίας αυτής θα αποτελούταν από άλλη καμέρα η οποία θα διέθετε καλύτερη εστίαση σε κοντινές αποστάσεις και θα αντιμετώπιζε καλύτερα χώρους με πολύ μικρή ή πολύ μεγάλη φωτεινότητα.

4.3 Μελλοντική Έρευνα

Παρότι η εργασία ολοκληρώθηκε, δεν είναι έτοιμη ακόμα για εμπορική αξιοποίηση. Η πρώτη σκέψη είναι ότι να τοποθετηθεί στη θέση του end effector μία αρπάγη (gripper) ώστε το συνολικό σύστημα να εξυπηρετεί pick 'n' place operations.

Όσον αφορά στην τοποθέτηση του ρομπότ σε ιπτάμενο όχημα, σίγουρα θα έπρεπε η πάνω βάση να αντικατασταθεί με άλλη ώστε να είναι συμβατή με την επιφάνεια του οχήματος αυτού.

Επίσης, μία άλλη εφαρμογή είναι η τοποθέτηση μιας κάμερας πάλι στη θέση του end effector. Η ύπαρξη μιας κάμερας σε συνδυασμό με την τοποθέτηση του ρομπότ σε ιπτάμενο όχημα θα έβρισκε εφαρμογές στην εξερεύνηση χώρου.

Τέλος, θα μπορούσε να πραγματοποιηθεί κινηματική ανάλυση στο ρομπότ σε περιπτώσεις όπου οι βάσεις δεν είναι παράλληλες με το έδαφος. Σε αυτές, το κέντρο βάρους μεταβάλλεται και το ρομπότ συμπεριφέρεται διαφορετικά.

ΠΑΡΑΡΤΗΜΑ

Α΄

ΛΟΓΙΣΜΙΚΑ ΕΡΓΑΣΙΑΣ

Το αναπτυχθέν λογισμικό, αναφορικά με τη σχεδίαση σε SolidWorks και την υλοποίηση των ελεγκτών σε περιβάλλον ROS μπορεί να βρεθεί στο `https://github.com/UPatras-ANeMoS/micro_Delta_Robot`.

ΒΙΒΛΙΟΓΡΑΦΙΑ

[1] https://en.wikipedia.org/wiki/Parallel_manipulator

[2] https://en.wikipedia.org/wiki/Stewart_platform

[3] https://en.wikipedia.org/wiki/Delta_robot

[4] Hexapod Surgery Assistant: Servo Motor Applications https://www.micromo.com/applications/medical-lab-automation-equipment/application-case-study-micro-precise-surgery-assistant

[5] Clavel R., "Conception D'un Robot Parallele Rapide a 4 Degres De Libertes", Ph.D. dissertation, Ecole Polytechnique Federale De Lausanne, Lausanne, 1991

[6] Delta Robot Kinematics, Trossen Robotics http://forums.trossenrobotics.com/tutorials/introduction-129/delta-robot-kinematics-3276/

[7] Hillier, N., Martin, S. (2009). "Characterisation of the Novint Falcon Haptic Device for Application as a Robot Manipulator", *in Australasian Conference on Robotics and Automation*, December 2-4, 2009

[8] http://www.peterkovesi.com/matlabfns/Rotations/invht.m

[9] CORKE, P. (2017) *ROBOTICS VISION AND CONTROL*. [S.I.]: SPRINGER

[10] https://en.wikipedia.org/wiki/Denavit%E2%80%93Hartenberg_parameters

[11] A. Gharahsofloo and A. Rahmani. "An Efficient Algorithm for Workspace Generation of Delta Robot". *International Journal of Robotics*, Vol. 5, No.2, pp. 48-53, 2015.

[12] Andrioaia D., Pascu M., Mihaila L. and Obrea C. F. "Determing the Workspace in case of the Robots with Parallel Structure Delta 3DOF", *in* $DAAAM\ INTERNATIONAL$, Vol. 23, No.1, pp. 1023-1026, 2012.

[13] Trucco E. and Verri A., Introductory Techniques for 3-D Computer Vision, Prentice Hall PTR, NJ 1998.

[14] http://wiki.ros.org/

[15] http://www.phidgets.com/products.php?category=11&product_id=1061_1

[16] http://hitecrcd.com/products/servos/sport-servos/analog-sport-servos/hs-422-deluxe-standard-servo/product

[17] https://en.wikipedia.org/wiki/PID_controller

Πανεπιστήμιο Πατρών, Πολυτεχνική Σχολή
Τμήμα Ηλεκτρολόγων Μηχανικών και Τεχνολογίας Υπολογιστών
Αργύριος Ματσούκας
© Μάιος 2017 – Με την επιφύλαξη παντός δικαιώματος.