

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

**ΣΧΕΔΙΑΣΗ ΚΑΙ OFF-LINE ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΣ ΡΟΜΠΟΤΙΚΟΥ
ΒΡΑΧΙΟΝΑ ΜΕ ΧΡΗΣΗ ΤΟΥ ΛΟΓΙΣΜΙΚΟΥ SOLIDWORKS**

**DESIGN AND OFF-LINE PROGRAMMING OF A ROBOTIC ARM
USING SOLIDWORKS SOFTWARE**

ΕΚΠΟΝΗΤΗΣ : ΝΙΚΟΠΟΥΛΟΣ ΣΤΑΜΑΤΗΣ

ΕΠΙΒΛΕΠΩΝ : Επίκ.Καθηγητής ΑΠΟΣΤΟΛΟΣ ΤΣΑΓΚΑΡΗΣ

ΘΕΣΣΑΛΟΝΙΚΗ 2017

Ευχαριστίες

Για το πέρας της παρούσας πτυχιακής εργασίας χρειάστηκε αρκετός χρόνος, υπομονή αλλά και εισαγωγή σε νέα πράγματα και έννοιες για το λόγω αυτό θα ήθελα να ευχαριστήσω θερμά τον καθηγητή κύριο Απόστολο Τσαγκάρη τόσο για το ενδιαφέρον θέμα που μου ανάθεσε όσο και για την καθοδήγηση και την βοήθεια που μου παρείχε καθ' όλη τη διάρκεια της υλοποίησης της πτυχιακής εργασίας μου.

Επίσης θα ήθελα να ευχαριστήσω τον φίλο και συνάδερφο Νικόλαο Μουστάκα για την βοήθεια του όλα τα χρόνια σπουδών μου στο τμήμα Μηχανικών Αυτοματισμού αλλά και την για την έμπνευση που μου έδωσε η οποία με οδήγησε να ασχοληθώ με ένα τέτοιο δύσκολο θέμα και με την ρομποτική γενικά.

Τέλος θα ήθελα να ευχαριστήσω τους γονείς μου για την υποστήριξη τους με οποιοδήποτε τρόπο έως ολοκλήρωση των προπτυχιακών μου σπουδών.

ΠΕΡΙΛΗΨΗ

Η αναπτυχθείσα εφαρμογή χρησιμοποιήθηκε ως εκπαιδευτικό εργαλείο για να ενισχύσει τόσο την εφαρμοζόμενη όσο και την πειραματική έρευνα πάνω σε θέματα της ρομποτικής επιστήμης επίσης προσφέροντας νέες ιδέες και ευκαιρίες για την ανάπτυξη και την βελτίωση διαφόρων θεμάτων ρομποτικής σε προπτυχιακό και μεταπτυχιακό επίπεδο.

Η εργασία αυτή έχει ως σκοπό την ανάπτυξη μιας εφαρμογής για τον off-line προγραμματισμό ένας εμπορικού ρομποτικού βραχίονα ο οποίος έχει ληφθεί ως μελέτη, δίνοντας τον χρήστη την ευκολία να εκτελέσει διαφορές κινήσεις τον προγραμματισμό μιας διαδικασίας. Χρησιμοποιώντας το λογισμικό SolidWorks σχεδιάστηκε το ρομπότ σε εικονικό περιβάλλον μεταφέροντας το αυτόσιο από την πραγματικότητα, και για την διεπαφή του με τον χρήστη χρησιμοποιήσαμε το λογισμικό LabVIEW ως εργαλείο για την ανάπτυξη και τη δοκιμή των κινητικών χαρακτηριστικών του ρομποτικού βραχίονα.

Ως εκ τούτου, το παρών θέμα στις μέρες μας είναι υψίστης σημασίας σημαντικό ως ένα ρομποτικό λογισμικό προσομοίωσης το οποίο αναλύσει και δίνει μεγάλη κατανόηση στους αλγόριθμους του ευθέως και αντίστροφου κινηματικού προβλήματος που είναι η βάση για ένα ρομποτικό σύστημα.

ABSTRACT

The developed application used as an educational tool to enhance both the applied and experimental research on issues of robotics science also offering new ideas and opportunities for the development and improvement of various robotics subjects at undergraduate and postgraduate level.

This study aims to develop an application for off-line programming of a commercial robotic arm which is taken as a study, giving the user an ease to perform different movements scheduling a procedure. Using SolidWorks software we designed the robot in a virtual environment by transferring it unchanged from reality, and to interface with the user we used the LabVIEW software as a tool for development and testing the kinetic characteristics of the robotic arm.

Therefore, the present issue nowadays is paramount important as a robot simulation software that analyze and gives great understanding to the algorithms of direct and inverse kinematic problem which is the basis for a robotic system.

Περιεχόμενα

Ευχαριστίες

Περίληψη

Abstract

1.Εισαγωγή.....σελίδα	
1.1 Γενικά	1
1.2 Ορισμός των ρομπότ	2
1.3 Δομή των ρομπότ σελίδα	4
1.3.1 Βαθμός ελευθερίας	5
1.3.2 Χώρος εργασίας –φάκελος εργασίας	6
1.4 Είδη Αρθρώσεων	7
1.5 Διεπαφές και μέθοδοι προγραμματισμού των ρομπότ	8
2.ΘεωρητικόΥπόβαθρο.....σελίδα	
2.1 Κινηματική ανάλυση	11
2.2 Ευθεία κινηματική ανάλυση	
2.2.1 Μητρώα Μετασχηματισμού	12
2.2.2 Παράμετροι Denavit-Hartenberg (D-H)	13
2.3 Αντίστροφη κινηματική ανάλυση	
2.3.1 Γεωμετρική Προσέγγιση	16
2.3.2 Αναλυτική αλγεβρική προσέγγιση	18
3.Μοντελοποίηση.....σελίδα	
3.1 Γενικά	23
3.2 Σχεδίαση αρπάγης	25
3.2.1 Βάση αρπάγης	26
3.2.2 Μηχανισμός κίνησης	27
3.2.3 Λαβή-Δαγκάνα	28
3.2.4 Στήριξη	29
3.3 Συναρμολόγηση αρπάγης	31
3.4 Σχεδίαση σέρβο κινητήρα ρομποτικού βραχίονα	41
3.5 Συναρμολόγηση ρομποτικού βραχίονα	47
4.Αυτοματοποίηση Διαδικασίας.....σελίδα	
4.1 Το πρόγραμμα LabVIEW	54
4.2 Δομή του περιβάλλοντος προγραμματισμού	56
4.2.1 Τα κυριότερα στοιχεία του LabVIEW	58
4.3 Επικοινωνία μεταξύ Λογισμικών SolidWorks LabVIEW & NI Soft Motion	63
4.3.1 Εισαγωγή και επικοινωνία με το μοντέλο του SolidWorks	66
4.4 Φόρμες του προγράμματος	

4.4.1 Φόρμα ευθείας κινηματικής του ρομποτικού βραχίονα	69
4.4.2 Φόρμα αντίστροφης κινηματικής του ρομποτικού βραχίονα	70
5.Εφαρμογή.....σελίδα	
5.1 Εκτέλεση Προγράμματος	71
5.2 Εξαγωγή Εκτελέσιμου Κώδικα	75
6.Αποτελέσματα-Συμπεράσματα.....σελίδα	
	79
7.Βιβλιογραφία-Αναφορές.....σελίδα	
	80
8.Παράρτημα.....σελίδα	
	81

1.Εισαγωγή

1.1 Γενικά

Η Ρομποτική είναι εκείνος ο κλάδος της επιστήμης του μηχανικού, που ασχολείται με τη σύλληψη, τη σχεδίαση, την κατασκευή, τη θεωρία και τις εφαρμογές των ρομπότ. Είναι γεγονός ότι η ρομποτική ωφελείται από τις εξελίξεις σε αρκετούς παραδοσιακούς κλάδους, όπως είναι η ηλεκτρολογία, η μηχανολογία και τα μαθηματικά. Με τη σειρά τους οι κλάδοι αυτοί ωφελούνται από την εξέλιξη της ,ειδικότερα , είναι σημαντικό να αναφερθούν οι ερευνητικές δραστηριότητες με στόχο την ανάπτυξη εξελιγμένων αλγορίθμων ελέγχου, καθώς και οι δραστηριότητες που σχετίζονται με την εξέλιξη συστημάτων τεχνητής νοημοσύνης.

Με μια πρόταση δηλαδή μπορούμε να ορίσουμε τη ρομποτική μια «διεπιστημονική περιοχή» που συνδυάζει τη φυσική, την ηλεκτρολογία, τη μηχανολογία, την πληροφορική, τις τηλεπικοινωνίες, τη θεωρία συστημάτων, τον αυτόματο έλεγχο, την τεχνητή νοημοσύνη, την τεχνολογία των αισθητήριων διατάξεων (δύναμης, αφής, όρασης, κλπ), την εικονική πραγματικότητα, την επεξεργασία σήματος, την υπολογιστική όραση και την τεχνητή ζωή.

Η ρομποτική αποτελεί αυτοδύναμο τεχνολογικό κλάδο και ανήκει στον ευέλικτο αυτοματισμό και είναι μια τεχνολογία με μέλλον και για το μέλλον. Η ανάπτυξη της ξεκίνησε με τον εμπλουτισμό των χωρικών μηχανισμών με επενεργητές και αργότερα με αισθητήρες και "εγκέφαλο" (συνήθως ηλεκτρονικός υπολογιστής (H/Y)). Ο "εγκέφαλος" αξιοποιεί τις πληροφορίες των αισθητήρων για να δώσει κατάλληλες εντολές στους επενεργητές, ώστε ο χωρικός μηχανισμός να εκτελέσει επιθυμητές εργασίες. Η έρευνα στην περιοχή της ρομποτικής εκτείνεται κυρίως σε τρεις κατευθύνσεις. Η πρώτη αφορά στην εφαρμογή και την ανάπτυξη τεχνικών ελέγχου για την βελτίωση της απόδοσης των ρομπότ. Η δεύτερη, αφορά στην εφαρμογή και την ανάπτυξη λογισμικού για τη διαχείριση των εργασιών των ρομπότ. Η τρίτη αφορά τη σχεδίαση υλικού υπολογιστών για την εκτέλεση του λογισμικού και την καλύτερη επικοινωνία με τους αισθητήρες και τους επενεργητές των ρομπότ.

1.2 Ορισμός των ρομπότ

Σύμφωνα με το Ινστιτούτο Ρομποτικής της Αμερικής, ρομπότ είναι ένας αναπρογραμματιζόμενο και προλειτουργικό σύστημα αυτόματου ελέγχου σχεδιασμένο να μετακινεί υλικά, αντικείμενα, εργαλεία ή εξειδικευμένες συσκευές με κατάλληλες μεταβλητά προγραμματιζόμενες κινήσεις που στοχεύουν στη βελτίωση της απόδοσης μιας σειράς εργασιών, του οποίου οι εμφάνιση και οι κινήσεις συχνά δίνουν την εντύπωση πως ενεργεί κατά βούληση.

Ο όρος βιομηχανικό ρομπότ (industrial robot) καθιερώθηκε το 1954 από τον G.C. Devol (ΗΠΑ). Ο Devol περιέγραψε πώς μπορεί να κατασκευαστεί ένα ελεγχόμενο μηχανικό χέρι, το οποίο μπορεί να εκτελεί διάφορες εργασίες στη βιομηχανία. Το πρώτο βιομηχανικό ρομπότ κατασκευάστηκε και τέθηκε σε λειτουργία το 1961 από την εταιρία Unimation. Για να μπορεί λοιπόν να θεωρηθεί ένα μηχάνημα ρομπότ, θα πρέπει να έχει την ικανότητα να λειτουργεί αυτόματα, και αυτόνομα. Αυτό σημαίνει ότι πρέπει να υπάρχει ενσωματωμένη ευφυΐα, ή προγραμματιζόμενη μνήμη, ή απλά μια διάταξη ρυθμιζόμενων μηχανισμών που ελέγχουν τους χειρισμούς. Το βασικό στοιχείο στον παραπάνω ορισμό είναι ο αναπρογραμματισμός των ρομπότ, δηλαδή η πολυχρηστικότητα και προσαρμοστικότητα τους εξαιτίας της σύνδεσης τους με ηλεκτρονικού υπολογιστή. Το βιομηχανικό ρομπότ είναι ένας βραχίονας με ευέλικτο πρόγραμμα, αυτόνομος που διαθέτει μεγάλη κινητικότητα η οποία επιτρέπει την εκτέλεση ανεξάρτητων κινήσεων μεταξύ τους, σε οποιαδήποτε σειρά. Το πρόγραμμα τροποποιείται ανάλογα με τις συνθήκες του περιβάλλοντος και τις συνθήκες εργασίας.



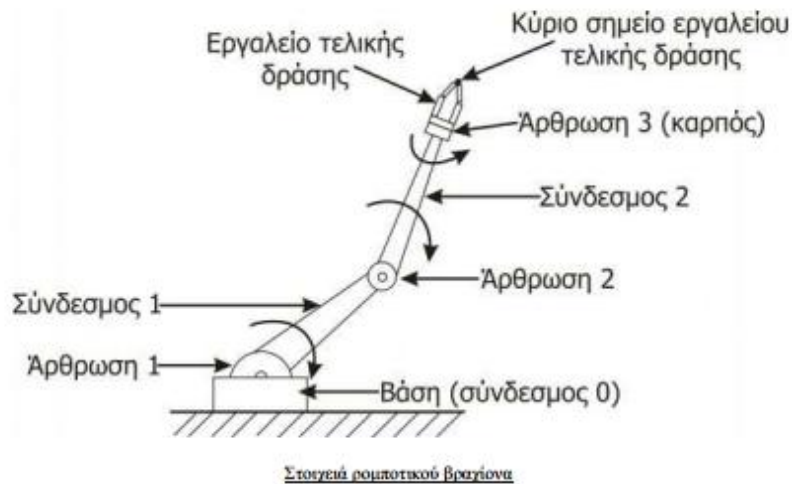
Σήμερα τα ρομπότ χρησιμοποιούνται κυρίως για να εκτελούν βαριές, ανιαρές, επικίνδυνες και επαναλαμβανόμενες εργασίες, κυρίως στη βιομηχανία. Υπολογίζεται ότι το 90% των ρομπότ είναι εγκατεστημένα σε βιομηχανικούς χώρους. Για παράδειγμα, οι βραχίονες που συναρμολογούν και συγκολλούν τον σκελετό ενός αυτοκινήτου στη γραμμή παραγωγής. Στόχους της εφαρμογής των ρομποτικών συστημάτων στη βιομηχανία αποτελούν η υποκατάσταση της χειρονακτικής εργασίας η βελτίωση και η σταθεροποίηση της ποιότητας και η αύξηση της παραγωγικότητας. Άλλοι τομείς, όπου χρησιμοποιούνται ρομπότ, είναι Διαστημικά προγράμματα, Στρατιωτικές εφαρμογές, Ιατρικές εφαρμογές, Ερευνητικά Ιδρύματα και πολλοί άλλοι. Τα τελευταία χρόνια έχουν αναπτυχθεί ρομπότ που προσπαθούν να μιμηθούν ανθρώπινα χαρακτηριστικά, όπως το περπάτημα και την όραση.

Λόγω του ότι τα βιομηχανικά ρομπότ ελέγχονται από υπολογιστές ή μικροεπεξεργαστές, αυτά μπορούν εύκολα να αναπρογραμματιστούν για διάφορες εργασίες. Τοιουτοτρόπως δεν απαιτείται η αντικατάσταση αυτής της μηχανής εάν αλλάξει η προδιαγραμμένη εργασία. Αυτός ο τύπος αυτοματισμού καλείται ευέλικτος αυτοματισμός.

1.3 Δομή των Ρομπότ

Το βιομηχανικό ρομπότ είναι ένας προγραμματιζόμενος μηχανικός χειριστής, ικανός να κινείται προς διάφορες διευθύνσεις, εφοδιασμένος στο άκρο του με μια διάταξη εργασίας, που ονομάζεται τελικό σημείο δράσης (ή εργαλείο), και ικανός να εκτελεί διάφορες βιομηχανικές εργασίες, που συνήθως γίνονται από τον άνθρωπο. Ένα σύγχρονο ρομπότ περιλαμβάνει τουλάχιστον τρία βασικά δομικά μέρη:

- 1) τον χειριστή (βραχίονα, χέρι), που είναι το κινούμενο μηχανικό τμήμα
- 2) τα στοιχεία δράσης (κινητήρες), που ενεργοποιούν τις αρθρώσεις του βραχίονα
- 3) τον υπολογιστή, που αποθηκεύει και εκτελεί τα προγράμματα εργασίας και ελέγχει τις κινήσεις του ρομπότ.



Ο βραχίονας του ρομπότ περιέχει το κύριο σώμα και τον καρπό, που έχει στο τέλος του το εργαλείο (τελικό στοιχείο δράσης). Το εργαλείο μπορεί να είναι μια κεφαλή συγκόλλησης, ένα πιστόλι χρωματίσματος, ένα μηχανικό εργαλείο ή μια αρπάγη, που ανοιγοκλείνει ανάλογα με την εφαρμογή, στην οποία πρόκειται να χρησιμοποιηθεί το ρομπότ. Επειδή όλα τα εργαλεία στερεώνονται στο τέλος των ρομπότ, λέγονται και "τελικά στοιχεία δράσης- end effectors".

Το κύριο σώμα (δηλαδή ο κορμός) του ρομπότ αποτελείται από μια διαδοχή συνδέσμων ή μελών (links), που συνδέονται με αρθρώσεις (joints). Οι αρθρώσεις ελέγχουν τις κινήσεις των συνδέσμων. Η ομάδα των αρθρώσεων, που ελέγχει την κίνηση του εργαλείου, ονομάζεται "καρπός". Κάθε μια από τις αρθρώσεις του βραχίονα και του καρπού παρέχει έναν βαθμό ελευθερίας στην κίνηση του τελικού στοιχείου δράσης. Έτσι, ένα ρομπότ με n βαθμούς ελευθερίας περιέχει n αρθρώσεις ή n άξονες κίνησης γενικά.

Η κίνηση του τελικού στοιχείου δράσης ρυθμίζεται ελέγχοντας τη θέση και την ταχύτητα των αξόνων κίνησης του ρομπότ. Στη ρομποτική ένας άξονας κίνησης ισοδυναμεί με έναν βαθμό ελευθερίας, ως προς τον οποίο μπορεί να κινηθεί το ρομπότ. Για να μπορέσει ένα ρομπότ να φτάσει ένα αυθαίρετο σημείο (μέσα στο χώρο εργασίας του) με έναν επιθυμητό προσανατολισμό του εργαλείου, χρειάζεται να έχει έξι άξονες (βαθμούς ελευθερίας) κίνησης. Έστω και μόνο ένας διαφορετικός προσανατολισμός του εργαλείου μπορεί να αλλάξει εντελώς τη θέση του βραχίονα του ρομπότ.

1.3.1 Βαθμός Ελευθερίας

Βασικό γνώρισμα κάθε ρομπότ αποτελεί ο Βαθμός Ελευθερίας (Degree of Freedom DOF), δηλαδή το πόσο ευκίνητο είναι ένα ρομπότ στο χώρο.

Ο ορισμός λέει: "Ο αριθμός των ανεξάρτητων παραμέτρων, που προσδιορίζουν τη θέση ενός σώματος στο χώρο, ονομάζεται Βαθμός Ελευθερίας"

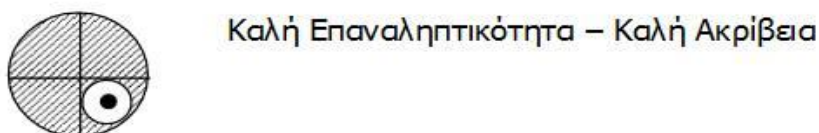
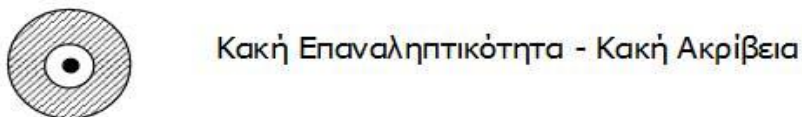
Για να περιγράψουμε ακριβώς τη θέση ενός στερεού σώματος στο χώρο, χρειαζόμαστε 6 μεταβλητές, 3 για την θέση και 3 για τον προσανατολισμό του. Για να μπορεί ένα ρομπότ να κινηθεί οπουδήποτε στο χώρο με οποιοδήποτε προσανατολισμό, πρέπει να έχει τουλάχιστον 6 βαθμούς ελευθερίας. Τρεις βαθμοί ελευθερίας αφορούν τη θέση του συστήματος (translational DOF) και τρεις αφορούν τον προσανατολισμό του (rotational DOF) στους 3 άξονες. Κατά συνέπεια, για να μπορεί ένα ρομπότ να κινηθεί απολύτως ελεύθερα στον χώρο, σε οποιοδήποτε σημείο και με οποιονδήποτε προσανατολισμό, πρέπει να έχει τουλάχιστον 6 βαθμούς ελευθερίας.

Ο ανθρώπινος βραχίονας υπολογίζεται ότι έχει 7 βαθμούς ελευθερίας. Στα βιομηχανικά ρομπότ σπάνια συναντάμε πάνω από 6 βαθμούς ελευθερίας, αφού ναί μεν θα βελτιωνόταν η ευελιξία τους, αλλά γινόταν πιο περίπλοκος ο αλγόριθμος ελέγχου τους χωρίς να επεκτείνεται ο χώρος δράσης τους.

1.3.2 Χώρος Εργασίας ή φάκελος εργασίας

Ως χώρος εργασίας ορίζεται ο τρισδιάστατος χώρος τον οποίο μπορεί να σαρώσει η άκρη του ρομποτικού μηχανισμού. Το μέγεθος και η γεωμετρική μορφή του χώρου αυτού εξαρτώνται από την κατασκευαστική δομή του ρομπότ. Από τα πιο σημαντικά μεγέθη ενός βιομηχανικού βραχίονα είναι η επαναληψιμότητα και η ακρίβεια. Πιο συγκεκριμένα τα παραπάνω μεγέθη αναφέρονται στα εξής:

- **Επαναληψιμότητα:** εκφράζει τη δυνατότητα του βραχίονα να γυρίσει στο ίδιο σημείο μετά από αρκετές επαναλήψεις και δίνεται ως εύρος μέσα στο οποίο ο βραχίονας θα τερματίσει την κίνηση. Η απόκλιση οφείλεται στο ότι κατά τη λειτουργία του το ρομπότ είναι δυνατό να χάσει λίγο από τη μέτρηση της θέσης με αποτέλεσμα να μη μπορεί να επιστρέψει στη συγκεκριμένη θέση μετά από ορισμένους κύκλους λειτουργίας. Δεομένου ότι στις συνήθεις βιομηχανικές εφαρμογές οι επιθυμητές κινήσεις διδάσκονται στο ρομπότ, αντιλαμβάνεται κανείς τη σπουδαιότητα της επαναληψιμότητας.
- **Ακρίβεια:** είναι η ικανότητα του ρομπότ να πηγαίνει ακριβώς στη θέση που του έχει δοθεί εντολή να πάει. Η ακρίβεια εξαρτάται κυρίως από τη διακριτότητα των εξαρτημάτων ελέγχου, τη μηχανολογική σύνδεση των μελών του και το ελάχιστο επιτρεπόμενο σφάλμα που επιβάλλει η ευστάθεια της λειτουργίας των σέρβο. Η ακρίβεια επηρεάζεται από το είδος και το μέγεθος του εκάστοτε φορτίου, σε αντίθεση με την ,επαναληψιμότητα γι' αυτό και ορισμένοι κατασκευαστές προδιαγράφουν μόνο την τελευταία.



1.4 Είδη Αρθρώσεων

Οι άξονες κίνησης ενός ρομποτικού βραχίονα μπορεί να είναι άξονες στροφικής κίνησης (περιστροφικές αρθρώσεις) ή γραμμικής μεταφορικής κίνησης (πρισματικές αρθρώσεις). Ένας στροφικός άξονας οδηγείται άμεσα από έναν ηλεκτρικό κινητήρα ή έμμεσα από ένα σύστημα αλυσίδας ή οδοντωτών τροχών. Η κίνηση κατά μήκος ενός γραμμικού άξονα πραγματοποιείται από ένα πρισματικό ζεύγος ή μέσω μιας κοχλιωτής λειτουργίας. Ένα πρισματικό ζεύγος διαθέτει ένα υδραυλικό ή πνευματικό έμβολο, ενώ ένας κοχλίας μετατρέπει την περιστροφική κίνηση ενός ηλεκτρικού κινητήρα σε γραμμική κίνηση κατά μήκος του αντίστοιχου άξονα του βραχίονα.

Άρθρωση (joint) καλείται η διάταξη εκείνη η οποία είναι διαμορφωμένη καταλλήλως και συνδέει δύο σώματα μεταξύ τους κατά τρόπο έτσι ώστε να εμποδίζει την εκτέλεση μίας ή πλειόνων από τις 6 μεταβλητές κίνησης, κάτι το οποίο οδηγεί στη μείωση της ελευθερίας κίνησης του ενός σε σχέση προς το άλλο σώμα.

Τα πιο σημαντικά είδη αρθρώσεων είναι:

Η περιστροφική άρθρωση είναι άρθρωση που επιτρέπει σχετική στροφή μεταξύ δύο γειτονικών συνδέσμων. Δίνει ένα βαθμό ελευθερίας αφήνοντας το σώμα να περιστραφεί σε ένα επίπεδο και αποκόπτει κάθε άλλη δυνατότητα κίνησης.

Η πρισματική (ή τηλεσκοπική) άρθρωση είναι άρθρωση που επιτρέπει σχετική μετατόπιση (σε ευθεία γραμμή) μεταξύ δύο γειτονικών συνδέσμων. Δίνει και αυτή ένα βαθμό ελευθερίας αφήνοντας το σώμα να μετατοπίζεται στη διεύθυνση ενός από τους άξονες, και αποκόπτει κάθε άλλη δυνατότητα κίνησης

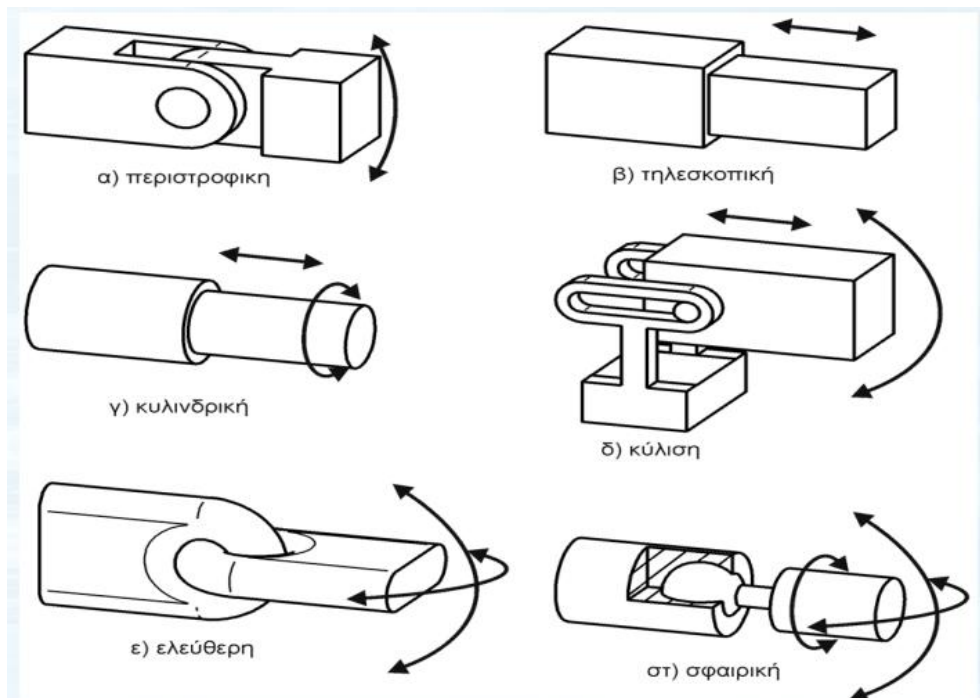
ΣΥΝΘΕΤΕΣ ΑΡΘΡΩΣΕΙΣ:

Η κυλινδρική άρθρωση δίνει δύο βαθμούς ελευθερίας αφήνοντας μία μεταφορική κίνηση στη διεύθυνση ενός άξονα και μία περιστροφική γύρω από τον άξονα αυτό.

Η άρθρωση της κύλισης δίνει και αυτή δύο βαθμούς ελευθερίας, δηλαδή μία μεταφορική και μία περιστροφική κίνηση, άλλα σε αυτή την περίπτωση ο άξονας της περιστροφικής κίνησης είναι κάθετος στη διεύθυνση του άξονα που πραγματοποιείται η μεταφορική κίνηση.

Η ελεύθερη άρθρωση δίνει δύο βαθμούς ελευθερίας αφήνοντας δύο περιστροφικές κινήσεις και εμποδίζοντας όλες τις υπόλοιπες

Η σφαιρική άρθρωση δίνει τρεις βαθμούς ελευθερίας αφήνοντας και τις τρεις περιστροφικές κινήσεις ελεύθερες και εμποδίζοντας όλες τις μεταφορικές.



1.5 Διεπαφές και μέθοδοι προγραμματισμού των ρομπότ

Η εγκατάσταση ή ο προγραμματισμός των κινήσεων και των ακολουθιών για ένα βιομηχανικό ρομπότ συνήθως διδάσκονται με τη σύνδεση του ελεγκτή του ρομπότ σε ένα φορητό υπολογιστή, επιτραπέζιο υπολογιστή ή (εσωτερικού ή Διαδικτυακού) υπολογιστή.

Ένα ρομπότ και μια συλλογή μηχανών ή περιφερειακών αναφέρονται ως κελί εργασίας, ή κελί. Ένα τυπικό κελί μπορεί να περιέχει έναν τροφοδότη των τμημάτων, μια μηχανή σχεδίασης και ένα ρομπότ. Οι διάφορες μηχανές είναι ένα ολοκληρωμένο σύστημα και ελέγχονται από έναν υπολογιστή ή ένα PLC. Το πώς αλληλεπιδρά το ρομπότ με τις άλλες μηχανές στο κελί πρέπει να προγραμματιστεί, τόσο όσον αφορά τις θέσεις τους στο κελί όσο και τον συγχρονισμό τους.

Λογισμικό: Ο υπολογιστής εγκαθίσταται με αντίστοιχο λογισμικό διασύνδεσης. Η χρήση ενός υπολογιστή απλοποιεί σημαντικά τη διαδικασία προγραμματισμού. Εξειδικευμένο ρομποτικό λογισμικό λειτουργεί είτε στον ελεγκτή του ρομπότ ή στον υπολογιστή ή και τα δύο ανάλογα με το σχεδιασμό του συστήματος.

Υπάρχουν δύο βασικές ενότητες που πρέπει να διδαχθούν (ή να προγραμματιστούν): τα δεδομένα θέσης και διαδικασίας. Για παράδειγμα, σε μια εργασία μετακίνησης μιας βίδα από έναν τροφοδότη σε μια τρύπα οι θέσεις του τροφοδότη και της τρύπας πρέπει πρώτα να διδαχθούν ή να προγραμματιστούν. Δεύτερον, η διαδικασία να πάρει τη βίδα από τον τροφοδότη στην τρύπα πρέπει να προγραμματιστεί, μαζί με κάθε είσοδο/έξοδο (I/O) που εμπλέκονται, για παράδειγμα, ένα σήμα για να δείξει πότε είναι η βίδα στον τροφοδότη έτοιμη ώστε να παραληφθεί. Ο σκοπός του ρομποτικού λογισμικού είναι να διευκολύνει και τις δύο αυτές εργασίες προγραμματισμού.

Η διδασκαλία των θέσεων των ρομπότ μπορεί να επιτευχθεί με διάφορους τρόπους:

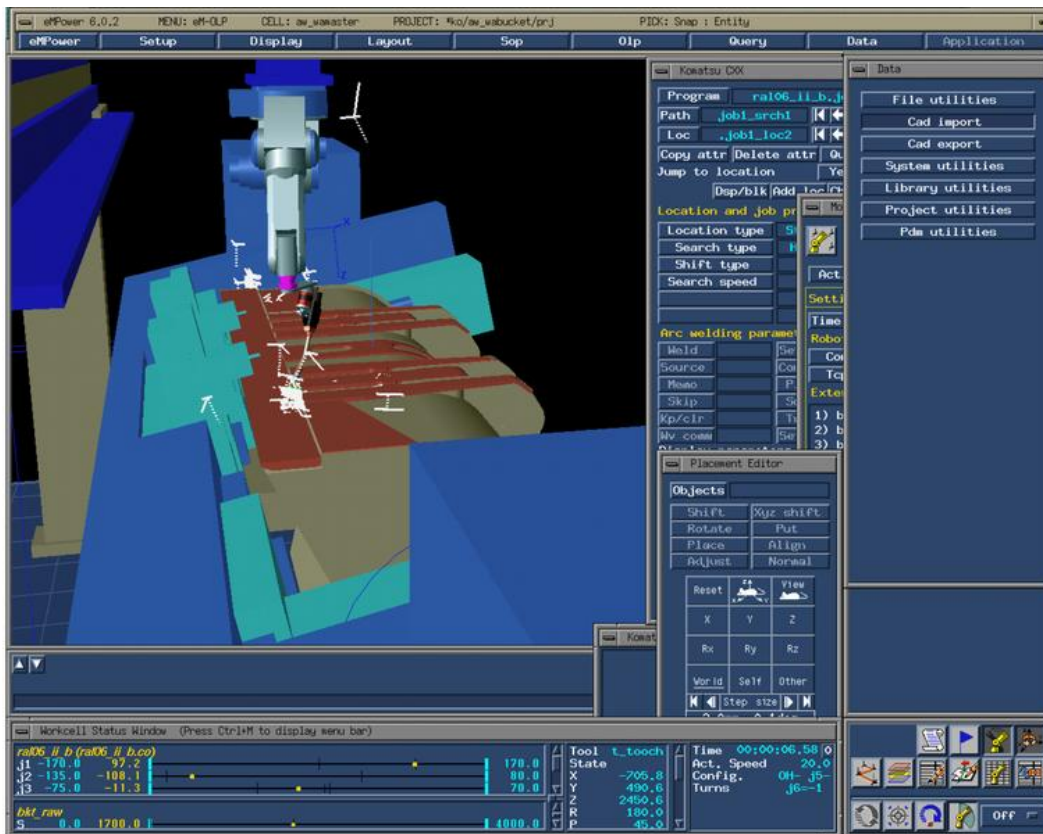
Εντολές θέσεων: Το ρομπότ μπορεί να κατευθυνθεί προς την επιθυμητή θέση χρησιμοποιώντας ένα περιβάλλον βασισμένο σε κείμενο ή εντολές στις οποίες η απαιτούμενη θέση X-Y-Z μπορούν να προσδιορίζονται και να επεξεργάζονται.

Διδακτικό μέσο (Teaching pendant): Οι θέσεις των ρομπότ μπορούν να διδαχθούν μέσω ενός εκπαιδευτικού μέσου. Αυτό είναι μια φορητή μονάδα ελέγχου και προγραμματισμού. Τα κοινά χαρακτηριστικά των εν λόγω μονάδων είναι η δυνατότητα να στέλνουν τα ρομπότ σε μια επιθυμητή θέση, ή "ίντσας" ή "επαφής" για να ρυθμιστεί μια θέση. Έχουν επίσης έναν τρόπο για να αλλάζουν την ταχύτητα από μια χαμηλή ταχύτητα που συνήθως απαιτείται για την προσεκτική τοποθέτηση, ή κατά τη διάρκεια δοκιμών μιας νέας ή τροποποιημένης ρουτίνας. Ένα μεγάλο κουμπί ακινητοποίησης έκτακτης ανάγκης συνήθως περιλαμβάνεται. Συνήθως όταν το ρομπότ έχει προγραμματιστεί δεν υπάρχει πλέον λόγος για τη χρήση διδάσκοντος μέσου.

Οδηγούμενο από τη μύτη (Leading by nose): είναι μια τεχνική που προσφέρονται από πολλούς κατασκευαστές ρομπότ. Σε αυτή τη μέθοδο, ένας άνθρωπος κρατάει το μηχανισμό χειρισμού του ρομπότ, ενώ κάποιος άλλος εισάγει μια εντολή που απενεργοποιεί το ρομπότ προκαλώντας το να χαλαρώσει. Ο χρήστης μετακινεί το ρομπότ στη συνέχεια με το χέρι στην απαιτούμενη θέση και/ή κατά μήκος μιας απαιτούμενης πορείας, ενώ το λογισμικό καταγράφει αυτή τη θέση στη μνήμη. Το πρόγραμμα μπορεί να τρέξει στη συνέχεια το ρομπότ σε αυτές τις θέσεις ή κατά μήκος της διαδρομής που έχει διδαχθεί. Αυτή η τεχνική είναι δημοφιλής για καθήκοντα όπως η βαφή.

Προγραμματισμός εκτός σύνδεσης(off-line): είναι όπου όλο το κελί, το ρομπότ και όλα τα μηχανήματα ή τα εργαλεία στον χώρο εργασίας αντιστοιχίζονται γραφικά. Το ρομπότ μπορεί στη συνέχεια να μεταφερθεί στην οθόνη και η διαδικασία να προσομοιωθεί. Η τεχνική αυτή αν και είναι περιορισμένη διότι βασίζεται σε ακριβή μέτρηση των θέσεων του σχετιζόμενου εξοπλισμού και επίσης στηρίζεται στην ακρίβεια θέσης του ρομπότ που μπορεί να μην είναι σύμφωνη με ό, τι έχει προγραμματιστεί.

Τα πλεονεκτήματα του *off-line* προγραμματισμού όμως είναι πολλά. Το πρώτο και βασικότερο είναι ότι δεν χρειάζεται να διακοπεί η λειτουργία του βραχίονα για τον προγραμματισμό μίας νέας τροχιάς. Αυτό έχει ως αποτέλεσμα την βελτίωση της παραγωγικότητας διότι μειώνονται οι περίοδοι αδράνειας του βραχίονα. Ο προγραμματισμός γίνεται πολύ πιο γρήγορα σε σχέση με τους άλλους τρόπους. Με την πλήρη και ρεαλιστική προσομοίωση μπορεί να γίνει έλεγχος για τυχόν συγκρούσεις με διάφορα αντικείμενα του περιβάλλοντα χώρου. Ο προγραμματισμός γίνεται από τον ηλεκτρονικό υπολογιστή από το γραφείο που έχει σαν αποτέλεσμα τη βελτίωση των συνθηκών εργασίας και την αύξηση της ασφάλειας.



Άλλου είδους: χειριστές μηχανών χρησιμοποιούν συχνά συσκευές διεπαφής χρήστη, συνήθως οθόνες αφής, οι οποίες χρησιμεύουν ως ο πίνακας ελέγχου του χειριστή. Ο χειριστής μπορεί να εναλλάσσει από πρόγραμμα σε πρόγραμμα, να κάνει προσαρμογές στο πλαίσιο ενός προγράμματος, αλλά και να λειτουργεί μια σειρά από περιφερειακές συσκευές που μπορούν να ενταχθούν στο ίδιο ρομπωτικό σύστημα. Τέτοιες μπορεί να είναι μηχανισμοί τερματισμού, τροφοδότες που προμηθεύουν εξαρτήματα το ρομπότ, ιμάντες, όργανα διακοπής λειτουργίας εκτάκτου ανάγκης, συστήματα μηχανικής όρασης υπολογιστή, ασφάλεια κλειδώματος συστήματος, εκτυπωτές bar code και μια σχεδόν άπειρη σειρά από άλλες βιομηχανικές συσκευές, οι οποίες ελέγχονται μέσω του πίνακα ελέγχου χειρισμού.

Το εκπαιδευτικό μέσο ή το PC συνήθως αποσυνδέεται μετά τον προγραμματισμό και το ρομπότ τρέχει στη συνέχεια το πρόγραμμα που έχει εγκατασταθεί στον ελεγκτή του. Ωστόσο, υπολογιστές χρησιμοποιούνται συχνά για να «εποπτεύουν» το ρομπότ και τα τυχόν περιφερειακά, ή να παρέχουν πρόσθετο αποθηκευτικό χώρο για πρόσβαση σε πολλές πολύπλοκες διαδρομές και ρουτίνες.

2. Θεωρητικό Υπόβαθρο

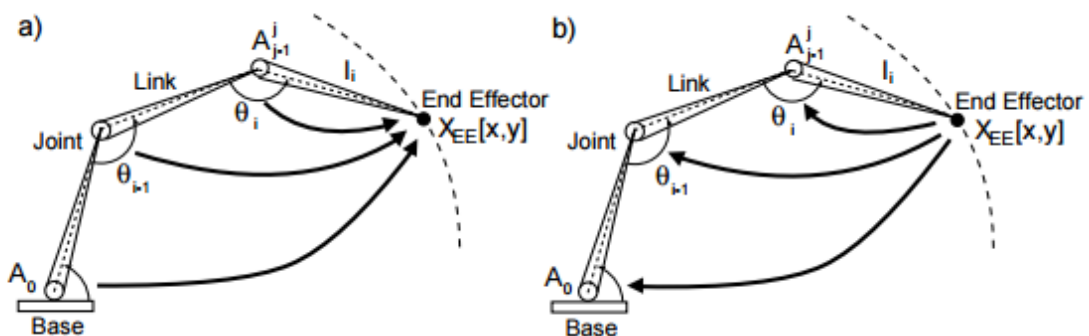
2.1 Κινηματική Ανάλυση

Ο όρος Κινηματική “Kinematics” στο τομέα της ρομποτικής ονομάζεται η επιστήμη που εξετάζει αναλυτική κίνηση ενός οποιοδήποτε μηχανισμού χωρίς να παίρνει υπόψη τις δυνάμεις και τις ροπές που την προκαλούν. Η εφαρμογή της κινηματικής σε ένα ρομποτικό βραχίονα έχει να κάνει καθαρά με τις γεωμετρικές και χρονικά εξαρτώμενες ιδιότητες της κίνησης του.

Η μελέτη της κινηματικής συμπεριφοράς ενός ρομπότ είναι απαραίτητη για τον προγραμματισμό και έλεγχο γιατί δίνει τις απαραίτητες για τη χωρική του διάταξη αλλά και την αλληλεπίδραση που μπορεί να έχει με το περιβάλλον.

Γενικά, το πρόβλημα της κινηματικής μπορεί να χωριστεί σε 2 υπό-προβλήματα. Το πρόβλημα της **ορθής κινηματικής ή ευθές πρόβλημα (Forward Kinematics)** και το πρόβλημα της **αντίστροφης κινηματικής (Inverse Kinematics)**.

Το ευθύ κινηματικό πρόβλημα συνίσταται την εύρεση της θέσης και του προσανατολισμού του άκρου του βραχίονα ως προς την βάση του, όταν γνωρίζουμε την θέση κάθε άρθρωσης, ενώ το αντίστροφο κινηματικό πρόβλημα συνίσταται στην εύρεση της θέσης της κάθε άρθρωσης, η οποία τοποθετεί το άκρο του βραχίονα σε μια συγκεκριμένη θέση και προσανατολισμό σε σχέση με την βάση του. Ο βαθμός δυσκολίας επίλυσης των προβλημάτων αυτών εξαρτάται από την γεωμετρία του βραχίονα, για αυτό το λόγω στην ρομποτική προτιμάμε ρομπότ με περιττούς βαθμούς ελευθερίας.



2.2 Ευθεία κινηματική ανάλυση

2.2.1 Μητρώα μετασχηματισμού

Οι τρεις πρώτες στήλες στη μήτρα του πίνακα μετασχηματισμού αντιπροσωπεύουν τον προσανατολισμό του άκρου, ενώ η τελευταία στήλη αντιπροσωπεύει τη θέση του άκρου. Ο προσανατολισμός και η θέση του άκρου μπορεί να υπολογιστεί σε γωνίες των αρθρώσεων.

$$T = \begin{bmatrix} r_{11} & r_{21} & r_{31} & dx \\ r_{21} & r_{22} & r_{32} & dy \\ r_{31} & r_{23} & r_{33} & dz \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Μητρώο Περιστροφής

Μητρώο Μεταφοράς

Βασικοί Μετασχηματισμοί

✓ Περιστροφή ως προς X

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos \theta & -\sin \theta & 0 \\ 0 & \sin \theta & \cos \theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

✓ Περιστροφή ως προς Y

$$\begin{bmatrix} \cos \theta & 0 & \sin \theta & 0 \\ 0 & 1 & 0 & 0 \\ -\sin \theta & 0 & \cos \theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

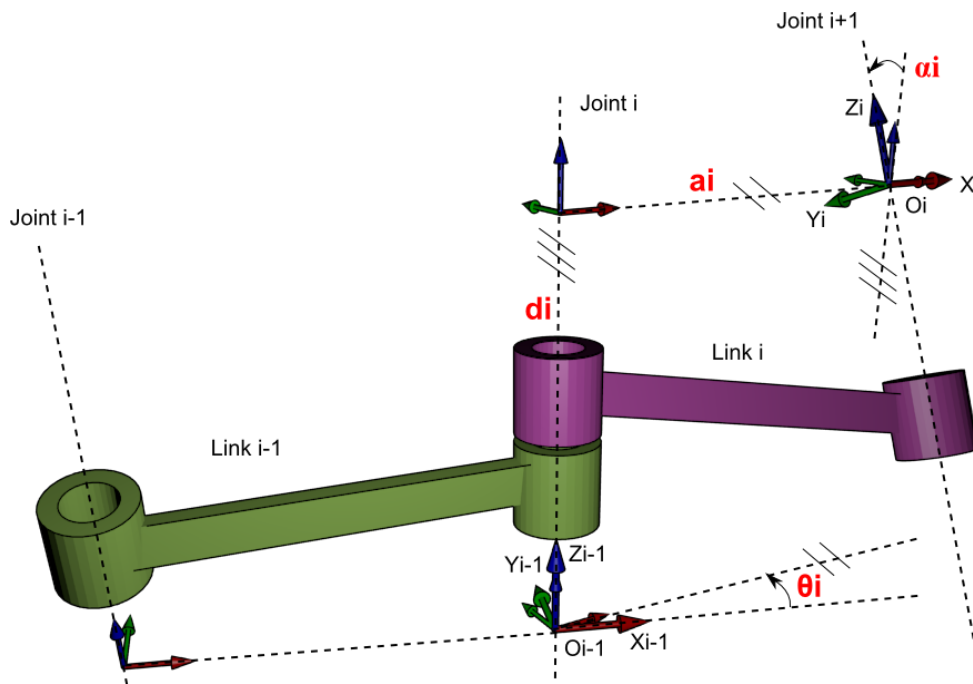
✓ Περιστροφή ως προς Z

$$\begin{bmatrix} \cos \theta & -\sin \theta & 0 & 0 \\ \sin \theta & \cos \theta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

2.2.2 Παράμετροι Denavit- Hartenberg (D-H)

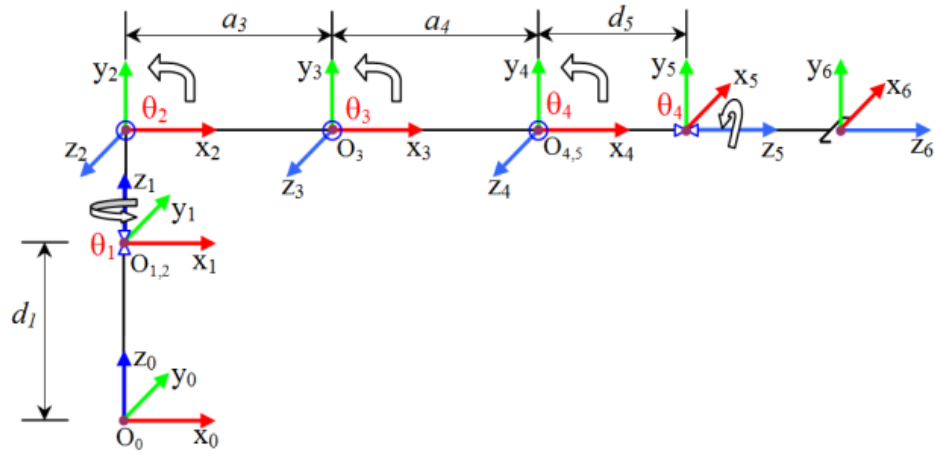
Για την ανάλυση της ορθής και της αντίστροφης κινηματικής λειτουργίας ενός ρομποτικού βραχίονα, μπορούμε να ακολουθήσουμε τη μέθοδο Denavit–Hartenberg (D–H). Η μέθοδος αυτή χρησιμοποιεί ένα ελάχιστο πλήθος παραμέτρων για την πλήρη περιγραφή της κινηματικής αλυσίδας.

Αυτές οι παράμετροι περιγράφουν τη θέση ενός ρομπότ δηλαδή ενός συνδέσμου-πλαίσιο από μια προηγούμενη θέση μέσω της ακολουθίας των μεταφορών και περιστροφών.



- a_i = η απόσταση μεταξύ των Z_i και Z_{i+1} μετρούμενη κατά μήκος του X_i
- α_i = η γωνία μεταξύ των Z_i και Z_{i+1} μετρούμενη ως προς τον X_i
- d_i = η απόσταση μεταξύ των X_{i-1} και X_i μετρούμενη κατά μήκος του Z_i
- θ_i = η γωνία μεταξύ των X_{i-1} και X_i μετρούμενη ως προς Z_i

Οι παράμετροι d_i και θ_i προσδιορίζουν τη σχετική θέση των γειτονικών μελών, ενώ οι παράμετροι a_i και α_i προσδιορίζουν τη δομή του μέλους i . Εάν οι τέσσερις αυτές παράμετροι έχουν καθορισθεί για κάθε άρθρωση του μηχανισμού, τότε η γεωμετρία του είναι πλήρως προσδιορισμένη.



Παραπάνω βλέπουμε την μηχανική ανάλυση του ρομποτικού βραχίονα κατά D-H. Αφού πλέον λες οι αρθρώσεις καθορίζεται με τις αρχές της σύμβασης D-H, δημιουργούμε τον πίνακα D-H του βραχίονα.

Μέλος	$\alpha_{i-1} (^{\circ})$	$a_{i-1} (\text{cm})$	$d_i (\text{cm})$	$\theta_i (^{\circ})$
1	0	0	d_1	θ_1
2	90°	0	0	θ_2
3	0	a_3	0	θ_3
4	0	a_4	0	$(\theta_4 - 90^{\circ})$
5	-90°	0	d_5	θ_5
6	0	0	0	θ_6

Αντικαθιστώντας τις παραμέτρους μπορούμε να εξάγουμε τους πίνακες μετασχηματισμού όπως φαίνεται παρακάτω

$$T_1^0 = \begin{bmatrix} c\theta_1 & -s\theta_1 & 0 & 0 \\ s\theta_1 & c\theta_1 & 0 & 0 \\ 0 & 1 & 0 & d_1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (1)$$

$$T_2^1 = \begin{bmatrix} c\theta_2 & -s\theta_2 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ s\theta_2 & c\theta_2 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2)$$

$$T_3^2 = \begin{bmatrix} c\theta_3 & -s\theta_3 & 0 & a_3 \\ s\theta_3 & c\theta_3 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3)$$

$$T_4^3 = \begin{bmatrix} c\theta_4 & -s\theta_4 & 0 & a_4 \\ s\theta_4 & c\theta_4 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (4)$$

$$T_5^4 = \begin{bmatrix} c\theta_5 & -s\theta_5 & 0 & 0 \\ 0 & 0 & 1 & d_5 \\ -s\theta_4 & c\theta_5 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (5)$$

$$T_6^5 = \begin{bmatrix} c\theta_5 & -s\theta_4 & 0 & 0 \\ s\theta_4 & c\theta_5 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (6)$$

Σε αυτό το σημείο πρέπει να αναφέρουμε πως το μέλος 6 του ρομπότ δηλαδή η αρπάγη δεν εκτελεί περιστροφική κίνηση αλλά κίνηση άνοιγμα-κλείσιμο, γι αυτό και ο πίνακας μετασχηματισμού δεν χρειάζεται παρά μόνο στο σχεδιαστικό πρόγραμμα.

Η θέση και ο προσανατολισμός κάθε μέλους προκύπτει από τους διαδοχικούς πολλαπλασιασμούς των μητρώων μετασχηματισμού όπως φαίνεται στις παρακάτω σχέσεις (1) έως (6) σύμφωνα με τον τύπο:

$$T_0^i = A_0^1 * A_1^2 * \dots * A_{i-2}^{i-1} * A_{i-1}^i$$

$${}^0_5T = T_1^0 T_2^1 T_3^2 T_4^3 T_5^4 = \begin{bmatrix} nx & ox & ax & px \\ ny & oy & ay & py \\ nz & oz & az & pz \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (7)$$

$$\begin{aligned} nx &= ((c1c2c3 - c1s2s3)c4 + (-c1c2s3 - c1s2c3)s4)c5 + s1s5 \\ ny &= ((s1c2c3 - s1s2s3)c4 + (-s1c2s3 - s1s2c3)s4)c5 - c1s5 \\ nz &= ((s2c3 + c2s3)c4 + (-s2s3 + c2c3)s4)c5 \end{aligned} \quad (8)$$

$$\begin{aligned} ox &= -((c1c2c3 - c1s2s3)c4 + (-c1c2s3 - c1s2c3)s4)s5 + s1c5 \\ oy &= -((s1c2c3 - s1s2s3)c4 + (-s1c2s3 - s1s2c3)s4)s5 - c1c5 \\ oz &= ((c2c3 - s2s3)s4)s5 - (-s2c3 + c2s3)c4 \end{aligned} \quad (9)$$

$$\begin{aligned} ax &= -(c1c2c3 - c1s2s3)s4 + (-c1c2s3 - c1s2c3)c4 \\ ay &= -(s1c2c3 - s1s2s3)s4 + (-s1c2s3 - s1s2c3)c4 \\ az &= (c2c3 - s2s3)c4 - (-s2c3 + c2s3)s4 \end{aligned} \quad (10)$$

$$\begin{aligned} dx &= -(c1c2c3 - c1s2s3)s4 + (-c1c2s3 - c1s2c3)c4)d5 + (c1c2c3 - c1s2s3)a4 + c1c2a3 \\ dy &= -(s1c2c3 - s1s2s3)s4 + (-s1c2s3 - s1s2c3)c4)d5 + (s1c2c3 - s1s2s3)a4 + s1c2a3 \\ dz &= (-s2c3 + c2s3)s4 + (-s2s3 + c2c3)c4)d5 + (s2s3 + c2s3)a4 + s2a3 + d1 \end{aligned} \quad (11)$$

όπου, $C_i = \cos \theta_i$ $S_i = \sin \theta_i$

2.3 Αντίστροφη κινηματική ανάλυση

Το αντίστροφο κινηματικό πρόβλημα είναι πολύ πιο δύσκολο από ό, τι το ευθύ. Όταν ένας ρομποτικός βραχίονας έχει περισσότερους βαθμούς ελευθερίας από τρεις ($\text{dof} > 3$), αντίθετα με την προς τα εμπρός κινηματική, στην αντίστροφη κινηματική υπάρχουν πολλαπλές λύσεις.

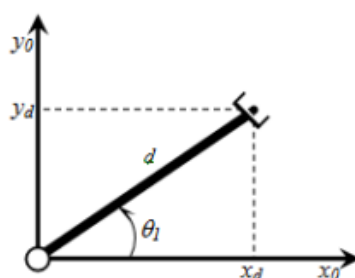
Κάθε μέλος χρειάζεται μια ιδιαίτερη μέθοδο λαμβάνοντας υπόψη τη δομή και τους περιορισμούς του συστήματος. Υπάρχουν δύο προσεγγίσεις η γεωμετρική και αλγεβρική που χρησιμοποιούνται για την εξαγωγή λύσης στην αντίστροφη κινηματική. Μερικοί περιορισμοί μπορούν να χρησιμοποιηθούν για να μειώσουν τον αριθμό των λύσεων για λόγους απλότητας.

2.3.1 Γεωμετρική προσέγγιση

Χρησιμοποιώντας το καρτεσιανό σύστημα συντεταγμένων, ο χρήστης καθορίζει την επιθυμητή θέση του στόχου της αρπάγης σε Καρτεσιανό χώρο ως (x, y, z) , όπου το z είναι το ύψος, και η γωνία ψ της αρπάγης σε σχέση με το έδαφος, διατηρείται σταθερή. Αυτή η σταθερά ψ επιτρέπει στους χρήστες να μετακινούνε αντικείμενα χωρίς να αλλάζουν τον προσανατολισμό του αντικειμένου (για παράδειγμα κρατώντας ένα φλιτζάνι με υγρό).

Τα μήκη d_1, a_3, a_4 και d_5 αντιστοιχεί στο ύψος βάσης, άνω μήκος του βραχίονα, το μήκος του αντιβραχίου(πήγης) και το μήκος της αρπάγης αντίστοιχα είναι όλες σταθερές. Οι γωνίες $\theta_1, \theta_2, \theta_3, \theta_4$ και θ_5 αντιστοιχούν στην περιστροφή του ώμου, βραχίονα, το αντιβράχιο, τον καρπό, και της αρπάγης, αντίστοιχα. Αυτές οι γωνίες ενημερώνεται καθώς η καθορισμένη θέση αλλάζει στο χώρο.

βλέπουμε καθαρά ότι $\theta_1 = (\text{atan2 } y, x)$ και η καθορισμένη ακτινική απόσταση από τη βάση d σχετίζονται με το x και y .



$$\begin{aligned} d &= \sqrt{x_d^2 + y_d^2} \\ x_d &= d \cos(\theta_1) \\ y_d &= d \sin(\theta_1) \end{aligned} \quad (12)$$

Προχωρώντας τώρα στην επίπεδη όψη βρίσκουμε μια σχέση μεταξύ της κοινής γωνίας θ_2 , θ_3 και θ_4 και ψ ως εξής:

$$\psi = \theta_2 + \theta_3 + \theta_4 \quad (13)$$

Εφόσον το ψ δίνεται, μπορούμε να υπολογίσουμε την ακτινική απόσταση και το ύψος του καρπού :

$$\begin{aligned} r_4 &= r_d - a_5 \cos(\psi) \\ z_4 &= z_d - a_5 \sin(\psi) \\ &\text{ή} \\ r_4 &= a_2 \cos(\theta_2) + a_4 \cos(\theta_2 + \theta_3) \\ z_4 &= a_3 \sin(\theta_2) + a_4 \sin(\theta_2 + \theta_3) + d_1 \end{aligned} \quad (14)$$

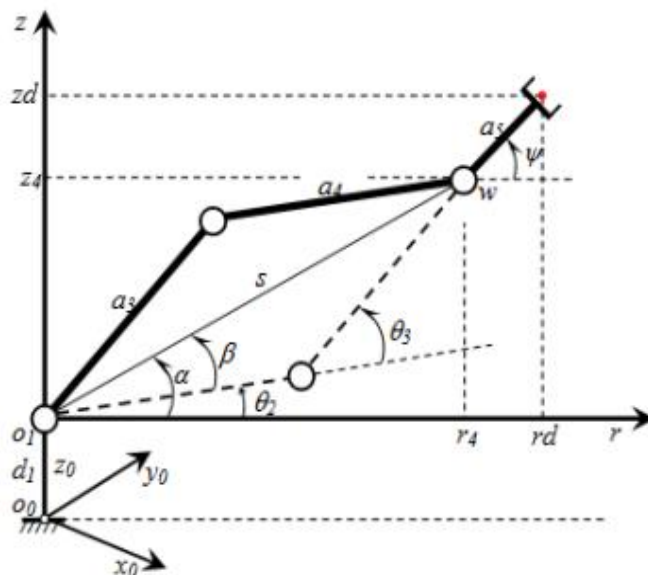
Τώρα θέλουμε να καθορίσουμε τις θ_2 και θ_3 . Λύνουμε πρώτα για α , β και s (από το Σχήμα 3.4) χρησιμοποιώντας το νόμο των συνημίτονων όπως:

$$\begin{aligned} \beta &= \alpha \tan 2(s^2 + a_3^2 - a_4^2, 2a_3s) \\ \alpha &= \alpha \tan 2(z_4 - d_1, r_4) \end{aligned} \quad (15)$$

$$s = \sqrt{(z_4 - d_1)^2 + r_4^2}$$

Με αυτές τις ενδιάμεσες τιμές, μπορούμε τώρα να βρούμε τις υπόλοιπες τιμές γωνιών ως:

$$\begin{aligned} \theta_2 &= \alpha \pm \beta \\ \theta_3 &= \alpha \tan 2(s^2 - a_3^2 - a_4^2, 2a_3a_4) \\ \theta_4 &= \psi - \theta_2 - \theta_3 \end{aligned} \quad (16)$$



2.3.2 Αναλυτική αλγεβρική Προσέγγιση

Χρησιμοποιώντας το X, Y και Z συνισταμένες που έχουμε πάρει στην ευθεία κινηματική (σχέση 11) έχουμε:

$$x = c_1[a_3c_2 + a_4c_{23} + d_5c_{234}] \quad (17)$$

$$y = s_1[a_3c_2 + a_4c_{23} + d_5c_{234}] \quad (18)$$

$$z = [d_5c_{234} + a_4s_{23} + a_3s_2] + d_1 \quad (19)$$

Η απλοποιημένη εξίσωση παίρνει την μορφή:

$$(17)^2 + (18)^2 \Rightarrow a_3c_2 + a_4c_{23} = \pm \sqrt{x^2 + y^2} - d_5c_{234} \quad (20)$$

Η πρώτη κίνηση της άρθρωσης, που ορίζεται από την γωνία θ_1 , μπορεί να υπολογιστεί απλά με τη χρήση γεωμετρικών παραμέτρων μόνο:

$$\therefore \theta_1 = a \tan 2(y, x)$$

Τώρα μπορούμε να υπολογίσουμε την γωνία θ_3 , χρησιμοποιώντας την εξίσωση (19)

$$(19) \Rightarrow a_3s_2 + a_4s_{23} = z - d_5s_{234} - d_1 \quad (21)$$

$$(20)^2 + (21)^2 \Rightarrow c_3 = \frac{(z - d_5s_{234} - d_1)^2 + (\pm \sqrt{x^2 + y^2} - d_5c_{234})^2 - a_3^2 - a_4^2}{2a_3a_4} \quad (22)$$

$$c_\psi = c_{234}, s_\psi = s_{234}$$

όπου

$$c_{234} = \cos(\theta_2 + \theta_3 + \theta_4), s_{234} = \sin(\theta_2 + \theta_3 + \theta_4)$$

$$\therefore c_3 = \frac{(z - d_5s_\psi - d_1)^2 + (\pm \sqrt{x^2 + y^2} - d_5c_\psi)^2 - a_3^2 - a_4^2}{2a_3a_4} \quad (23)$$

$$s_3 = \pm \sqrt{1 + c_3^2}$$

$$\therefore \theta_3 = a \tan 2(s_3, c_3) \quad (24)$$

Μετά από υπολογισμό της γωνίας θ_3 μπορούμε να βρούμε την θ_2 από:

$$\therefore \theta_2 = a - \theta \quad (24)$$

$$\begin{aligned} a &= a \tan 2 \left(z - d_5 s_\psi - d_1, \pm \sqrt{x^2 + y^2} - d_5 c_\psi \right) \\ \theta &= a \tan 2 \left(a_4 s_3, a_3 + a_4 c_3 \right) \end{aligned} \quad (25)$$

$$\theta_2 = a \tan 2 \left(z - d_5 s_\psi - d_1, \pm \sqrt{x^2 + y^2} - d_5 - a \tan 2 \left(a_4 s_3, a_3 + a_4 c_3 \right) \right) \quad (26)$$

$$\theta_\psi = \theta_2 + \theta_3 + \theta_4 \quad (27)$$

$$\therefore \theta_4 = \theta_\psi + \theta_2 + \theta_3 \quad (28)$$

Μπορούμε να βρούμε την γωνία θ_5 χρησιμοποιώντας συνολικό πίνακα μετασχηματισμού στην εξίσωση (7):

$$\begin{aligned} s_5 &= s_1 r_{11} - c_1 r_{21} \\ c_5 &= s_1 r_{12} - s_1 r_{22} \\ r_{11} &= ((c_1 c_2 c_3 - c_1 s_2 s_3) c_4 + (-c_1 c_2 s_3 - c_1 s_2 c_3) s_4) c_5 + s_1 s_5 \\ r_{12} &= -((c_1 c_2 c_3 - c_1 s_2 s_3) c_4 + (-c_1 c_2 s_3 - c_1 s_2 c_3) s_4) s_5 + s_1 c_5 \\ r_{21} &= ((s_1 c_2 c_3 - s_1 s_2 s_3) c_4 + (-s_1 c_2 s_3 - s_1 s_2 c_3) s_4) c_5 - c_1 s_5 \\ r_{22} &= -((s_1 c_2 c_3 - s_1 s_2 s_3) c_4 + (-s_1 c_2 s_3 - s_1 s_2 c_3) s_4) s_5 - c_1 c_5 \end{aligned} \quad (29)$$

$$\therefore \theta_5 = a \tan 2(s_5, c_5) \quad (30)$$

Άλλη μια αλγεβρική λύση χρησιμοποιώντας την συνολική μήτρα μετασχηματισμού στην εξίσωση (7), μπορούμε να βρούμε τη λύση στην αντίστροφη κινηματική για το βραχίονα.

$${}^0T_5 = T_1^0 T_2^1 T_3^2 T_4^3 T_5^4 = G = \begin{bmatrix} r_{11} & r_{21} & r_{31} & x \\ r_{21} & r_{22} & r_{32} & y \\ r_{31} & r_{23} & r_{33} & z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (31)$$

Για να βρούμε τη λύση στην αντίστροφη κινηματική για την πρώτη άρθρωση θ_1 ως συνάρτηση των γνωστών στοιχείων $T_{base}^{end-effector}$ ο αντίστροφος μετασχηματισμός του συνδέσμου πολλαπλασιάζεται όπως φαίνεται στην εξίσωση:

$$\left[T_0^1\right]^{-1} * T_0^5 = \left[T_0^1\right]^{-1} * \left[T_0^1\right] * T_1^2 * T_2^3 * T_3^4 * T_4^5 \quad (32)$$

Όπου

$$\left[T_0^1\right]^{-1} * \left[T_0^1\right] = I$$

Σε αυτήν την περίπτωση, η παραπάνω εξίσωση δίνεται:

$$\left[T_0^1\right]^{-1} * T_0^5 = T_1^2 * T_2^3 * T_3^4 * T_4^5 \quad (33)$$

η λύση της εξίσωσης αυτής εξηγείται:

$$\begin{bmatrix} c_1 r_{11} + s_1 r_{21} & c_1 r_{12} + s_1 r_{22} & c_1 r_{13} + s_1 r_{23} & c_1 x + s_1 y \\ -s_1 r_{11} + c_1 r_{21} & -s_1 r_{12} + c_1 r_{22} & -s_1 r_{13} + c_1 r_{23} & -s_1 x + c_1 y \\ r_{31} & r_{23} & r_{33} & z - d_1 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & a_{31} & a_{14} \\ a_{21} & a_{22} & a_{32} & a_{24} \\ a_{31} & a_{23} & a_{33} & a_{34} \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (34)$$

$$a_{11} = ((c_2 c_3 - s_2 s_3) (c_4 - 1) + (-c_2 s_3 - s_2 c_3) s_4) c_5$$

$$a_{12} = -((c_2 c_3 - s_2 s_3) (c_4 - 1) + (-c_2 s_3 - s_2 c_3) s_4) s_5$$

$$a_{13} = -((c_2 c_3 - s_2 s_3) s_4 + (-c_2 s_3 - s_2 c_3) c_4)$$

$$a_{14} = -((c_2 c_3 - s_2 s_3) s_4 + (-c_2 s_3 - s_2 c_3) c_4) d_5 + (c_2 c_3 - s_2 s_3) a_4 + c_2 a_3$$

$$a_{21} = -s_5$$

$$a_{22} = -c_5$$

$$a_{23} = 0$$

$$a_{24} = 0$$

$$a_{31} = ((s_2 s_3 + c_2 c_3) (c_4 - 1) + (c_2 c_3 - s_2 s_3) s_4) c_5$$

$$a_{32} = -((s_2 c_3 + c_2 s_3) (c_4 - 1) + (c_2 c_3 - s_2 s_3) s_4) s_5$$

$$a_{33} = -(s_2 c_3 + c_2 s_3) s_4 + (c_2 c_3 - s_2 s_3) c_4$$

$$a_{34} = -(s_2 c_3 + c_2 s_3) s_4 + (c_2 c_3 - s_2 s_3) c_4) d_5 + (s_2 c_3 + c_2 s_3) a_4 + s_2 a_3$$

=

$$\begin{bmatrix} ((c_{23} (c_4 - 1) - s_{234}) c_5 & -((c_{23} (c_4 - 1) - s_{234}) s_5 & -s_{234} & -s_{234} + c_{23} a_4 + s_2 a_3 \\ -s_5 & -c_5 & 0 & 0 \\ (s_{23} (c_4 - 1) - c_{23} s_4) c_5 & -(s_{23} (c_4 - 1) + c_{23} s_4) s_5 & c_{234} & c_{234} d_5 + s_{23} a_4 + s_2 a_3 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

(35)

Από την εξίσωση (34), (35) βρίσκουμε:

$$\begin{aligned} c_1 y - s_1 x &= 0 \\ \therefore \theta_1 &= a \tan 2(y, x) \end{aligned} \quad (36)$$

Για να βρούμε τις άλλες μεταβλητές, οι ακόλουθες εξισώσεις λαμβάνεται με παρόμοιο τρόπο.

$$\left[T_0^1 * T_1^2 \right]^{-1} * T_0^5 = T_2^3 * T_3^4 * T_4^5$$

$$\left[T_0^1 * T_1^2 * T_2^3 \right]^{-1} * T_0^5 = T_3^4 * T_4^5$$

$$\left[T_0^1 * T_1^2 * T_2^3 * T_3^4 \right]^{-1} * T_0^5 = T_4^5$$

Τώρα, μπορούμε να υπολογίσουμε την θ_2 από την παραπάνω εξίσωση:

$$\begin{aligned} c_1 x + s_1 y &= -s_{234} d_5 + c_{23} a_4 + c_2 a_3 \\ c_1 x + s_1 y &= -s_{234} d_5 + c_2 (c_3 a_4 + a_3) \\ c_2 &= \frac{c_1 x + s_1 y + s_{234} d_5}{(c_3 a_4 + a_3)} \end{aligned}$$

$$\begin{aligned} z - d_1 &= c_{234} d_5 + s_{23} a_4 + s_2 a_3 \\ z - d_1 &= c_{234} d_5 + s_2 (s_3 a_4 + a_3) \\ s_2 &= \frac{z - d_1 - c_{234} d_5}{(s_3 a_4 + a_3)} \end{aligned}$$

$$\therefore \theta_2 = a \tan 2(s_2, c_2) \quad (37)$$

Παρόμοια μπορούμε να βρούμε την θ_3 :

$$\begin{aligned} c_1 x + s_1 y &= -s_{234} d_5 + c_{23} a_4 + c_2 a_3 \\ z - d_1 &= c_{234} d_5 + s_{23} a_4 + s_2 a_3 \end{aligned}$$

$$\begin{aligned}
[c_1 x + s_1 y - s_{234} d_5]^2 &= [c_{23} a_4 + c_2 a_3]^2 \\
[z - d_1 + c_{234} d_5]^2 &= [s_{23} a_4 + s_2 a_3]^2 \\
c_2 &= \frac{[c_1 x + s_1 y - s_{234} d_5]^2 + [z - d_1 + c_{234} d_5]^2 - a_3 a_4}{2a_3 a_4} \\
s_3 &= \pm \sqrt{1 - c_3^2} \\
\therefore \theta_3 &= a \tan 2(s_3, c_3) \tag{38}
\end{aligned}$$

$$\begin{aligned}
s_{234} &= -(c_1 r_{13} + s_1 r_{23}) \\
c_{234} &= r_{33} \\
\therefore \theta_{234} &= a \tan 2(s_{234}, c_{234}) \tag{39}
\end{aligned}$$

Επίσης:

$$\begin{aligned}
s_5 &= s_1 r_{11} - c_1 r_{21} \\
c_5 &= s_1 r_{12} - c_1 r_{22} \\
\therefore \theta_5 &= a \tan 2(s_5, c_5) \tag{39}
\end{aligned}$$

Παρόμοιες διαδικασίες πολλαπλασιασμού για την άρθρωση 4:

$$\begin{aligned}
(T_{01} * T_{12} * T_{23})^{-1} G &= T_{34} * T_{45} \tag{40} \\
\begin{bmatrix} * & * & c_{123} r_{13} + c_{23} s_1 r_{23} + s_{23} r_{33} & * \\ * & * & -s_{23} c_1 r_{13} - s_{123} r_{23} + c_{23} r_{33} & * \\ * & * & * & * \\ 0 & 0 & 0 & 1 \\ (c_4 - 1)c_5 & -(c_4 - 1)s_5 & -s_4 & -s_4 d_5 + d_4 \\ s_4 c_5 & -s_4 s_5 & c_4 & c_4 d_5 \\ s_5 & c_5 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} &= \\
s_4 &= -(c_{123} r_{13} + c_{23} s_1 r_{23} + s_{23} r_{33}) \\
c_4 &= -(s_{23} c_1 r_{13} - s_{123} r_{23} + c_{23} r_{33})
\end{aligned}$$

Που έχει ως αποτέλεσμα:

$$\therefore \theta_4 = a \tan 2(s_4, c_4) \tag{41}$$

3.Μοντελοποίηση

3.1 Γενικά

Βαδίζοντας στον ψηφιακό κόσμο, τα μηχανολογικά σχέδια είχαν την ανάγκη να μεταφερθούν από δυσδιάστατη απεικόνιση τους στην τρισδιάστατη. Διάφορες εταιρίες ανέπτυξαν τρισδιάστατα λογισμικά προγράμματα με σκοπό να αυξήσουν την παραγωγικότητα τους και ασφαλώς τα κέρδη τους. Το Solidworks είναι ένα από αυτά. Το λογισμικό με το οποίο πραγματοποιείται η διαδικασία είναι το ένα ολοκληρωμένο σχεδιαστικό πακέτο το οποίο βρίσκει ευρεία εφαρμογή στην σύγχρονη Βιομηχανία. Συγκεκριμένα το SolidWorks αποτελεί ένα πακέτο 3D CAD μοντελοποίησης, το οποίο υποστηρίζεται από το λειτουργικό σύστημα Microsoft Windows και έχει αναπτυχθεί από την εταιρία Dassault Systèmes SolidWorks Corp., θυγατρική της Dassault Systèmes, S. A. Το SolidWorks χρησιμοποιείται σήμερα από περισσότερους από 1,3 εκ. μηχανικούς και σχεδιαστές και περισσότερες από 130.000 εταιρίες ανά τον κόσμο. Ιστορικά το SolidWorks εμφανίστηκε στη διεθνή αγορά το Δεκέμβριο του 1993 από τον Jon Hirschtick, οποίος με έδρα το Concord, Μασαχουσέτη, ΗΠΑ προσέλαβε μια ομάδα μηχανικών για την δημιουργία μιας εταιρείας, η οποία κα ανέπτυξε 3D CAD λογισμικό το οποίο ήταν εύκολο στη χρήση, προσιτό και ικανό να λειτουργήσει σε έναν συμβατικό προσωπικό υπολογιστή. Το 1997 Η Dassault Systèmes, εταιρία γνωστή στο χώρο των λογισμικών μοντελοποίησης για το CATIA CAD λογισμικό της, απέκτησε την εταιρεία και σήμερα κατέχει το 100% των μετοχών της.

Τα μοντέλα που παράγει μπορούν να χρησιμοποιηθούν με πολλούς τρόπους, όπως την προσομοίωση της πραγματικής διαδικασίας στησίματος μίας μηχανολογικής κατασκευής κομμάτι- κομμάτι, ελέγχοντας και τη γεωμετρία.

Αξίζει να αναφερθεί πως η σχεδίαση σε δύο διαστάσεις συνεχίζεται επειδή είναι η πιο βασική μορφή τεχνικής σχεδίασης που είναι γνωστή σε όλες τις ειδικότητες των μηχανικών. Μερικά από τα βασικά πλεονεκτήματα της 3D σχεδίασης είναι:

- 1) Το τελικό προϊόν εκτιμείται καλύτερα, έτσι έχουμε λιγότερα πρωτότυπα για την ανάπτυξή του.
- 2) Κατά τη σχεδίαση του προϊόντος υπάρχουν περισσότερες πληροφορίες σχετικά με τα χαρακτηριστικά του προϊόντος
- 3) Περιορισμός λαθών, πιο γρήγορη εξαγωγή του προϊόντος στην αγορά
- 4) Ο σχεδιασμός γίνεται πολύ πιο γρήγορα
- 5) Το γεωμετρικό μοντέλο μπορεί να συνδεθεί με συστήματα CAM και FEA για προσδιορισμό τεχνικών και λειτουργικών χαρακτηριστικών καθώς και της διαδικασίας παραγωγής του.
- 6) Μπορούμε να κατασκευάσουμε το 3D μοντέλο μας και αν για κάποιο λόγο υπάρχει ανάγκη απεικόνισης του σε 2D μπορεί να γίνει πολύ εύκολα στο Solidworks
- 7) Στο assembly μπορούμε να αποφύγουμε collision detection

Επίσης μπορούμε να χρησιμοποιήσουμε το API(Application Programming Interface) για να λύσουμε μηχανολογικά προβλήματα αλλά και να ενσωματώσουμε κώδικα σε διάφορες γλώσσες προγραμματισμού (κυρίως Visual C ++,VBA (Visual Basic for Applications)).

Πλατφόρμες που ενσωματώνει το πακέτο του SolidWorks είναι:

- Το **SolidWorks Simulation**, το οποίο αποτελεί ένα εργαλείο για την επικύρωση του μηχανολογικού σχεδιασμού. Το εργαλείο αυτό αναδεικνύει στους μηχανικούς τη συμπεριφορά του σχεδιασμένου αντικείμενου στο φυσικό περιβάλλον για το οποίο προορίζεται. Συνεπώς με την πλατφόρμα του SolidWorks Simulation μπορεί να μελετηθεί η συμπεριφορά του σχεδιασμένου αντικείμενου υπό την επίδραση δυναμικών και θερμικών φορτίων, να ευρεθεί η ίδιο συχνότητα του.

Το **SolidWorks Motion**, που αποτελεί ένα εικονικό εργαλείο προτυποποίησης, το οποίο παρέχει δυνατότητες προσομοίωσης της κίνησης ενός μηχανισμού, ελέγχοντας τον τρόπο αυτόν την ομαλή και επιδιωκόμενη λειτουργία του.

- Το **SolidWorks Flow Simulation**, είναι ένα πλήρες εικονικό εργαλείο προσομοίωσης της ροής ρευστό και της μεταφοράς θερμότητας, προσφέροντας τη δυνατότητα στους μηχανικούς να διεξάγουν δοκιμές σε εικονικά πρωτότυπα.

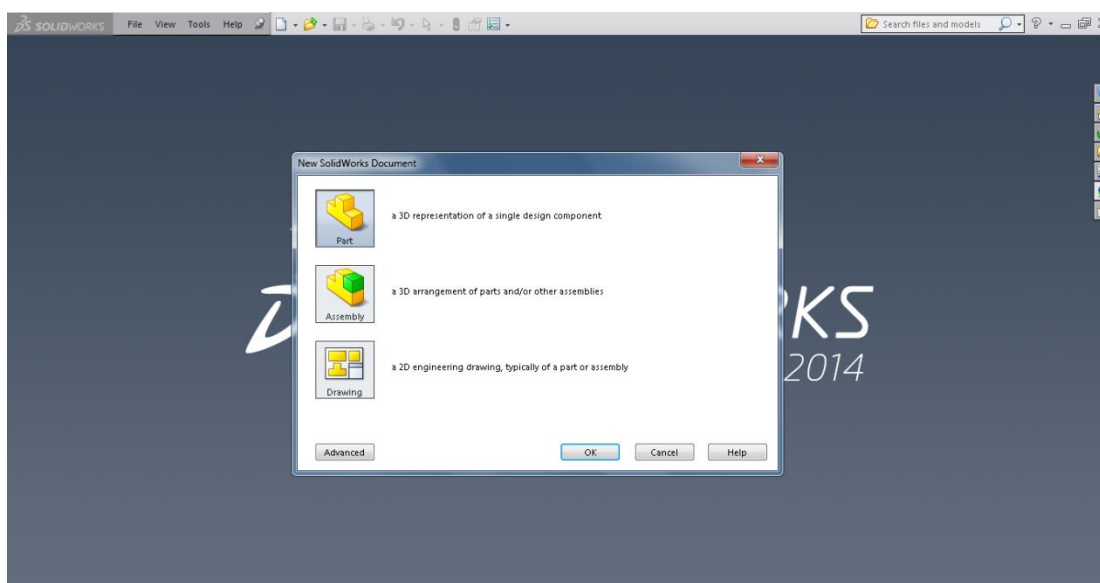
Σε αυτό το στάδιο ο ρομποτικός βραχίονας μοντελοποιείτε χρησιμοποιώντας το SolidWorks 2014, αυτό το βήμα είναι σημαντικό για διάφορους λόγους. Ένας από τους σημαντικότερους είναι ότι επιτρέπει την χρήση του ρομποτικού βραχίονα ουσιαστικά ακόμη και αν δεν είναι φυσικά διαθέσιμος.

Το βήμα μοντελοποίηση αφορούσε την αντίστροφη μηχανική του ρομποτικού βραχίονα, έπρεπε να γίνει με την αποσυναρμολόγηση του μοντέλου για να αποφευχθούν τυχόν παραβλήματα βραχίονας δεν είναι βιομηχανικός αλλά εκπαιδευτικός, δεν υπάρχει κάποιο manual για το ακριβές σχέδιο του έτσι ένα παχύτερο χρησιμοποιήθηκε για την απόκτηση ακριβής μέτρησης για τα μεμονωμένα μέλη-εξαρτήματα(**parts**) τα οποία στη συνέχεια συνδέονται μεταξύ τους σε ένα υποσυνγκρότημα(**subassemblies**) και τελικά προστίθενται σε μία πλήρη συναρμολόγηση(**assembly**) του βραχίονα.

3.2 Σχεδίαση αρπάγης

Ξεκινώντας το πιο σημαντικό αλλά και πιο σύνθετο μέλος του ρομποτικού βραχίονα είναι η αρπάγη (ακρραία απόληξη) γι αυτό το λόγω επιλέχθηκε να αναπτυχθεί αναλυτικότερα. Τα υπόλοιπα μέλη δημιουργούνται με την ίδια διαδικασία και σχεδόν με τις ίδιες εντολές.

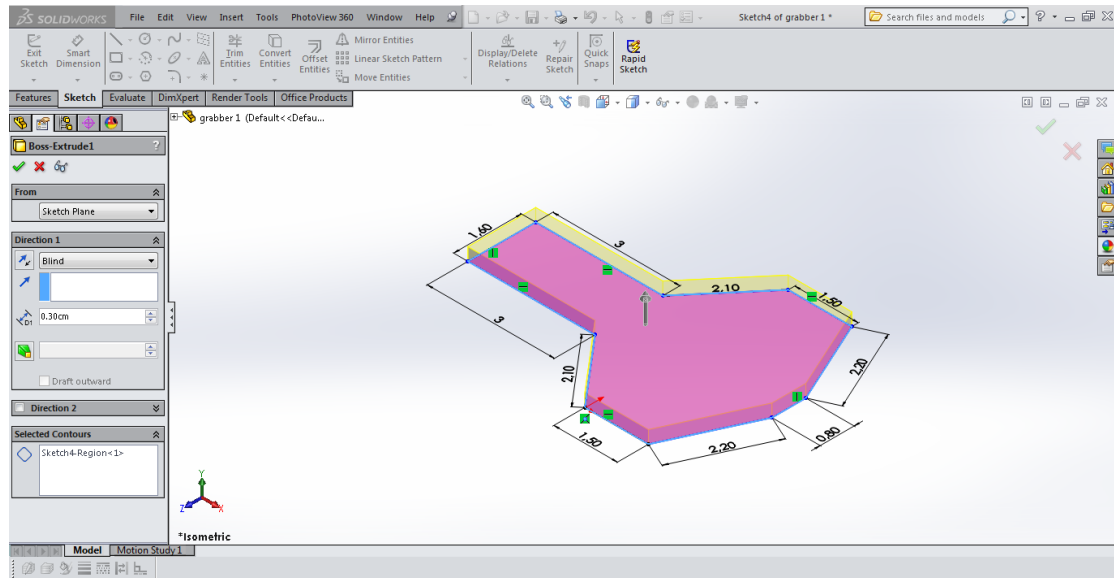
Επίσης μέσω του PhotoView 360 ένα πρόσθετο το οποίο παρέχεται με το SolidWorks είναι δυνατή η φωτορεαλιστική απεικόνιση των αντικειμένων. Τονίζεται ακόμη ότι υπάρχουν πολλοί και διαφορετικοί τρόποι να σχεδιάσουμε ένα 3D αντικείμενο.



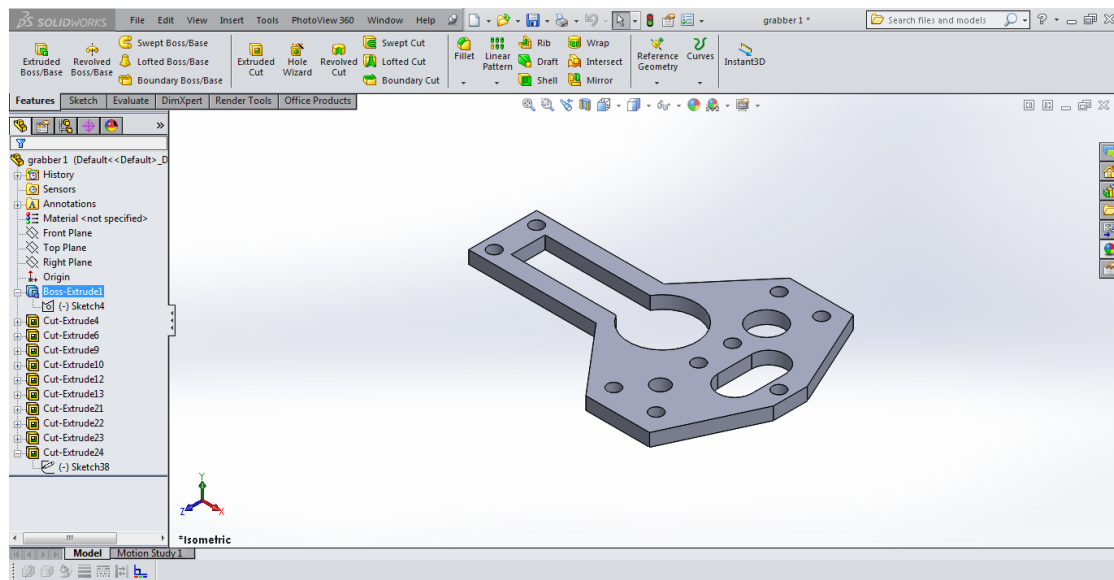
Δημιουργία νέου part.

3.2.1 Βάση αρπάγης

Αφού έχουμε σχεδιάσει το περίγραμμα της βάσης (sketch) σε “Top Plane” με την εντολή “Extruded Boss/Base” την μεταφέρουμε σε 3D

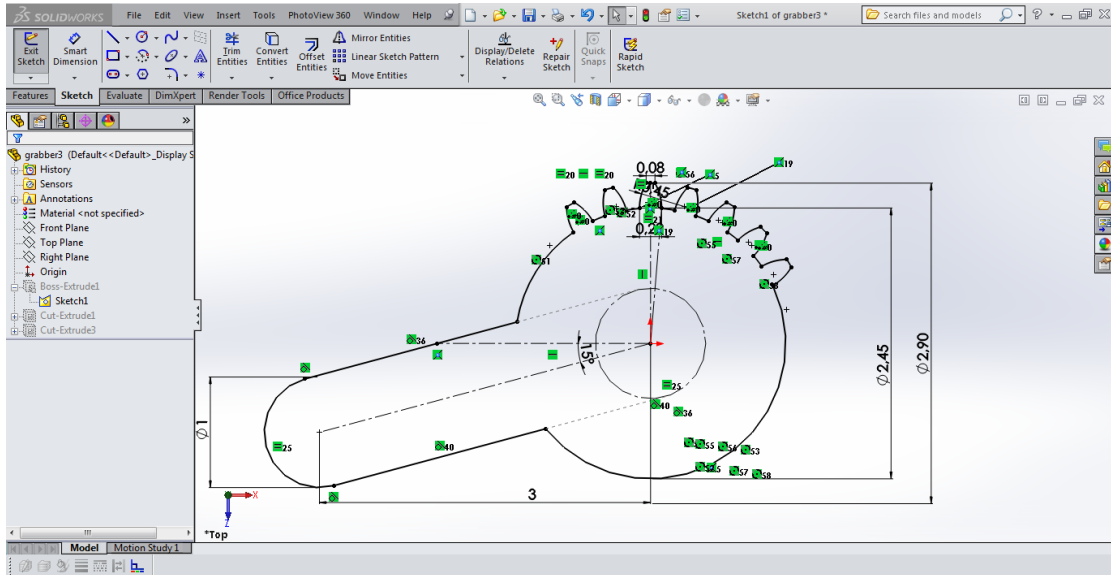


Με την εντολή “Cut Extrude” αφού σχεδιάζουμε τις οπές πάνω στην επιφάνεια κάνουμε τα ανοίγματα-κοψίματα.

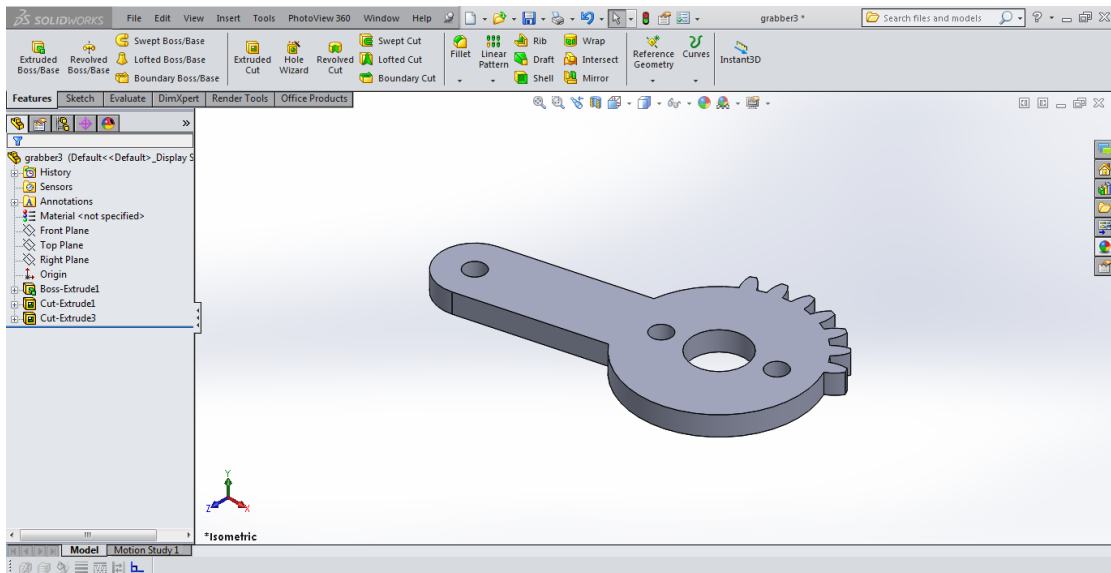


3.2.2 Μηχανισμός κίνησης

Στο μηχανισμό κίνησης χρειάστηκε περισσότερη λεπτομέρεια που δεν θα μπορούσαμε να δημιουργήσουμε εύκολα απευθείας σε τρισδιάστατο σχέδιο για αυτό το λόγω βλέπουμε πόσο μας βοηθάει η καλή σχεδίαση σε δυο διαστάσεις.

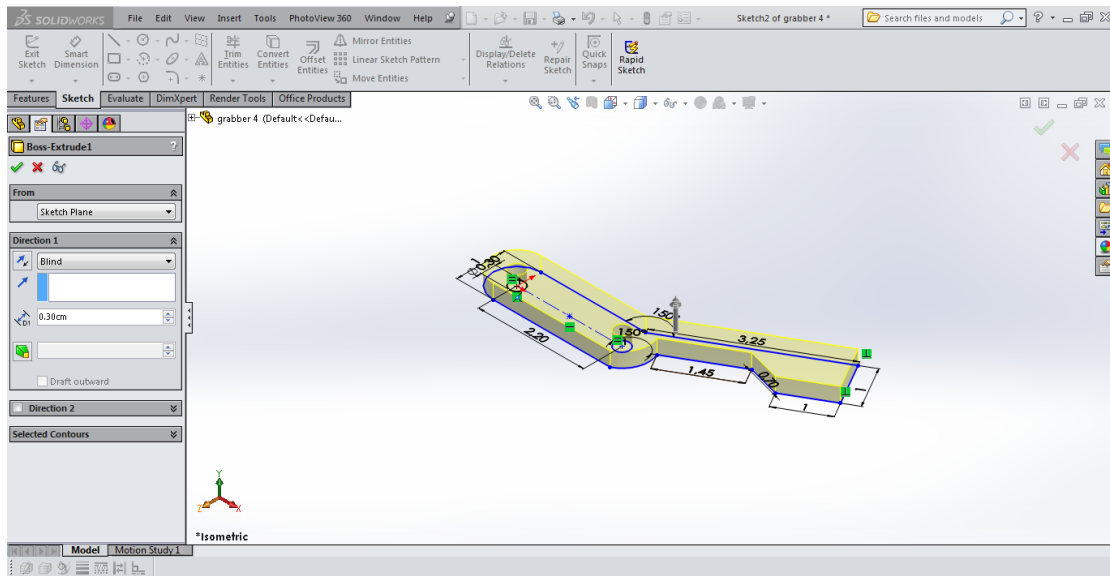


Με τη εντολή “Extruded Boss/Base” την μεταφερόμαστε σε 3D και με την “Cut Extrude” ανοίγουμε τις 3 μικρές οπές.

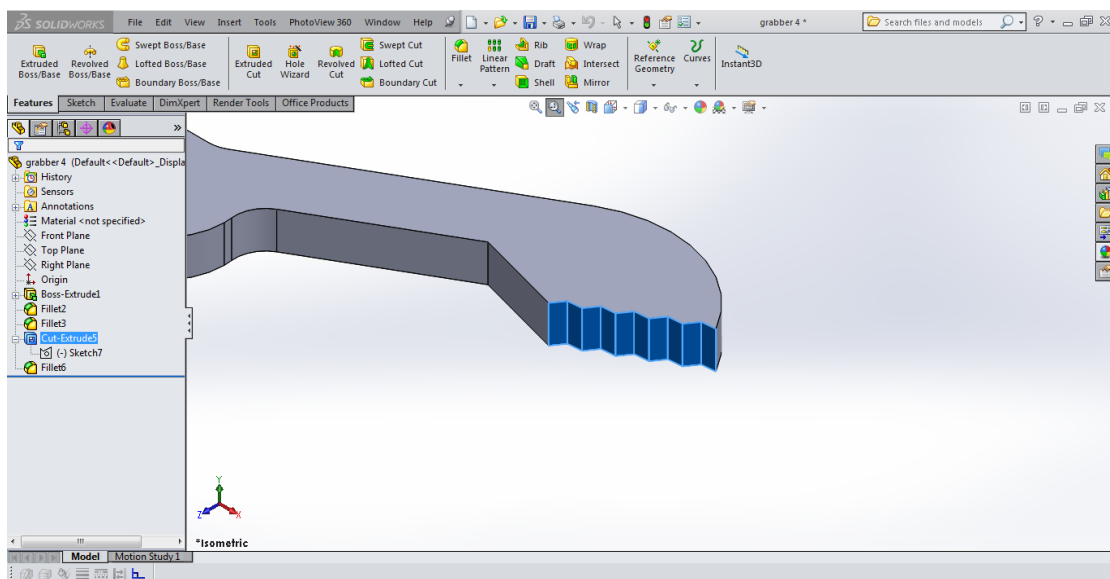


3.2.3 Λαβή-Δαγκάνα

Η λαβή δημιουργήθηκε επίσης σε “Top Plane” και μια η πιο σημαντική λεπτομέρεια είναι η προιονωτή απόληξη που έχει σχεδιαστεί όπως ακριβός και στην πραγματική λαβή.

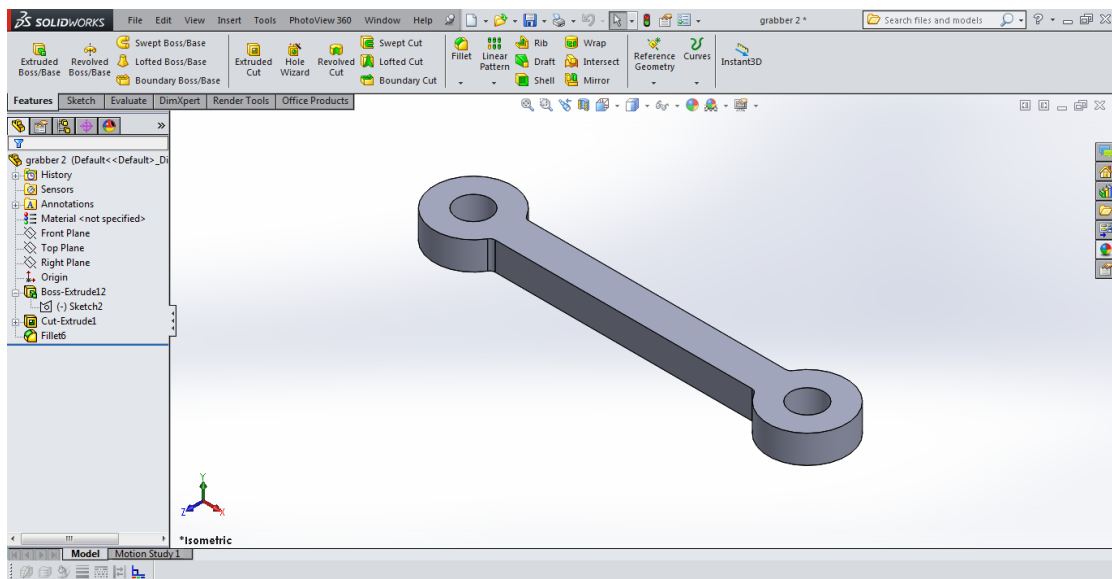
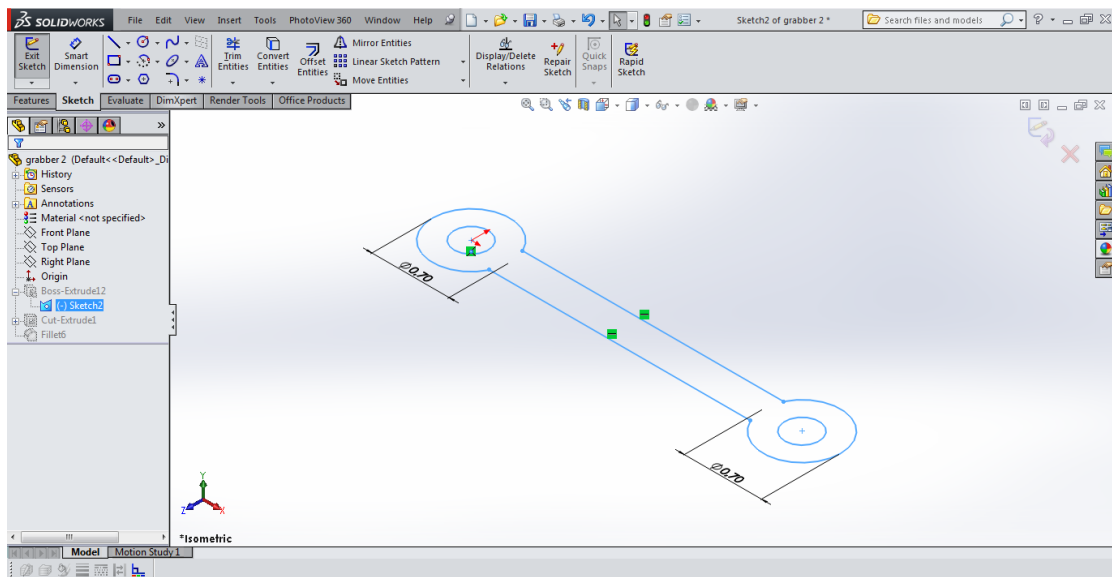


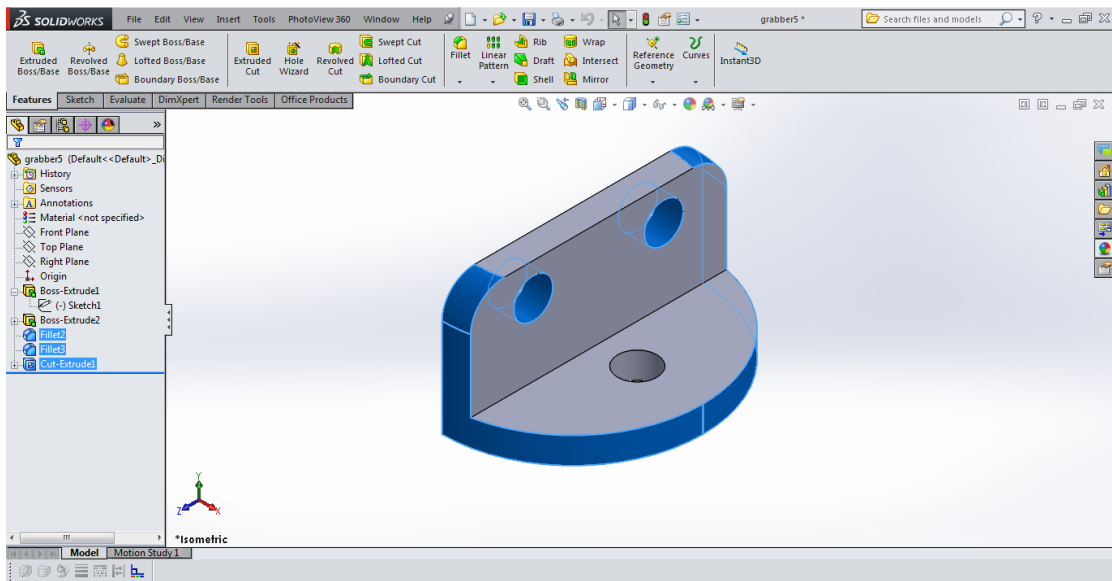
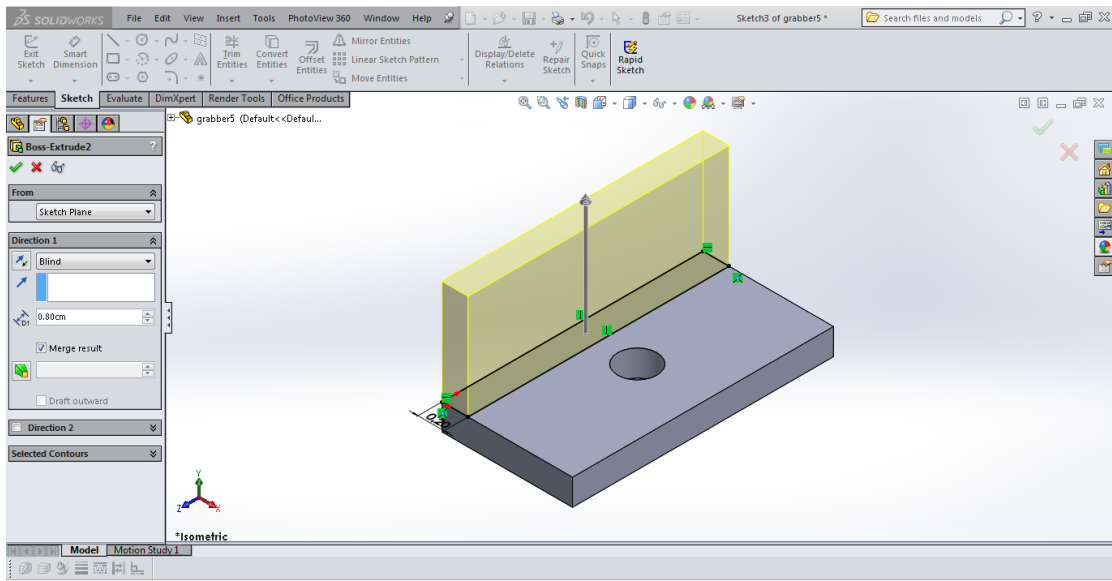
Μία σημαντική εντολή είναι η “Fillet” δηλαδή η στρογγυλοποίηση μιας εσωτερικής ή εξωτερικής γωνίας ενός σχεδιασμού μέρους



3.2.4 Στήριξη

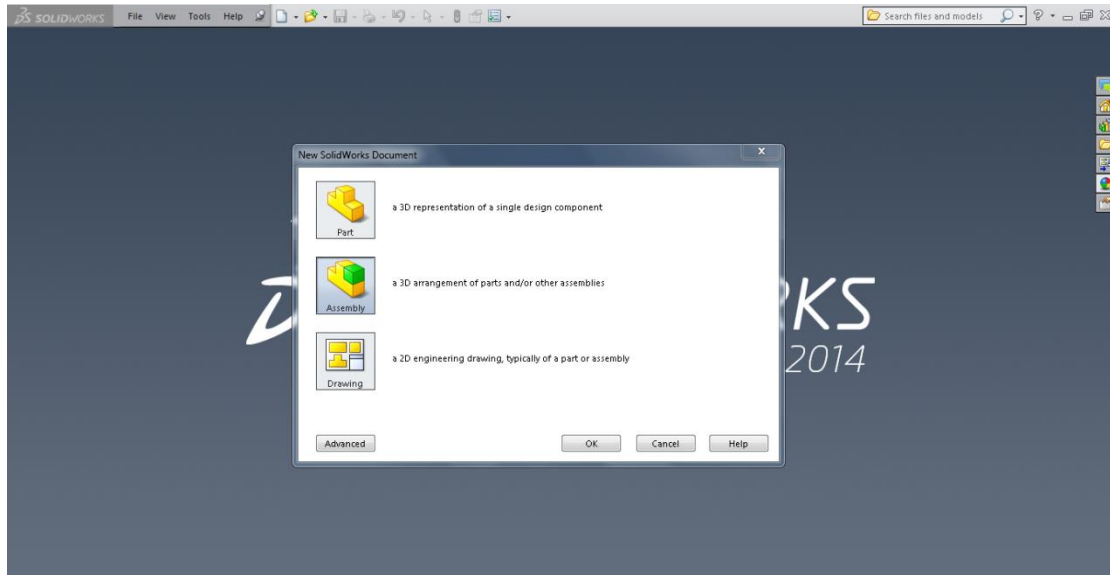
Η στήριξη παρομοίως δημιουργείται εύκολα όπως φαίνεται στις παρακάτω εικόνες. Τα κομμάτια που θα δημιουργήσουμε είναι για την στήριξη της λαβής αλλά και για την στήριξη του καρπού της αρπάγης.



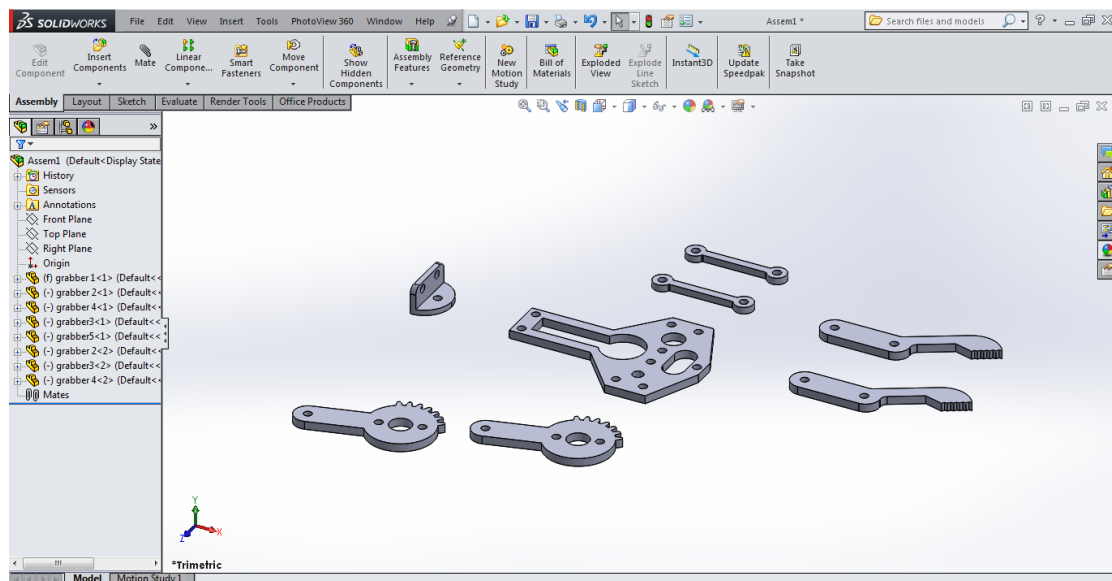


3.3 Συναρμολόγηση Αρπάγης

Αφού έχουμε σχεδιάσει όλα τα εξαρτήματα προχωράμε προς την τελική μορφή της αρπάγης δηλαδή στην ένωση όλων των παραπάνω ,ανοίγοντας λοιπόν ένα νέο παράθυρο επιλέγουμε “Assembly”



Με την επιλογή “insert component” μπορούμε να εισάγουμε οποιοδήποτε εξάρτημα θέλουμε για να συναρμολογήσουμε το κάθε ένα ξεχωριστά ή ακόμη και όλα μαζί.



Είναι σημαντικό να γνωρίζουμε πώς το πρώτο εξάρτημα που βάζουμε στο παράθυρο της συναρμολόγησης “assembly” είναι πάντα από προεπιλογή “fixed”, δηλαδή δεν μπορεί να κινηθεί σε σχέση με την υπόλοιπη συναρμολόγηση ωστόσο μπορούμε να το κάνουμε “float” σε οποιαδήποτε στιγμή δηλαδή να το ελευθερώσουμε στο χώρο και σχετικά με τα υπόλοιπα εξαρτήματα. Αυτό δίνει ένα πλαίσιο αναφοράς για όλα τα αλλά mates, και βοηθά στην πρόληψη απροσδόκητων κινήσεων των εξαρτημάτων όταν προστίθενται καινούργια μελή.

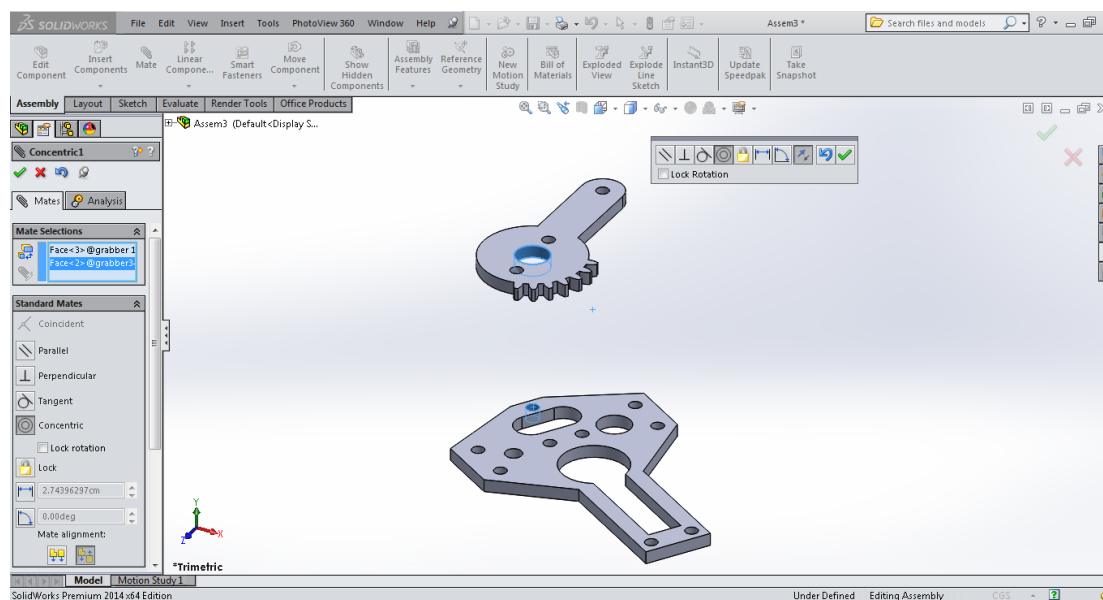
1) Ένα σταθερό συστατικό έχει ένα (f) πριν από το όνομά του στο “Feature Manager Tree”.

2) Ένα υπό καθορισμό συστατικό έχει ένα (-) πριν από το όνομά του στο “Feature Manager Tree”.

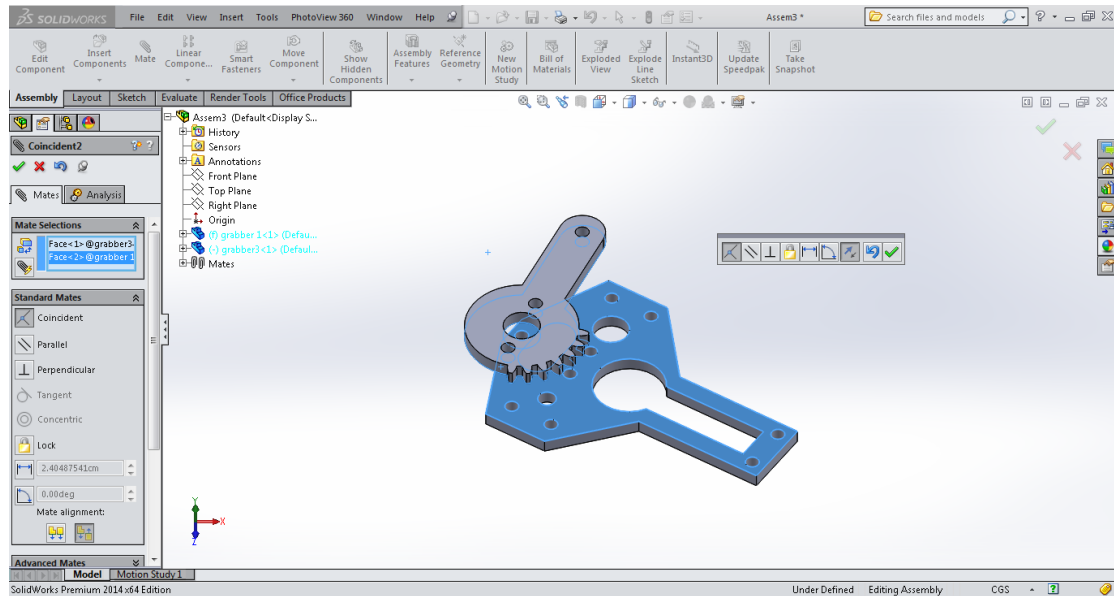
3) Ένα πλήρως ορισμένο συστατικό δεν έχει πρόθεμα.

Για να συναρμολογήσουμε την αρπάγη εισάγουμε πρώτα την βάση με την εντολή “insert components” και έπειτα τον μηχανισμό κίνησης και επιλέγουμε τα κατάλληλα “mates” σε σημεία, επιφάνειες, ακμές, πλευρές, προσόψεις ή κυκλικά μέρη των κομματιών μας που θέλουμε να ταιριάξουν.

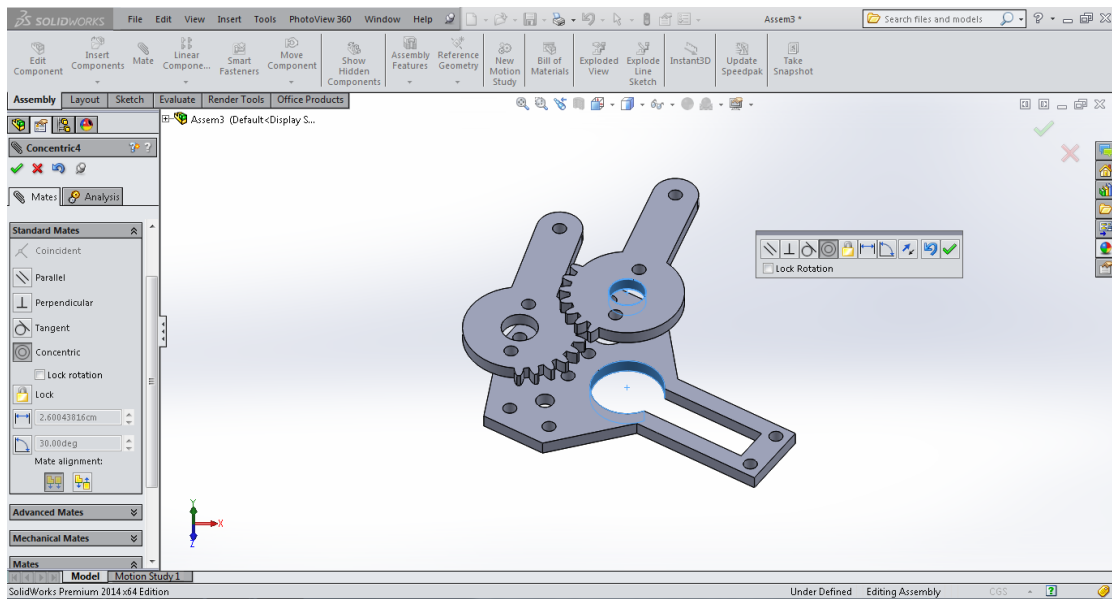
Στην εικόνας παρακάτω βλέπουμε με μπλε χρώμα τα “mate”, το πρώτο μεταξύ των δυο οπών με την εντολή “concentric” (ομόκεντρος).



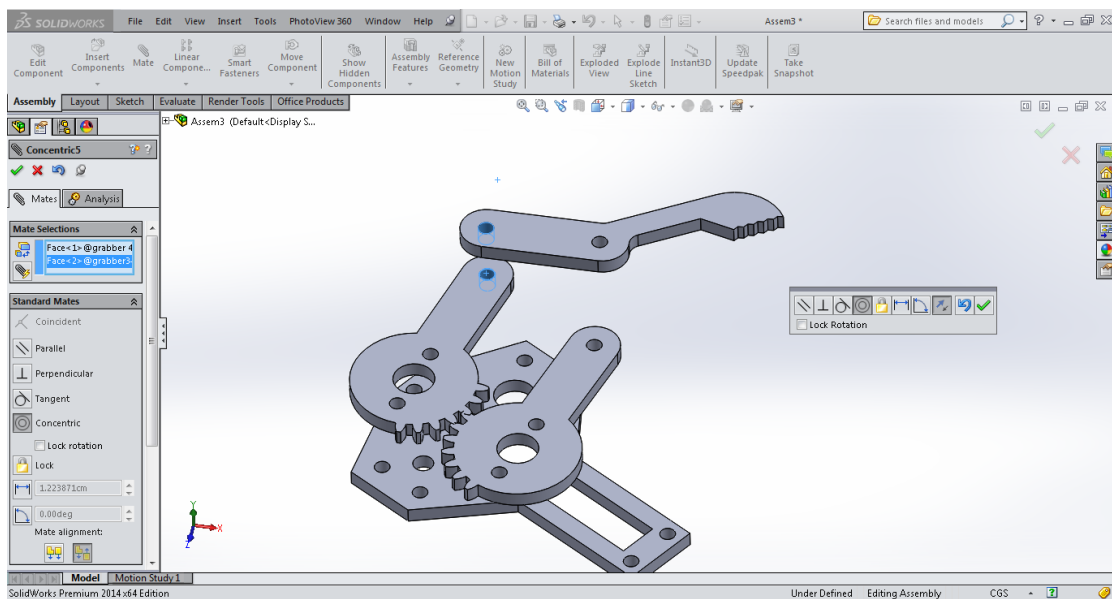
Αμέσως μετά με την εντολή “coincident” (συμπίπτων) επικολούμε τον μηχανισμό στήριξης στην βάση. Πλέον η μόνη ελευθερία που έχει ο μηχανισμός είναι αυτός της περιστροφής.



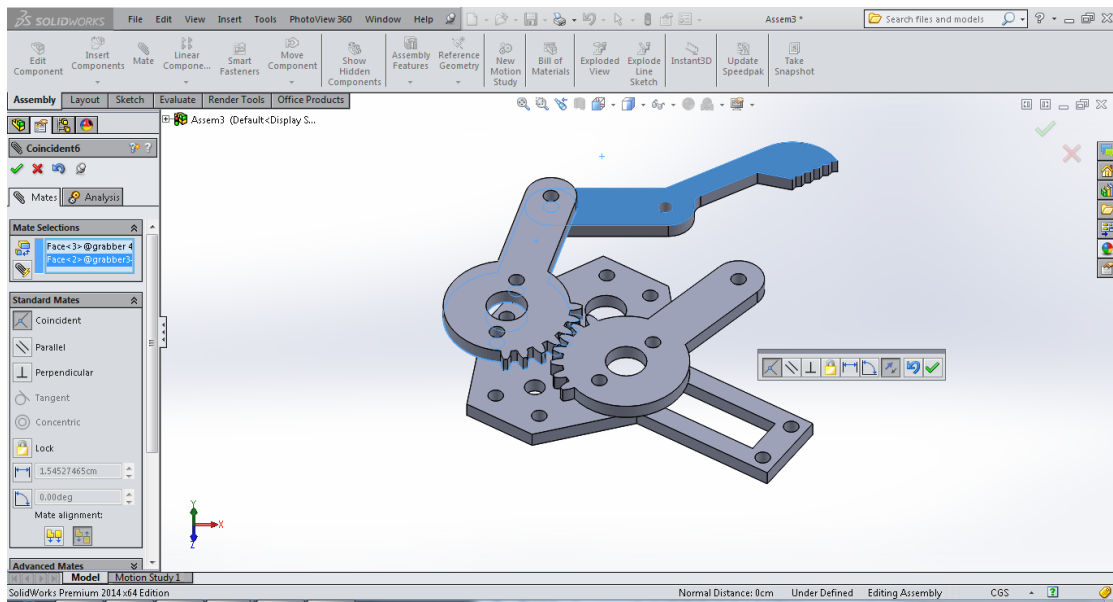
Σε επόμενη φάση εισάγουμε το ίδιο μέλος του μηχανισμού κίνησης και το επικολούμε επάνω στην βάση με ακριβώς της δυο ίδιες εντολές που χρησιμοποιήσαμε στο πρώτο αλλά με την μόνη διαφορά ότι στο δεύτερο μέλος κάνουμε “flip alignment” (αναστροφή της ευθυγράμμισης).



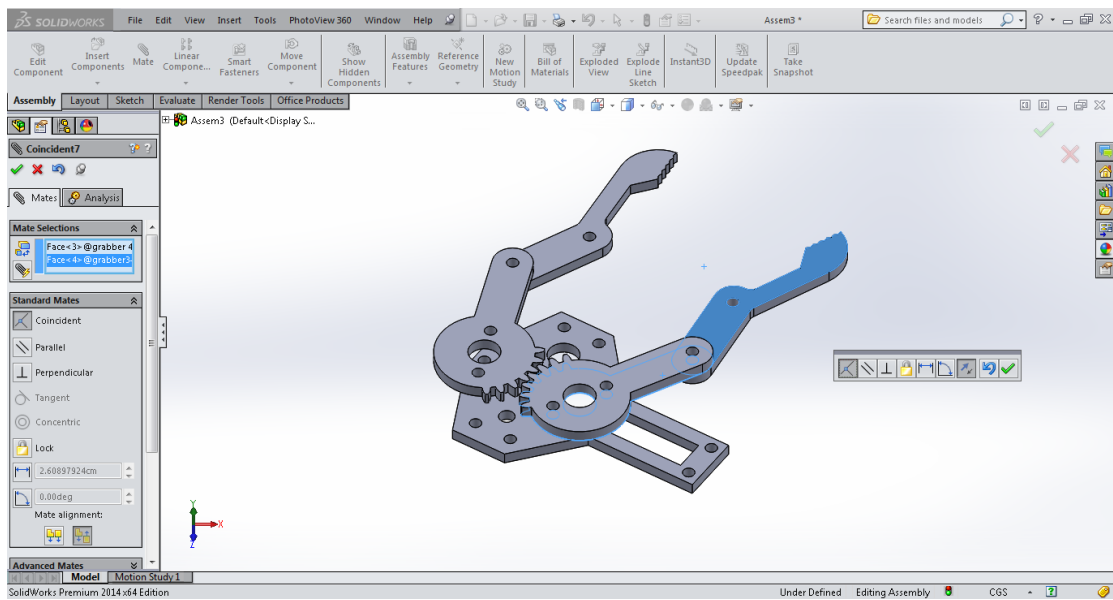
Όπως και προηγουμένως εισάγουμε την λαβή και χρησιμοποιούμε την εντολή “concentric” .



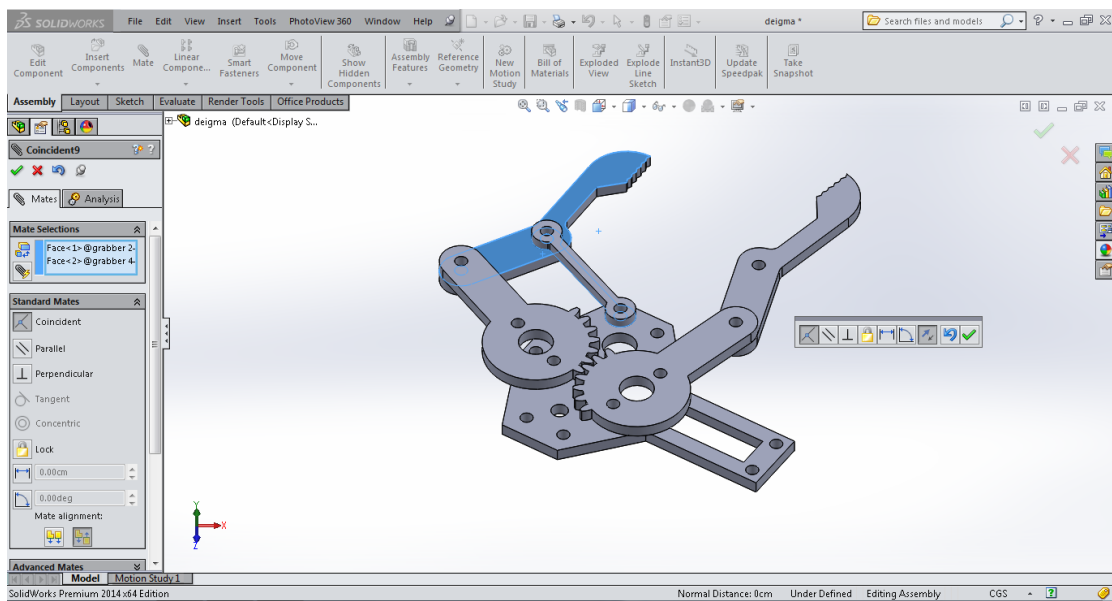
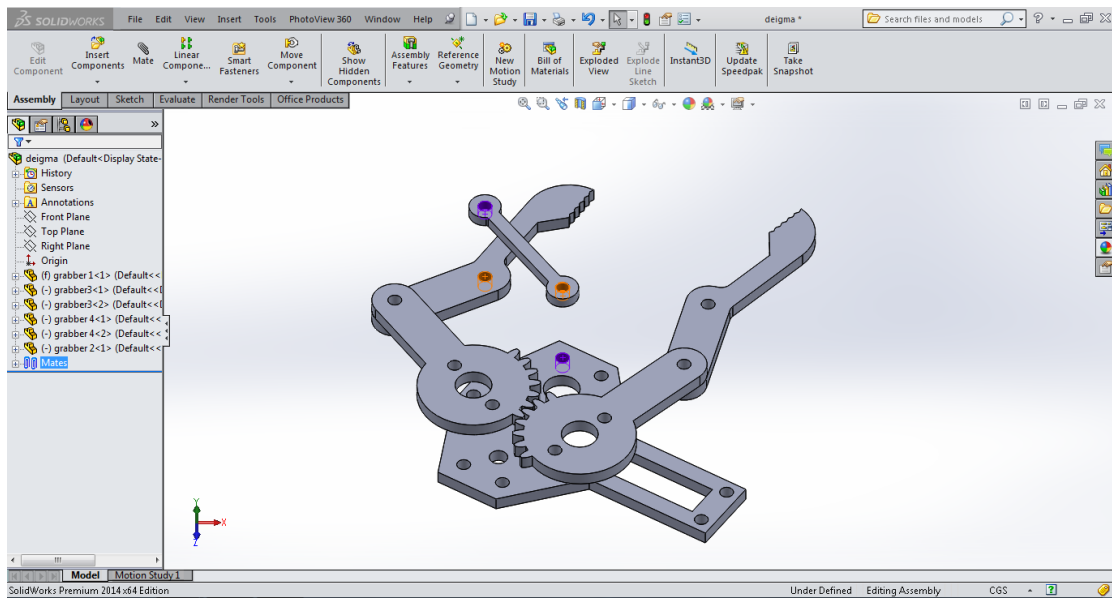
Με την εντολή “coincident” προσέχουμε απλά να επικολήσουμε την λαβή από το κάτω μέρος της βασης όπως ακριβος είναι και η πραγματική.



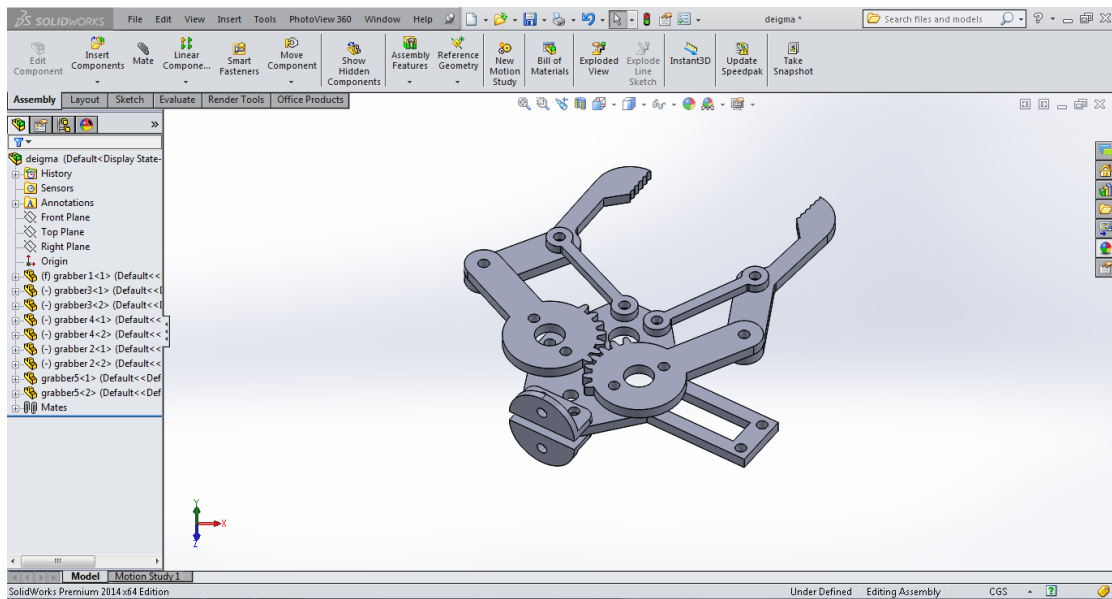
Σε αυτό το στάδιο εισάγουμε ξανά το μέλος της λαβής και αφού κάνουμε “ flip alignment” με τις δυο πλεον βασικές εντολές “concentric και coincident” επικολούμε την λαβή στο μηχανισμό κίνησης.



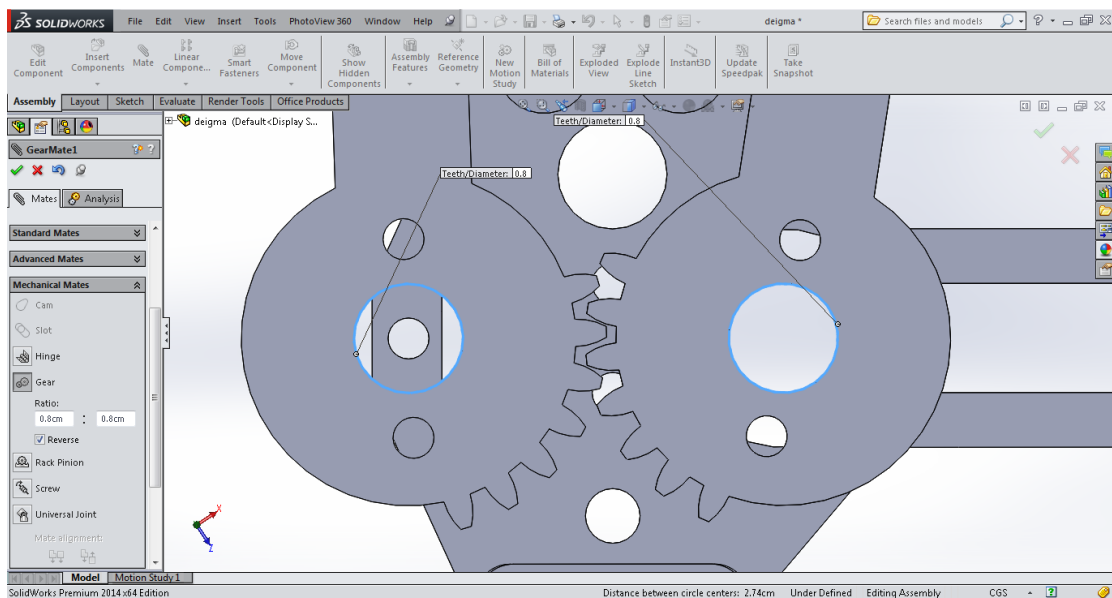
Τέλος εισάγουμε την στήριξη με ακριβώς την ίδια διαδικασία.



Στην εικόνα βλέπουμε την αρπάγη ολοκληρωμένη σχεδιαστικά αλλά όπως είναι τώρα ο μηχανισμός κίνησης της δεν είναι συγχρονισμένος δηλαδή ο κάθε μηχανισμός κινείται μόνος του και ανεξάρτητα από τον άλλο.

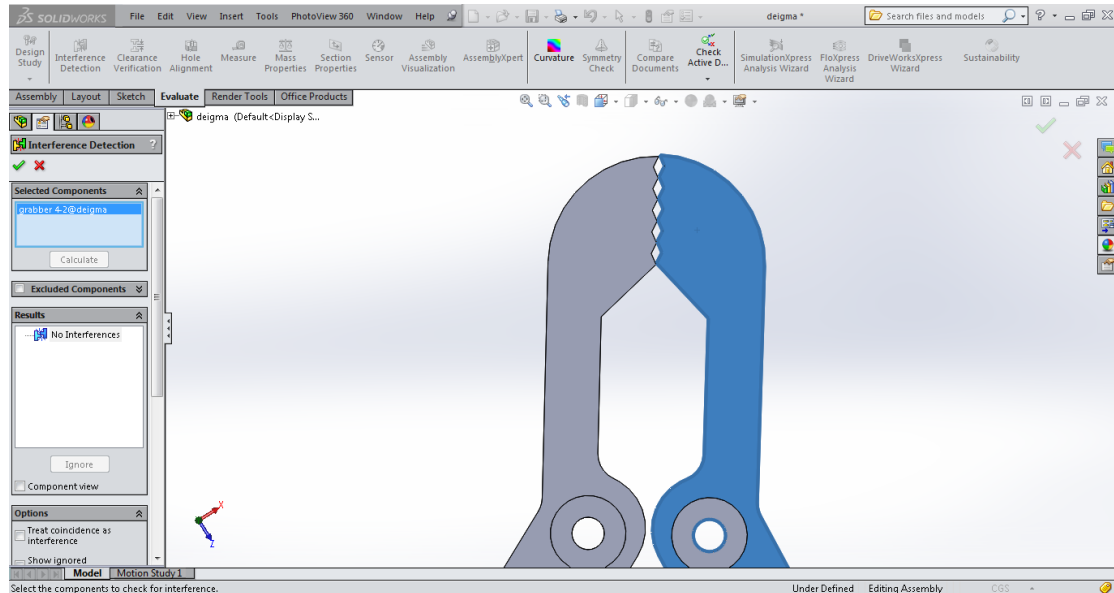


Έτσι λοιπόν θα χρησιμοποιηθούμε ακόμη μια εντολή την “Gear mate” (μηχανισμός) από τα “Mechanical mates”. Η “Gear mate” αναγκάζει δύο μέρη να περιστρέφονται το ένα σε σχέση με το άλλο σχετικά με τους επιλεγμένους άξονες. Σωστές επιλογές για τον άξονα περιστροφής περιλαμβάνουν κυλινδρικές και κωνικές επιφάνειες, άξονες, και γραμμική άκρες.

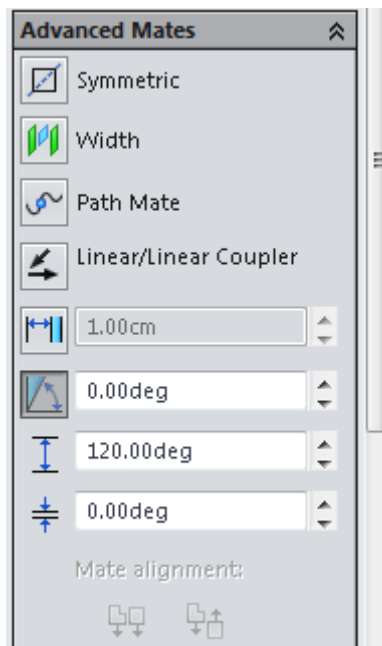


Πολύ σημαντικό είναι η ανίχνευση παρεμβολών “interference detection” μεταξύ των μελών που μας βοηθά να εξετάσουμε, να διορθώσουμε και να αξιολογήσουμε αυτές τις παρεμβολές. Η ανίχνευση παρεμβολών είναι χρήσιμο σε σύνθετα συγκροτήματα, όπου μπορεί να είναι δύσκολο να καθοριστεί οπτικά κατά πόσο τα

συστατικά αλληλεπιδρούν μεταξύ τους. Στην περίπτωση μας την ανίχνευση παρεμβολών την χρησιμοποιήσαμε και συνολικά για όλη την κατασκευή της αρπαγής αλλά και για να δούμε που ακριβώς είναι το σημείο όπου η λαβή είναι κλειστή.



Ακόμη μια εντολή που εισάγουμε στην λαβή είναι η "angle" (γωνία) από τα "advanced mates". Δίνουμε λοιπόν μέγιστο και ελάχιστο όριο στην γωνία δηλαδή στο άνοιγμα που θα έχει η λαβή έτσι ώστε να μην έχουμε σύγκρουση των μελών.



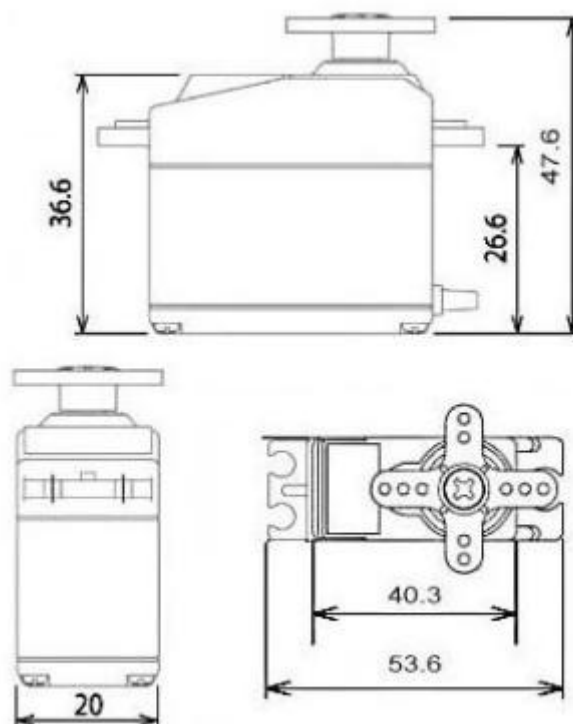
Η Αρπάγη σε πραγματική και φωτορεαλιστική απεικόνιση.





3.4 Σχεδίαση Σέρβο κινητήρα ρομποτικού βραχίονα

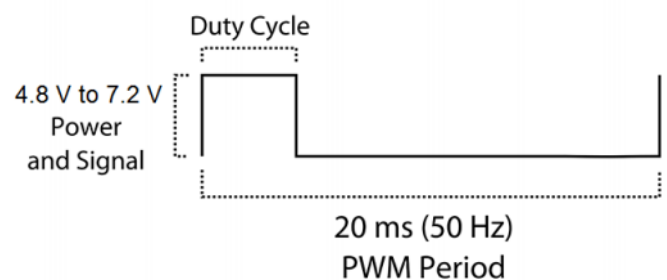
Αυτό το υψηλής ροπής MG996R ψηφιακός σέρβο κινητήρας διαθέτει μεταλλικά γρανάζια με αποτέλεσμα μια εξαιρετικά υψηλή ροπή στα 10kg μέσα σε ένα μικρό πακέτο. Το MG996R διαθέτει αναβαθμισμένη θωράκιση και ένα επανασχεδιασμένο PCB και IC σύστημα ελέγχου που κάνουν πολύ πιο ακριβή στις κινήσεις του. Οι οδοντωτοί τροχοί του κινητήρα έχουν επίσης αναβαθμιστεί για να βελτιώσουν το νεκρό εύρος ζώνης και το κεντράρισμα. Ο σέρβο μηχανισμός μπορεί να περιστραφεί περίπου 120 μοίρες (60 σε κάθε κατεύθυνση). Από το manual του βρίσκουμε το σχέδιο όπου έχει τις απαραίτητες διαστάσεις για την σχεδίαση του. Είναι αναγκαία η σχεδίαση του διότι μας βοηθά στην τελική συναρμολόγηση του βραχίονα.



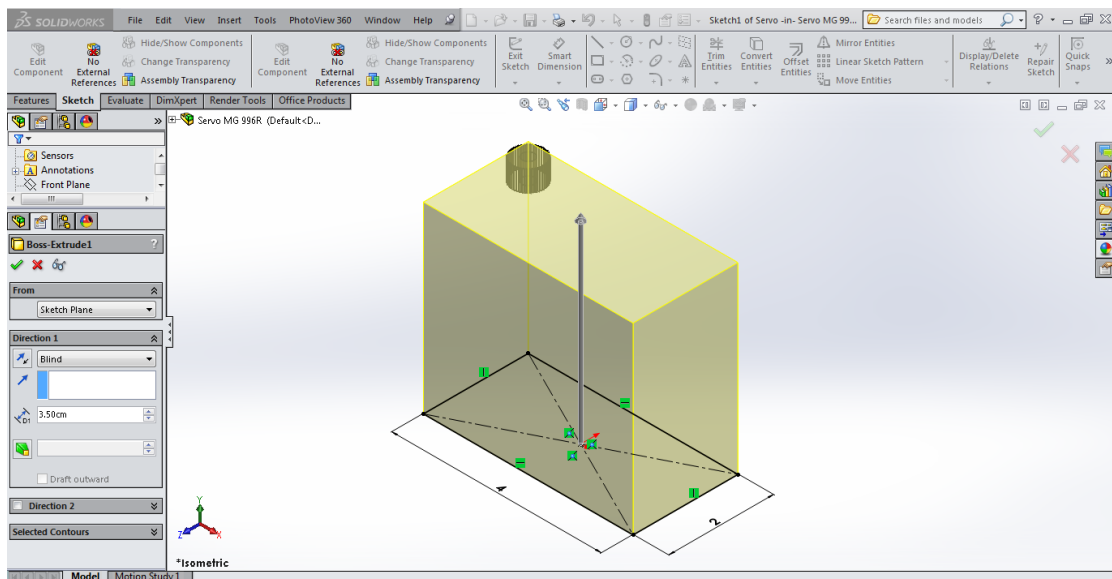
ΠΡΟΔΙΑΓΡΑΦΕΣ

- Βάρος: 55 g
- Διαστάσεις: 40.7 x 19.7 x 42.9 mm
- Ροπή σε στάση: 9.4 kgf·cm (4.8 V), 11 kgf·cm (6 V)
- Ταχύτητα λειτουργίας: 0.17 s/60° (4.8 V), 0.14 s/60° (6 V)
- Τάση λειτουργίας: 4.8 V a 7.2 V
- Ρεύμα λειτουργίας: 500 (mA) –900 (mA) (6V)
- Ρεύμα σε στάση: 2.5 A (6V)
- Νεκρό εύρος ζώνης: 5 μs
- Θερμοκρασία: 0°C – 55 °C

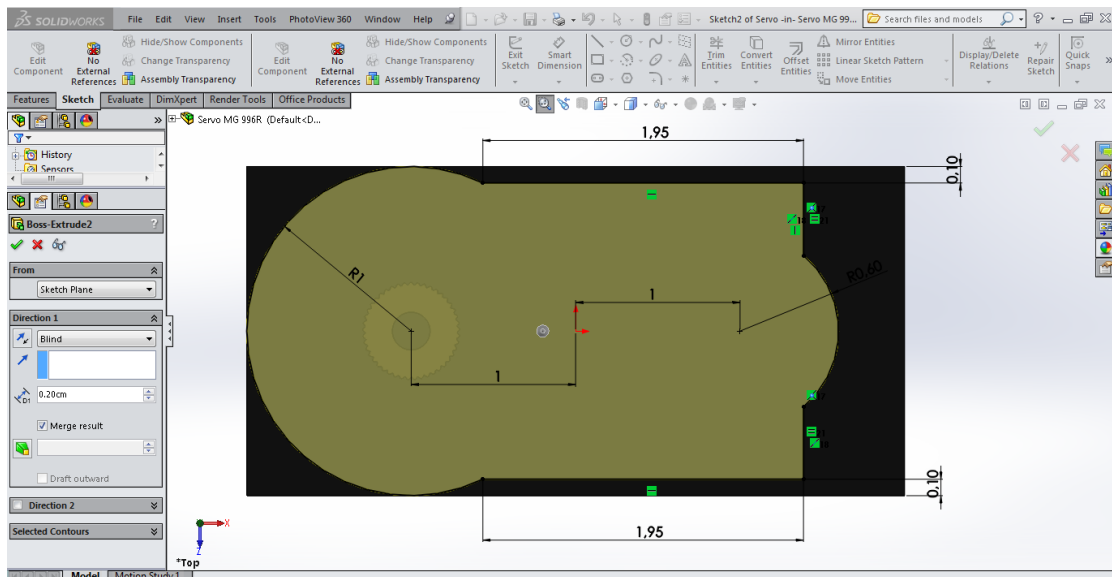
PWM=Orange (⏏)
Vcc = Red (+)
Ground=Brown (-)



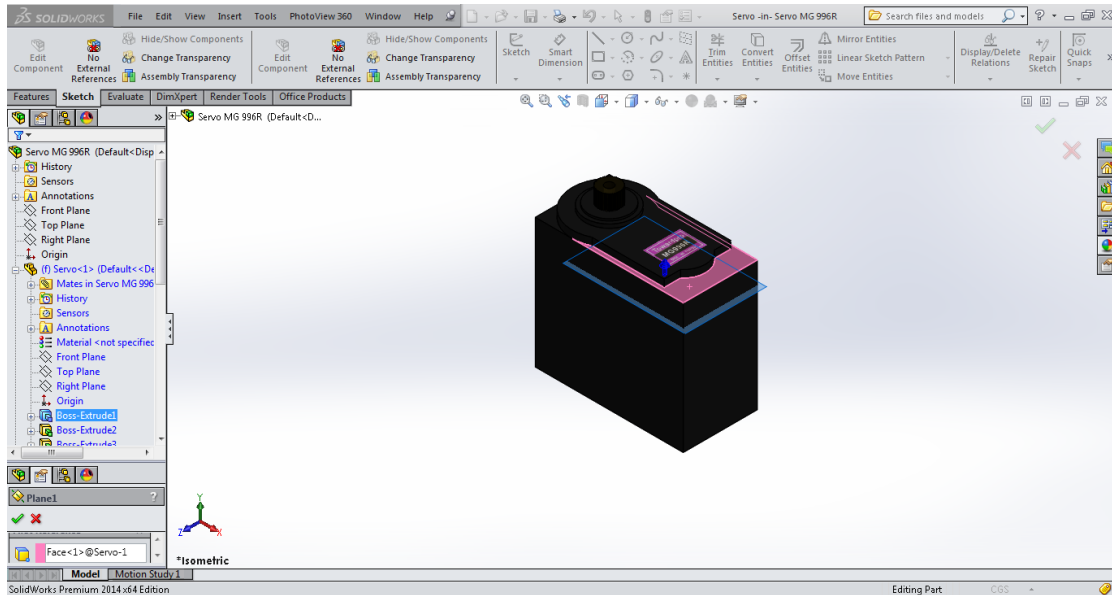
Σχεδιάζουμε πρώτα την βάση σε “top plane” και την κάνουμε “extrude” .



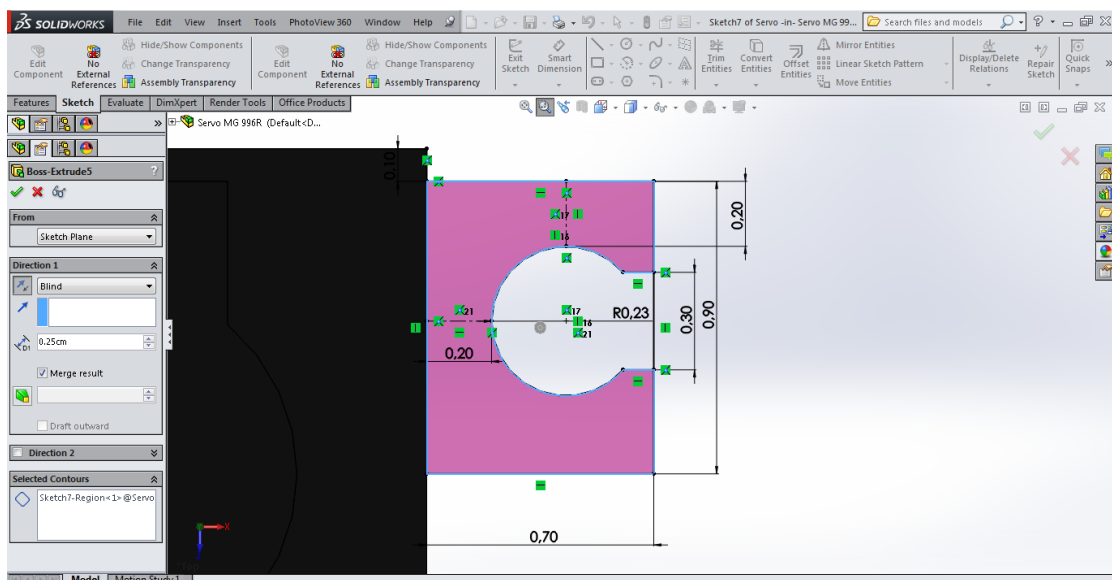
Μετά σχεδιάζουμε πάνω στην βάση το περίγραμμα στο οποίο θα τοποθετηθεί ο άξονας.



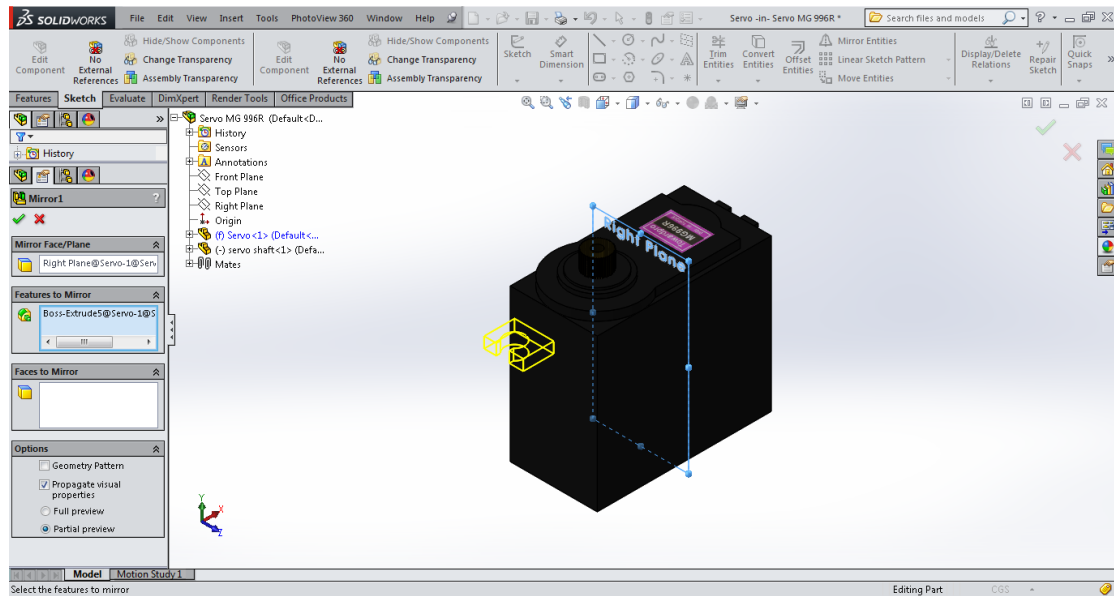
Μετατοπίζουμε το πλάνο μας από την κορυφή όπου σχεδιάσαμε προς τα κάτω κατά 10 cm.



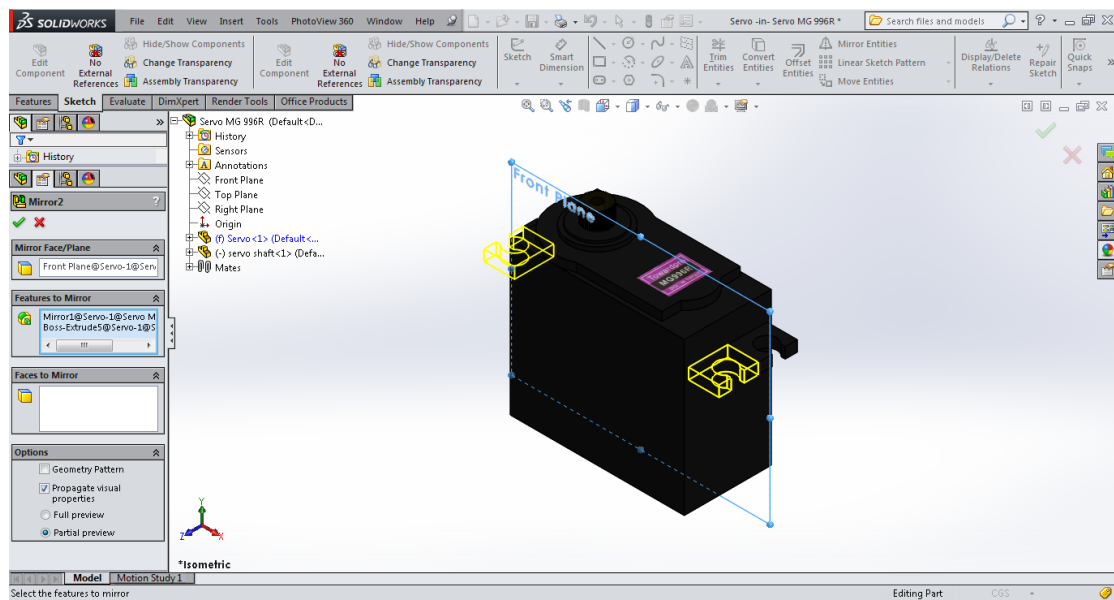
Έπειτα σχεδιάζουμε την πλαϊνή στήριξη στην οποία βιδώνουμε των κάθε σέρβο κινητήρα.



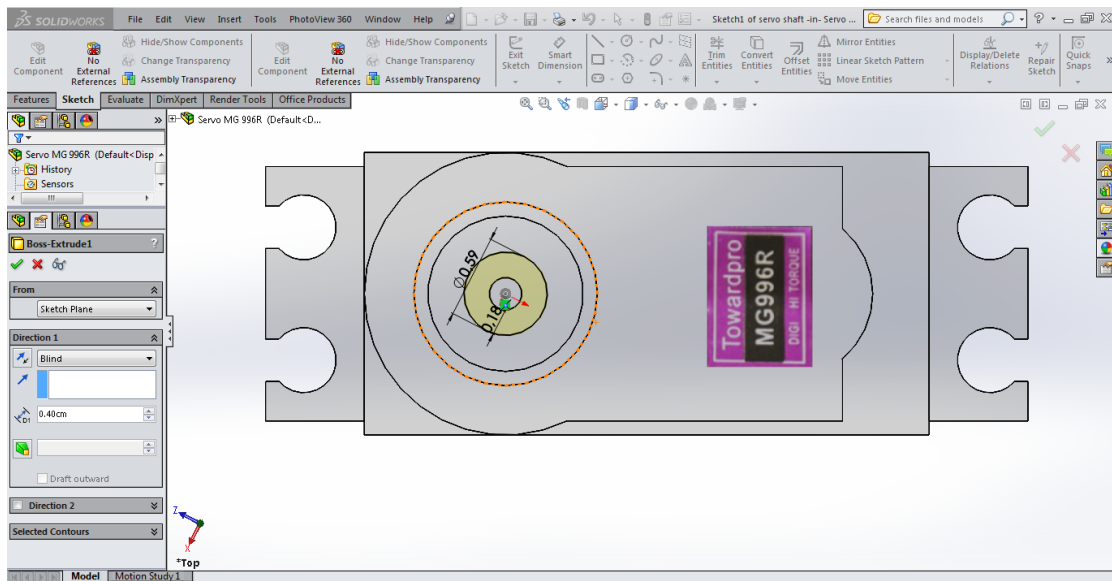
Αφού σχεδιάσαμε την πλαϊνή στήριξη διαλέγουμε “Right Plane” και την εντολή “Mirror” (καθρέφτης) δημιουργούμε αμέσως την στήριξη κα από την απέναντι πλευρά.



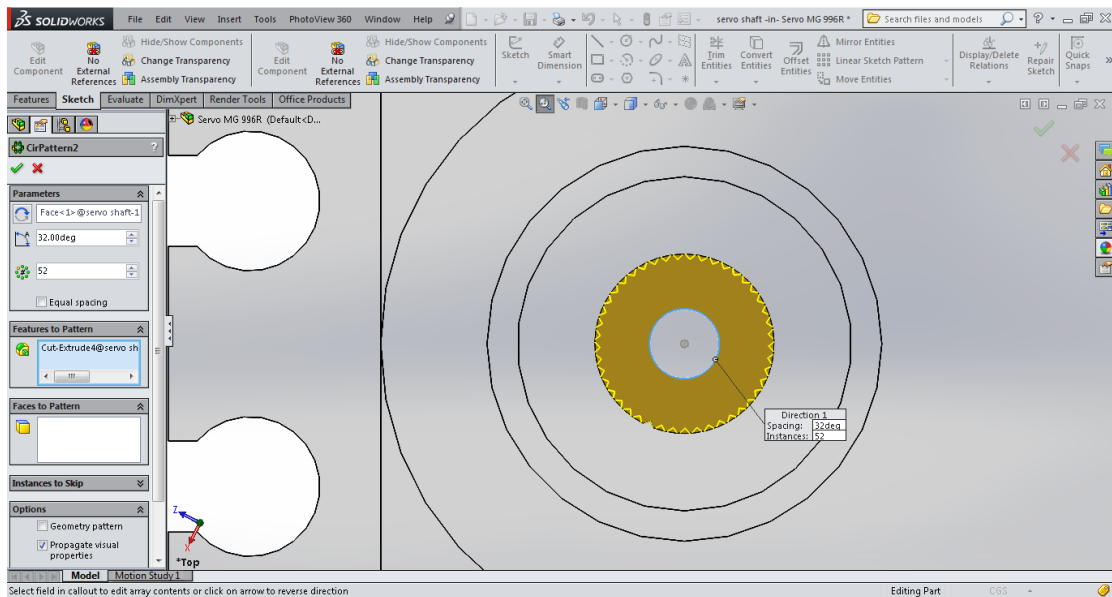
Αμέσως μετά διαλέγουμε “Front Plane” και αντιγράφουμε και μετατοπίζουμε με την εντολή “Mirror” την στήριξη.



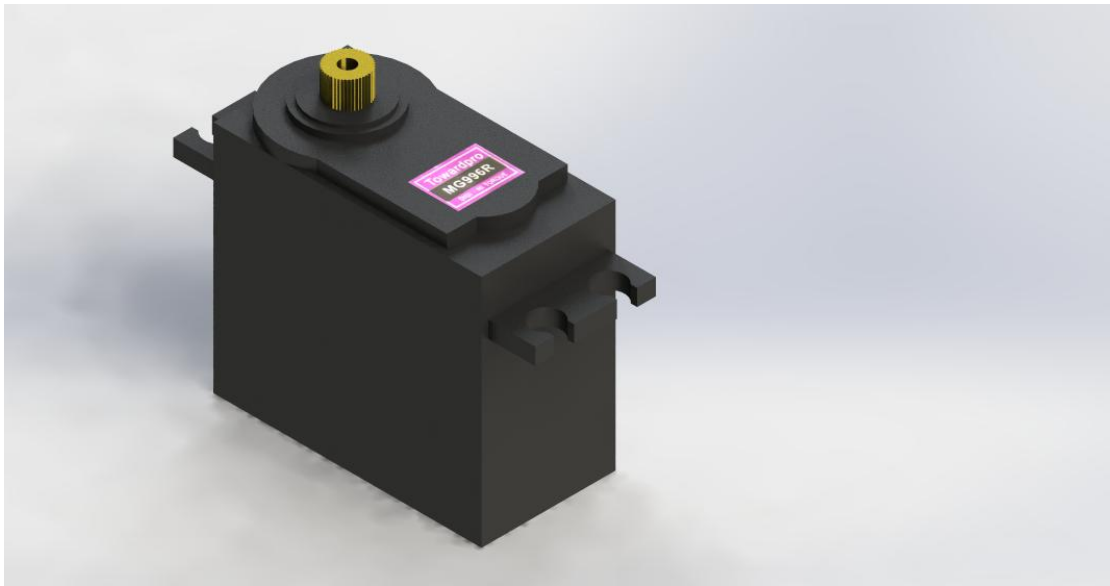
Τέλος σχεδιάζουμε τον άξονα του κινητήρα και εισάγουμε και το “decal” (αυτοκόλλητο) ΤΟΥ.



Με μία ακόμη εντολή την “Circular Pattern” (κυκλικό μοτίβο) δημιουργούμε την οδοντωτή επιφάνεια του άξονα.



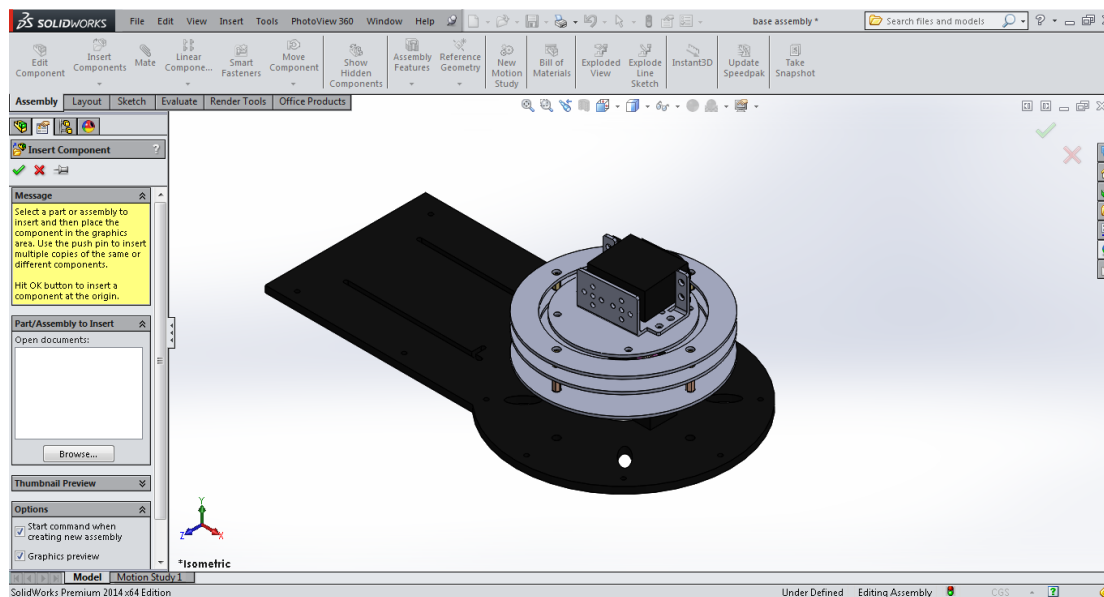
Ο σέρβο κινητήρας σε πραγματική και φωτορεαλιστική απεικόνιση.



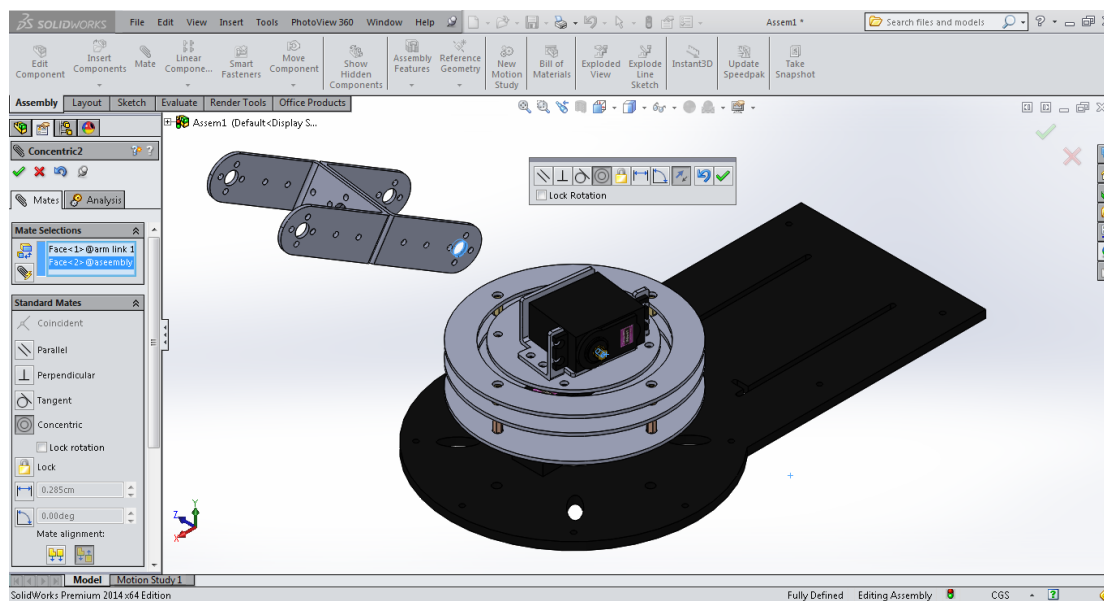
3.5 Συναρμολόγηση Ρομποτικού βραχίονα

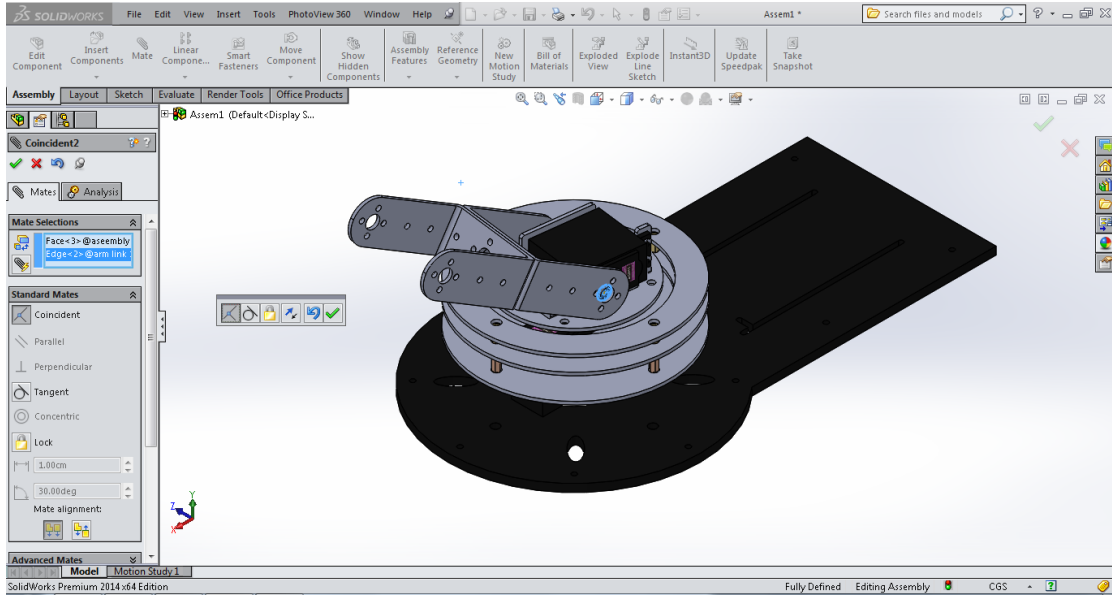
Στην συναρμολόγηση του ρομποτικού βραχίονα λόγω του όγκου των υλικών που σχεδιάστηκαν για να χρησιμοποιηθούν, δημιουργήσαμε “Subassemblies” και τα εισάγαμε σε ένα τελικό “assembly” έτσι ώστε να φτάσουμε την τελική μορφή του βραχίονα. Επίσης η συναρμολόγηση έγινε άπλα με δυο εντολές σε όλα τα εξαρτήματα (coincident-concentric mates).

BHMA 1°

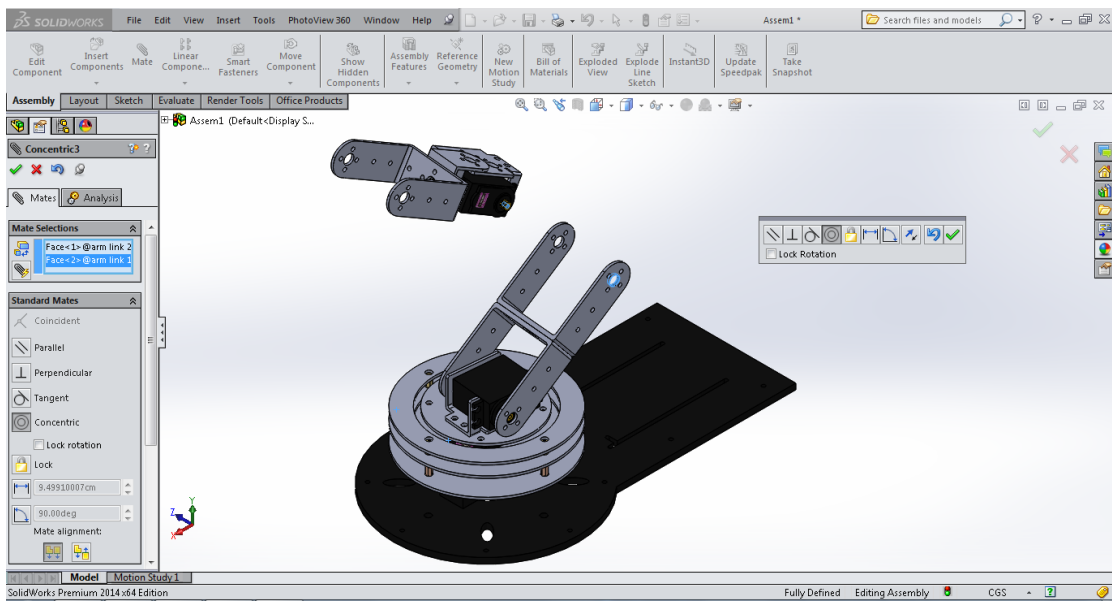


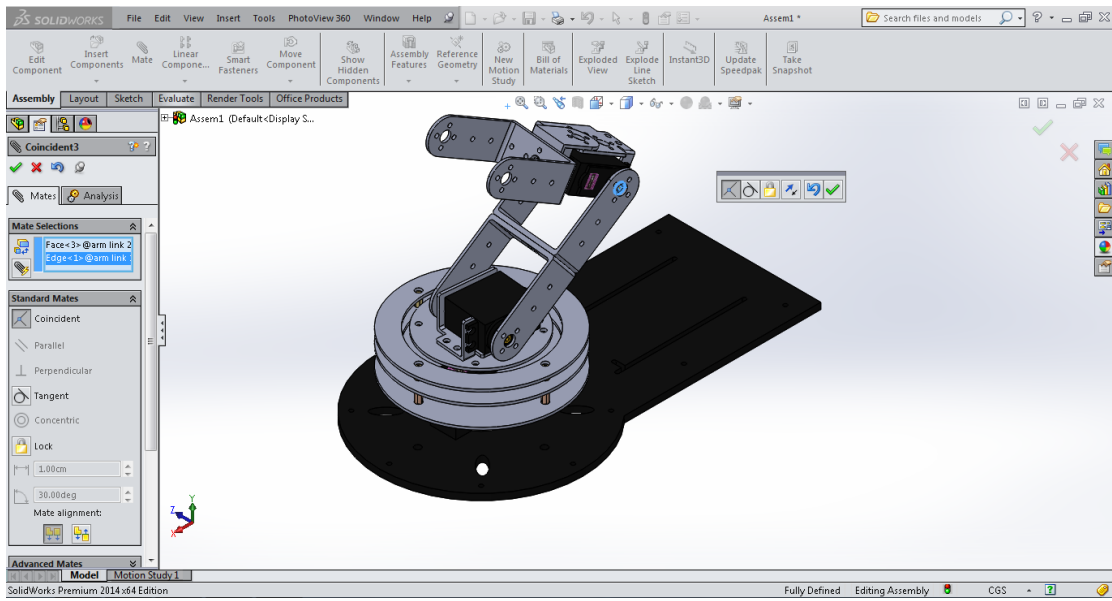
BHMA 2°



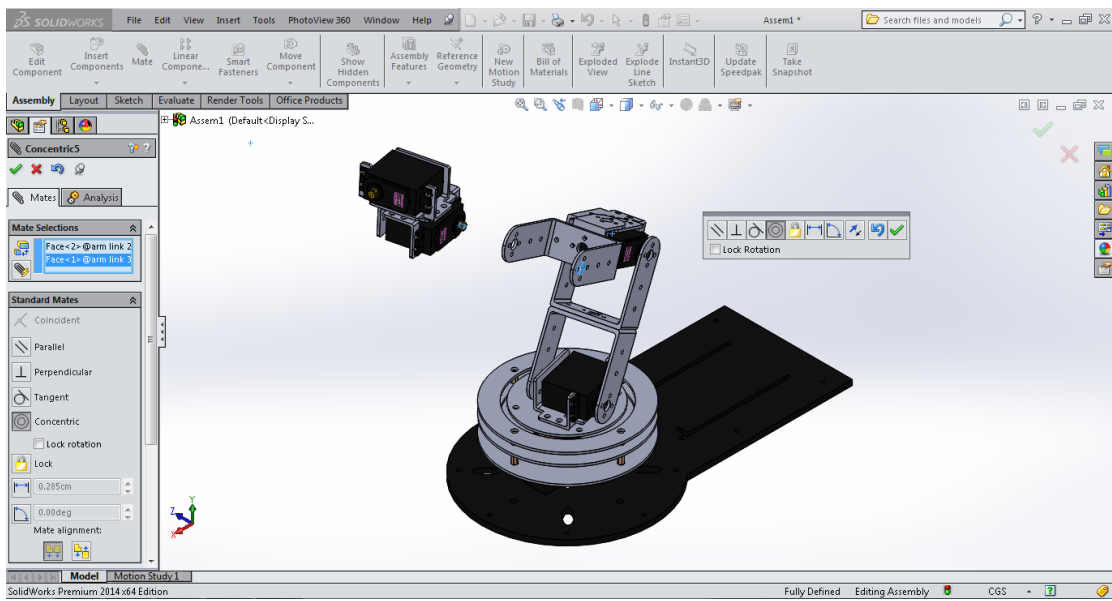


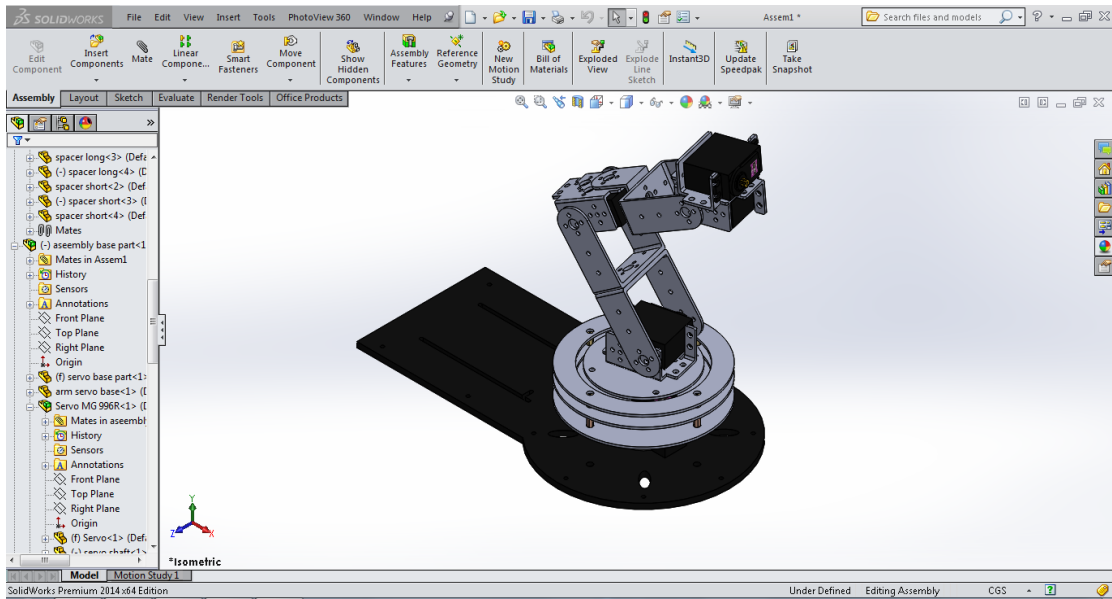
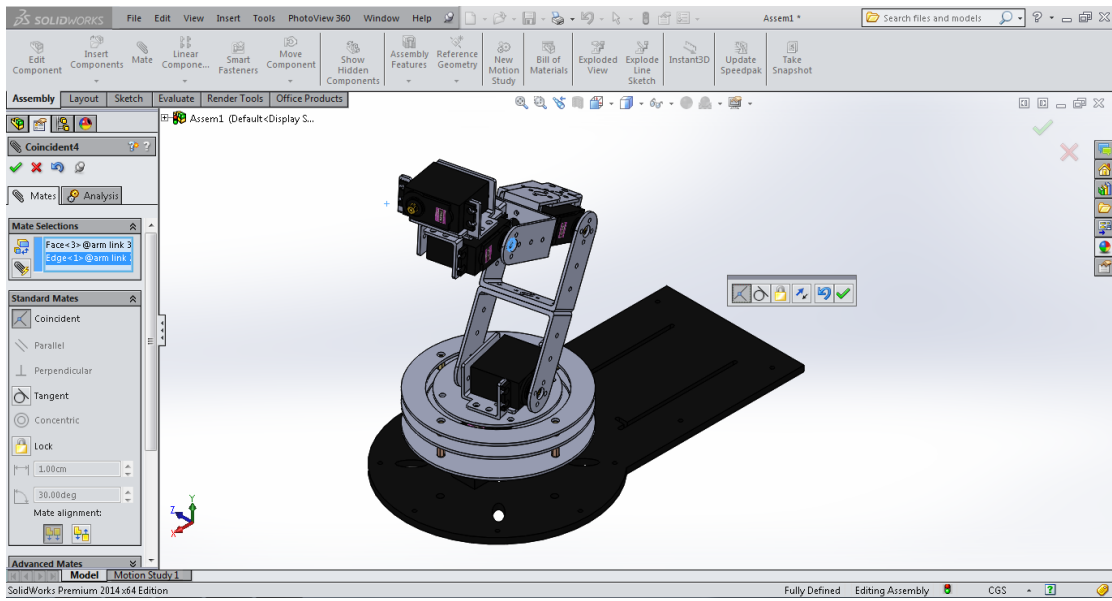
BHMA 3°



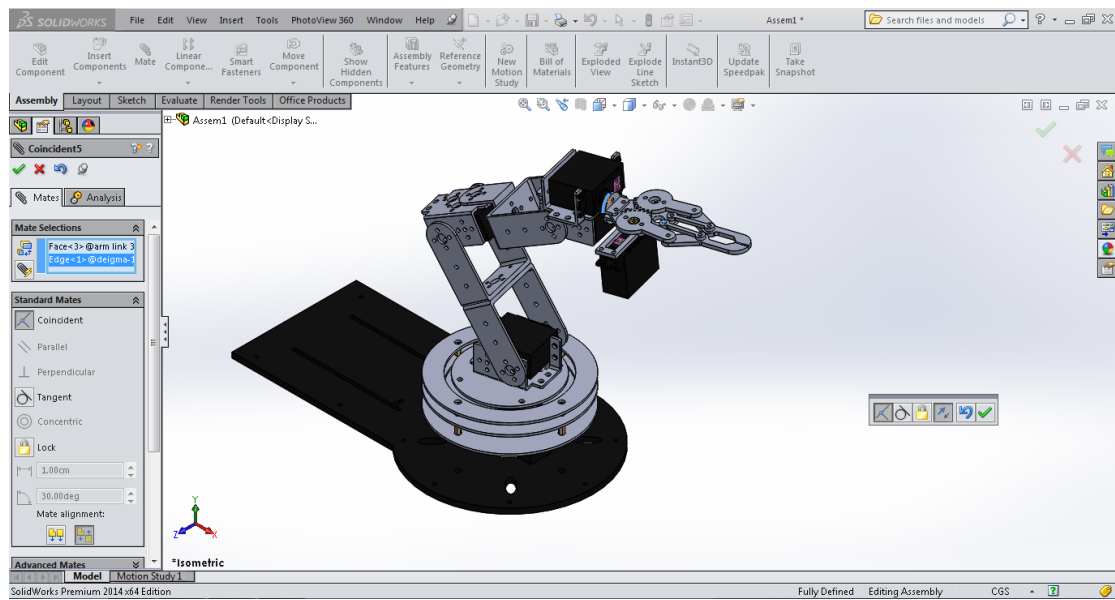
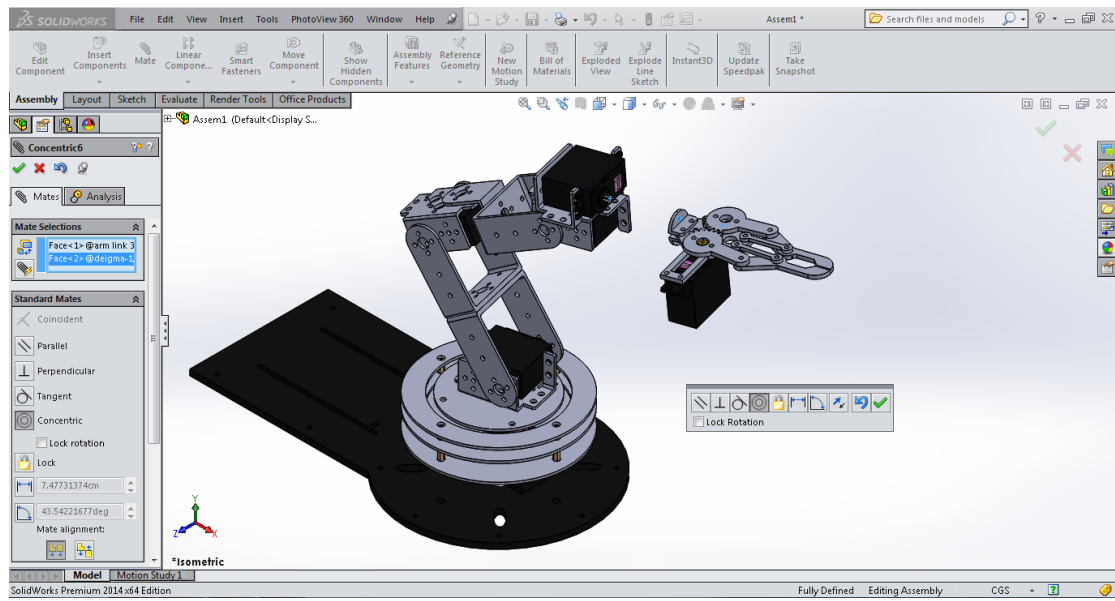


BHMA 4°



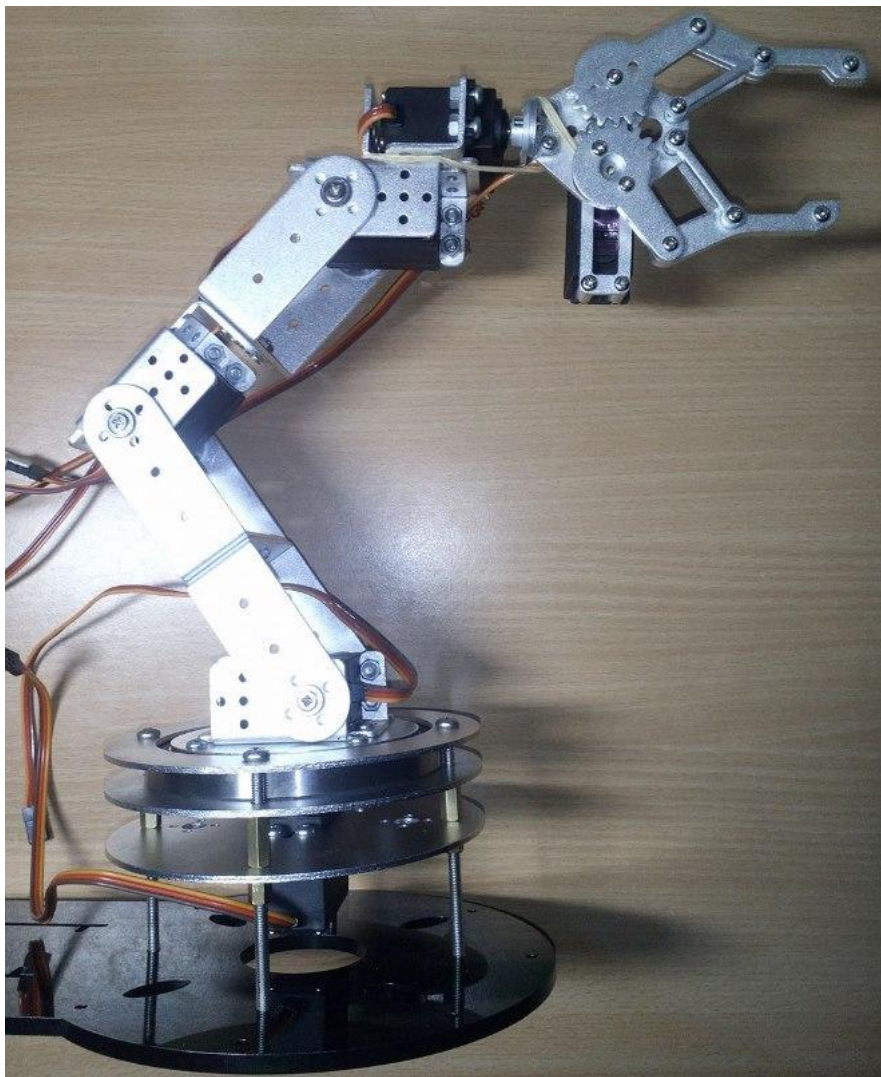


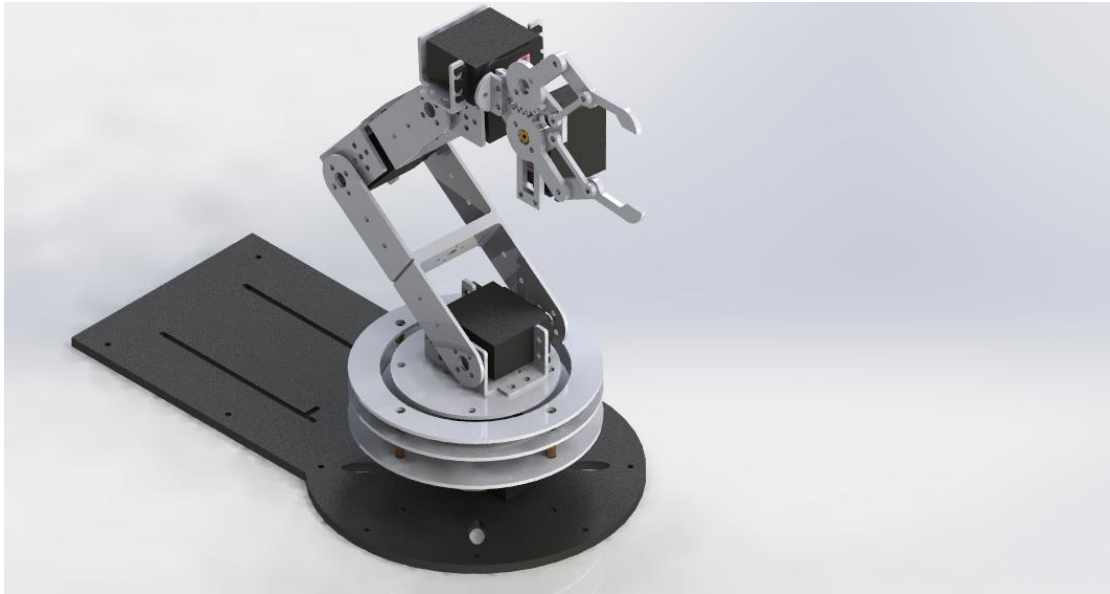
BHMA 5°





Ο ρομποτικός βραχίονα σε πραγματική και φωτορεαλιστική απεικόνιση.





4. Αυτοματοποίηση Διαδικασίας

4.1 Το πρόγραμμα LabVIEW

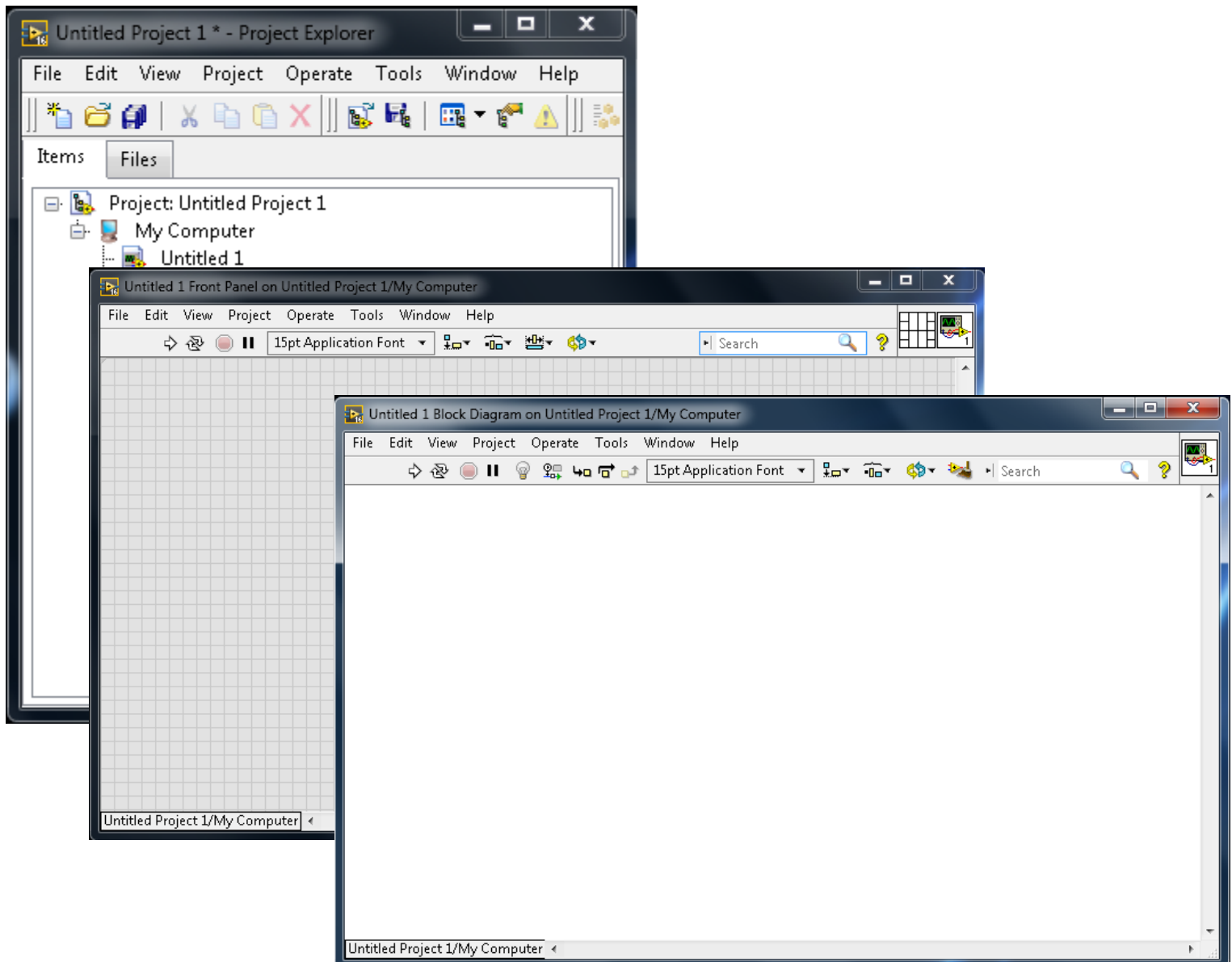
(LABoratory Virtual Instrument Engineering Workbench)

Για χρόνια το LabVIEW επέτρεψε μηχανικούς και τους επιστήμονες να αναπτύξουν πολύπλοκα αυτόνομα συστήματα. Στον πυρήνα του το LabVIEW χρησιμοποιείται ευρέως για την συνδεσιμότητα του με αισθητήρες και ενεργοποιητές και προσφέρει σήμερα πάνω από 8000 οδηγούς για την μέτρηση συσκευών, παρέχοντας ένα ενιαίο περιβάλλον που είναι ένα πλαίσιο με συνδυασμό γραφικών και κειμένου κώδικα. Το LabVIEW σας δίνει την ελευθερία να ενσωματώσουμε πολλαπλές προσεγγίσεις για τον προγραμματισμό, την ανάλυση και την ανάπτυξη αλγορίθμων. Επιπλέον, με τις νέες του βιβλιοθήκες μας δίνει μια εντελώς νέα σουίτα για την ρομποτική ειδικά, παρέχοντας όλα τα απαραίτητα εργαλεία για την ανάπτυξη της .

Το LabVIEW (**L**aboratory **V**irtual **I**nstrument **E**ngineering **W**orkbench) είναι μια πλατφόρμα του συστήματος σχεδιασμού και περιβάλλον ανάπτυξης για μια οπτική γλώσσα προγραμματισμού από την National Instruments. Η γραφική γλώσσα ονομάζεται "G" .Δεν πρέπει να συγχέεται με G-code. Αρχικά κυκλοφόρησε για το Apple Macintosh, το 1986, χρησιμοποιείται συνήθως για την απόκτηση δεδομένων, τον έλεγχο οργάνων και βιομηχανικού αυτοματισμού σε μια ποικιλία λειτουργικών συστημάτων , συμπεριλαμβανομένων των Microsoft Windows, διάφορες εκδόσεις του Unix, Linux και MacOS. Η τελευταία έκδοση του LabVIEW είναι 2016,την οποία και χρησιμοποιήσαμε.

Τα Ρομπότ σημαίνουν πολλά πράγματα για πολλούς ανθρώπους, έτσι και η National Instruments προσφέρει ένα πακέτο με παραγωγικά εργαλεία σχεδιασμού για τα πάντα, από το σχεδιασμό αυτόνομων οχημάτων μέχρι και τη διδασκαλία των αρχών σχεδιασμού της ρομποτικής. Η γραφική γλώσσα προγραμματισμού αλλά και τα πρόσθετα "modules" που παρέχει καθιστά εύκολο να προγραμματίσουμε σύνθετες εφαρμογές ρομποτικής, παρέχοντας υψηλό επίπεδο επικοινωνίας με τους αισθητήρες, την αποφυγή εμποδίων, το σχεδιασμού διαδρομής, την κινηματική, το σύστημα διεύθυνσης, και πολλά άλλα.

Η γλώσσα προγραμματισμού που χρησιμοποιείται στο LabVIEW, που ονομάζεται G, είναι μια "dataflow" γλώσσα προγραμματισμού. Η εκτέλεση καθορίζεται από την δομή ενός γραφικού διαγράμματος μπλοκ (source-code) επί του οποίου ο προγραμματιστής συνδέει διαφορετικής λειτουργίας-κόμβους. Ενώνοντας τα μπλοκ με καλώδια διαδίδουμε τις μεταβλητές και κάθε κόμβος μπορεί να εκτελεστεί το συντομότερο καταστούν διαθέσιμα όλα τα δεδομένα εισόδου. Δεδομένου ότι αυτό θα μπορούσε να είναι και η περίπτωση για πολλαπλούς κόμβους ταυτόχρονα, μπορεί να εκτελεστεί εγγενώς παράλληλα. Ένα άλλο σημαντικό πλεονέκτημα του, είναι ότι επιτρέπει την ανάπτυξη ενός πολύ φιλικού προς τον χρήστη προγράμματος διεπαφής (User Interface) χωρίς να απαιτείται ιδιαίτερη ενασχόληση με το σχεδιασμό των γραφικών που θα απαιτηθούν. Στο LabVIEW το User Interface κατασκευάζεται με τη χρήση ενός συνόλου εργαλείων και αντικειμένων (objects). Αυτό είναι γνωστό ως **Front Panel**. Στη συνέχεια, προστίθεται ο κώδικας με τη χρήση γραφικών αναπαραστάσεων των συναρτήσεων ώστε να ελεγχθούν τα αντικείμενα του Front Panel. Το **Block Diagram** περιέχει αυτόν τον κώδικα. Γενικά, το Block Diagram θα μπορούσε να θεωρηθεί ως ένα είδος διαγράμματος ροής.



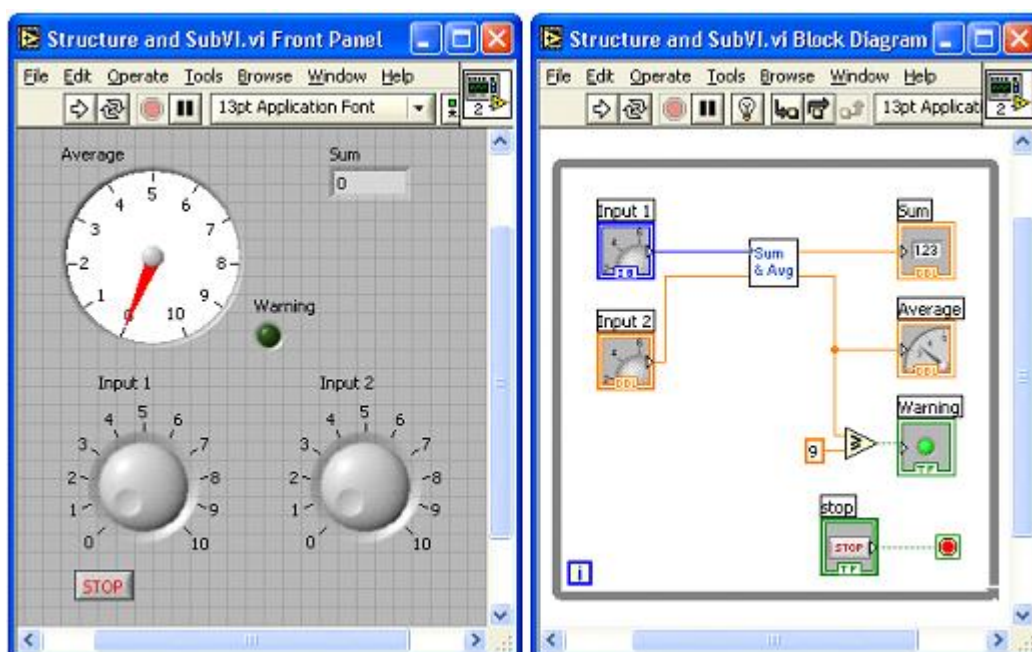
4.2 Δομή του περιβάλλοντος προγραμματισμού

Το LabVIEW ενσωματώνει τη δημιουργία διεπαφών χρήστη στον κύκλο ανάπτυξης του. Τα προγράμματα και οι υπορουτίνες στο LabVIEW ονομάζονται εικονικά όργανα VI's (Virtual Instruments). Κάθε VI έχει τρία συστατικά: ένα μπλοκ διάγραμμα (Block Diagram), ένα εμπρόσθιο πάνελ(Front Panel) και ένα πάνελ συνδετήρα (Project Explorer).

1)Ο **Project Explorer** χρησιμοποιείται για να αντιπροσωπεύσει τα VI's από άλλα καλούμενα VI's και γενικά είναι το παράθυρο στο οποίο στηρίζονται όλα.

2)Το **Front Panel** το οποίο αποτελεί και το μέσο επικοινωνίας με το χρήστη (user interface) είναι χτισμένο χρησιμοποιώντας ρυθμιστές(controls) και δείκτες(indicators). Οι ρυθμιστές είναι **είσοδοι** επιτρέπουν σε ένα χρήστη να παράσχει πληροφορίες για την VI. Οι δείκτες είναι **έξοδοι**: δείχνουν, ή εμφανίζουν, τα αποτελέσματα με βάση τις εισόδους που δόθηκαν στο VI.

3)Το **Block Diagram ή Back Panel**, το οποίο είναι ένα μπλοκ διάγραμμα, περιέχει το γραφικό πηγαίο κώδικα που καθορίζει τον τρόπο με τον οποίο λειτουργεί το VI. Όλα τα αντικείμενα που τοποθετούνται στο Front Panel εμφανίζονται στο Back Panel ως τερματικά. Η πίσω πλευρά περιέχει επίσης τις δομές και τις λειτουργίες οι οποίες εκτελούν εργασίες πάνω στους ρυθμιστές και προσφέροντας στοιχεία στους δείκτες.



Συλλογικά οι ρυθμιστές, οι δείκτες, οι δομές και οι λειτουργίες αναφέρονται ως κόμβοι. Οι κόμβοι συνδέονται μεταξύ τους με τη χρήση καλωδίων. Έτσι, ένα εικονικό όργανο μπορεί να λειτουργήσει είτε ως πρόγραμμα, με το Front Panel είτε να λειτουργεί ως διεπαφή χρήστη. Αυτό συνεπάγεται κάθε VI μπορεί να ελεγχθεί εύκολα προτού ενσωματωθεί ως υπορουτίνα σε ένα μεγαλύτερο πρόγραμμα.

Το περιβάλλον προγραμματισμού LabVIEW, απλά με drag & drop το καθιστά απλό για να δημιουργήσουμε πολλές εφαρμογές. Τα πιο προηγμένα συστήματα ανάπτυξης LabVIEW προσφέρουν τη δυνατότητα για την κατασκευή (stand-alone) εφαρμογές. Επιπλέον, είναι δυνατή η δημιουργία κατανεμημένων εφαρμογών, οι οποίες επικοινωνούν με ένα μοντέλο (client-server), και έτσι είναι ευκολότερο να εφαρμοστεί, λόγω του εγγενώς παράλληλη φύση του γραφικού κώδικα "G".

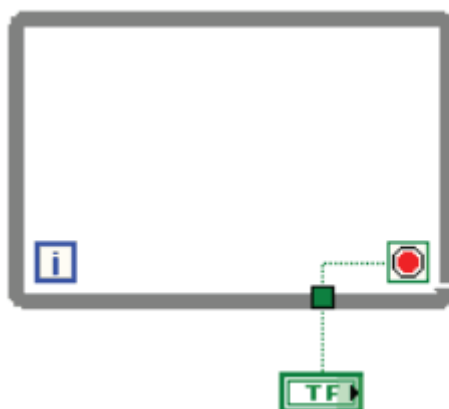
4.2.1 Τα κυριότερα στοιχεία του LabVIEW

While Loop

Το while loop είναι μια δομή που επαναλαμβάνει ένα μέρος του κώδικα του προγράμματος μέχρι να εκπληρωθεί μια συνθήκη. Στο LabVIEW ένα while loop παριστάνεται με ένα ορθογώνιο σχήμα του οποίου τις διαστάσεις μπορεί να αλλάξει ο χρήστης. Εντός του loop υπάρχει το iteration terminal που περιέχει ανά πάσα στιγμή τον αριθμό των επαναλήψεων του βρόγχου.

Πρέπει να ληφθεί υπ' όψιν ότι επειδή η αρχική του τιμή είναι πάντα μηδενική αν ο βρόγχος εκτελεστεί μια φορά το iteration terminal περιέχει την τιμή 0. Εντός του βρόγχου επίσης υπάρχει και το conditional terminal το οποίο λειτουργεί σαν είσοδος και αποτελεί ουσιαστικά τη συνθήκη ελέγχου του βρόγχου. Πρόκειται για μια Boolean μεταβλητή εισόδου η οποία όταν πάρει την τιμή FALSE τερματίζεται η λειτουργία του while loop.

Επιλέγεται από το functions palette < structures < while loop.

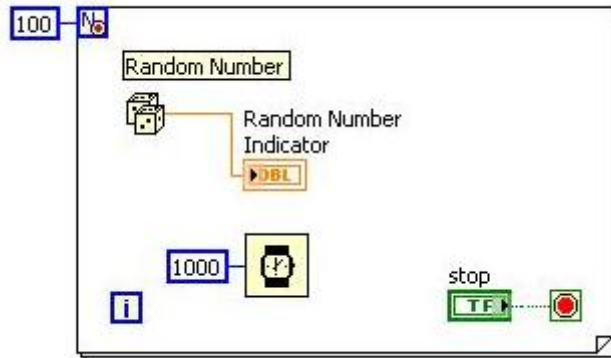


For Loop

Το for loop είναι μια δομή η οποία εκτελεί ένα μέρος του προγράμματος για έναν καθορισμένο αριθμό επαναλήψεων. Όπως το while loop, παριστάνεται με ένα ορθογώνιο σχήμα του οποίου οι διαστάσεις μπορούν να αλλαχθούν.

Περιλαμβάνει δύο terminals, το count terminal το οποίο λειτουργεί σαν είσοδος και καθορίζει τον αριθμό των επαναλήψεων και το iteration terminal που περιέχει τον αριθμό των επαναλήψεων που έχουν εκτελεστεί.

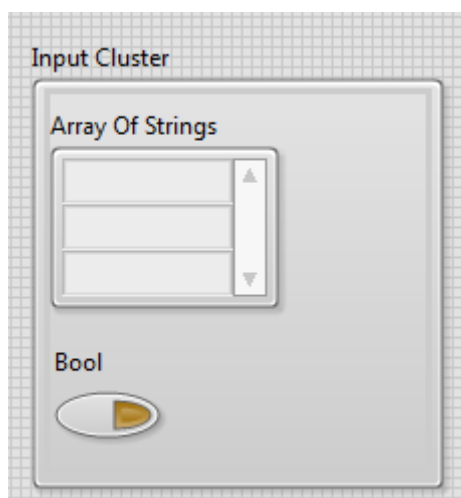
Επιλέγεται από το functions palette < structures < for loop



Arrays & Clusters

Οι πίνακες (arrays) είναι ένα σύνολο στοιχείων του ίδιου τύπου. Μπορεί να έχουν μια ή περισσότερες διαστάσεις. Τα στοιχεία του πίνακα μπορούν να είναι οποιουδήποτε τύπου. Το LabVIEW περιλαμβάνει πολλές λειτουργίες που σχετίζονται με τους πίνακες και ονομάζονται array functions. Οι λειτουργίες αυτές διευκολύνουν την και αξιοποιούν τη χρήση των πινάκων και βρίσκονται στη Functions Palette και συγκεκριμένα στο μενού Array.

Το cluster είναι ένας τύπος δεδομένου ο οποίος περιέχει πολλαπλά στοιχεία δεδομένων τα οποία μπορούν να είναι διαφορετικού τύπου. Αυτό είναι και το στοιχείο που διαφοροποιεί ένα cluster από ένα array μιας και το δεύτερο ενώ περιλαμβάνει πολλαπλά στοιχεία αυτά θα πρέπει να είναι του ίδιου τύπου. Ο μόνος περιορισμός που υπάρχει ως προς τα στοιχεία του cluster είναι ότι θα πρέπει να είναι όλα ή controls ή indicators.

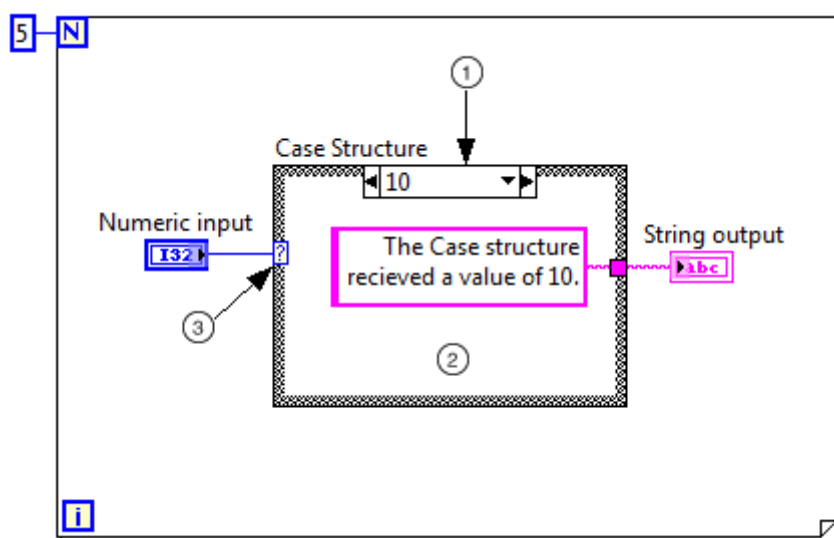


Shift Registers

Οι shift registers χρησιμοποιούνται στο while και στο for loop και μεταφέρουν τιμές από μία επανάληψη του βρόγχου στην επόμενη. Για να δημιουργηθεί ένα νέο shift register γίνεται left click στην δεξιά ή αριστερή πλευρά των loops και από το μενού που εμφανίζεται επιλέγεται **Add Shift Register**. Ένας shift register αποτελείται από δύο terminals τα οποία είναι τοποθετημένα το ένα στη δεξιά και το άλλο στην αριστερή πλευρά του βρόγχου. Το δεξί terminal αποθηκεύει την τιμή μιας μεταβλητής κατά την ολοκλήρωση μιας επανάληψης του βρόγχου. Η τιμή αυτή εμφανίζεται στη συνέχεια στο αριστερό terminal κατά την έναρξη της επόμενης επανάληψης του βρόγχου. Μπορεί να αποθηκεύσει όλων των ειδών τα δεδομένα –αριθμητικά, Boolean, string, array κ.α.

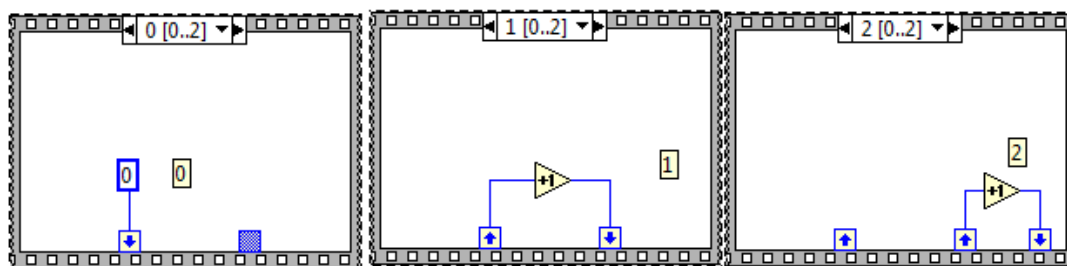
3.3.1.4 Case Structure

Η δομή Case εμπεριέχει δύο ή περισσότερα subdiagrams ή αλλιώς cases από τα οποία εκτελείται μόνο το ένα κάθε φορά που εκτελείται η δομή. Στην αριστερή πλευρά του συμβόλου της δομής case υπάρχει ένας επιλογέας που ανάλογα με την τιμή που παίρνει σαν είσοδο διαλέγει και εκτελεί το ανάλογο subdiagram. Η τιμή της εισόδου μπορεί να είναι ακέραια, Boolean ή string. Για να προστεθούν cases γίνεται δεξί click στις πλευρές της case structure και επιλέγεται **Add Case**. Επιλέγεται από το functions palette < structures < case



Sequence Structure

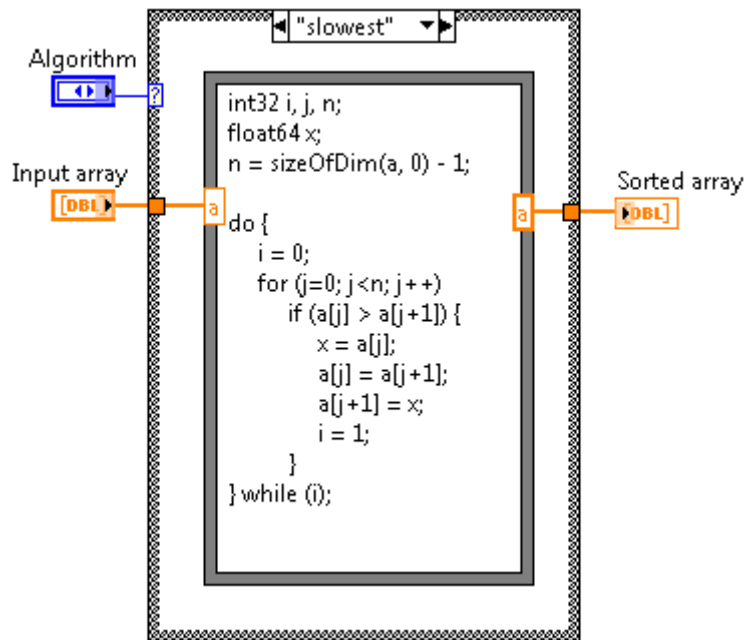
Η sequence structure εκτελεί ακολουθιακά έναν αριθμό block diagrams. Μοιάζει με τη δομή case διότι περιέχει πολλά στιγμιότυπα. Σε αυτή την περίπτωση όμως δεν εκτελείται μόνο το diagram που περιέχεται στο στιγμιότυπο που επιλέχθηκε αλλά όλα τα diagrams με την σειρά που έχουν τοποθετηθεί. Έτσι πρώτα θα εκτελεστεί το diagram που περιέχεται στο στιγμιότυπο 0, ακολούθως αυτό που περιέχεται στο στιγμιότυπο 1 κ.ο.κ. Η δομή αυτή είναι χρήσιμη στον έλεγχο της ροής των δεδομένων αποφεύγοντας έτσι πιθανά λάθη από την εκτέλεση κάποιου κόμβου πριν την επιθυμητή σειρά εκτέλεσης του. Για να προστεθεί ένα frame γίνεται δεξί click στις πλευρές του sequence structure και στη συνέχεια επιλέγεται **Add Frame Before** ή **Add Frame After** ανάλογα με την επιθυμητή θέση τοποθέτησης. Επιλέγεται από το functions palette < structures < sequence



Formula Node

Πρόκειται για μια δομή μέσα στην οποία μπορούν να εισαχθούν απ' ευθείας αλγεβρικές φόρμουλες. Αυτό είναι πολύ χρήσιμο γιατί όταν απαιτείται μια πολύπλοκη έκφραση, η υλοποίηση της με τα αριθμητικά functions που περιέχει το LabVIEW καταλαμβάνει μεγάλη έκταση στο χώρο και γενικά κάνει το block diagram περίπλοκο. Επίσης υποστηρίζει και εκφράσεις παρόμοιες με αυτές της C. Οι built-in συναρτήσεις που υποστηρίζει είναι οι παρακάτω: abs, acos, acosh, asin, asinh, atan, atanh, ceil, cos, cosh, cot, csc, exp, expm1, floor, getexp, getman, int, intrz, ln, ln1, log, log2, max, min, mod, rand, rem, sec, sign, sin, sinc, sinh, sqrt, tan, tanh. Για να δημιουργηθούν τα input και output terminals αρχικά γίνεται δεξί click στις πλευρές του Formula Node και στη συνέχεια επιλέγεται **Add Input** ή **Add Output**. Κατόπιν εισάγονται τα ονόματα των μεταβλητών στα input και output boxes. Ακολούθως συνδέεται το επιθυμητό control στο input box καθώς και ένα indicator στο output box για να λαμβάνονται τα αποτελέσματα.

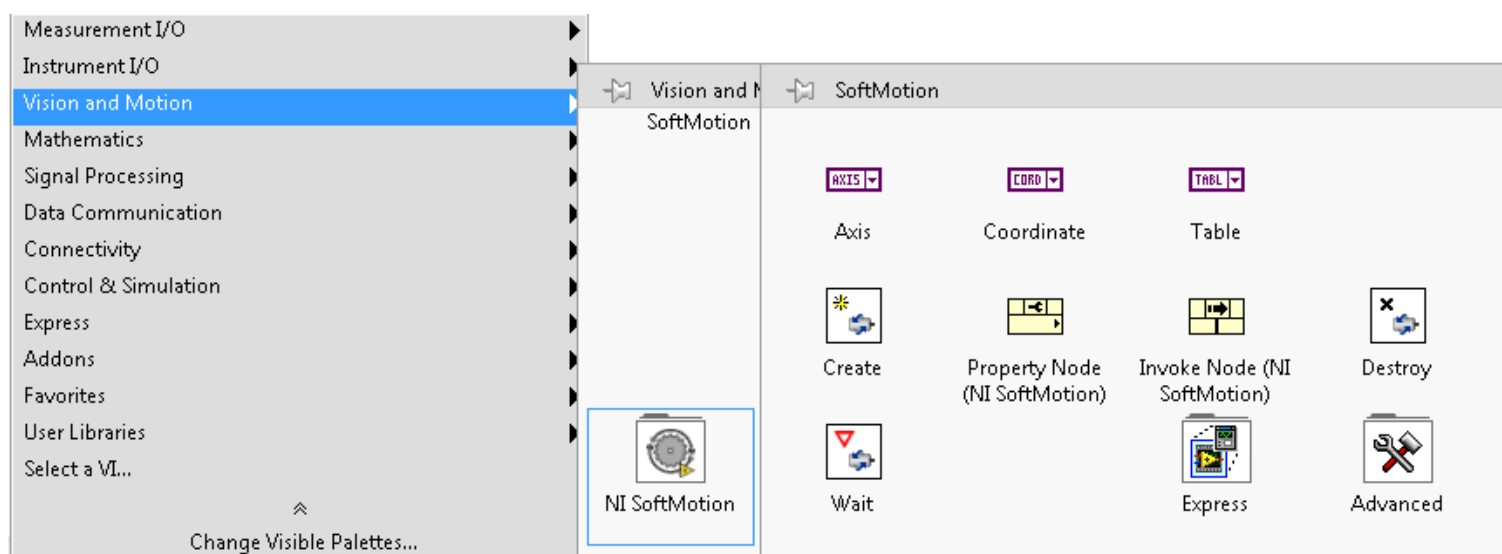
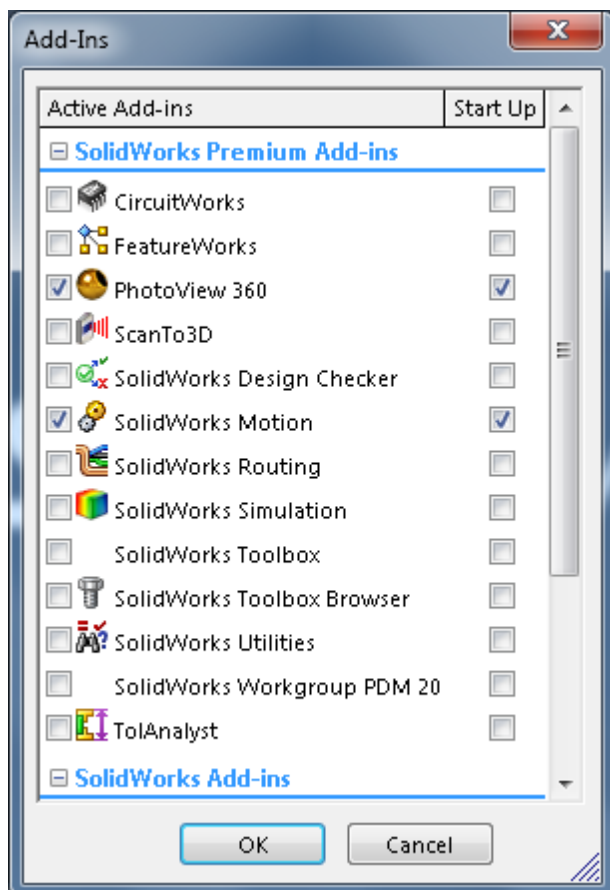
Επιλέγεται από το functions palette < structures < formula node



Συνοψίζοντας λοιπόν, στην παλέτα συναρτήσεων υπάρχουν έτοιμες κατασκευασμένες λειτουργίες όπως συνθήκες (while και if), βρόχοι επανάληψης, άθροιση αριθμών οποιουδήποτε είδους, λογικές μεταβλητές και πράξεις, πίνακες, μεταβλητές string, συγκρίσεις, έτοιμα προγράμματα 67 για άνοιγμα και γράψιμο αρχείων, πρότυπες κυματομορφές και προγράμματα δημιουργίας και τέλεσης πράξεων μεταξύ κυματοφορμών, στοιχεία ανώτερων μαθηματικών όπως ολοκληρώματα και αθροίσματα σειρών, ειδικός κατάλογος για την επικοινωνία με όργανα και άλλος για τη διασύνδεση οργάνων καθώς και μία σειρά εργαλεία για τον έλεγχο της εφαρμογής. Τέλος, δίνεται η δυνατότητα στο χρήστη να δημιουργήσει κάποιες δικές του βιβλιοθήκες και να επιλέξει κάποιες δικές του συναρτήσεις. Στην παλέτα ελεγκτών (Controls Palette) υπάρχει μία σειρά από controls για τον έλεγχο λογικών και αριθμητικών μεταβλητών, πινάκων, γραφικών παραστάσεων, strings, διευθύνσεων οργάνων, είσοδο αρχείων, δημιουργία ActiveX αντικειμένων καθώς και οριζμένων από το χρήστη αρχείων. Εκτός από τα Controls για όλα τα παραπάνω υπάρχουν μαζί και οι αντίστοιχοι Indicators. Τέλος στην παλέτα εργαλείων υπάρχουν εργαλεία για χρήση του κέρσορα στην εισαγωγή κειμένου, στην επιλογή και αλλαγή των τιμών των αντικειμένων του Front Panel, για καλωδίωση των αντικειμένων του Block Diagram, για επιλογή και εισαγωγή χρωμάτων καθώς και για ειδικές λειτουργίες όπως η εισαγωγή breakpoint.

4.3 Επικοινωνία μεταξύ Λογισμικών SolidWorks LabVIEW & NI Soft Motion

Το SoftMotion είναι ένα πρόσθετο λογισμικό που επιτρέπει στο SolidWorks και το LabVIEW να συνεργαστούν και έρχεται σαν ένα μικρό module για το LabVIEW και σαν (add-in) για να SolidWorks.

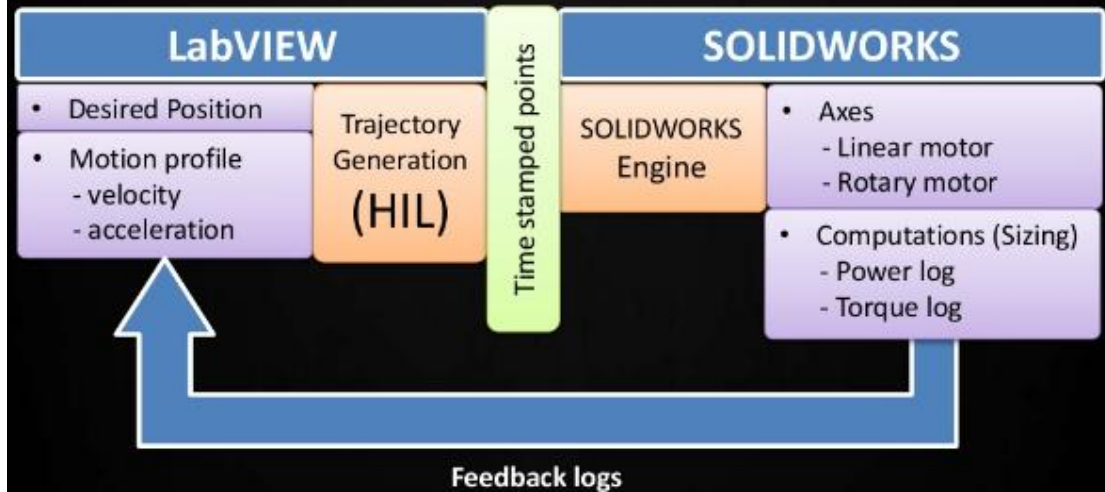


Με τη χρήση του SoftMotion με το SolidWorks μπορούμε να προσομοιώσουμε το σύστημα μας με το πραγματικό προφίλ κίνησης, μπορούμε εξίσου να προσομοιώσουμε τη μηχανική δυναμική, συμπεριλαμβανομένης της μάζας, τριβής και βαρύτητας, τους χρόνους κύκλου, τις ατομικές επιδόσεις ενός συστατικού και τη σύνδεση με ένα πραγματικό αλγόριθμο ελέγχου. Με την εικονική προτυποποίηση επίσης γνωστή και ως ψηφιακή προτυποποίηση μας προσφέρει τη δυνατότητα να απεικονίσουμε και να βελτιστοποιήσουμε το σχεδιασμό. Ενσωματώνοντας την προσομοίωση της κίνησης με το CAD μοντέλο απλοποιεί το σχεδιασμό, επειδή η προσομοίωση χρησιμοποιεί τις πληροφορίες που ήδη υπάρχουν στο μοντέλο CAD, όπως τα (assembly mates), συνδέσμους, και τις ιδιότητες των υλικών.

Τυπικές εφαρμογές για την LabVIEW SoftMotion με SolidWorks περιλαμβάνουν τα ακόλουθα:

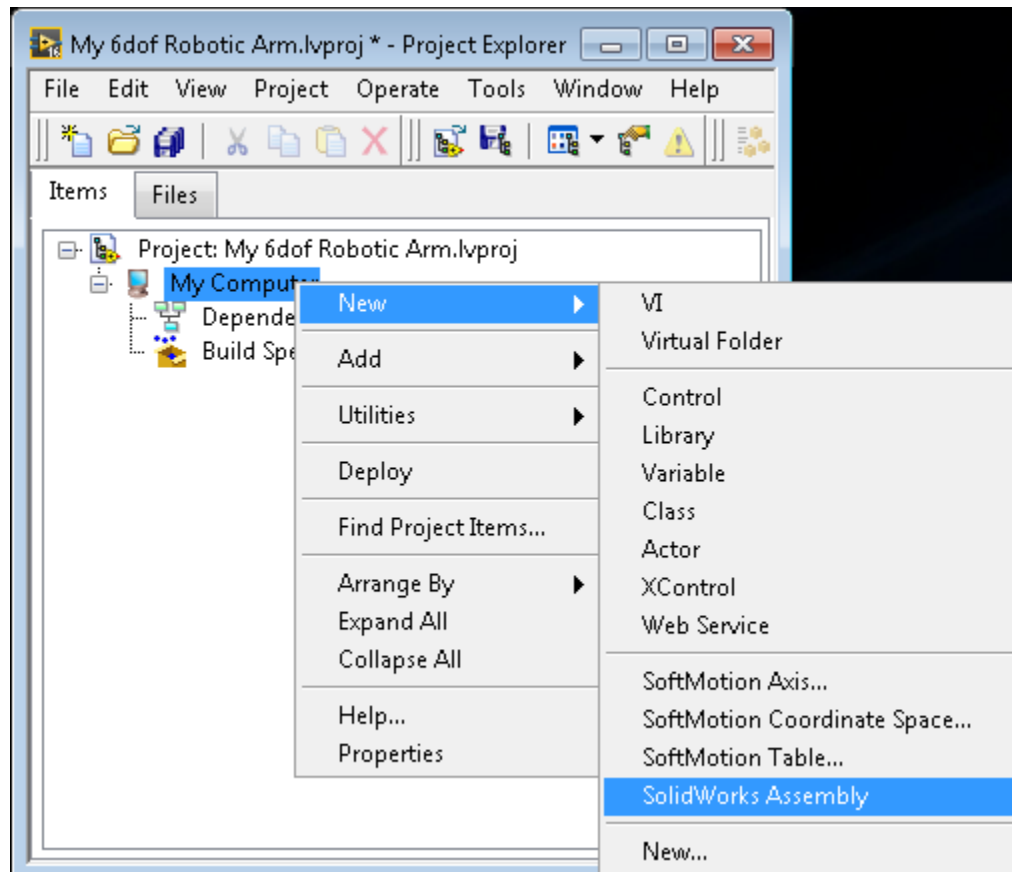
- ***Motion trajectory design σχεδιασμός κίνησης τροχιάς*** – Μπορούμε να δημιουργήσουμε πολύπλοκες κινήσεις που περιέχουν μια σειρά ενεργειών διαδοχικών ή ταυτόχρονων κινήσεων που αποτελούνται από ευθείες κινήσεις, κινήσεις τόξου ακόμα και πολύπλοκες κινήσεις όπως οδοντωτών τροχών και έκκεντρων.
- ***Visualization Οπτικοποίηση*** - Το μηχάνημα λειτουργεί ως ένα εικονικό πρωτότυπο και μας βοηθά να επικυρώσουμε τη συνολική προκαταρκτική του μελέτη πολύ πιο νωρίς από την ανάπτυξη του αξιολογώντας γρήγορα τη σκοπιμότητα του συνολικού εννοιολογικού σχεδιασμού του.
- ***Collision detection Ανίχνευση Σύγκρουσης*** - Με τη λειτουργία της ανίχνευσης σύγκρουσης μπορούμε να επικυρώσουμε διάφορα σχέδια προφίλ κίνησης, χρησιμοποιώντας το πραγματικό μοντέλο 3D CAD. Για την πρόληψη των συγκρούσεων μπορούμε να ελέγξουμε για παρεμβολές μεταξύ των εξαρτημάτων αλλά και με τον περιβάλλοντα χώρο, να αποφύγουμε κινήσεις κλειδώματος των μελών, να ελαχιστοποιήσουμε τις περιττές και άσκοπες κινήσεις, γρήγορα να αξιολογήσουμε (what-if) σενάρια, να βελτιστοποιήσουμε το προφίλ κίνησης μας και με ασφάλεια να δοκιμαστούν νέες λογικές του συστήματος ελέγχου χωρίς τον κίνδυνο να καταστραφεί η φυσική μηχανή μας αλλά και να μην διακοπή η λειτουργία λόγω λάθη προγραμματισμού.
- ***Throughput time studies Μελέτη χρόνου απόδοσης***- Με την επικύρωση του σχεδιασμού του συστήματος κίνησης μας χρησιμοποιώντας μια προσομοίωση που περιλαμβάνει τους πραγματικούς περιορισμούς του προφίλ κίνησης μπορούμε να υπολογίσουμε με ακρίβεια τον χρόνο κύκλου (cycle time) και την απόδοση της μηχανής μας.
- ***Transmission sizing Μέτρηση μετάδοσης*** - Μπορούμε να υπολογίσουμε την απαιτούμενη ροπή του κινητήρα που χρησιμοποιούμε όπως και τα διαγράμματα ταχύτητας, επιτάχυνσης και ωφέλιμου φορτίου για το προφίλ κίνησης.

LabVIEW - solidworks

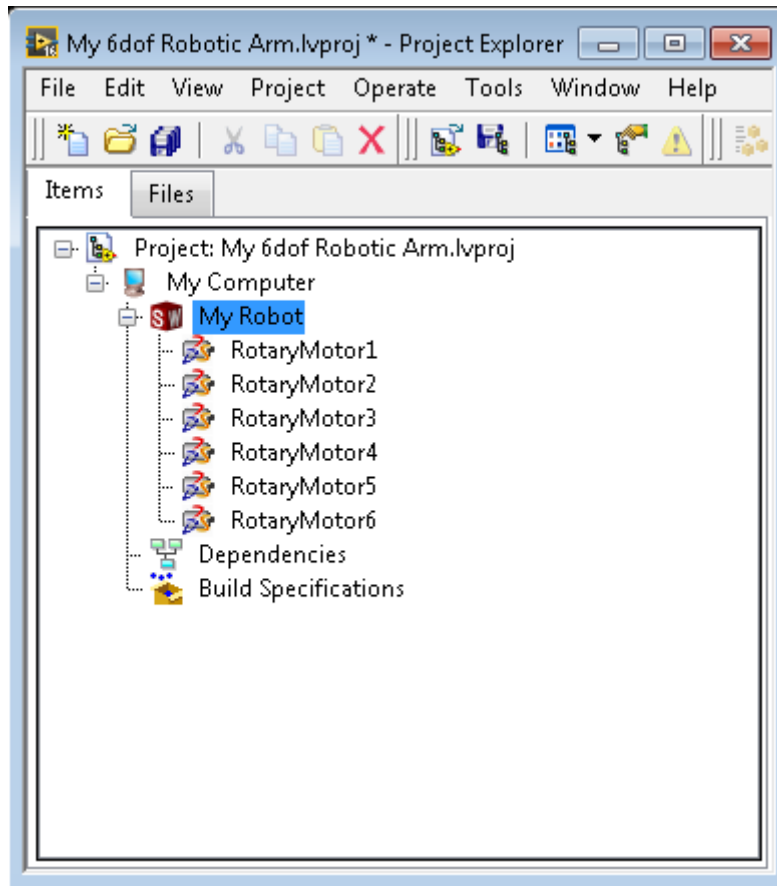


4.3.1 Εισαγωγή και επικοινωνία με το μοντέλο του SolidWorks

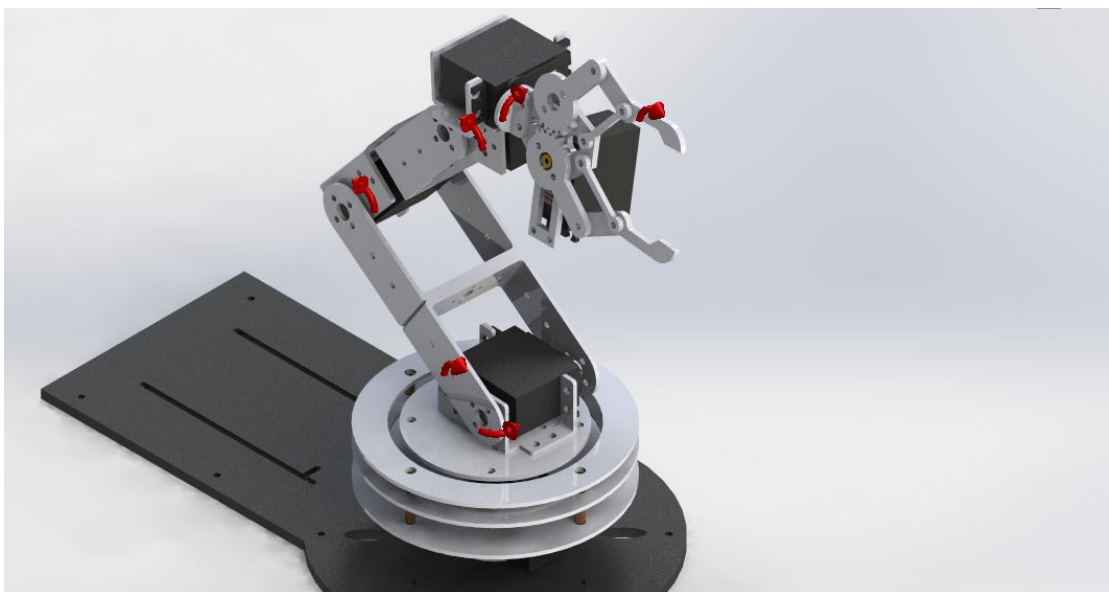
Αφού έχουμε συνδέσει τα δυο λογισμικά με την βοήθεια του “Soft Motion” εισάγουμε στο “LabVIEW” το τελικό μοντέλο του ρομποτικού βραχίονα με την επιλογή “SolidWorks Assembly”.



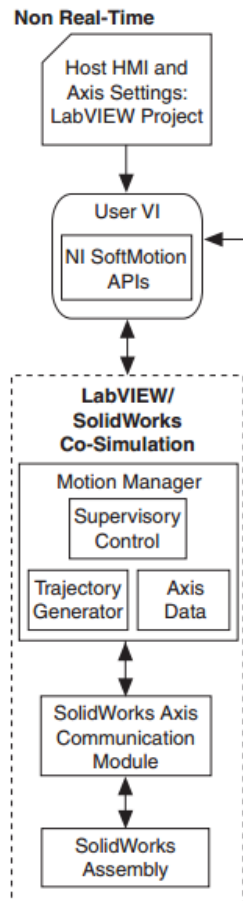
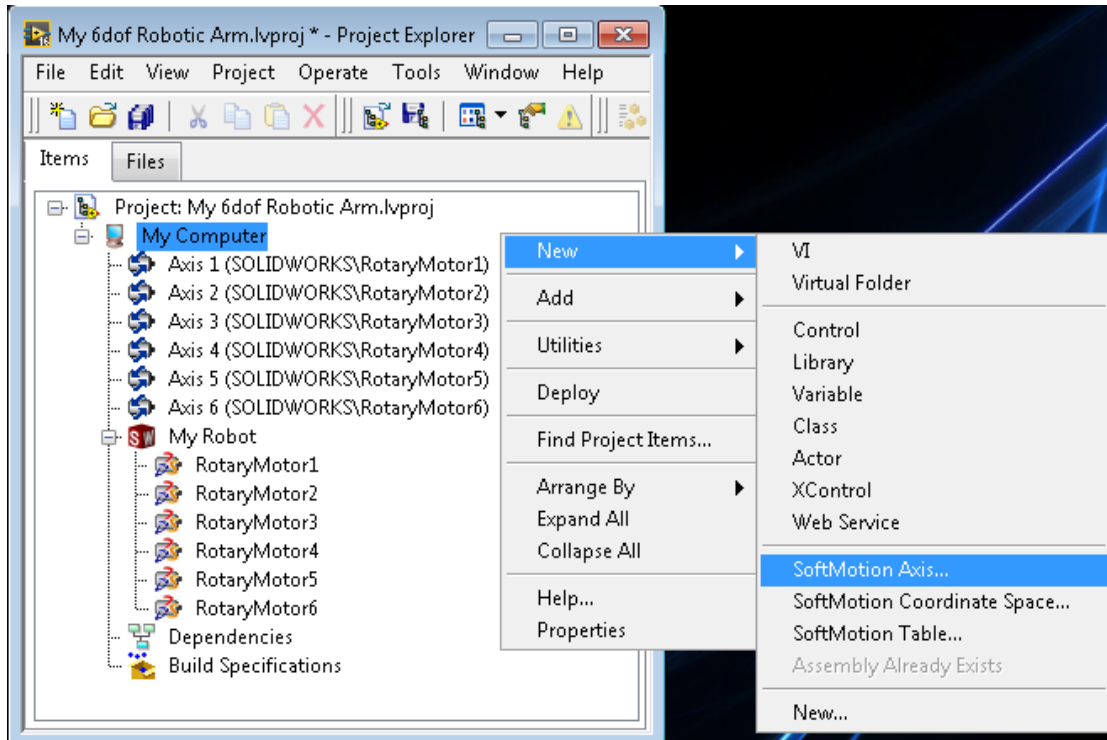
Όμως μόνο η εισαγωγή του μοντέλου δεν αρκεί για την επικοινωνία με τον βραχίονα πρέπει να εισάγουμε για κάθε μέλος και τον αντίστοιχο κινητήρα.



Οι τιμές που θα μπορούν να πάρουν οι γωνίες του κάθε μέλους θα παραμετροποιηθούν αργότερα στο πρόγραμμα έτσι η φορά των κινητήρων έχει οριστεί τυχαία δεξιόστροφα ή αριστερόστροφα.



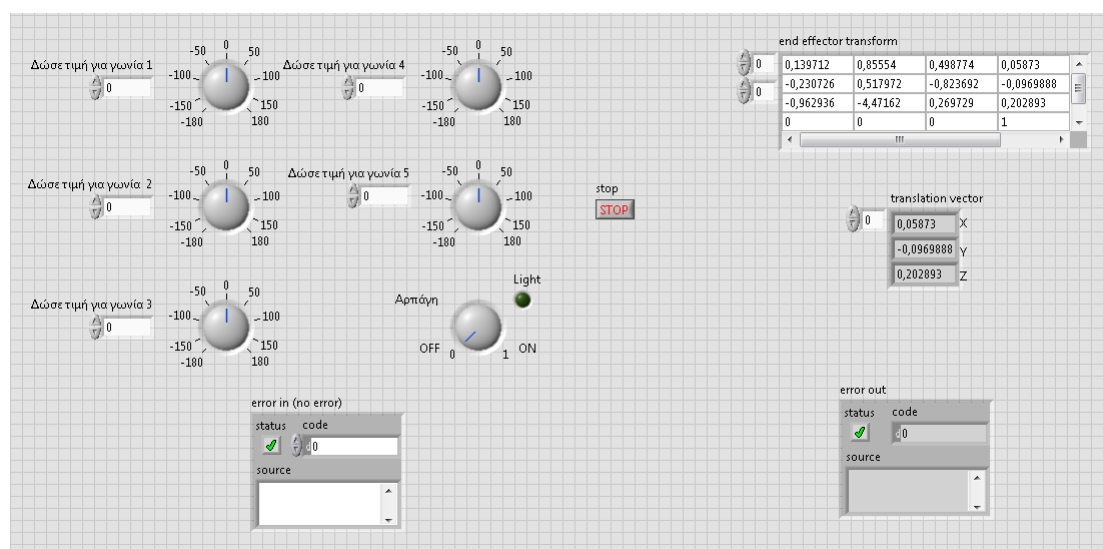
Επίσης για κάθε κινητήρα για να μπορέσει το LabVIEW να λάβει δεδομένα από το SolidWorks πρέπει όλοι οι κινητήρες ανατεθούν σε ξεχωριστό άξονα(Axis).



4.4 Φόρμες του Προγράμματος

4.4.1 Φόρμα ευθείας κινηματικής του ρομποτικού βραχίονα

Στην κυρίως φόρμα της ευθείας κινηματικής ο χρήστης έχει την επιλογή να δώσει τιμή σε κάθε γωνία του βραχίονα είτε με απευθείας πληκτρολόγηση είτε με ένα εικονικό ποτενσιόμετρο όπως φαίνεται στα αριστερά της εικόνας. Η αρπάγη έχει οριστεί αλλά και σχεδιαστεί να παίρνει 2 τιμές την μηδενική που σημαίνει ότι είναι κλειστή (κατάσταση OFF) και τις 90° μοίρες που σημαίνει ότι είναι ανοιχτή (κατάσταση ON). Για να είμαστε σίγουροι για το πρόγραμμα που δημιουργήσαμε, πρέπει να προβλέψουμε κάθε πρόβλημα που ένας χρήστης μπορεί να συναντήσει έτσι ώστε να υπάρχει έλεγχος για τυχόν σφάλματα έχει προστεθεί ένα "button Stop" που λειτουργεί δείχνοντας τον χρήστη εάν η τιμές που έχει εισάγει είναι εντός ορίων ή εάν η κίνηση του βραχίονα είναι εφικτή. Στα δεξιά της εικόνας βλέπουμε τον πίνακα μετασχηματισμού της αρπάγης εκεί όπου βρίσκεται κάθε φορά το τελικό σημείο του βραχίονα επίσης από εκεί εξάγουμε και το τελικό σημείο X,Y,Z που είναι και η τέταρτη στήλη του πίνακα μετασχηματισμού.



4.4.2 Φόρμα αντίστροφης κινηματικής του ρομποτικού βραχίονα

Στην κυρίως φόρμα της αντίστροφης κινηματικής ο χρήστης εισάγει το τελικό σημείο στο οποίο θέλει να φτάσει ο ρομποτικός βραχίονας στο χώρο.

Η αρπάγη ενεργοποιείται όπως και στην φόρμα της ευθείας κινηματικής ακόμη έχουμε προσθέσει κατάλληλο κώδικα για τυχόν σφάλματα. Το πρόγραμμα παίρνει το X,Y,Z που δίνει ο χρήστης τα εισάγει στον πίνακα μετασχηματισμού και δίνει τις γωνίες του βραχίονα $\theta_1, \theta_2, \theta_3, \theta_4, \theta_5$.

Διάγραμμα μετασχηματισμού

0	24	Δώσε X
	-32	Δώσε Y
	136	Δώσε Z

Αρπάγη OFF 0 ON 1 Light

stop STOP

end effector transform

0	-0,891	0,454	0	24
0	-0,454	-0,891	0	-32
	0	0	1	136
	0	0	0	0

error in (no error)

status	code
✓	0
source	

error out

status	code
✓	0
source	

θ_1 5,47521

θ_2 -18384,6

θ_3 37081,9

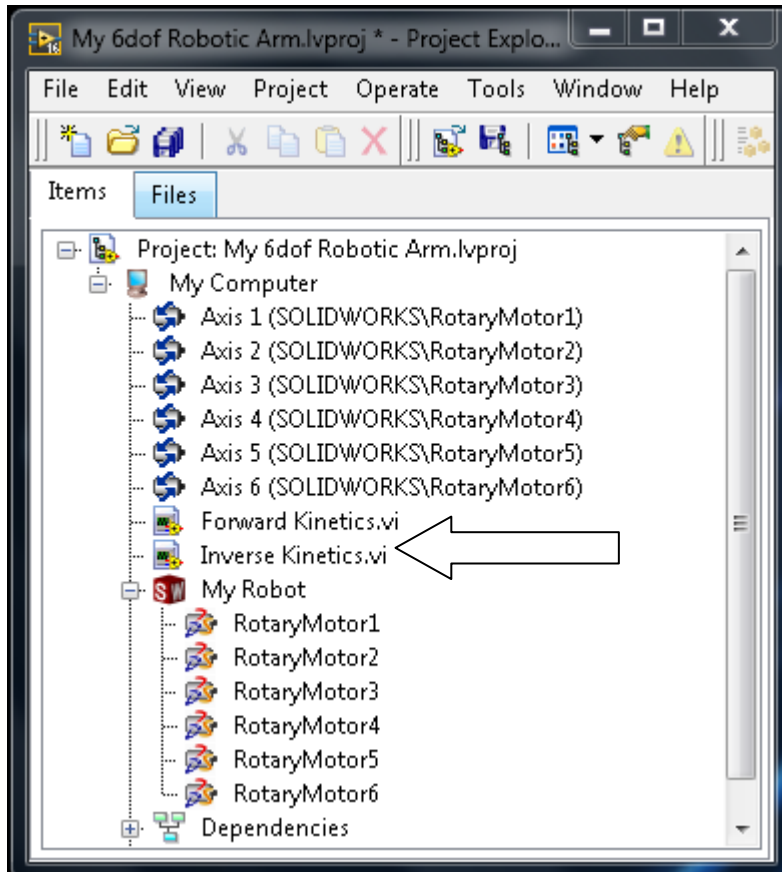
θ_4 -18697,3

θ_5 -5,47521

5.Εφαρμογή

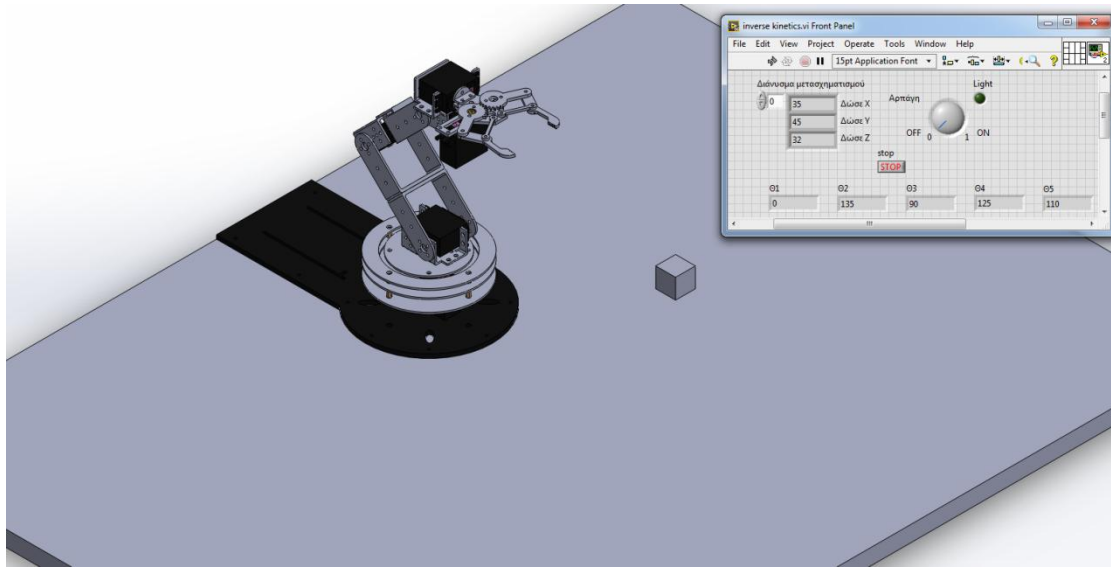
5.1 Εκτέλεση Προγράμματος

Ανοίγοντας το πρόγραμμα του LabVIEW ο χρήστης μπορεί να επιλέξει με ποίον τρόπο θέλει να κινήσει τον ρομποτικό βραχίονα πατώντας στην φόρμα Forward Kinetics (ευθεία κινηματική) ή Inverse Kinetics (αντίστροφη κινηματική).Εμείς έχουμε επιλέξει την φόρμα "Inverse Kinetics" η οποία είναι και η δυσκολότερη σε επίπεδο υλοποίησης.

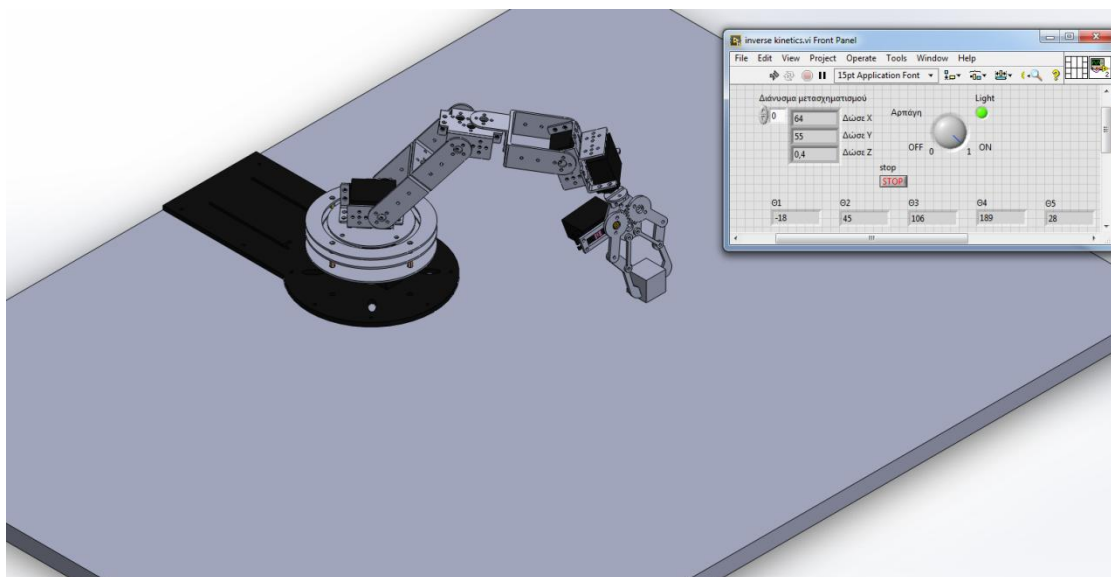


Η αρχική θέση του βραχίονα έχει οριστεί η παρακάτω όπως φαίνεται στην εικόνα για την καλύτερη επίδειξη του προγράμματος χρειάστηκε να σχεδιάσουμε επίσης ένα τραπέζι και ένα κύβο. Λειτουργώντας την εφαρμογή σαν "pick & place" θα μεταφέρουμε τον κύβο από την μια μεριά του τραπέζιού στην άλλη απλά δίνοντας στην εφαρμογή το σημείο στο οποίο το οποίο είναι ο κύβος και στο σημείο που θέλουμε να πάει.

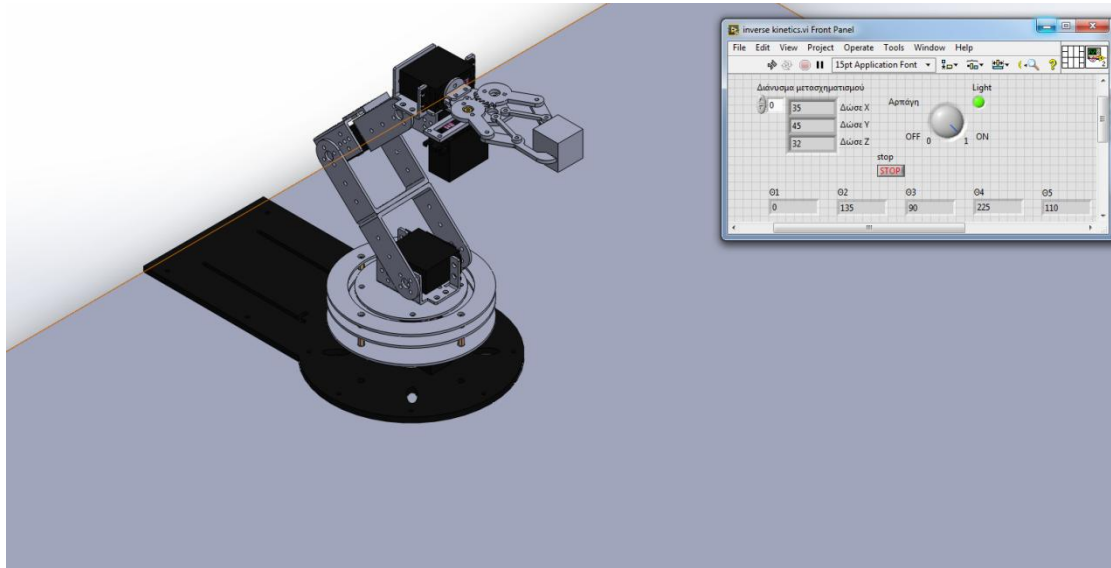
Ξεκινώντας από την αρχική θέση και γνωρίζοντας την απόσταση του κύβου από την βάση του ρομποτικού βραχίονα δίνουμε το σημείο X,Y,Z.



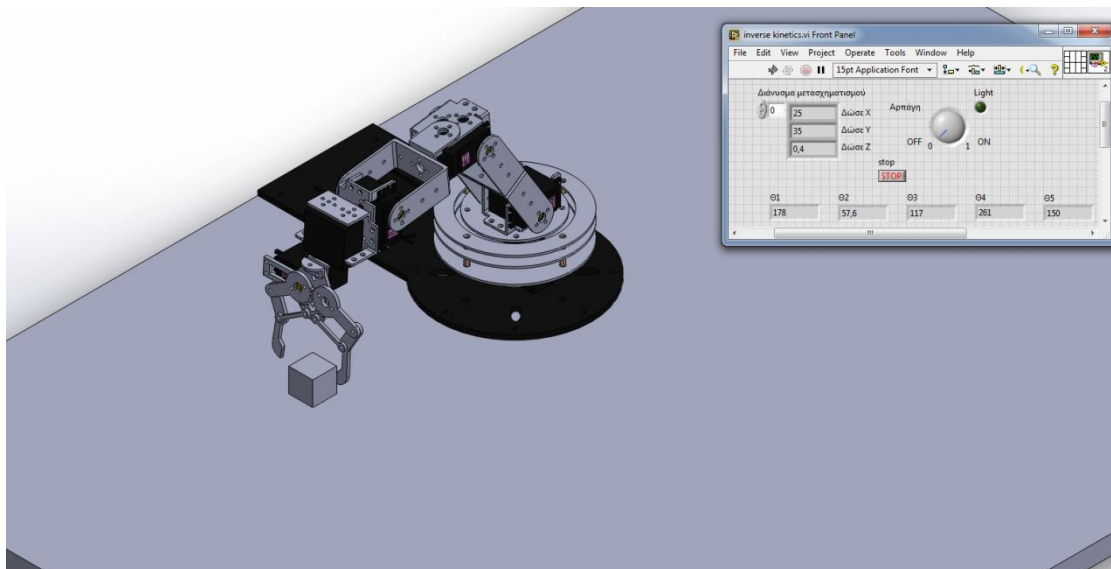
Βλέπουμε ότι όλα σχεδόν τα μέλοι δηλαδή οι γωνίες του ρομπότ έχουν πάρει την κατάλληλη θέση ώστε η αρπάγη να φτάσει το σημείο που ορίσαμε. Ακόμη ενεργοποιούμε την αρπάγη η οποία κλείνει μέχρι να σηναντήσει κάποιο αντικείμενο.



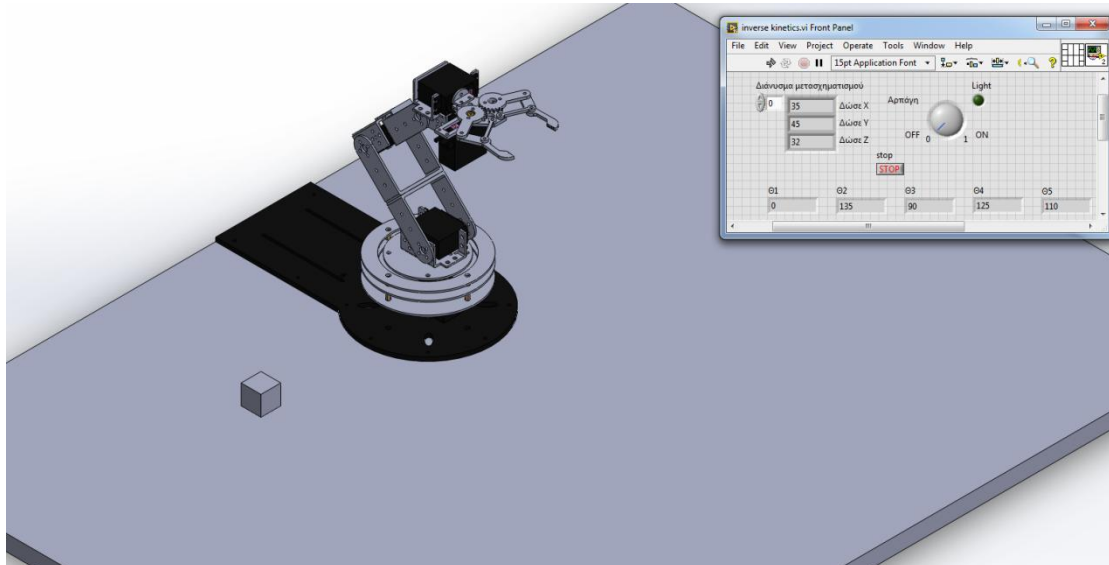
Στην εικόνα φαίνεται το ρομπότ στην ενδιάμεση θέση τοποθέτησης του αντικειμένου



Ο ρομποτικός βραχίονας τοποθετεί το αντικείμενο στην τελική επιθυμητή θέση, και απενεργοποιούμε την αρπάγη



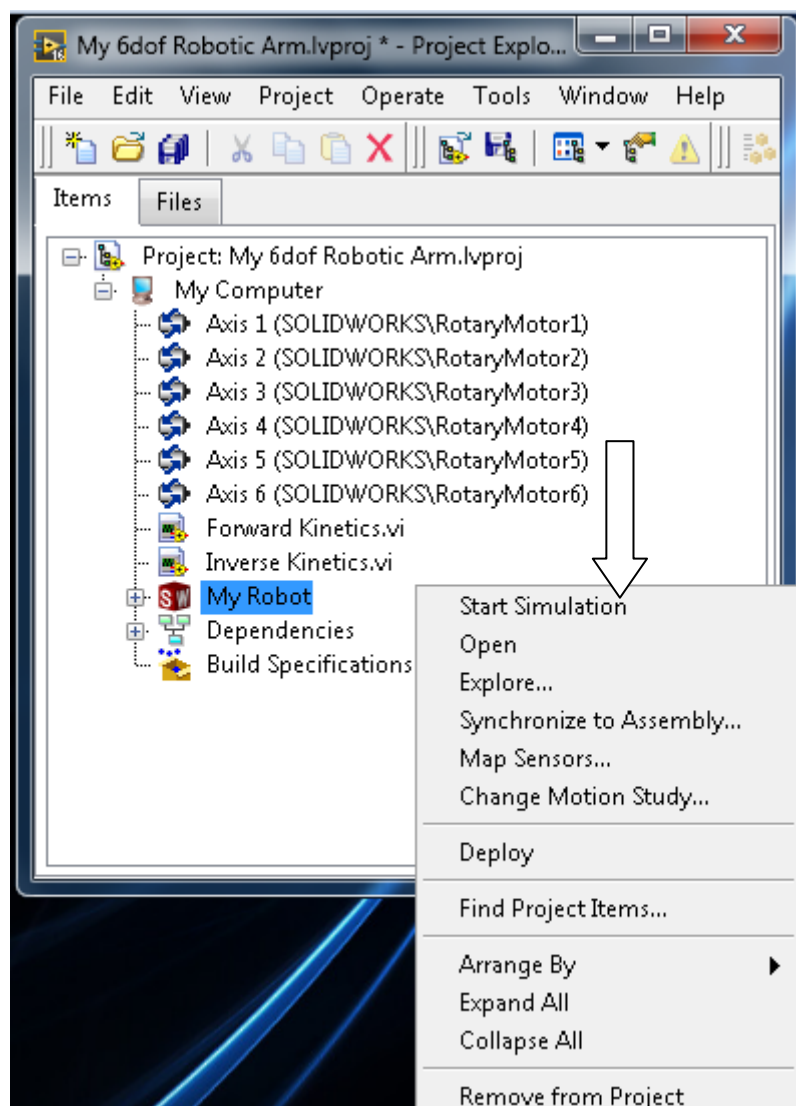
Στη συνέχεια ο βραχίονας επιστρέφει στην αρχική του θέση όπως φαίνεται στην εικόνα



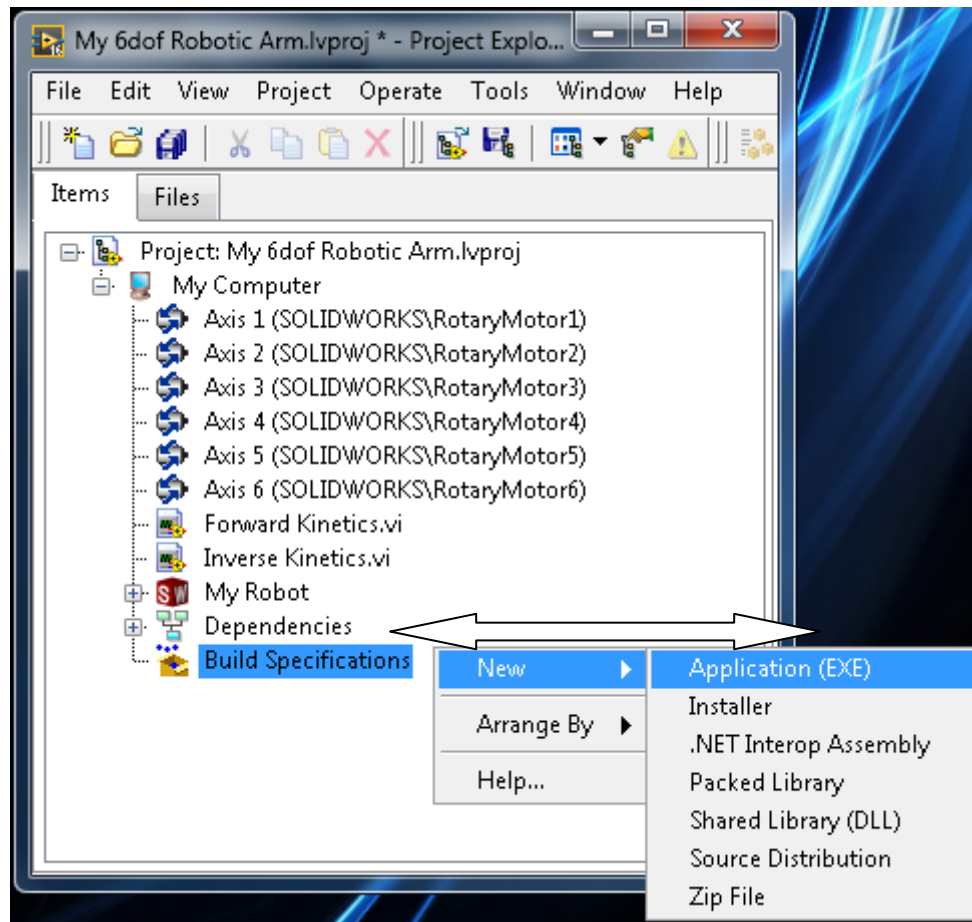
Αφού τελειώσουμε με οποιαδήποτε κίνηση μας το "SolidWorks" μας δίνει την επιλογή να δούμε ολοκληρωμένη την κίνηση του ρομποτικού βραχίονα όπως έχει προγραμματιστεί μέσω "animation". Εκεί μπορούμε να δούμε για τυχόν συγκρούσεις με αλλά μελή ή εμπόδια.

5.2 Εξαγωγή Εκτελέσιμου Κώδικα

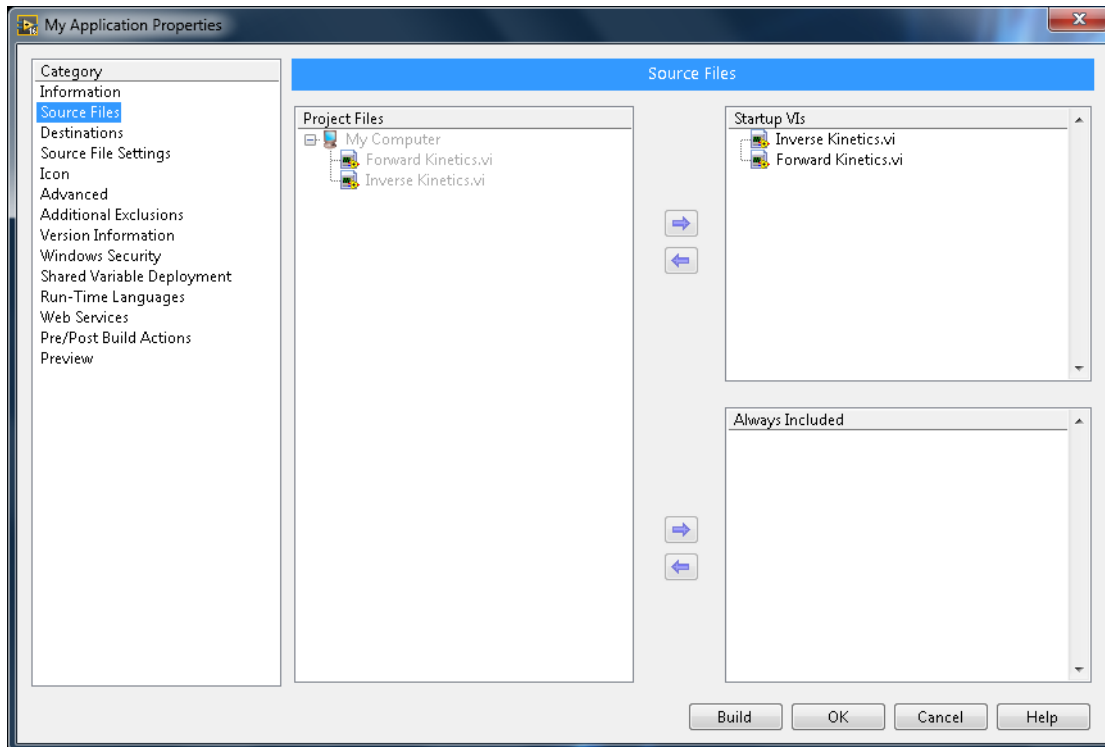
Αφού τελειώσουμε με οποιαδήποτε κίνηση μας στη συνέχεια γίνεται εκτέλεση της εντολής “Start Simulation” και βλέπουμε τον ρομποτικό βραχίονα όπως έχει προγραμματιστεί. Στο σημείο αυτό μπορούμε να δούμε για τυχόν συγκρούσεις με άλλα μελή ή εμπόδια. Εφόσον δεν εντοπισθεί κάποια σύγκρουση τότε ο χρήστης μπορεί να εξάγει τον κώδικα για τον προγραμματισμό του ρομπότ.



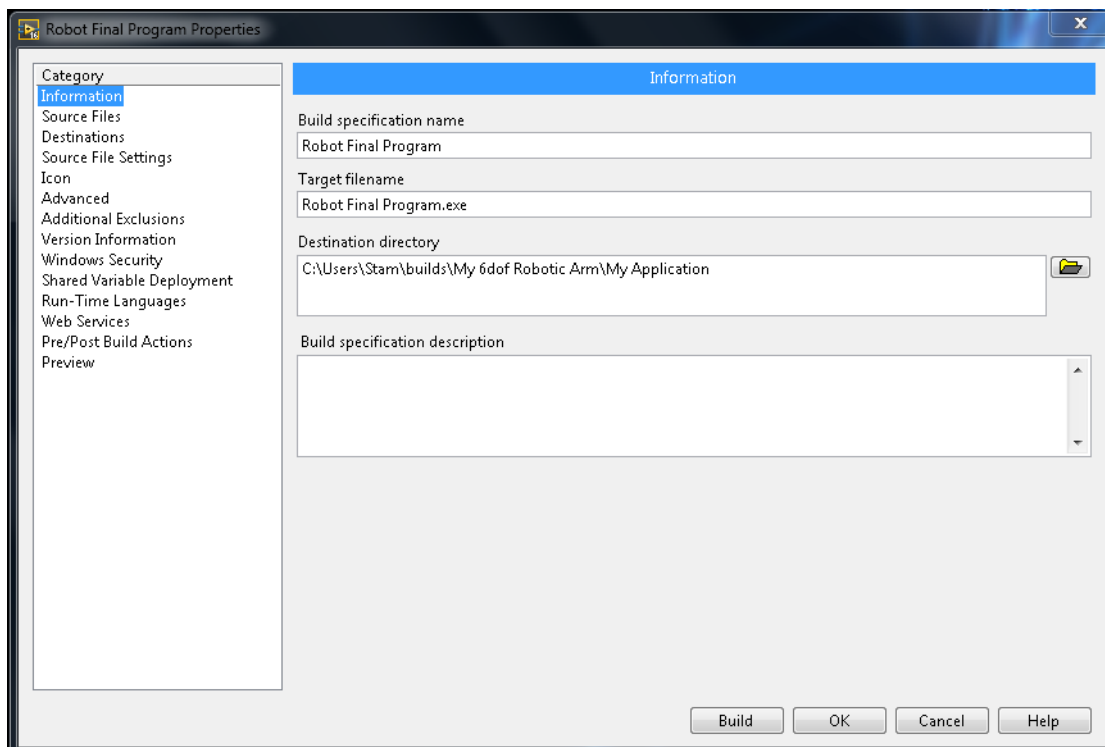
Αφού έχουμε τρέξει το “Start Simulation” το πρόγραμμα έχει αποθηκεύσει την κίνηση του ρομποτικού βραχίονα. Το LabVIEW μέσω του “Build Specifications” μας δίνει την δυνατότητα να παράγουμε απευθείας εκτελέσιμο κώδικα σε μορφή (.EXE) ή ακόμη και σε μορφή (.DLL) όπως ακριβώς θα κάναμε “compile” σε οποιοδήποτε άλλο πρόγραμμα.



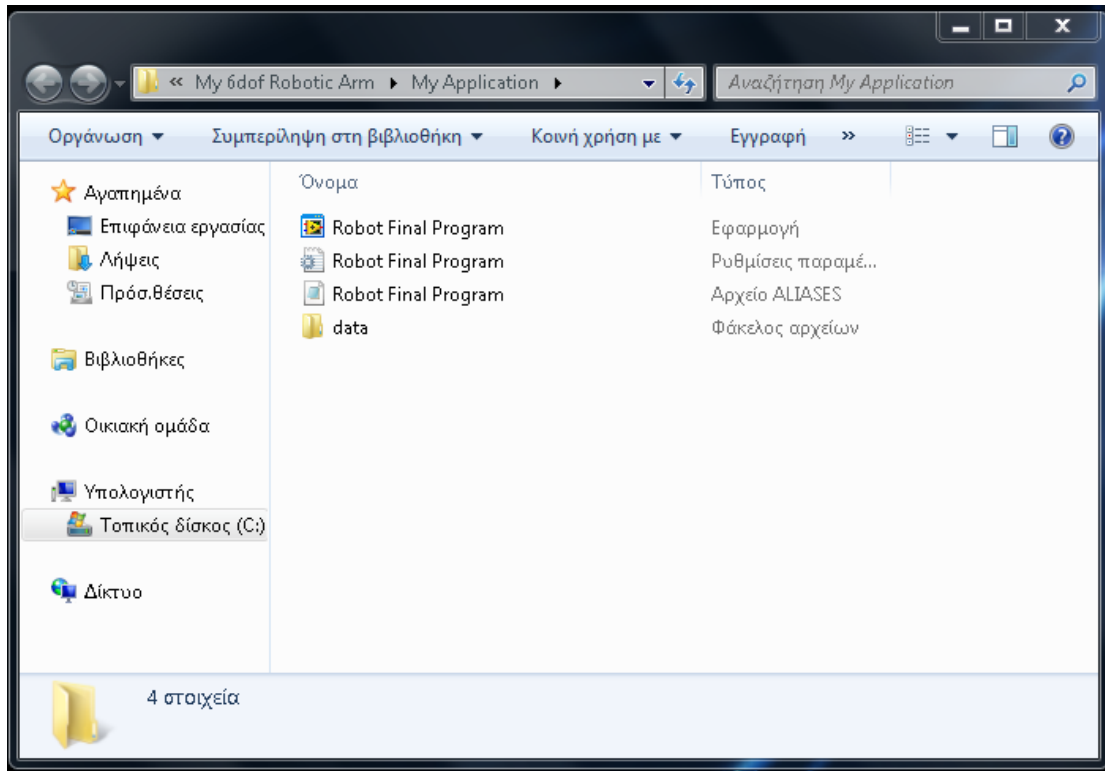
Στην καρτέλα “Source File” εισάγουμε την φόρμα στην οποία κάναμε το “Simulation” στην περίπτωση μας το “Inverse Kinetics”.



Στην καρτέλα “Information” εισάγουμε το όνομα του προγράμματος μας και κάνουμε “Build”.



Αφού κάνουμε “Build” η εφαρμογή μας είναι πλέον εκτελέσιμη παρακάτω φαίνεται ένα μέρος του κώδικα σε μορφή “.txt”. Τέλος κατεβάζοντας το αρχείο με τον κώδικα στην μονάδα ελέγχου του ρομποτικού βραχίονα το ρομπότ εκτελεί ακριβώς την κίνηση την οποία έχουμε προγραμματίσει.



```
Robot Final Program - Σημειωματάριο
Αρχείο  Επεξεργασία  Μορφή  Προβολή  Βοήθεια
Option Explicit
Dim swApp As Sldworks.Sldworks
Dim swModel As Sldworks.ModelDoc2
Dim swModelDocExt As Sldworks.ModelDocExtension
Dim swSketchManager As Sldworks.SketchManager
Dim swSketchSegment As Sldworks.SketchSegment
Dim swSketch As Sldworks.Sketch
Dim status As Boolean
Dim exApp As Excel.Application
Dim sheet As Excel.Worksheet
Sub main()
    Set swApp = Application.Sldworks
    'Open new part document
    Set swModel = swApp.NewDocument("C:\ProgramData\Solidworks\Solidworks 2015\templates\Part.prt", 0, 0, 0)
    'Insert 3D sketch
    Set swSketchManager = swModel.SketchManager
    swSketchManager.Insert3DSketch True
    Set swSketchSegment = swSketchManager.CreateCenterLine(-0.082642, 0.005659, 0#, -0.049926, 0.045073, 0#)
    Set swSketch = swSketchManager.ActiveSketch
    status = swSketch.SetWorkingPlaneOrientation(0, 0, 0, 1, 0, 0, 1, 1, 0, 0)
    Set swSketchSegment = swSketchManager.CreateCenterLine(-0.049926, 0.045073, 0#, -0.049926, -0.022634, -0.065874)
    Set swSketch = swSketchManager.ActiveSketch
    status = swSketch.SetWorkingPlaneOrientation(0, 0, 0, 0, 1, 1, 0, 0, 1, 0)
    swModel.ClearSelection2 True
    swSketchManager.InsertSketch True
    swModel.ActivateSelectedFeature
    swModel.ClearSelection2 True
    swSketchManager.InsertSketch True
    Set swModelDocExt = swModel.Extension
    status = swModelDocExt.SelectByID2("Front Plane", "PLANE", 0, 0, 0, False, 0, Nothing, 0)
    swModel.ClearSelection2 True
    Set swSketchSegment = swSketchManager.CreateCircle(-0.056401, 0.005985, 0#, -0.054697, -0.005141, 0#)
    swModel.ClearSelection2 True
    swSketchManager.InsertSketch True
    swModel.ClearSelection2 True
    'Insert a 3D sketch plane
    swSketchManager.Insert3DSketch True
    status = swModelDocExt.SelectByID2("Line1@3DSketch1", "EXTSKETCHSEGMENT", -5.656096142099999E-02, 3.70796232466087E-02, 0, True, 0, Nothing, 0)
    status = swModelDocExt.SelectByID2("Point2@sketch1", "EXTSKETCHPOINT", -5.64010297276809E-02, 5.98490302365917E-03, 0, True, 0, Nothing, 0)
    status = swSketchManager.CreateSketchPlane(9, 9, 0)
    status = swModelDocExt.SelectByID2("Plane1", "SKETCHSURFACES", 0, 0, 0, False, 0, Nothing, 0)
    swModel.ActivateSelectedFeature
    swModel.ClearSelection2 True
    swSketchManager.InsertSketch True
    Set swApp = GetObject(, "sldworks.application")
    If Not swApp Is Nothing Then
        Set doc = swApp.ActiveDoc
        If Not doc Is Nothing Then
            If doc.GetType = swDocPART Then
                Set part = doc
                Set sm = doc.SelectionManager
                If Not part Is Nothing And Not sm Is Nothing Then
```

6. Αποτελέσματα-Συμπεράσματα

Συνοψίζοντας βλέπουμε με την συνεργασία δυο προγραμμάτων καταφέραμε να πετύχουμε τον “off-line” προγραμματισμό του ρομποτικού βραχίονα. Αυτό έχει πολλά ωφέλει για τον προγραμματιστή-χρήστη, ένα από τα κυριότερα είναι ότι μπορεί να προγραμματίσει το ρομπότ από απόσταση χωρίς να χρειάζεται η φυσική του παρουσία στο χώρο εργασίας. Επίσης μειώνεται η χρονοβόρα διαδικασία και το σταμάτημα της παραγωγικής διαδικασίας για τον προγραμματισμό του ρομπότ για μια άλλη κατεργασία. Εύκολα ο προγραμματιστής με απλές γνώσεις σχεδίου μπορεί να δημιουργήσει μια καινούργια αρπάγη ώστε το ρομπότ να χρησιμοποιηθεί για παράδειγμα σε εφαρμογή συγκόλλησης (welding) χωρίς να χρειαστεί να πειράξει το κυρίως πρόγραμμα του βραχίονα.

Όσο αφορά το πρόγραμμα του SolidWorks το βασικότερο είναι ότι μπορούμε να μεταφέρουμε οτιδήποτε από την πραγματικότητα στο 3D περιβάλλον και με μεγάλη λεπτομέρεια. Ιδιαίτερο ενδιαφέρον παρουσιάζουν τα πρόσθετα του Solid Works στα οποία μπορούμε να κάνουμε στατικές και δυναμικές αναλύσεις όπως και να εξάγουμε το μηχανολογικό σχέδιο κάθε εξαρτήματος ώστε να το κατεργαστούμε σε κάποιο CNC. Το Solid Works επίσης μπορεί να μας δώσει το λογαριασμό του κάθε εξαρτήματος αναλόγως το υλικό που χρησιμοποιούμε πάνω του πχ χαλκός, αλουμίνιο, χρυσός, ξύλο κ.α.

Τέλος το LabVIEW είναι ένα πρόγραμμα το οποίο συνεργάζεται αρκετά ικανοποιητικά με το SolidWorks προσθέτοντας ένα γραφικό περιβάλλον προγραμματισμού διαφορετικό από το κλασικό συντακτικό κώδικα. Έχει δυνατότητες επεξεργασίας σημάτων, επικοινωνία με hardware και συμπεριλαμβάνει αρκετά μεγάλη ποικιλία μαθηματικών συναρτήσεων οι οποίες μας βοήθησαν στην ανάπτυξη των κινηματικών αναλύσεων χωρίς να χρειαστεί η περεταίρω βοήθεια από άλλο πρόγραμμα υπολογισμού όπως το Wolfram Mathematica

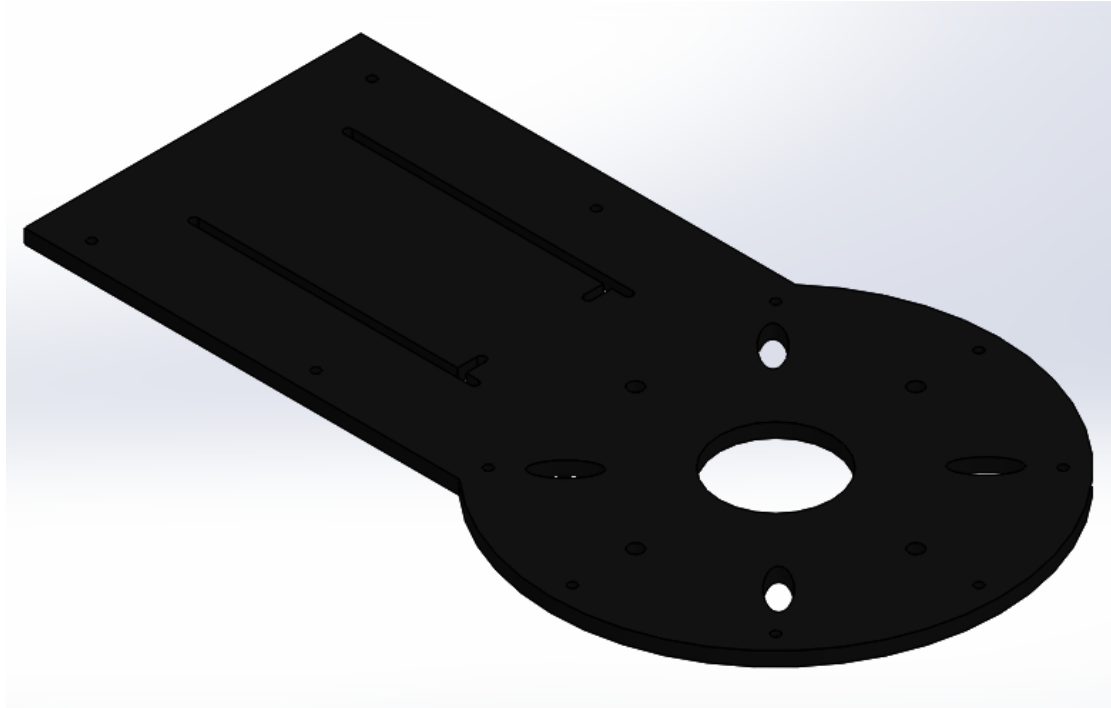
7.Βιβλιογραφία-Αναφορές

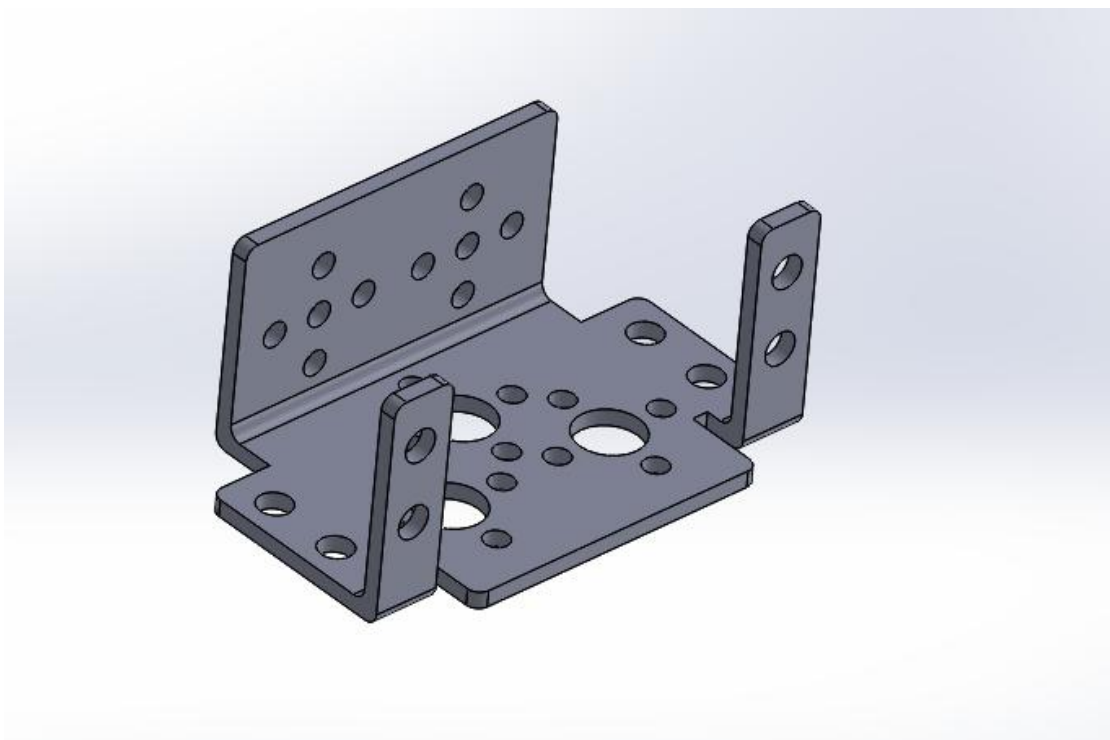
1. John J. Craig Εισαγωγή στην Ρομποτική-Εισαγωγή και Αυτόματος Έλεγχος Εκδόσεις Τζιόλα 2009
2. Labview για Μηχανικούς Προγραμματισμός Συστημάτων DAQ, Κωνσταντίνου ΙΠ. Καλοβρέκτη, Εκδόσεις Τζιόλα 2005
3. Δουλγέρη Ζωή. “Ρομποτική- κινηματική, δυναμική και έλεγχος αρθρωτών βραχιόνων”, Αθήνα(2007): Εκδόσεις Κριτική
4. "Robot Dynamics and Control", Mark W. Spong, Seth Hutchinson, and M. Vidyasagar, Second Edition, 2004
5. Τσελές Δημήτριος CAD-CAM Αθήνα Σύγχρονη Εκδοτική 2003
6. Νικόλαος Α. Μπιλάλης, Εμμανουήλ Μαραβελάκης,(2009) Συστήματα CAD/ CAM και τρισδιάστατη μοντελοποίηση
7. Matt Lombard, SolidWorks 2010 Bible, Wiley Publishing Inc
8. https://www.researchgate.net/publication/238739709_Software_Development_for_the_Kinematic_Analysis_of_a_Lynx_6_Robot_Arm
9. <http://www.cds.caltech.edu/~murray/books/MLS/pdf/mls94-complete.pdf>
10. <https://www.youtube.com/channel/UCjlg6Eq1XVxz95qJOQlycCQ/videos> Stem education
11. http://www.intechopen.com/books/industrial_robotics_theory_modelling_and_control/robot_kinematics_forward_and_inverse_kinematics
12. <http://www.ni.com/tutorial/9566/en/>
13. <http://www.cds.caltech.edu/~murray/books/MLS/pdf/mls94-complete.pdf>
14. <http://www.solidworks.com/sw/support/downloads.htm> Dassault Systemes SolidWorks 3D Design Software
15. <https://www.ni.com/downloads/products/> National Instruments LabVIEW Test, Measurement and Embedded Systems Design
16. <http://www.ni.com/labview/softmotion/> National Instruments Softmotion Module

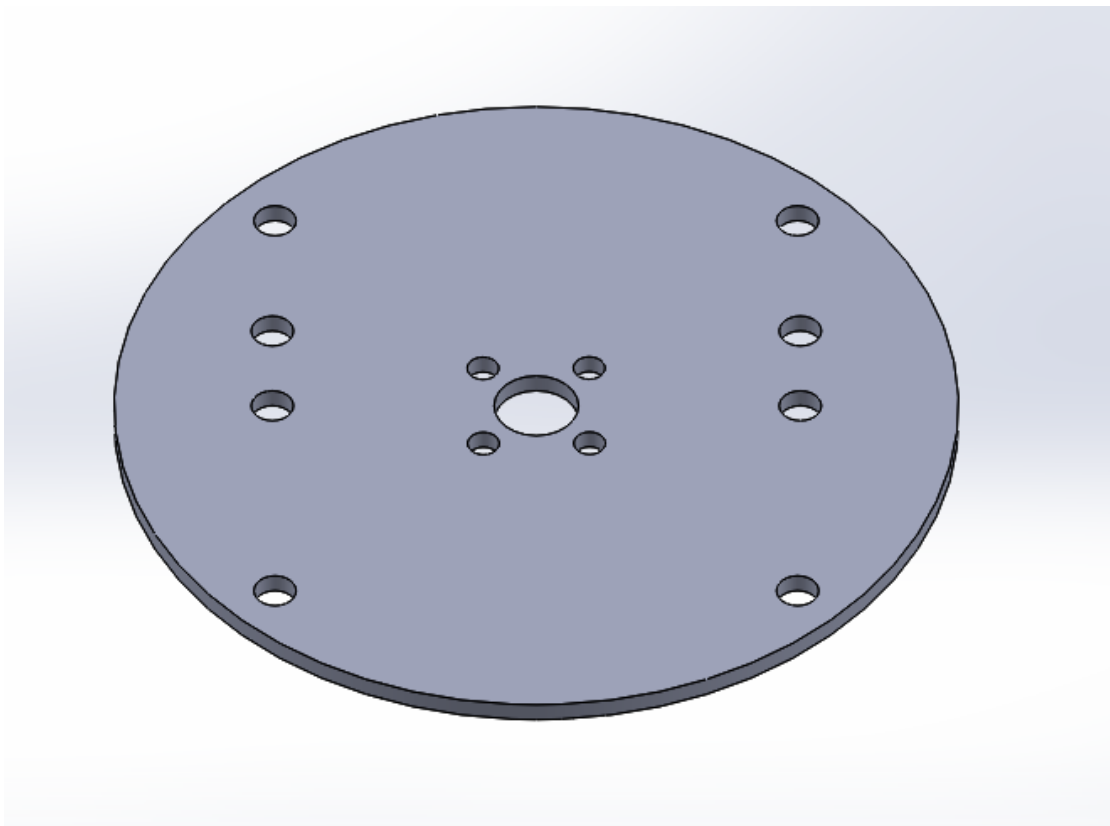
8.Παράρτημα

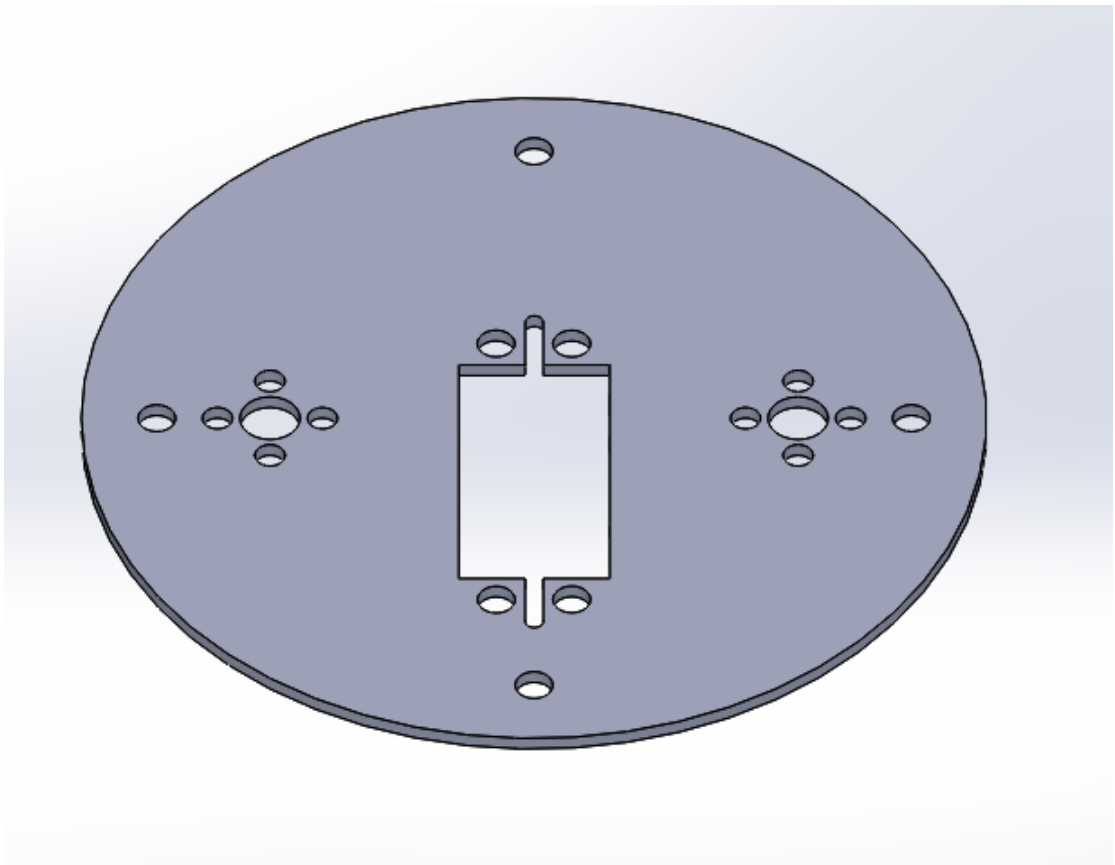
8.1 Φωτογραφικό Υλικό Ρομποτικού βραχίονα και 3D σχέδια SolidWorks

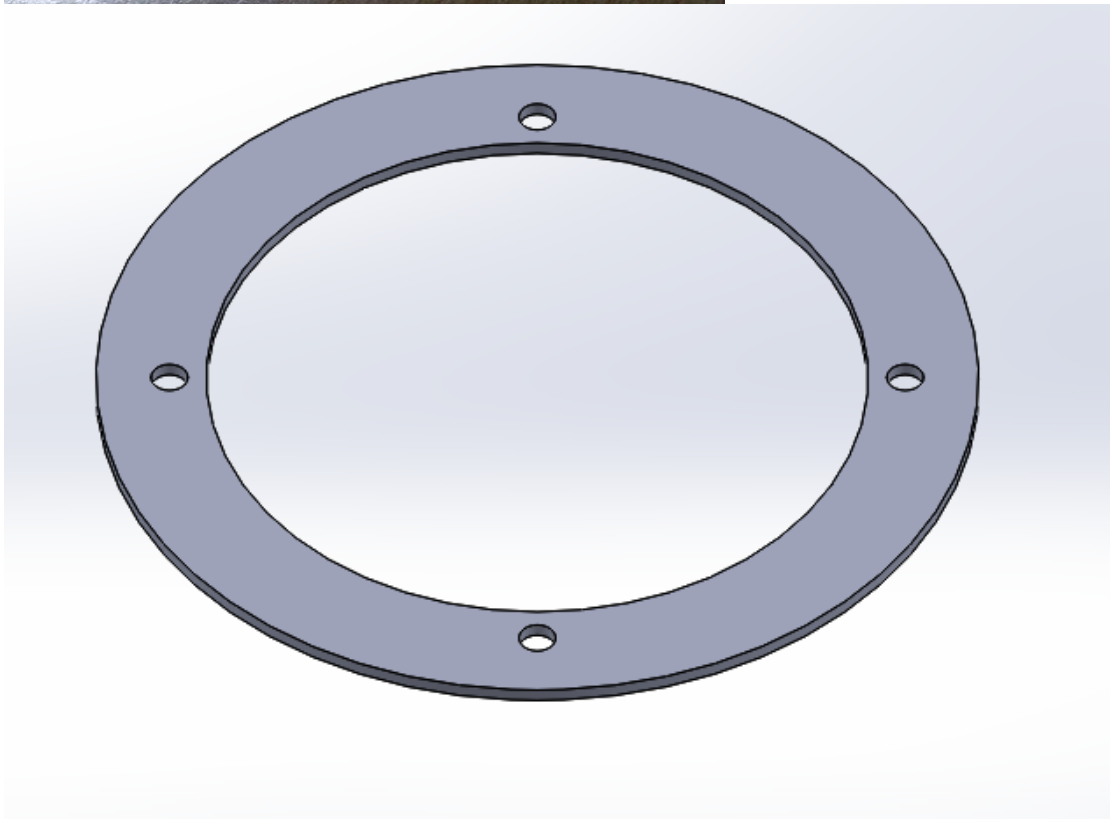
Παρακάτω ακολουθούν φωτογραφίες των κομματιών, και εικόνες από το 3D σχεδιασμό τους

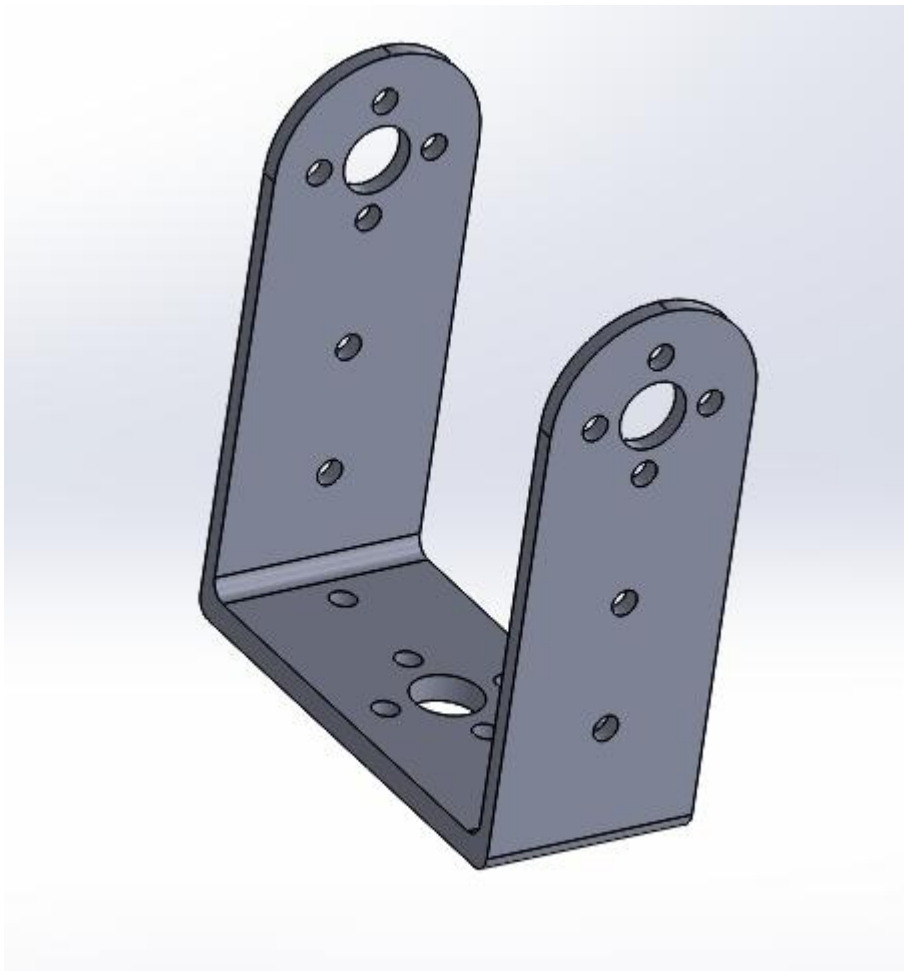


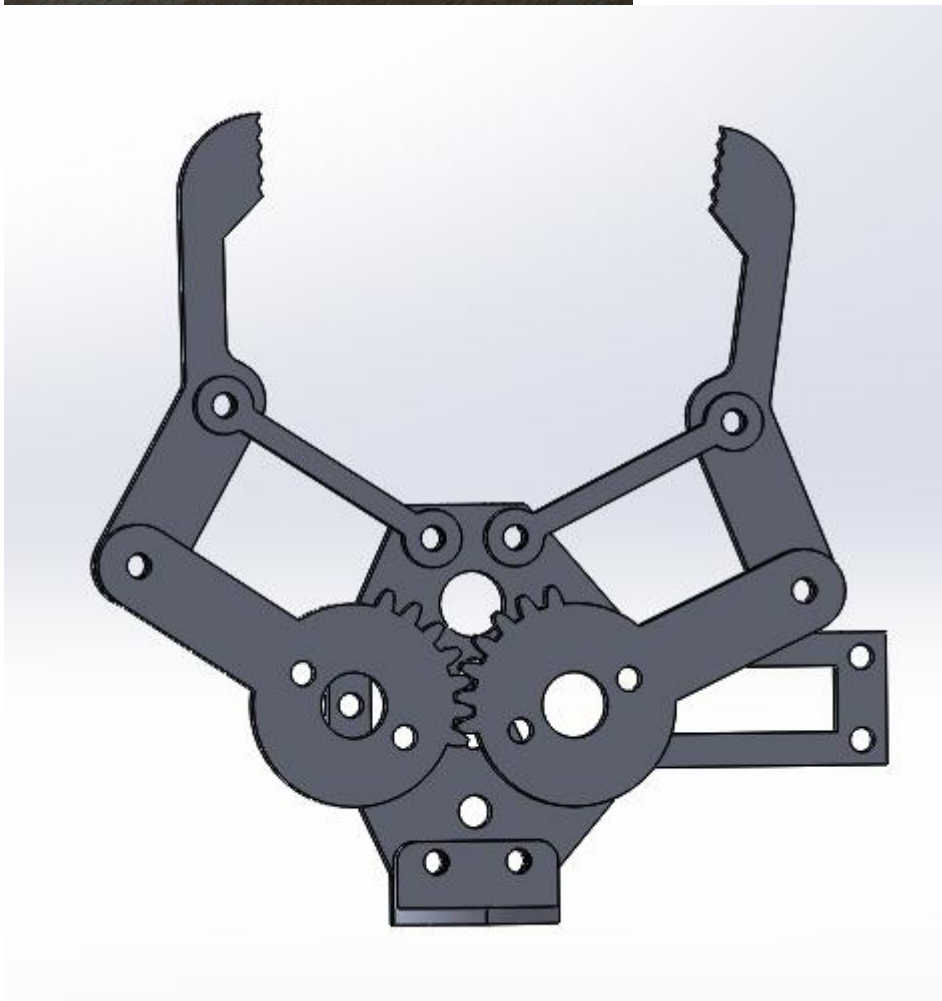


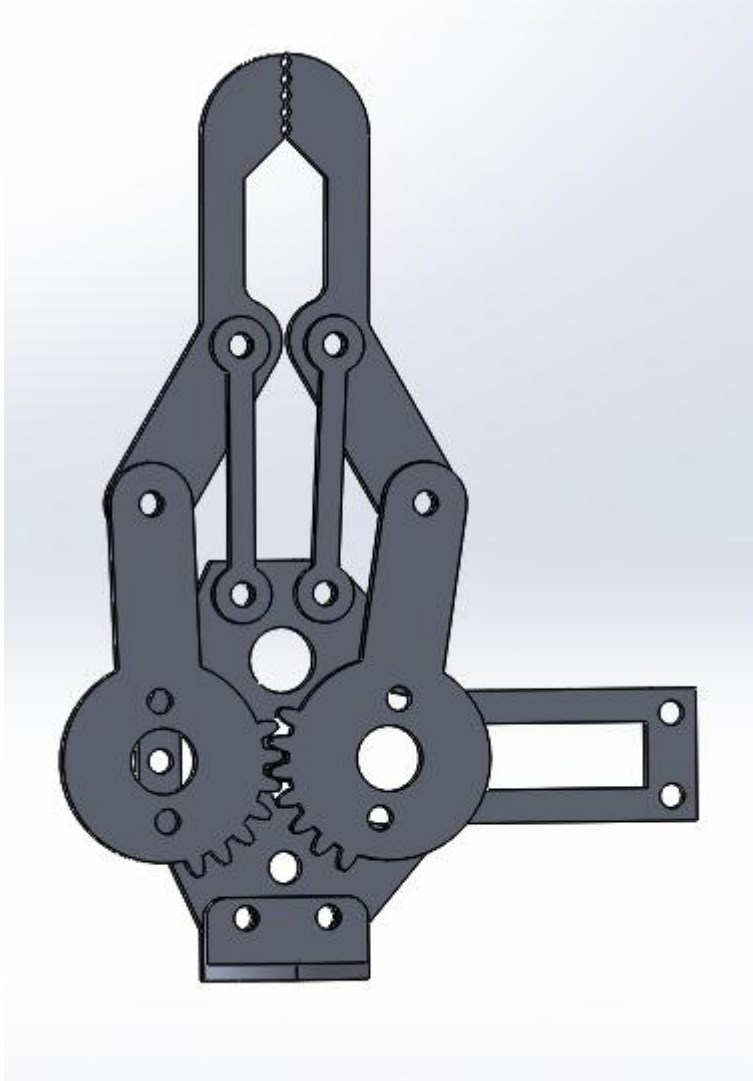


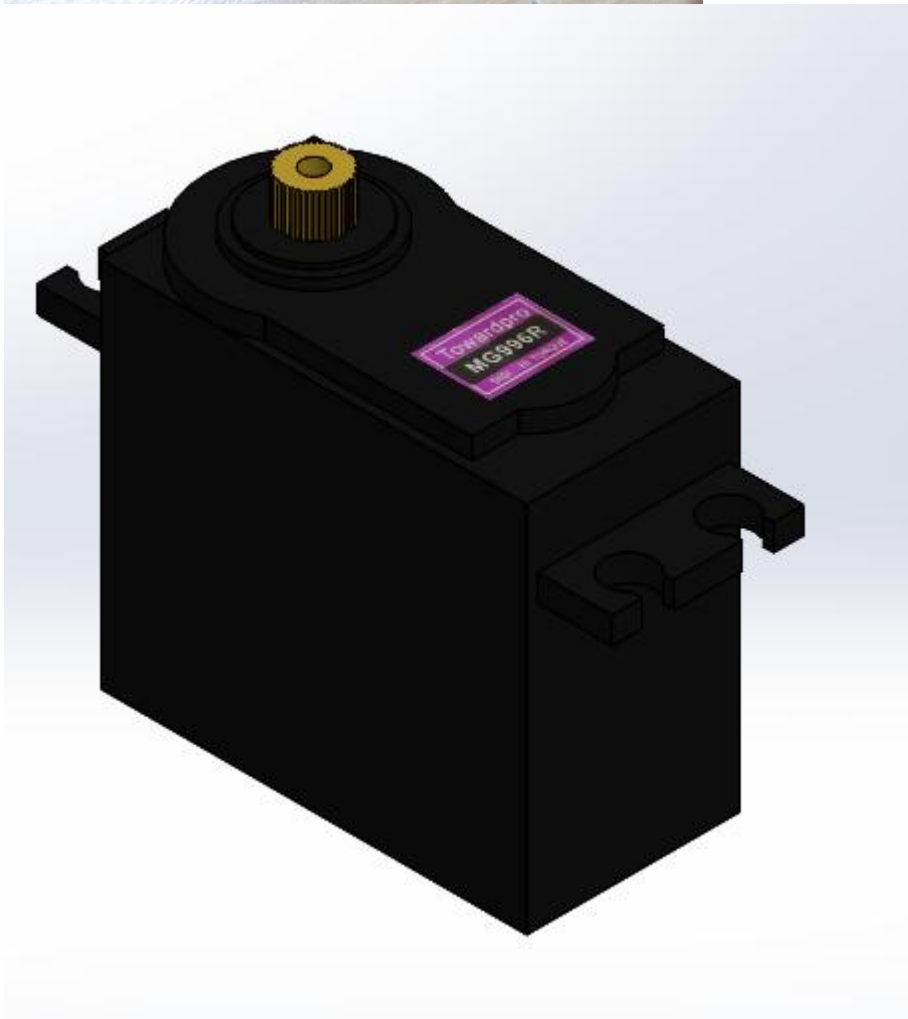




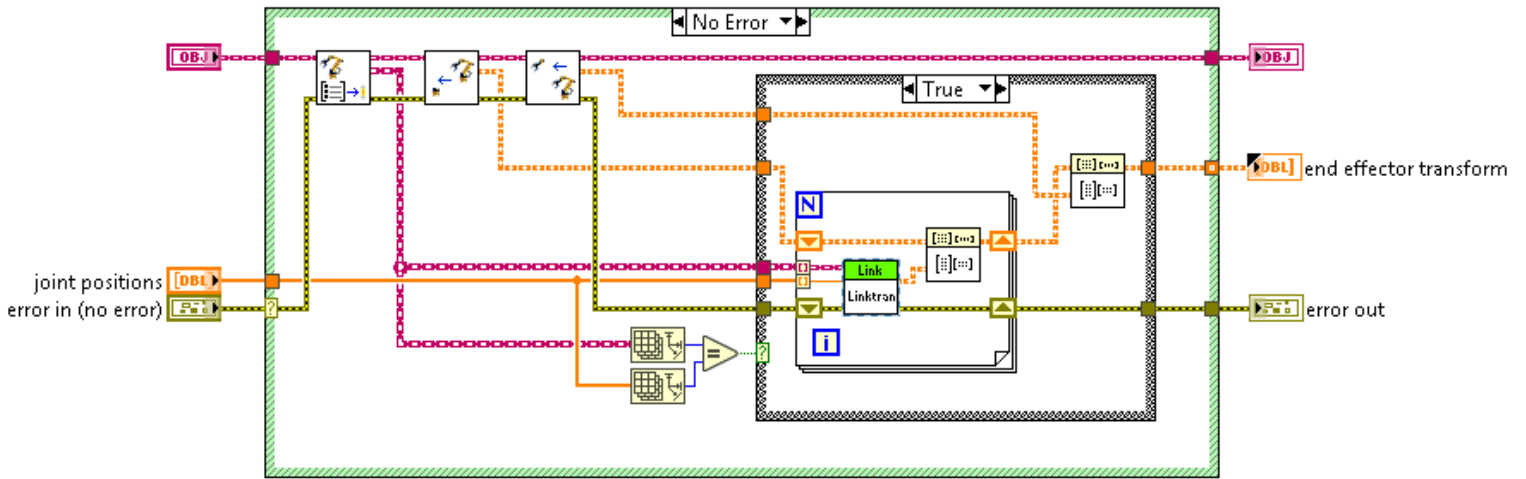




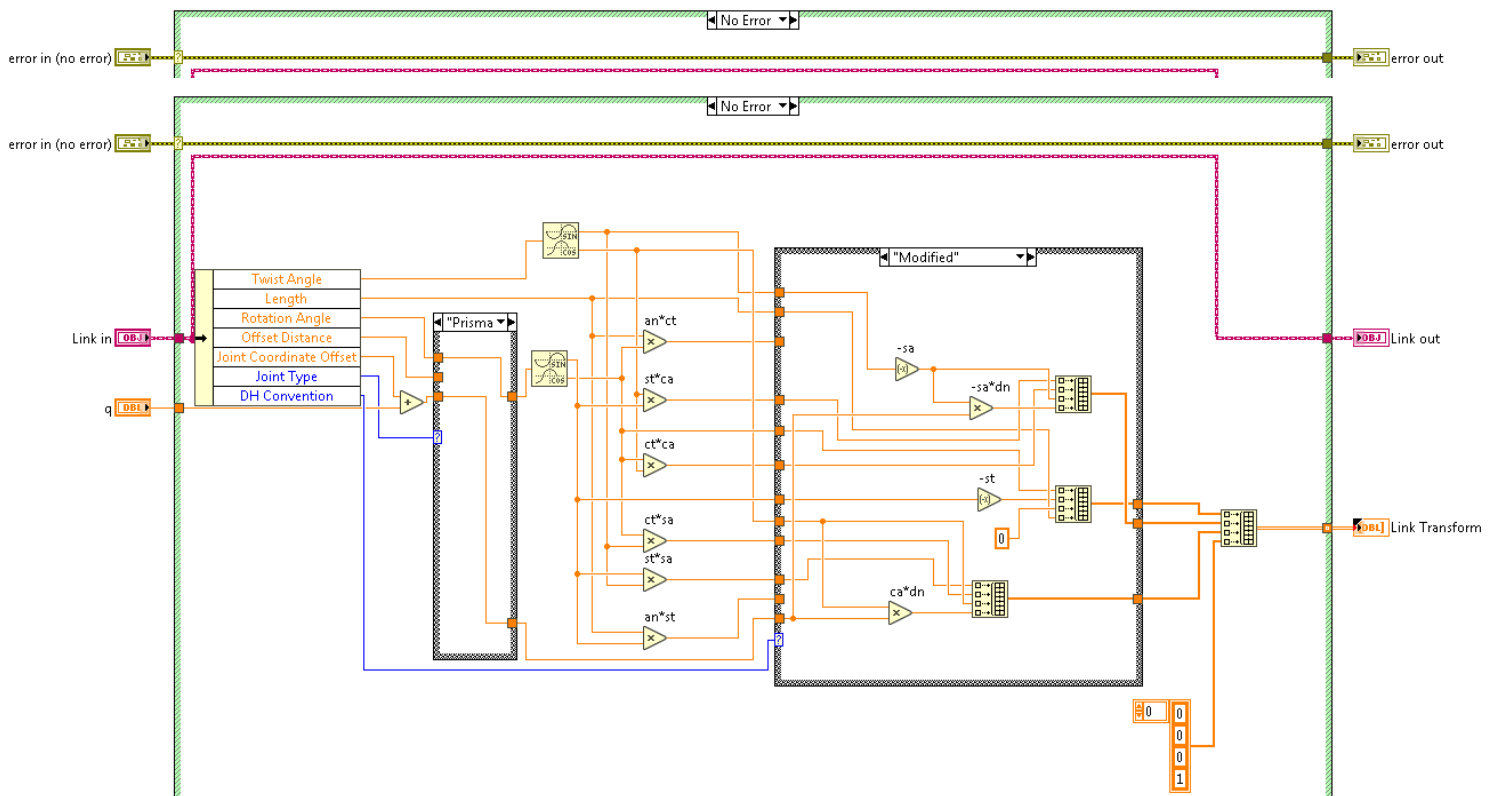




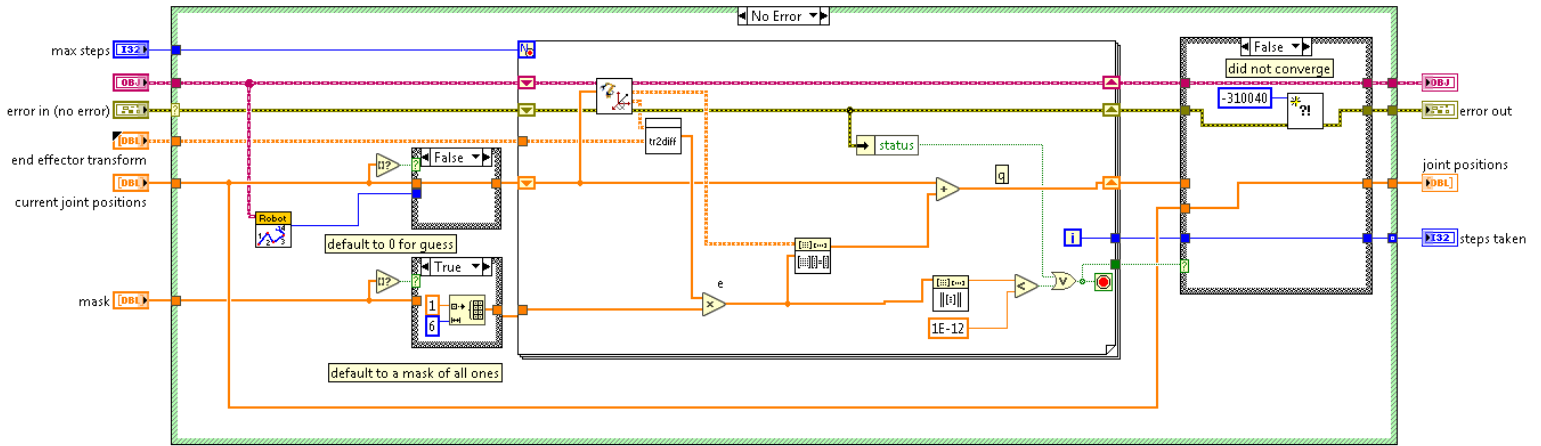
8.2 Προγράμματα κινηματικής



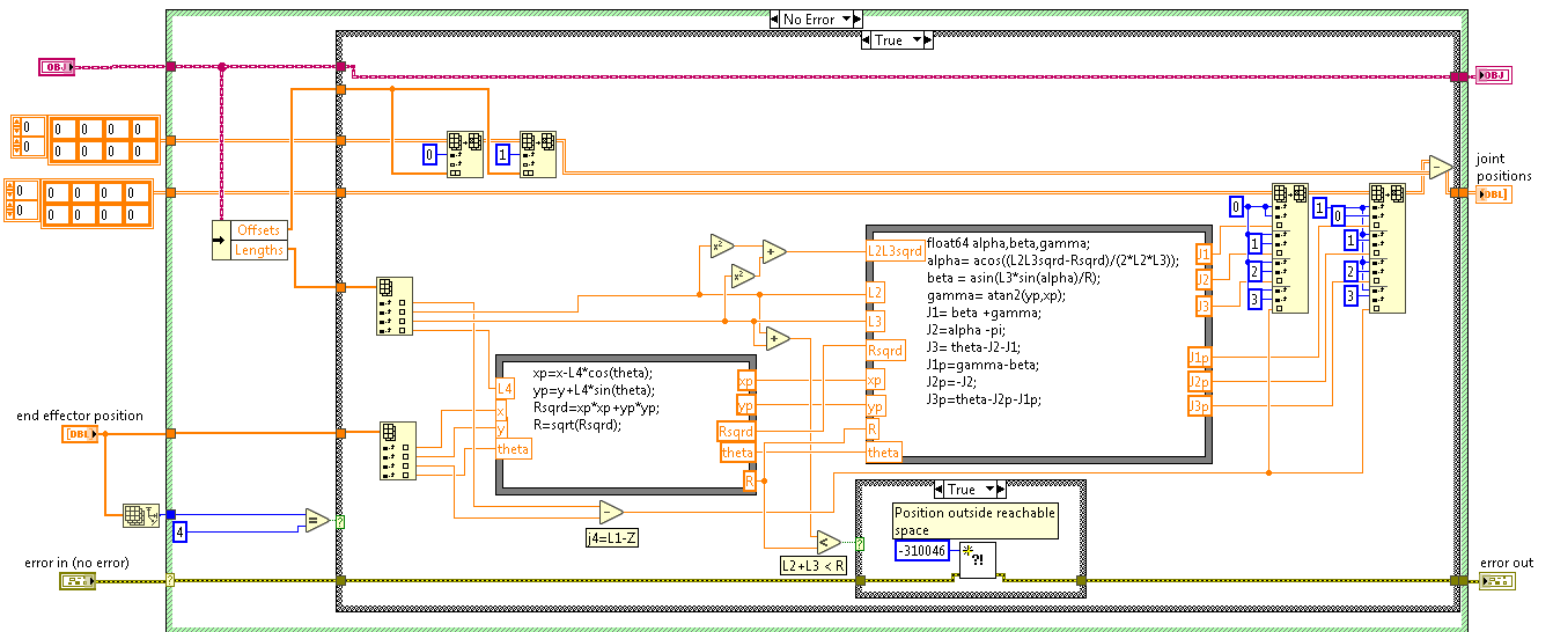
Κυρίως φόρμα ευθείας κινηματικής



Υπορουτίνα ευθείας κινηματικής



Κυρίως φόρμα αντίστροφης κινηματικής



Υπορούτινα αντίστροφης κινηματικής

8.3 Hardware

Η παρακάτω πλακέτα πωλείτε μαζί ή και ξεχωριστά από τον ρομποτικό βραχίονα είναι ένας ελεγκτής με δυνατότητα κινήσεις 32 σερβοκινητήρων. Χρησιμοποιεί δυο ATMEL 168PA micro controllers και τέσσερα 74HC595 για την οδήγηση των κινητήρων.

