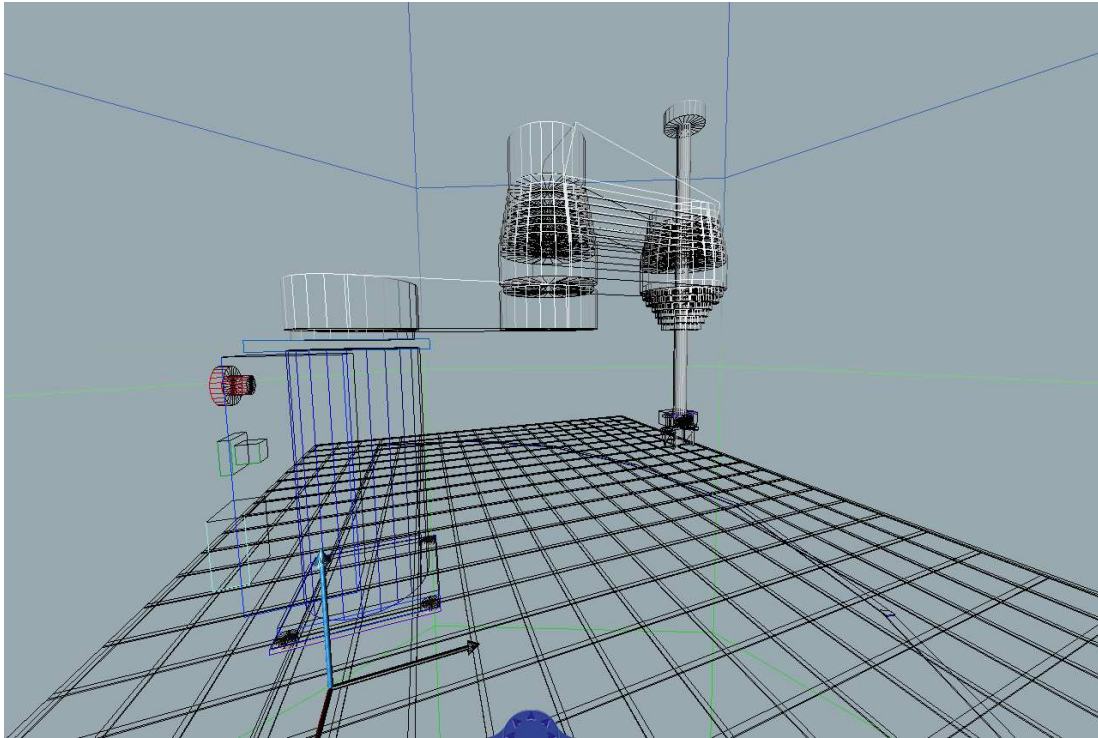




ΤΕΧΝΟΛΟΓΙΚΟ ΕΚΠΑΙΔΕΥΤΙΚΟ ΙΔΡΥΜΑ ΚΡΗΤΗΣ
ΤΜΗΜΑ ΗΛΕΚΤΡΟΝΙΚΗΣ



ΑΝΑΠΤΥΞΗ ΕΚΠΑΙΔΕΥΤΙΚΟΥ ΛΟΓΙΣΜΙΚΟΥ ΓΙΑ ΠΡΟΣΟΜΟΙΩΣΗ ΡΟΜΠΟΤΙΚΟΥ ΒΡΑΧΙΟΝΑ ΤΥΠΟΥ SCARA

Επιβλέπων Καθηγητής:

Δρ. Ε. Δοϊτσίδης

Επίκουρος Καθηγητής

υπό

Παπούλια Σπυρίδωνα και Παπαδάτο Αριστείδη

Χανιά, 2013

ΕΥΧΑΡΙΣΤΙΕΣ

Στο σημείο αυτό θα θέλαμε να ευχαριστήσουμε αρχικά τον καθηγητή μας κ. Δοϊτσιδίδη Ελευθέριο για την επίβλεψη του στην εκπόνηση της πτυχιακής μας εργασίας. Ήταν πάντα πρόθυμος σε όποια απορία και να είχαμε και έδειξε σοβαρότητα και ζήλο στην προσπάθεια μας αυτή. Στην συνέχεια θα θέλαμε να ευχαριστήσουμε τις οικογένειες μας για την στήριξη τους όλα αυτά τα χρόνια στην προσπάθεια μας να ολοκληρώσουμε τις σπουδές μας ώστε να κλείσουμε αυτό το σημαντικό κομμάτι της ζωής μας. Χωρίς αυτούς για μας θα ήταν πάρα πολύ δύσκολο και ευχαριστούμε για την εμπιστοσύνη τους και την στήριξη τους στον αγώνα αυτό. Ακόμη να ευχαριστήσουμε φίλους γνωστούς και λοιπούς συγγενείς που ήταν και αυτοί δίπλα μας όλα αυτά τα χρόνια και ο καθένας έπαιξε διαφορετικό ρόλο το διάστημα αυτό στην προσπάθεια μας. Τέλος πολλά ευχαριστώ αξίζουν στην Ηλιάνα για την υπομονή της και την βοήθεια της στην προσπάθεια μας. Αποτέλεσε σημαντικό κομμάτι της ζωής μου και ως πηγή έμπνευσης έπαιξε σημαντικό ρόλο. Την ευχαριστώ για την στήριξη της και τον τρόπο που μου έδειξε τον σωστό δρόμο σε δύσκολες στιγμές (Σπύρος). Τέλος πολλά ευχαριστώ αξίζουν στην σύζυγο μου Γεωργία για την στήριξη της και αυτής με την σειρά της όλα αυτά τα χρόνια. Ήταν δίπλα μου όλες τις στιγμές ευχάριστες και δυσάρεστες και με βοήθησε πολύ σε πολύ μεγάλο βαθμό στην ζωή μου. Την ευχαριστώ και για τα 2 μας παιδιά που μου χάρισε που για μένα αποτέλεσαν σημαντική πηγή έμπνευσης για την πορεία μου ως τώρα (Άρης)

ΠΕΡΙΛΗΨΗ ΠΤΥΧΙΑΚΗΣ

Η πτυχιακή εργασία αυτή έχει ως αντικείμενο την μοντελοποίηση, ανάπτυξη και έλεγχο ενός εικονικού ρομποτικού βραχίονα τύπου Scara. Πραγματοποιήθηκε μελέτη και κατασκευή του ρομποτικού βραχίονα σε εικονικό περιβάλλον (VRML) και στην συνέχεια κατασκευάστηκε γραφικό περιβάλλον διεπαφής (GUI) για τον χειρισμό του ρομπότ.



ΠΕΡΙΕΧΟΜΕΝΑ

ΚΕΦΑΛΑΙΟ 1	5
Εισαγωγή	5
1.1 Αντικείμενο της εργασίας	5
1.2 Είδη ρομπότ	6
1.3 Βιομηχανικοί ρομποτικοί βραχίονες: βασικές έννοιες και είδη	10
1.4 Χώρος εργασίας	10
1.5 Ωφέλιμο φορτίο – επαναληψιμότητα – ακρίβεια	10
1.6 Ταξινόμηση βραχιόνων βάσει της γεωμετρικής διαμόρφωσής τους.....	11
1.7 Λογισμικά ελέγχου	14
ΚΕΦΑΛΑΙΟ 2	18
Γλώσσα μοντελοποίησης VRML	18
2.1 Εισαγωγή.....	18
2.2 Γενικά για την vrml.....	18
2.3 Εναλλακτικά Εργαλεία μοντελοποίησης τρισδιάστατων αντικειμένων	20
2.4 Σχεδιασμός εικονικών κόσμων με χρήση του v-realm builder.....	21
2.5 Εγκατάσταση και αρχικοποίηση του v-realm builder.....	22
2.6 Εισαγωγή και επεξεργασία αντικειμένων στο v-realm builder	23
2.7 Έλεγχος εικονικών κόσμων μέσω simulink.....	31
ΚΕΦΑΛΑΙΟ 3	36
Σχεδιασμός και Ανάπτυξη του Εικονικού Μοντέλου.....	36
3.1 Εισαγωγή.....	36
3.2 Αλγόριθμος Denavit-Hartenberg (DH)	36
3.3 Κινηματική αναλυση του ρομποτικού βραχίονα τυπου Scara	39
3.4 Σχεδιασμός βάσης	41
3.5 Σχεδιασμός βάσης 2	42
3.6 Σχεδιασμός κουμπιών	42
3.7 Σχεδιασμός Βραχίονα.....	43
3.8 Σχεδιασμός άκρου εργασίας.....	43
3.9 Σχεδιασμός δαπέδου.....	44

ΠΕΡΙΕΧΟΜΕΝΑ

3.10 Σχεδιασμός φόντου	45
3.11 Συνολική εικόνα ρομποτικού βραχίονα τύπου scar	46
3.12 Διασύνδεση του εικονικού μοντελου με simulink	46
3.13 Αναπτυξη γραφικου περιβαλλοντος διεπαφης.....	49
3.14 Γραφικό Περιβάλλον Διεπαφης για τον ελεγχο του ρομποτικου Βραχιονα Τύπου Scara 53	
ΚΕΦΑΛΑΙΟ 4	55
Ενδεικτικά Σενάρια λειτουργίας.....	55
4.1 Σεναρια λειτουργιασ	55
ΒΙΒΛΙΟΓΡΑΦΙΑ	58
ΠΑΡΑΡΤΗΜΑ.....	60

ΚΕΦΑΛΑΙΟ 1

Εισαγωγή

1.1 ΑΝΤΙΚΕΙΜΕΝΟ ΤΗΣ ΕΡΓΑΣΙΑΣ

Στον τομέα των ηλεκτρονικών σήμερα παρατηρούμε μια ραγδαία ανάπτυξη της τεχνολογίας. Αυτό μας δίνει την δυνατότητα να παράγουμε νέα μέσα και εφαρμογές που θα διευκολύνουν την καθημερινότητα μας. Ένας τομέας της ηλεκτρονικής και των αυτοματισμών είναι η ρομποτική.

Ένα **ρομπότ** είναι μια μηχανική συσκευή η οποία μπορεί να υποκαθιστά τον άνθρωπο σε διάφορες εργασίες. Ένα ρομπότ μπορεί να δράσει κάτω από τον απευθείας έλεγχο ενός ανθρώπου ή αυτόνομα κάτω από τον έλεγχο ενός προγραμματισμένου υπολογιστή.

Τα ρομπότ μπορούν να χρησιμοποιηθούν ώστε να κάνουν εργασίες οι οποίες είτε είναι δύσκολες ή επικίνδυνες για να γίνουν απευθείας από έναν άνθρωπο. Σε άλλες περιπτώσεις, χρησιμοποιούνται για να εκτελέσουν εργασίες ταχύτερα ή φθηνότερα απ' ό τι ο άνθρωπος. Έτσι, μπορούν να χρησιμοποιηθούν στην αυτόματη παραγωγή μεγάλων ποσοτήτων κάποιου προϊόντος και με χαμηλότερο κόστος (για παράδειγμα, στις αλυσίδες παραγωγής).

Η λέξη ρομπότ προέρχεται από το σλαβικό *robota* που σημαίνει εργασία. Καθιερώθηκε ως όρος με την σημερινή του έννοια το 1920 από τον Τσέχο θεατρικό συγγραφέα Karel Čapek στο έργο του "R.U.R." (Rossum's Universal Robots), όπου σατιρίζει την εξάρτηση της κοινωνίας από τους μηχανικούς εργάτες (ρομπότ) της τεχνολογικής εξέλιξης και που τελικά εξοντώνουν τους δημιουργούς τους. Σε πολλές σύγχρονες σλαβικές γλώσσες (π.χ. την πολωνική) χρησιμοποιείται σαν έκφραση της καθημερινότητας με την έννοια της σκληρής δουλειάς [1].

Σκοπός της παρούσας πτυχιακής εργασίας είναι:

1. Ο σχεδιασμός σε εικονικό περιβάλλον ενός ρομποτικού βραχίονα τύπου SCARA
2. Η κινηματική μελέτη του
3. Ο σχεδιασμός ενός γραφικού περιβάλλοντος διεπαφής που θα επιτρέπει στον χρήστη να τοποθετεί το ρομπότ σε μια επιλεγμένη θέση

Η ανάπτυξη του μοντέλου έγινε με χρήση της γλώσσας VRML, ενώ το λογισμικό ελέγχου αναπτύχθηκε σε MATLAB.

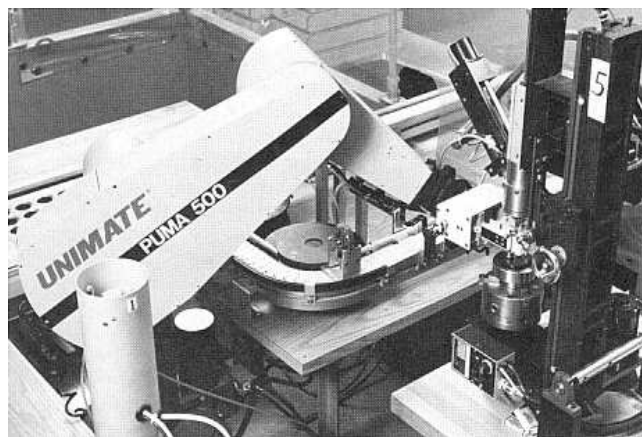
1.2 ΕΙΔΗ ΡΟΜΠΟΤ

Στο εμπόριο είναι διαθέσιμα πολλών και διαφορετικών τύποι ρομπότ για εκπαιδευτική χρήση. Οι διαφορές τους επικεντρώνονται στο τύπο, στους βαθμούς ελευθερίας, στα υλικά κατασκευής, στο μέγεθος και στις δυνατότητες ελέγχου τους.

Κατά την πολυετή εξέλιξη της επιστήμης της ρομποτικής προέκυψαν διάφορα είδη ρομποτικών μηχανισμών, οι οποίοι διαφέρουν σημαντικά στη μορφή, αποτελούνται όμως από αντίστοιχα επιμέρους υποσυστήματα. Αυτά είναι το μηχανολογικό υποσύστημα, το υποσύστημα αίσθησης και το σύστημα ελέγχου.

Τα βασικότερα είδη ρομπότ είναι τα παρακάτω:

Ρομπότ Σταθερής Βάσης: Τα ρομπότ αυτά αποτελούνται από διαδοχικά στερεά σώματα (σύνδεσμοι) που συνδέονται μέσω αρθρώσεων σχηματίζοντας μία κινηματική αλυσίδα. Η αλυσίδα αυτή έχει το ένα άκρο της (βάση) σταθερά συνδεδεμένο με κάποιο σημείο του περιβάλλοντος χώρου. Η μορφή αυτή ρομπότ είναι η παραδοσιακή μορφή ενός βιομηχανικού ρομποτικού βραχίονα, και περιλαμβάνει το βραχίονα, τον καρπό και το εργαλείο (Σχήμα 1.1)



Σχήμα 1.1 Ο Βιομηχανικός Ρομποτικός Βραχίονας PUMA 560 της Unimation Inc.

Κινούμενα Ρομπότ: Ως κινητά ρομπότ χαρακτηρίζονται όλα εκείνα τα ρομπότ που έχουν τη δυνατότητα να μετακινήσουν όλα τα σημεία του μηχανισμού τους. Η δυνατότητα αυτή προσφέρεται από ειδικά συστήματα προώθησης, τα οποία μπορεί να είναι είτε απλά (όπως τροχοί) είτε πολύπλοκα (όπως jet, προπέλες, μηχανικά πόδια). Τα κινούμενα ρομπότ διακρίνονται σε επιμέρους κατηγορίες ανάλογα με το βαθμό

αυτονομίας τους. Έτσι έχουμε: • AGVs: τα AGVs (Automatic Guided Vehicles) έχουν περιορισμένη αυτονομία

κίνησης, δεδομένου ότι η τροχιά τους είναι προκαθορισμένη μέσω καλωδίων στο έδαφος ή πομπών στον περιβάλλοντα χώρο (Σχήμα 1.2).



Σχήμα 1.2 AGV σε Βιομηχανικό Περιβάλλον

Αυτόνομα Έντροχα Ρομπότ: Τα ρομπότ αυτά λειτουργούν με υψηλό βαθμό αυτονομίας. Πιο συγκεκριμένα μπορούν και λειτουργούν χωρίς συνεχή εξωτερική επίβλεψη και είναι ικανά να εκτελούν εργασίες αυτόνομα δεχόμενα μόνο ορισμένες υψηλού επιπέδου εντολές (Σχήμα 1.3).



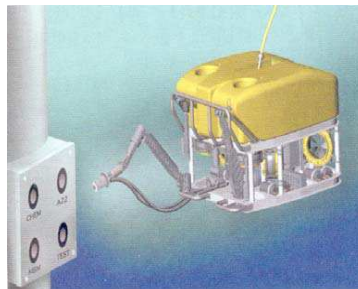
Σχήμα 1.3 Αυτόνομο Έντροχο Ρομπότ

Βαδίζοντα Ρομπότ: Τα ρομπότ αυτά χρησιμοποιούν μηχανικά πόδια για την κίνησή τους και όχι συμβατικούς τροχούς όπως στις προηγούμενες δύο κατηγορίες. Τα κυριότερα πλεονεκτήματα της συγκεκριμένης υλοποίησης είναι η μεγάλη δυνατότητα αποφυγής εμποδίων και η ικανότητα αναρρίχησης σε ανώμαλα εδάφη και μη επίπεδες επιφάνειες. Από τα πιο συνηθισμένα ρομπότ αυτής της κατηγορίας είναι τα δίποδα ενώ δεν αποκλείονται και εφαρμογές με περισσότερα από δύο πόδια, π.χ. ρομπότ που μοιάζουν και κινούνται όπως οι αράχνες (Σχήμα 1.4)



Σχήμα 1.4 Ο Dante II του Εργαστηρίου JPL της NASA κατά τη διάρκεια ανάβασης σε βουνό της Αλάσκα

ROVs: τα ROVs (Remotely Operated Vehicles) ανήκουν στην κατηγορία των μη επανδρωμένων υποβρύχιων ρομπότ. Όπως υποδηλώνει το όνομά τους δεν έχουν μεγάλο βαθμό αυτονομίας, μιας και είναι συνδεδεμένα με το μητρικό πλοίο μέσω καλωδίου, το οποίο και καλύπτει τις ανάγκες του ρομπότ σε ενέργεια και επικοινωνίες. Τα ρομπότ αυτού του τύπου έχουν σχήμα κουτιού και κινούνται γενικά σε χαμηλές ταχύτητες (Σχήμα 1.5).



Σχήμα 1.5 Απόδοση ενός ROV

AUVs: τα AUVs (Autonomous Underwater Vehicles), αντίθετα με τα ROVs, είναι πλήρως αυτόνομα και κατά συνέπεια δεν έχουν την ανάγκη καλωδίου. Για τις ανάγκες τροφοδοσίας (ενέργεια) χρησιμοποιούνται ειδικές μπαταρίες, κάτι όμως που θέτει και περιορισμούς στη λειτουργία των ρομπότ αυτών. Τα AUVs έχουν σχήμα τορπιλών και μπορούν να κινούνται με αρκετά μεγάλες ταχύτητες (Σχήμα 1.6).



Σχήμα 1.6 Καλλιτεχνική αναπαράσταση ενός AUV

Εναέρια ρομπότ: Πρόκειται για μη επανδρωμένα ιπτάμενα ρομπότ, όπως ελικόπτερα και αεροπλάνα. Τα ρομπότ αυτά έχουν διαρκώς αυξανόμενες εφαρμογές, όμως εξαιτίας της μειωμένης ακόμα σταθερότητας και ασφάλειας στη συμπεριφορά τους χρησιμοποιούνται για στρατιωτικούς κυρίως σκοπούς (Σχήματα 1.7 και 1.8).



Σχήμα 1.7 Ρομποτικό Ελικόπτερο του USC



Σχήμα 1.8 Το ρομπότ Helios της NASA

1.3 ΒΙΟΜΗΧΑΝΙΚΟΙ ΡΟΜΠΟΤΙΚΟΙ ΒΡΑΧΙΟΝΕΣ: ΒΑΣΙΚΕΣ ΕΝΝΟΙΕΣ ΚΑΙ ΕΙΔΗ

Ένας ρομποτικός βραχίονας αποτελείται από μία σειρά διαδοχικών στερεών σωμάτων που ονομάζονται σύνδεσμοι. Οι σύνδεσμοι συνδέονται ανά δύο μεταξύ τους μέσω αρθρώσεων σχηματίζοντας μία κινηματική αλυσίδα. Οι αρθρώσεις μπορεί να είναι :

Πρισματικές: Η κίνηση που πραγματοποιείται είναι σχετική μεταφορική κίνηση μεταξύ δύο διαδοχικών συνδέσμων.

Περιστροφικές: Η κίνηση που πραγματοποιείται είναι σχετική περιστροφική κίνηση μεταξύ δύο διαδοχικών συνδέσμων

Σφαιρικές: Η κίνηση που πραγματοποιείται είναι σφαιρική περιστροφική κίνηση μεταξύ δύο διαδοχικών συνδέσμων. και παρέχουν στην κατασκευή από έναν βαθμό ελευθερίας. Μία κινηματική αλυσίδα χαρακτηρίζεται ως ανοικτή όταν υπάρχει μία μόνο διαδοχή συνδέσμων που να συνδέει τα δύο άκρα του βραχίονα και κλειστή όταν οι σύνδεσμοι που τη συνιστούν σχηματίζουν βρόχο.

1.4 ΧΩΡΟΣ ΕΡΓΑΣΙΑΣ

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, κάτι που θα γίνει φανερό και στη συνέχεια.

1.5 ΩΦΕΛΙΜΟ ΦΟΡΤΙΟ – ΕΠΑΝΑΛΗΨΙΜΟΤΗΤΑ – ΑΚΡΙΒΕΙΑ

Από τα πιο σημαντικά μεγέθη ενός βιομηχανικού βραχίονα είναι το ωφέλιμο φορτίο, η επαναληψιμότητα και η ακρίβεια. Πιο συγκεκριμένα τα παραπάνω μεγέθη αναφέρονται στα εξής:

Ωφέλιμο Φορτίο: είναι το βάρος που μπορεί να μεταφέρει το άκρο του βραχίονα. Ως σημείο εφαρμογής του βάρους θεωρείται η φλάντζα του καρπού. Το προδιαγραφόμενο αυτό φορτίο δεν είναι σταθερό και εξαρτάται από την ταχύτητα με την οποία πρόκειται να κινηθεί ο καρπός.

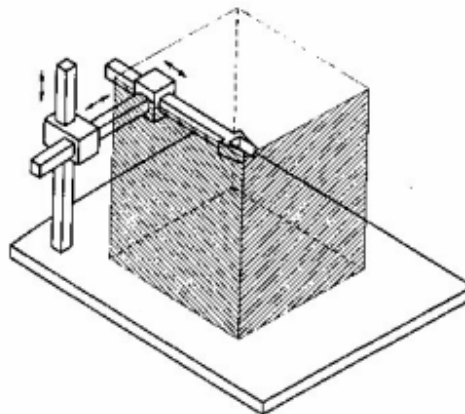
Επαναληψιμότητα: εκφράζει τη δυνατότητα του βραχίονα να γυρίσει στο ίδιο σημείο μετά από αρκετές επαναλήψεις και δίνεται ως εύρος μέσα στο οποίο ο βραχίονας θα τερματίσει την κίνηση. Η απόκλιση οφείλεται στο ότι κατά τη λειτουργία του το ρομπότ είναι δυνατό να χάσει λίγο από τη μέτρηση της θέσης με αποτέλεσμα να μη μπορεί να επιστρέψει στη συγκεκριμένη θέση μετά από ορισμένους κύκλους λειτουργίας. Δεδομένου 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.

1.6 ΤΑΞΙΝΟΜΗΣΗ ΒΡΑΧΙΟΝΩΝ ΒΑΣΕΙ ΤΗΣ ΓΕΩΜΕΤΡΙΚΗΣ ΔΙΑΜΟΡΦΩΣΗΣ ΤΟΥΣ

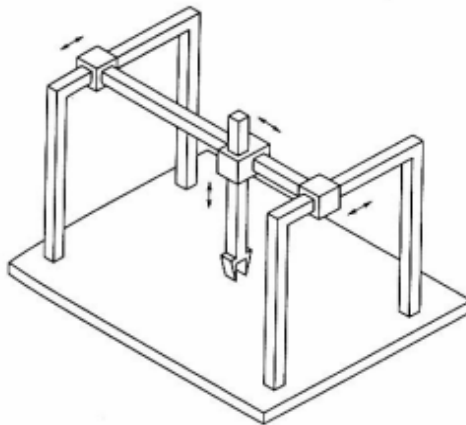
Ο τύπος και η διαδοχή των αρθρώσεων ενός βραχίονα επιτρέπει την ταξινόμησή των ρομποτικών βραχίωνων σε διάφορες κατηγορίες, οι οποίες αναφέρονται παρακάτω. Θα έχουμε λοιπόν τα εξής:

Καρτεσιανοί Βραχίονες (Σχήμα 1.8): Η καρτεσιανή γεωμετρία υλοποιείται με τρεις διαδοχικές πρισματικές αρθρώσεις. Οι άξονες των αρθρώσεων αυτών είναι ανά δύο κάθετοι μεταξύ τους. Η καρτεσιανή δομή παρέχει μεγάλη δυσκαμψία και σταθερή ακρίβεια σε ολόκληρο το χώρο εργασίας που είναι ένα παραλληλεπίπεδο. Βασικό μειονέκτημα της κατασκευής είναι η μειωμένη επιδεξιότητα κίνησης, λόγω της πρισματικής φύσης των αρθρώσεων.



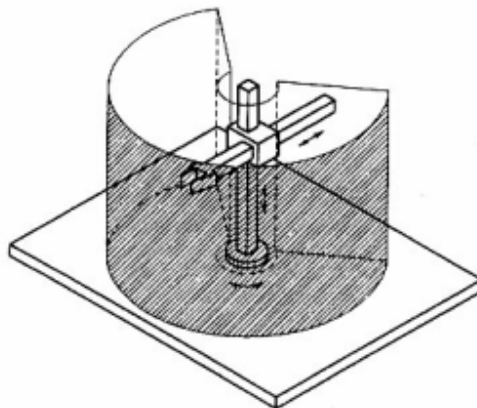
Σχήμα 1.8 Καρτεσιανός βραχίονας

Βραχίονες Gantry (Σχήμα 1.9): Οι βραχίονες τύπου Gantry είναι καρτεσιανοί, διαφέρουν όμως από τους τελευταίους στον τρόπο προσέγγισης τους αντικείμενου ενδιαφέροντος. Ειδικότερα ο βραχίονας Gantry προσεγγίζει το αντικείμενο από πάνω, τη στιγμή που ένας κλασικός καρτεσιανός βραχίονας προσεγγίζει το αντικείμενο από το πλάι. Άμεσες συνέπειες της διαφοροποίησης αυτής είναι η αύξηση του χώρου εργασίας και της δυσκαμψίας, καθώς επίσης και η δυνατότητα χειρισμού μεγάλων και βαριών αντικειμένων.



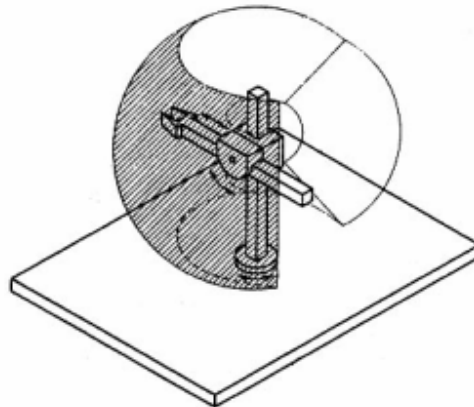
Σχήμα 1.9 Βραχίονας GANTRY

Κυλινδρικοί Βραχίονες (Σχήμα 1.10): Στους κυλινδρικούς βραχίονες η πρώτη πρισματική άρθρωση της καρτεσιανής δομής έχει αντικατασταθεί από μία περιστροφική άρθρωση. Οι συγκεκριμένοι βραχίονες χαρακτηρίζονται από καλή δυσκαμψία, όμως η ακρίβεια της θέσης του καρπού μειώνεται καθώς η οριζόντια μετατόπιση αυξάνεται. Ο χώρος εργασίας στην περίπτωση αυτή είναι τμήμα κυλίνδρου. Σημαντικό μειονέκτημα της συγκεκριμένης γεωμετρίας είναι το ότι ο βραχίονας εισέρχεται στο χώρο εργασίας και τον περιορίζει.



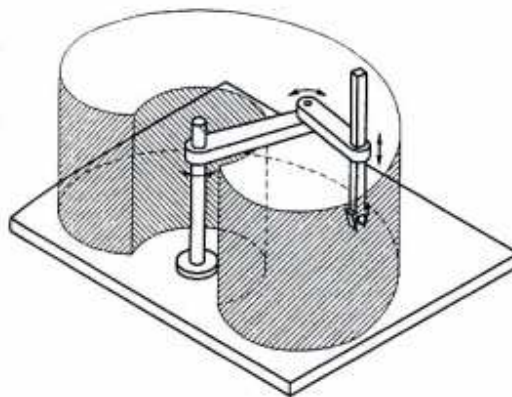
Σχήμα 1.10 Κυλινδρικός βραχίονας

Σφαιρικοί Βραχίονες (Σχήμα 1.11): Στους βραχίονες αυτούς αντικαθίσταται πλέον και η δεύτερη πρισματική άρθρωση της καρτεσιανής δομής με περιστροφική. Η μηχανολογική πολυπλοκότητα αυξάνει, ενώ η δυσκαμψία μειώνεται. Επιπλέον η ακρίβεια του καρπού μειώνεται με την αύξηση της ακτινικής απόστασης. Ο χώρος εργασίας είναι τμήμα σφαίρας και περιέχει ένα μέρος της βάσης με άμεση συνέπεια τη δυνατότητα χειρισμού αντικειμένων που βρίσκονται στο έδαφος.



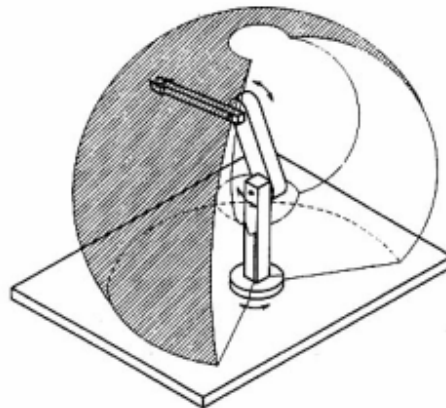
Σχήμα 1.11 Σφαιρικός βραχίονας

Βραχίονες SCARA (Σχήμα 1.12): Η γεωμετρία SCARA είναι ειδική και περιλαμβάνει δύο περιστροφικές και μία πρισματική άρθρωση τοποθετημένες κατά τέτοιο τρόπο ώστε οι άξονες κίνησης να είναι παράλληλοι μεταξύ τους. Το όνομα SCARA προέρχεται από τα αρχικά των λέξεων Selective Compliance Assembly Robot Arm. Η συγκεκριμένη γεωμετρία παρέχει μεγάλη δυσκαμψία σε κατακόρυφη φόρτιση και ελαστικότητα σε οριζόντια. Η ακρίβεια τοποθέτησης του καρπού μειώνεται με την αύξηση της απόστασης του από τον άξονα της πρώτης άρθρωσης.



Σχήμα 1.12 Βραχίονας τύπου Scara

Ανθρωπομορφικοί Βραχίονες (Σχήμα 1.13): Η ανθρωπομορφική γεωμετρία υλοποιείται με τρεις διαδοχικές περιστροφικές αρθρώσεις. Ειδικότερα, ο άξονας περιστροφής της πρώτης άρθρωσης είναι κατακόρυφος και κάθετος στους άξονες περιστροφής των επομένων δύο αρθρώσεων, οι οποίοι είναι παράλληλοι μεταξύ τους. Η συγκεκριμένη δομή παρέχει τη μεγαλύτερη επιδεξιότητα από όλες τις προηγούμενες, καθώς όλες οι αρθρώσεις είναι περιστροφικές. Ωστόσο η ακρίβεια του καρπού δεν είναι σταθερή εντός του χώρου εργασίας που έχει τη μορφή σφαίρας [2].



Σχήμα 1.13 Ανθρωπομορφικοί Βραχίονες

1.7 ΛΟΓΙΣΜΙΚΑ ΕΛΕΓΧΟΥ

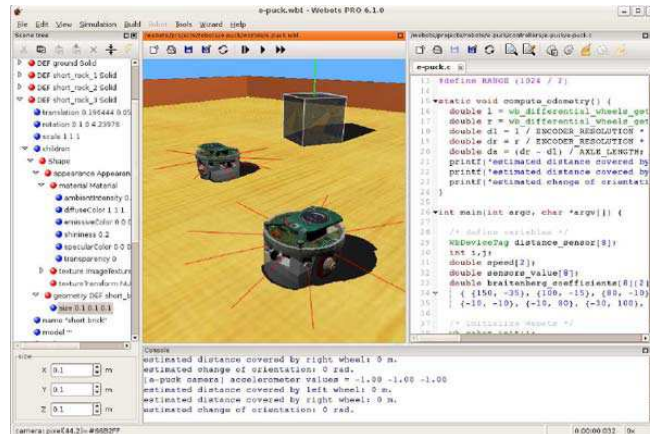
Για να γίνουν όλα αυτά κατανοητά όμως και να λειτουργήσουν και σε περιβάλλον προσομοίωσης αλλά και σε πραγματικό περιβάλλον υπάρχει το λογισμικό (software) δηλαδή προγράμματα οδήγησης που θα μας επιτρέψουν να λειτουργήσουμε ένα ρομπότ και να το χειριστούμε εύκολα και σε προσομοίωση αλλά και σε πραγματικό περιβάλλον.

Εκτός από το κατάλληλο Hardware για την λειτουργία ρομποτικών συστημάτων είναι αναγκαία η ύπαρξη κατάλληλου λογισμικού. Στην αγορά υπάρχουν διαθέσιμες πολλές λύσεις που μπορούν να χρησιμοποιηθούν τόσο για εκπαιδευτικούς όσο για και ερευνητικούς σκοπούς. Ενδεικτικά αναφέρονται τα παρακάτω:

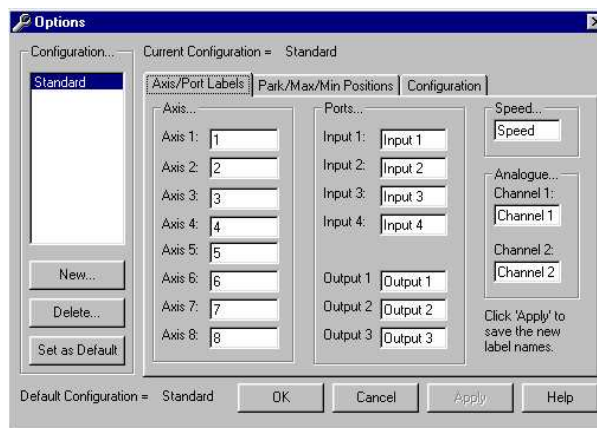
To Webots [3], είναι ένα πακέτο λογισμικού που επιτρέπει την προσομοίωση ρομποτικών συστημάτων. Προσφέρει ένα γρήγορο περιβάλλον διαμόρφωσης πρωτοτύπων, που επιτρέπει στο χρήστη να δημιουργεί τρισδιάστατους εικονικούς κόσμους με ιδιότητες φυσικής, όπως η μάζα, οι ενώσεις, ο συντελεστής τριβής, κλπ. Ο χρήστης μπορεί να προσθέσει ενεργά αντικείμενα και συγκεκριμένα ρομποτικές συσκευές διαφόρων τύπων. Αυτές, μπορούν να εξοπλιστούν με διαφορετικές συσκευές αισθητήρων και επενεργητών, όπως αισθητήρες απόστασης, ρόδες, κάμερες, servos, αισθητήρες αφής, κλπ. Ο χρήστης μπορεί να προγραμματίσει ρομποτικές συσκευές σε περιβάλλον προσομοίωσης και στη συνέχεια να μεταφέρει το λογισμικό στις πραγματικές συσκευές όπως το Khepera, Hemisson, LEGO Mindstorms, Aibo, κλπ.

To Robotica [4], είναι ένα λογισμικό πακέτο, για έλεγχο ρομποτικών βραχιόνων. Χρησιμοποιεί μια σειρά από λειτουργίες και ορισμούς, του προγράμματος μαθηματικών Mathematicasymbolic. Το Robotica, μπορεί να λειτουργήσει σε περιβάλλον Microsoft Windows και συνεργάζεται με το πρόγραμμα Mathematica. Το κύριο χαρακτηριστικό του Robotica, είναι η δυνατότητα να υπολογίζει, συμβολικά ή αριθμητικά, τις κινηματικές και δυναμικές εξισώσεις, ενός ρομποτικού συστήματος, χρησιμοποιώντας

την μέθοδο Denevit-Hartenburg. Στο Σχήμα 1.15, παρουσιάζονται μερικές ενδεικτικές εικόνες, από το γραφικό περιβάλλον του Robotica [5].

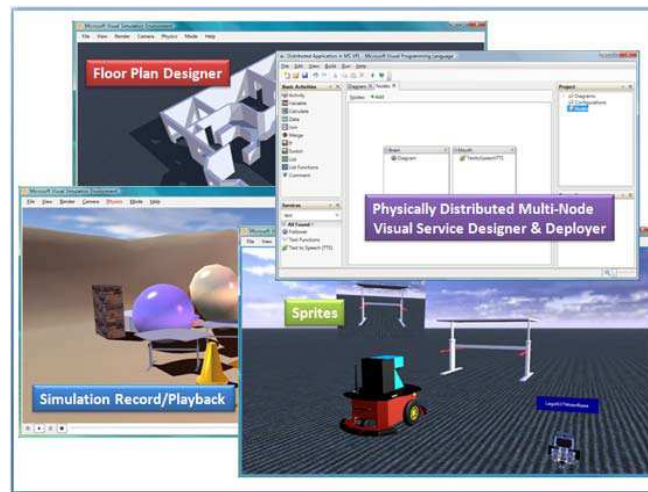


Σχήμα 1.14 Γραφικό περιβάλλον του λογισμικού ελέγχου ρομπότ Webots

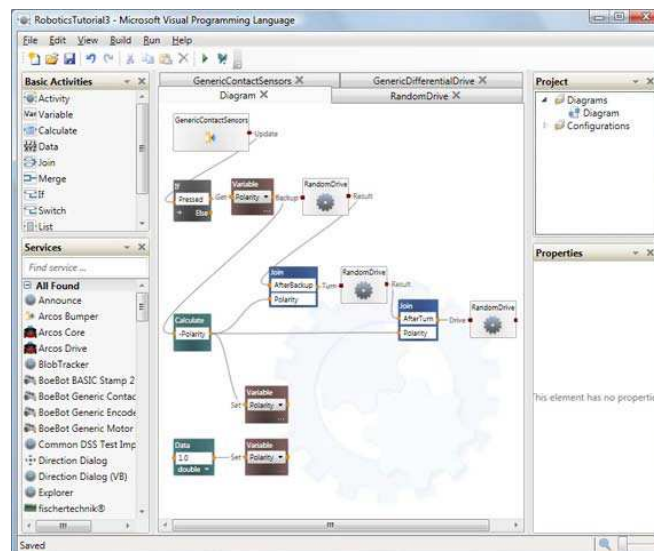


Σχήμα 1.15.α Γραφικό περιβάλλον του λογισμικού ελέγχου βραχίονα Robotica

Το **Microsoft Robotics Developer Studio** [6], είναι ένα λογισμικό πακέτο, για έλεγχο και προσομοίωση ρομπότ. Το πακέτο αυτό είναι βασισμένο στο περιβάλλον των Microsoft Windows και σχεδιάστηκε έτσι ώστε να επιτρέπει σε οποιονδήποτε χρήστη να δημιουργεί εφαρμογές για των έλεγχο ρομποτικών συστημάτων. Χρησιμοποιεί, ένα γραφικό εργαλείο προγραμματισμού (visual programming tool) και καθιστά εύκολη την πρόσβαση σε αισθητήρες και μηχανισμούς κίνησης. Υποστηρίζει ένα μεγάλο αριθμό από γλώσσες όπως C#, Visual Basic, .NET, Jscript και IronPython. Ακόμα, εμπεριέχει κάποια παραδείγματα όπως το “Soccer Simulation” και το “Sumo Competition” της Microsoft.

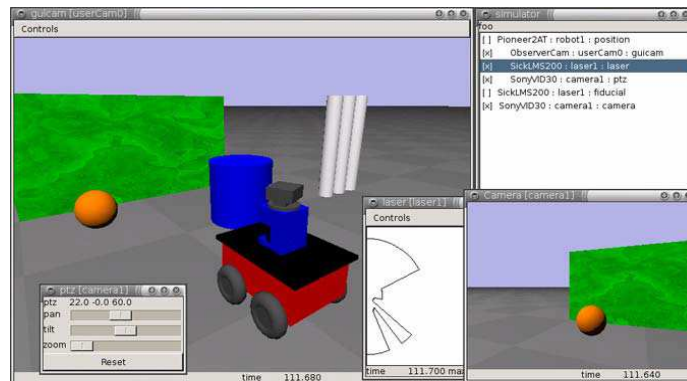


Σχήμα 1.16 Λογισμικό πακέτο ελέγχου ρομπότ Microsoft Robotics Developer Studio



Σχήμα 1.17 Γραφικό περιβάλλον του λογισμικού πακέτου ελέγχου και προσομοίωσης ρομπότ Microsoft Robotics Developer Studio

Το **“Player Project”** [7], είχε σαν σκοπό να αποτελέσει μια πλατφόρμα ανάπτυξης λογισμικού για ρομποτικές συσκευές και αισθητήρες. Το **“Stage”** του Player Project (Player/Stage), είναι ένας προσομοιωτής πολλαπλών ρομποτικών συσκευών (multiple robot simulator). Προσομοιώνει ομάδες ρομποτικών συσκευών που κινούνται και αισθάνονται σε ένα 3D χαρτογραφημένο περιβάλλον. Υπάρχουν διαθέσιμα ποικίλα μοντέλα αισθητήρων, όπως σόναρ (sonar), σαρωτές απόστασης λέιζερ (scanning laser rangefinder), κάμερα και οδόμετρο.



Σχήμα 2.8 Γραφικό περιβάλλον του λογισμικού εφαρμογών Project (Player/Stage)

ΚΕΦΑΛΑΙΟ 2

Γλώσσα μοντελοποίησης VRML

2.1 ΕΙΣΑΓΩΓΗ

Στο κεφάλαιο αυτό θα περιγραφεί αναλυτικά ο τρόπος ανάπτυξης εικονικών κόσμων με την βοήθεια της VRML (Virtual Reality Modeling Language - Γλώσσα Μοντελοποίησης Εικονικής Πραγματικότητας), τα βασικά της χαρακτηριστικά, καθώς και ο τρόπος αλληλεπίδρασης των εικονικών κόσμων με το λογισμικό MATLAB.

2.2 ΓΕΝΙΚΑ ΓΙΑ ΤΗΝ VRML

Η γλώσσα VRML, είναι ένα πρότυπο δημιουργίας τρισδιάστατων γραφικών για το παγκόσμιο ιστό (World Wide Web). Πρόκειται για τον ορισμό ενός συγκεκριμένου τύπου αρχείου (file format), που περιέχει μια ASCII περιγραφή των τρισδιάστατων σκηνών. Η VRML αποτέλεσε την πρώτη ουσιαστική προσπάθεια για τη δημιουργία τρισδιάστατης (3D) γλώσσας, για χρήση, κυρίως στο διαδίκτυο[8].

Με τη χρήση της VRML, ο προγραμματιστής είναι σε θέση να υλοποιεί μια ακολουθία εικόνων και να αλληλεπιδρά με αυτές. Για παράδειγμα, μπορεί κανείς να παρατηρεί ένα δωμάτιο ενός κτιρίου και χρησιμοποιώντας κατάλληλα χειριστήρια να μετακινείται στο δωμάτιο σαν να επρόκειτο για πραγματική περιπλάνησή του μέσα στο χώρο. Για την ορθή πλοήγηση μέσα σε έναν τέτοιο τρισδιάστατο κόσμο και για την πρακτική ανάγνωση ενός VRML αρχείου από τον υπολογιστή, είναι απαραίτητη η χρήση ενός φυλλομετρητή (browser) για εφαρμογές της VRML[9].

Οι εφαρμογές της VRML είναι πολλές: επιχειρήσεις, διαφημίσεις στον παγκόσμιο ιστό (e-commerce), ψυχαγωγία, εκπαίδευση, επικοινωνία, αρχιτεκτονική κλπ. Μπορεί να χρησιμοποιηθεί για τη δημιουργία τρισδιάστατων αναπαραστάσεων πολύπλοκων σκηνών, όπως εικονογραφήσεις, ορισμοί προϊόντων και παρουσιάσεις εικονικής πραγματικότητας. Η VRML αποτελεί ένα ανεξάρτητο πρότυπο, που μπορεί να χρησιμοποιηθεί στο διαδίκτυο ανεξάρτητα από την υπολογιστική πλατφόρμα, καθώς και για τον ορισμό τρισδιάστατων αντικειμένων και την παραμετροποίηση των χαρακτηριστικών τους (σχήμα, χρώμα, μέγεθος, κλπ)[10].

Η πρώτη έκδοση της VRML παρουσιάστηκε το Μάιο του 1995 και βασίστηκε πάνω σε ένα υποσύνολο του 3D μοντέλου της Silicon Graphics, με την ονομασία Ανοικτός Εφευρέτης (Open Inventor). Τον Ιανουάριο του 1996 παρουσιάστηκε η βελτιωμένη έκδοση VRML 1.0c. Το 1997 η έκδοση 2.0 της VRML προτυποποιήθηκε κατά ISO (ISO/IEC 14772-1:1997) και με ελάχιστες διαφορές από την έκδοση 2.0 ονομάστηκε VRML 97. Μεταξύ της πρώτης και της τελευταίας έκδοσης μεσολάβησαν τουλάχιστον 47 πρόχειρες εκδόσεις (drafts). Η VRML97 είναι ασύμβατη με την VRML 1 και έχει τις εξής βελτιώσεις:

- αυξημένες δυνατότητες διαδραστικότητας και αλληλεπίδρασης με τα μέρη του εικονικού κόσμου,
- υποστήριξη JAVA και Javascript,
- εντολές για ήχο,
- κίνηση και υποστήριξη video.

Το 1999 παρουσιάστηκε το νέο πρότυπο X3D (ISO/IEC 19775-1) που αποτελεί υπερέσυνολο της VRML και ενδεικτικά περιλαμβάνει:

- χρήση XML,
- περισσότερες εντολές, άρα και περισσότερες δυνατότητες,
- πιο αυστηρή δομή και κανονικοποίηση,
- δυνατότητα παραγωγής κώδικα σε δυαδική μορφή και ταυτόχρονη συμπίεση (αφού η VRML είναι σε μορφή κειμένου).

Η VRML ποτέ δεν εξαπλώθηκε όσο αναμενόταν διότι:

- όταν εμφανίστηκε απαιτούσε αρκετή επεξεργαστική ισχύ από τις κάρτες γραφικών (πλέον οι απλές σύγχρονες κάρτες γραφικών είναι αρκετά ισχυρές για να αποδώσουν πολύ ικανοποιητικά έναν σύνθετο εικονικό κόσμο)
- ήταν αρκετά πρωτοποριακή ιδέα για το γενικότερο επίπεδο χρηστών του διαδικτύου οι οποίοι, την περίοδο που παρουσιάστηκε η VRML, ζητούσαν την εύκολη και γρήγορη εύρεση πληροφοριών και την απλότητα στην επικοινωνία
- τα εμπορικά προϊόντα πλοήγησης σε εικονικούς κόσμους και απεικόνισης 3D γραφικών δεν βρήκαν απήχηση στο αγοραστικό κοινό με αποτέλεσμα οι εταιρίες να στραφούν σε άλλες μορφές προϊόντων λογισμικού και υλικού.

Ωστόσο, η VRML είναι μια γλώσσα με την οποία μπορούν πολύ εύκολα να δημιουργηθούν απλά τρισδιάστατα αντικείμενα (κύβος, κώνος, κύλινδρος και σφαίρα), να καλυφθούν με χρωματιστές υφές (textures) και να ενσωματωθούν σε άλλα

αντικείμενα. Το ίδιο απλή παραμένει και η μεταφορά, η περιστροφή, η μεγέθυνση-σμίκρυνση, ο χρωματισμός και η επικάλυψη των αντικειμένων με υφές. [11]

Τα αρχεία της VRML ονομάζονται εικονικοί κόσμοι και έχουν κατάληξη ".wrl" (ή ακόμα ".wrlz" για να δηλώνεται ότι είναι συμπιεσμένα). Τα αρχεία αυτά μπορούν να συγγραφούν από ένα οποιονδήποτε κειμενογράφο (text editor) ή μπορούν να εξαχθούν από κάποια εφαρμογή τρισδιάστατης μοντελοποίησης όπως 3D Studio Max, V-Realm Builder, FormZ, κλπ.. Η δομή τους είναι δενδρική και τους επιτρέπει την αλληλεπίδραση και την αλληλεξάρτηση των επιμέρους τμημάτων τους.

Ένα αρχείο VRML αποτελείται από ένα δένδρο κόμβων (VRML Tree). Οι επιμέρους κόμβοι μπορεί να είναι:

- i. Κόμβοι Σχήμα (Shape node), όπως ένα σύνολο από πολύγωνα, μία σφαίρα, ένας κύβος, κ.α.
- ii. Κόμβοι Ιδιότητας (Property node), όπως ένα σύνολο από Normals, από Materials, ένα Light ή ένας Transformation.
- iii. Κόμβοι τύπου Ομάδας (Group node), που περιέχει κόμβους παιδιά κ.ο.κ.

Σε ένα αρχείο τύπου VRML, ένας κόμβος αποτελείται από ένα προαιρετικό όνομα, ένα τύπο κόμβου και μια λίστα από πεδία (fields) ή ιδιότητες (properties). Προφανώς, όταν δοθεί ένα όνομα σε κάποιο κόμβο, αυτός μπορεί να χρησιμοποιηθεί στη συνέχεια ως έχει. [12]

2.3 ΕΝΑΛΛΑΚΤΙΚΑ ΕΡΓΑΛΕΙΑ ΜΟΝΤΕΛΟΠΟΙΗΣΗΣ ΤΡΙΣΔΙΑΣΤΑΤΩΝ ΑΝΤΙΚΕΙΜΕΝΩΝ

Για αναπαράσταση 3D αντικειμένων, εκτός από την VRML είναι δυνατόν να χρησιμοποιηθούν και εναλλακτικές γλώσσες ή εργαλεία. Ενδεικτικά αναφέρονται τα παρακάτω:

- Η 3DMLW (3D Mark-up Language for Web) [13] είναι μια γλώσσα ανοιχτού κώδικα (open-source), που βασίζεται σε αρχεία τύπου XML και χρησιμοποιείται για την αναπαράσταση τρισδιάστατων και δισδιάστατων αντικειμένων, με δυνατότητες αλληλεπίδρασης, για τον παγκόσμιο ιστό.
- Η COLLADA (COLLABorative Design Activity) [14] είναι μια γλώσσα προγραμματισμού, που βασίζεται σε ένα XML σχήμα και έχει κατοχυρωθεί ως ένα διεθνές πρότυπο, για τη δημιουργία 3D εφαρμογών στο Διαδίκτυο.
- Η O3D [15] είναι μια γλώσσα ανοιχτού κώδικα, τύπου JavaScript API. Αναπτύχθηκε από την Google, για την δημιουργία 3D εφαρμογών, με δυνατότητες αλληλεπίδρασης, έτσι ώστε οι τρισδιάστατες εφαρμογές, να λειτουργούν σε ένα φυλλομετρητή ιστού (web browser) ή σε μια XUL

εφαρμογή. Η O3D αναπτύσσεται σαν ένα πειραματικό πρόσθετο (plug-in) για φυλλομετρητές ιστού.

- Η Universal 3D (U3D) [16] είναι ένα συμπιεσμένο πρότυπο αρχείου, που χρησιμοποιείται για δεδομένα 3D γραφικών. Η “3D Industry Forum” είχε ορίσει το πρότυπο αυτό, για να διευκολύνει την ανταλλαγή δεδομένων. Το πρότυπο αυτό κατοχυρώθηκε από την “Ecma International” τον Αύγουστο του 2005 ως ECMA-363, με στόχο να γίνει ένα διεθνές πρότυπο για 3D δεδομένα όλων των ειδών. Στην U3D μορφή, μπορούν να εισαχθούν 3D αντικείμενα με δυνατότητες αλληλεπίδρασης, σε κείμενα τύπου PDF και να παρατηρηθούν μέσω του AcrobatReader.

Η X3D [17] είναι μια γλώσσα προγραμματισμού, που έχει κατοχυρωθεί ως πρότυπο ISO (ISO/IEC 19775-1), θεωρείται υπερσύνολο της VRML και για την αναπαράσταση εικονικών κόσμων, χρησιμοποιεί αρχεία τύπου XML. Τα χαρακτηριστικά της X3D επεκτείνουν τη VRML (π.χ. Humanoid Animation, NURBS, GeoVRML, κλπ) και της δίνουν την ικανότητα να κωδικοποιεί την εικόνα, χρησιμοποιώντας ένα XML συντακτικό αντίστοιχο, του Ανοιχτού Εφευρέτη της VRML97. Ακόμα, εμπλουτίζεται και το περιβάλλον προγραμματισμού των εφαρμογών (Application Programming Interface - APIs).

2.4 ΣΧΕΔΙΑΣΜΟΣ ΕΙΚΟΝΙΚΩΝ ΚΟΣΜΩΝ ΜΕ ΧΡΗΣΗ ΤΟΥ V-REALM BUILDER

Για τον σχεδιασμό VRML αντικειμένων, είναι δυνατόν να χρησιμοποιηθεί το πρόγραμμα μοντελοποίησης εικονικών κόσμων, V-Realm Builder (Version 2.0). Το V-Realm Builder είναι ένα ισχυρό προγραμματιστικό εργαλείο, με δυνατότητα δημιουργίας 3D αντικειμένων και “κόσμων”, που στην συνέχεια μπορούν να παρατηρηθούν, από ένα V-Realm φυλλομετρητή ή άλλο φυλλομετρητή συμβατό με VRML 2.0. Το V-Realm Builder και η VRML, δε δημιουργήθηκαν για να αντικαταστήσουν τα σημερινά εργαλεία μοντελοποίησης, που μπορούν να δημιουργήσουν ρεαλιστικά αντικείμενα, με 5 ή 10 εκατομμύρια πολύγωνα. Το V-Realm Builder, χρησιμοποιώντας VRML, έχει το πλεονέκτημα, να ελαχιστοποιεί το μέγεθος των αρχείων και να παρέχει ένα μέσο μοντελοποίησης σύνθετων αντικειμένων, που χρησιμοποιούν μικρό μέγεθος αρχείων.

Το V-Realm Builder έχει ένα φιλικό για τον χρήστη, γραφικό παραθυρικό περιβάλλον, που του δίνει τη δυνατότητα σχεδιασμού και ελέγχου ενός 3D κόσμου, χρησιμοποιώντας μόνο το ποντίκι, αφού δεν είναι απαραίτητη η γραφή κώδικα (hand-coding). Το γραφικό περιβάλλον (Graphical User Interface- GUI) που διαθέτει το V-Realm Builder, είναι προσαρμοσμένο ειδικά στη VRML, με ισχυρές επεμβατικές ικανότητες και λόγω των άμεσων στιγμιαίων οπτικών αναδράσεων, απλοποιεί τη διαδικασία δημιουργίας 3D κόσμων. [18]

2.5 ΕΓΚΑΤΑΣΤΑΣΗ ΚΑΙ ΑΡΧΙΚΟΠΟΙΗΣΗ ΤΟΥ V-REALM BUILDER

Οι εικονικοί κόσμοι που δημιουργούνται με την βοήθεια του V-Realm Builder, έχουν την δυνατότητα αλληλεπίδρασης και ελέγχου, από το λογισμικό MATLAB. Το V-Realm Builder συνοδεύει το λογισμικό MATLAB και μπορεί να εγκατασταθεί μέσω αυτού. Προκειμένου να γίνει αυτό, στο Command Window του λογισμικού MATLAB πληκτρολογούνται οι εντολές: `vrinstall` και `-install editor` ή για λόγους συντομίας μπορεί από την αρχή να πληκτρολογηθεί η εντολή: `vrinstall('-install','editor')`. Στην συνέχεια εμφανίζονται στο Command Window του λογισμικού MATLAB τα εξής:

Starting editor installation...

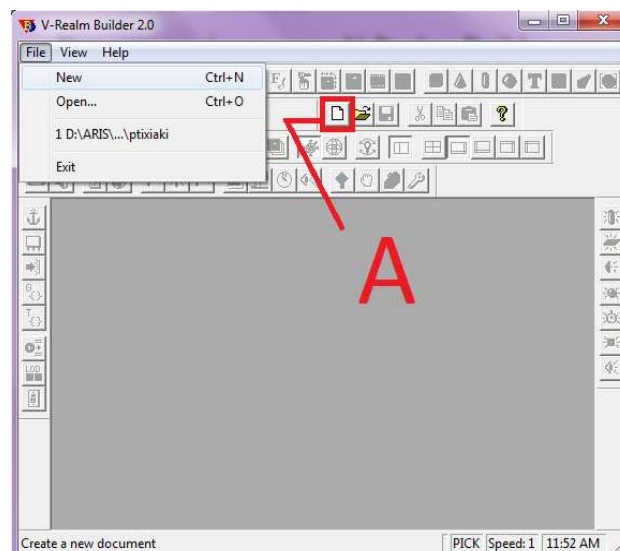
Done.

Για να ελεγχθεί εάν είναι επιτυχής η εγκατάσταση, στο Command Window του λογισμικού MATLAB πληκτρολογούνται οι εντολές: `vrinstall` και `-check` ή για λόγους συντομίας μπορεί από την αρχή να δοθεί η εντολή: `vrinstall('-check')`. Εάν η εγκατάσταση είναι επιτυχής, εμφανίζονται στο Command Window του λογισμικού MATLAB τα εξής:

VRML editor: installed

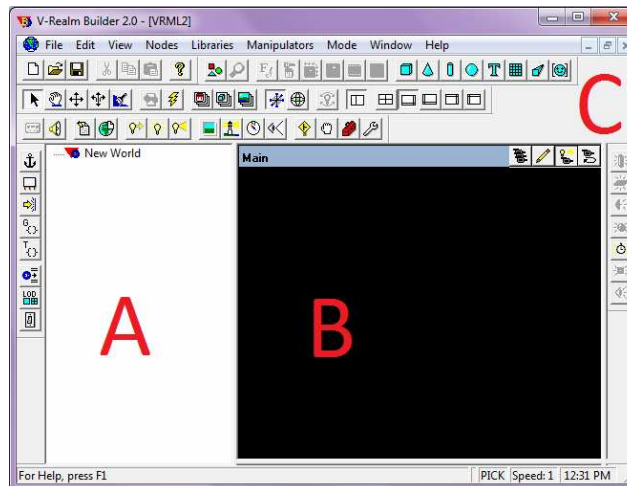
Για την εκκίνηση του V-Realm Builder στα Microsoft Windows, πρέπει να εκτελεστεί το αρχείο “`vrbuild2.exe`”. [19]

Για να σχεδιαστεί ένας νέος κόσμος με το V-Realm Builder, θα πρέπει στο μενού του V-Realm Builder να επιλεγεί το File και μετά το New ή να πληκτρολογηθεί `Ctrl+N` ή επιλογή του εικονίδιο New (Σχήμα 2.1, [A]).



Σχήμα 2.1 Μενού έναρξης του V-Realm Builder

Στο αριστερό παράθυρο του V-Realm Builder, υπάρχει ένα δέντρο κόμβων (Σχήμα 2.2, [A]) που καθορίζει την ιεραρχία των αντικειμένων και τις ιδιότητές τους, όπως τη θέση τους ως προς τον κόσμο και ως προς άλλα αντικείμενα, μέγεθος, χρώμα, περιστροφή κλπ. Στα δεξιά που είναι το βασικό παράθυρο του εικονικού κόσμου (Σχήμα 2.2, [B]), εμφανίζεται ένας άδειος εικονικός κόσμος που θα γεμίσει με τα διάφορα αντικείμενα που θα εισαχθούν σε αυτόν. Ακόμα, υπάρχουν και διάφορες εργαλειοθήκες για την επεξεργασία του εικονικού κόσμου (Σχήμα 2.2, [C]).



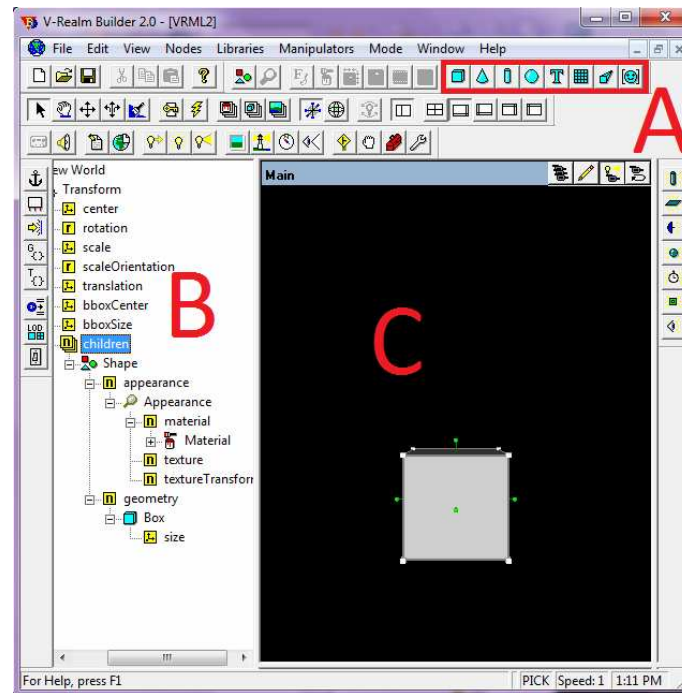
Σχήμα 2.2 Το γραφικό περιβάλλον του V-Realm Builder

Η ιεραρχία ενός τυπικού VRML δέντρου έχει την ακόλουθη δομή. Στην κορυφή βρίσκεται ο εικονικός κόσμος και στο αμέσως χαμηλότερο επίπεδο βρίσκονται διάφοροι κόμβοι (nodes). Κάθε κόμβος (node) αντιπροσωπεύει ένα από τα αντικείμενα που έχουν εισαχθεί και απαρτίζουν τον εικονικό κόσμο, μαζί με τις ιδιότητες που το χαρακτηρίζουν. Τα ομαδοποιημένα στοιχεία κάθε κόμβου λέγονται παιδιά (children). Για παράδειγμα, ένα κτίριο μιας πόλης, μπορεί να θεωρηθεί σαν μια ομάδα κόμβων που έχει ως παιδιά, παράθυρα, πόρτες και τοίχους. Αφού αυτά τα ομαδοποιημένα στοιχεία του κτιρίου, είναι ένα μέρος από την μεγαλύτερη δομή ενός κόσμου που έχει οριστεί, δηλαδή την πόλη. Δεν υπάρχει κάποιο όριο, στον αριθμό των παιδιών που μπορεί να έχει μια ομάδα, αλλά όλα τα παιδιά μοιράζονται υποχρεωτικά τα χαρακτηριστικά του γονέα (group node).

2.6 ΕΙΣΑΓΩΓΗ ΚΑΙ ΕΠΕΞΕΡΓΑΣΙΑ ΑΝΤΙΚΕΙΜΕΝΩΝ ΣΤΟ V-REALM BUILDER

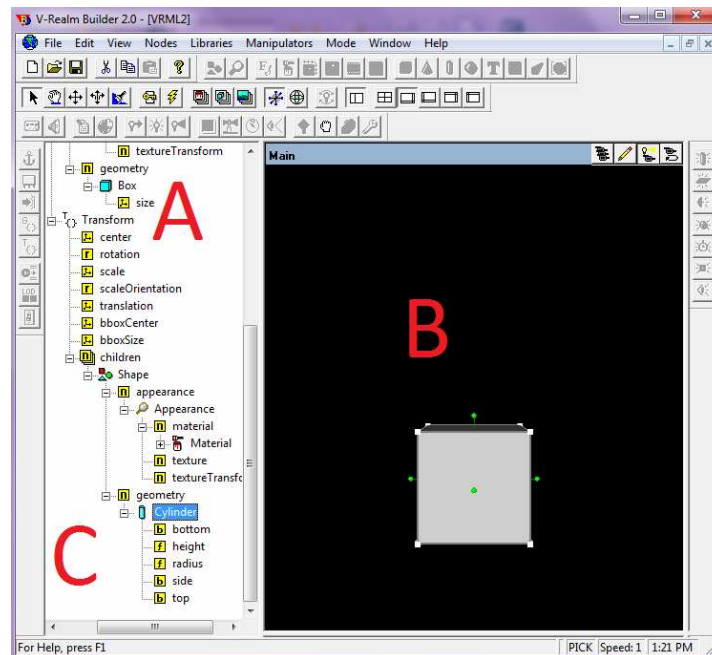
Για την εισαγωγή ενός τρισδιάστατου γεωμετρικού αντικειμένου, στον εικονικό κόσμο, όπως π.χ. ενός κύβου, πρέπει στην Geometry Node ToolBar (Σχήμα 3.3, [A]) να επιλεγεί, το αντίστοιχο επιθυμητό εικονίδιο.

Μετά την εισαγωγή του, στα αριστερά φαίνεται το δέντρο που περιγράφει τον κύβο (Σχήμα 2.3, [B]). Περιέχει πληροφορίες του αντικειμένου (Σχήμα 2.3, [C]) για τη θέση του στον χώρο, το σχήμα, την εμφάνιση, το υλικό και τη γεωμετρία του (Transform, Shape, Appearance, Material και geometry αντίστοιχα).



Σχήμα 2.3 Εισαγωγή αντικειμένου στο V-Realm Builder

Για να ενωθεί ένα αντικείμενο με ένα άλλο, όπως ένας κύβος με ένα κύλινδρο, θα πρέπει να επιλεγεί το children (παιδιά), από το δέντρο του πρώτου αντικειμένου, δηλαδή το κύβου (Σχήμα 2.4, [A]) και με τον ίδιο τρόπο, να εισαχθεί από την Geometry Node ToolBar το δεύτερο αντικείμενο, δηλαδή ο κύλινδρος. Όπως φαίνεται παρακάτω, έχει εισαχθεί ο κύλινδρος (Σχήμα 3.4, [B]), που είναι πλέον συνδεδεμένος, με τον κύβο. Με την εισαγωγή του κύλινδρου, εμφανίζεται αυτόματα και το VRML δέντρο του (Σχήμα 2.4, [C]), που εμπεριέχει τις πληροφορίες και τα χαρακτηριστικά του.



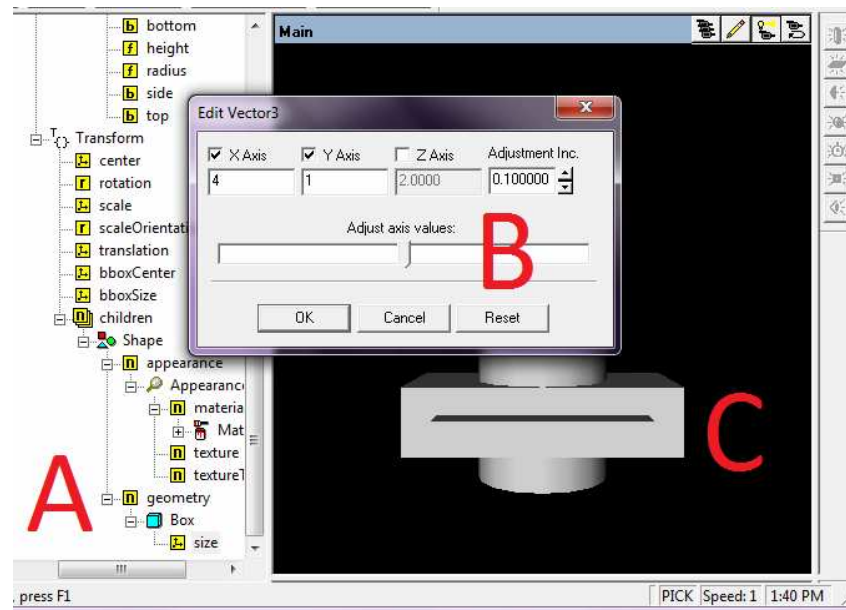
Σχήμα 2.4 Ένωση αντικειμένων στο V-Realm Builder

Επαναλαμβάνουμε την ίδια διαδικασία για να εισάγουμε έναν κύβο ακόμα ενωμένο με τον κύλινδρο.

Για την επεξεργασία των διαστάσεων των κύβων, πρέπει στο δέντρο του καθε κύβου (Σχήμα 2.5, [A]) να ανοίξει:

- i. Ο κόμβος children, που έχει τα χαρακτηριστικά του σχήματος του αντικειμένου.
- ii. Ο κόμβος Shape, που έχει τα γεωμετρικά χαρακτηριστικά και τα χαρακτηριστικά του υλικού.
- iii. Ο κόμβος geometry που έχει το γεωμετρικό σχήμα.
- iv. Ο κόμβος Box που είναι το σχήμα που είχε επιλεγεί παραπάνω.
- v. Το size που ορίζει το μέγεθος του γεωμετρικού σχήματος ενός αντικειμένου.

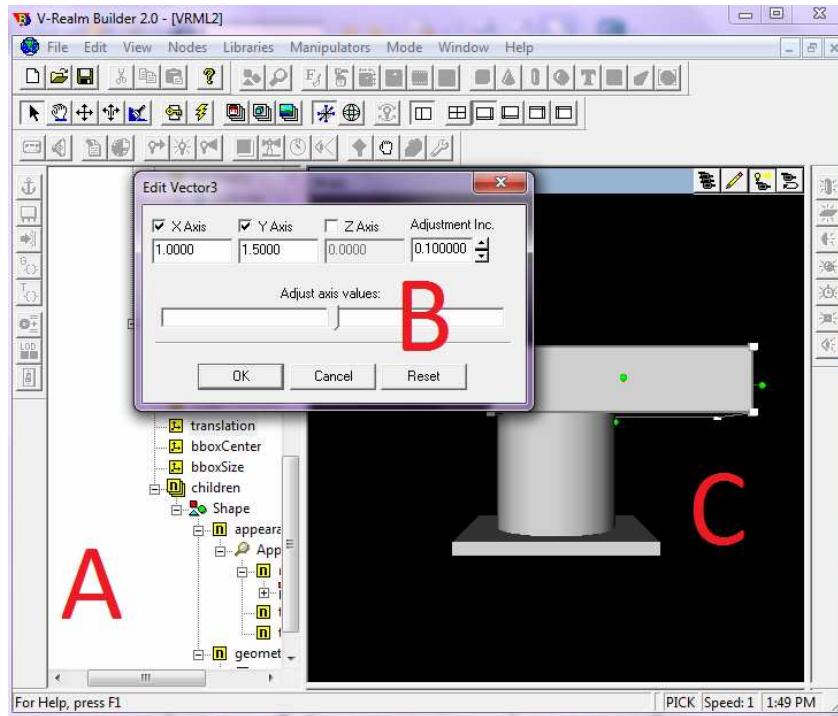
Στη συνέχεια εμφανίζεται ένα παράθυρο με όνομα “Edit Vector 3” (Σχήμα 2.5,[B]), για να αλλάξουν οι τιμές κατά τους άξονες X, Y και Z. Όπως φαίνεται παρακάτω, ο κύβος έχει μετατραπεί σε ένα ορθογώνιο παραλληλόγραμμο (Σχήμα 2.5,[C]), λόγω των αλλαγών που έχει υποστεί στις διαστάσεις του.



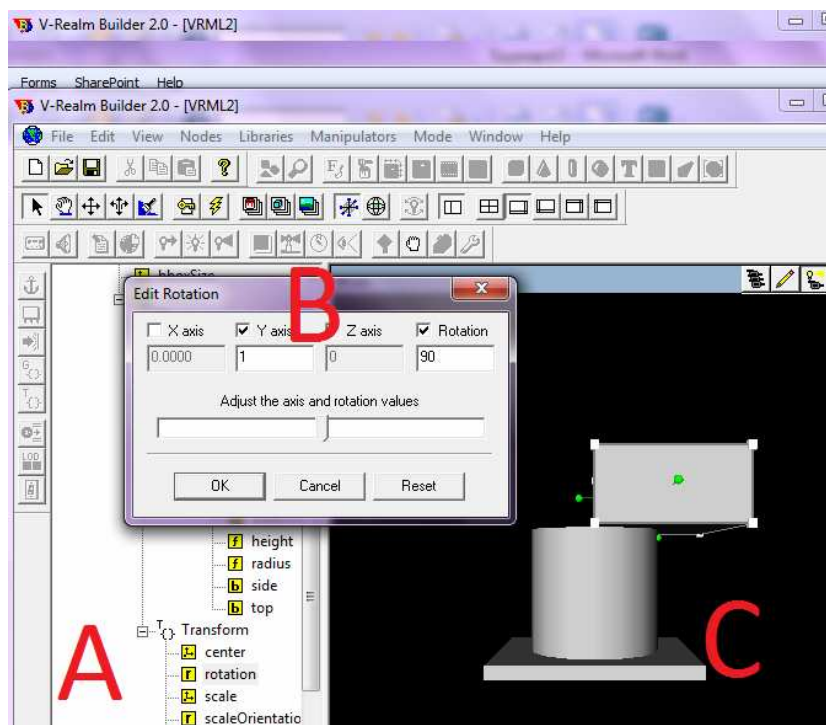
Σχήμα 2.5 Επεξεργασία διαστάσεων αντικειμένων στο V-Realm Builder

Για την επεξεργασία των σχετικών θέσεων των αντικειμένων και συγκεκριμένα, της θέσης του δεύτερου αντικειμένου σε σχέση με το πρώτο, πρέπει να μεταβληθούν οι τιμές της παραμέτρου Transform, στο δέντρο του δεύτερου αντικειμένου (Σχήμα 2.6, [A]). Δηλαδή, στο Transform του δέντρου, του ορθογώνιου παραλληλογράμμου, πρέπει να επιλεγεί το translation, που ορίζει τη θέση των αντικειμένων (Transform- translation). Στο παράθυρο που θα εμφανιστεί (Σχήμα 2.6, [B]) πρέπει να αλλαχθούν οι τιμές κατά τους άξονες X, Y και Z. Όπως φαίνεται, το αντικείμενο έχει τοποθετηθεί στην κορυφή του κυλίνδρου και προς τα δεξιά (Σχήμα 2.6, [C]).

Για την επεξεργασία της γωνίας και του άξονα περιστροφής των αντικειμένων, συγκεκριμένα του δεύτερου αντικειμένου σε σχέση με το πρώτο, θα πρέπει στο Transform του δέντρου, του ορθογώνιου παραλληλογράμμου (Σχήμα 2.7.α, [A]), να επιλεγεί το rotation που ορίζει τις γωνίες και τον άξονα περιστροφής (Transform rotation). Θα εμφανιστεί ένα παράθυρο με όνομα “Edit Rotation”(Σχήμα 2.7.α, [B]), για να επιλεγεί η γωνία και ο άξονας περιστροφής του αντικειμένου. Όπως φαίνετε παρακάτω, το αντικείμενό έχει περιστραφεί ως προς άλλο άξονα (Σχήμα 2.7.α, [C]). Συγκεκριμένα, ο διαμορφωμένος κύβος έχει περιστραφεί 90° ως προς τον Y άξονα.

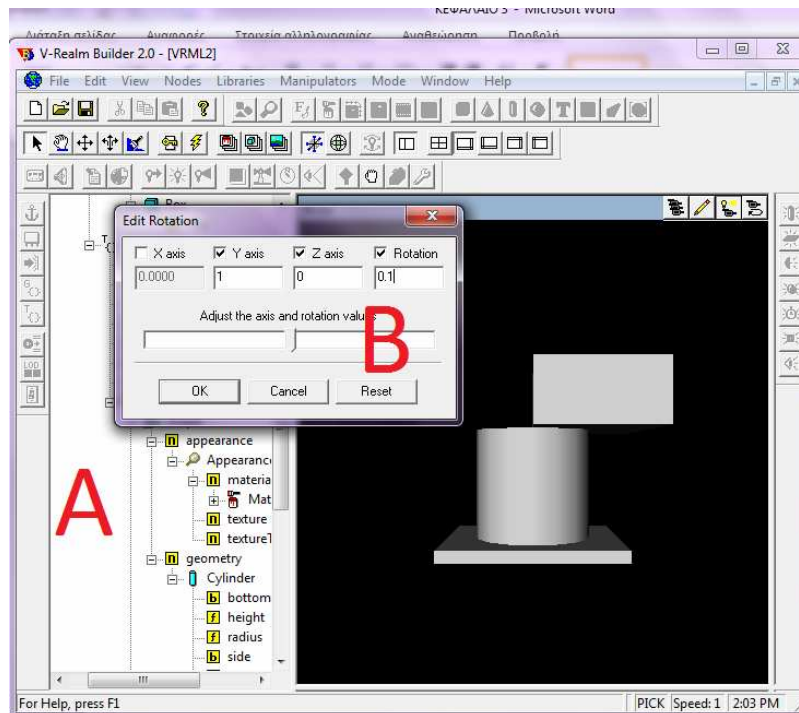


Σχήμα 2.6 Επεξεργασία θέσης αντικειμένων στο V-Realm Builder



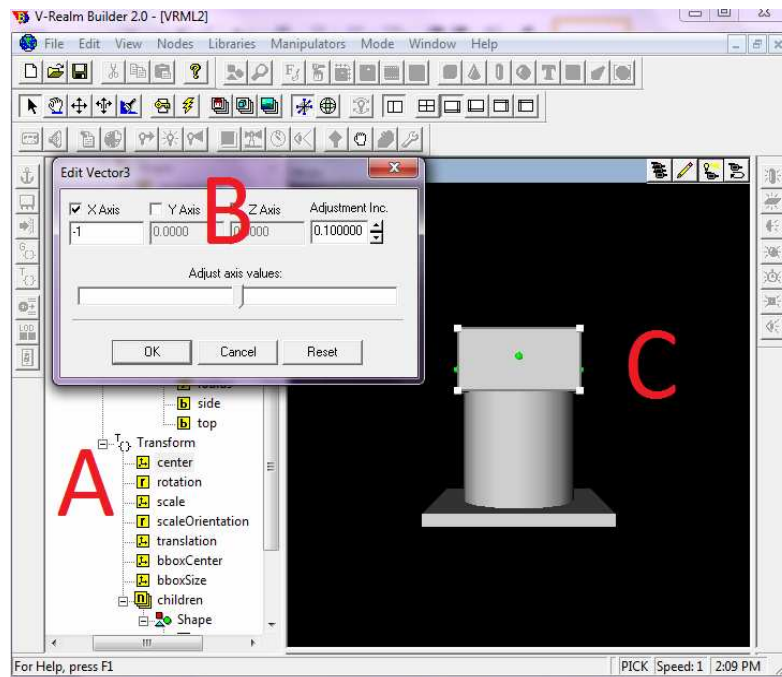
Σχήμα 2.7.α Επεξεργασία γωνίας και άξονα περιστροφής αντικειμένων στο V-Realm Builder

Για να επιλεγεί ο ίδιος άξονας περιστροφής και για τα τρία αντικείμενα, θα πρέπει στο Transform του δέντρου, του κυλίνδρου (Σχήμα 2.7.β, [A]), να επιλεγεί το rotation που ορίζει τις γωνίες και τον άξονα περιστροφής (Transform-rotation) και στο παράθυρο που θα εμφανιστεί με όνομα “Edit Rotation” (Σχήμα 2.7.β, [B]) να επιλεγεί ο επιθυμητός άξονας (Y άξονας). Αν δεν επιλεγεί κοινός άξονας περιστροφής, το αντικείμενο θα περιστρέφεται ως προς δύο άξονες και η κίνηση δεν θα είναι ομαλή.



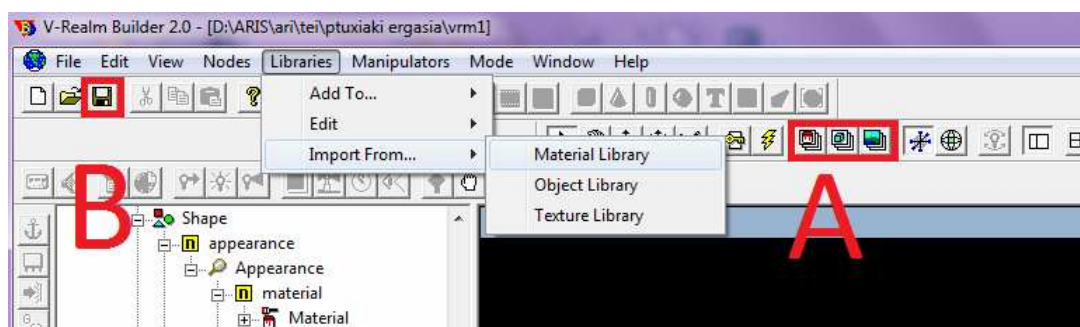
Σχήμα 2.7.β Επεξεργασία γωνίας και άξονα περιστροφής αντικειμένων στο V-Realm Builder

Για να επιλεγεί το ίδιο κέντρο και για τα δυο αντικείμενα, πρέπει στο Transform του δέντρου, του ορθογώνιου παραλληλογράμμου (Σχήμα 2.8, [A]), να επιλεγεί το center, που ορίζει το κέντρο ενός αντικειμένου και στο παράθυρο που εμφανίζεται (Σχήμα 3.7, [B]), να μεταβληθούν οι τιμές του κέντρου κατά τους άξονες X, Y και Z. Όπως φαίνεται, το κέντρο του αντικειμένου έχει μετατοπιστεί (Σχήμα 2.8, [C]).



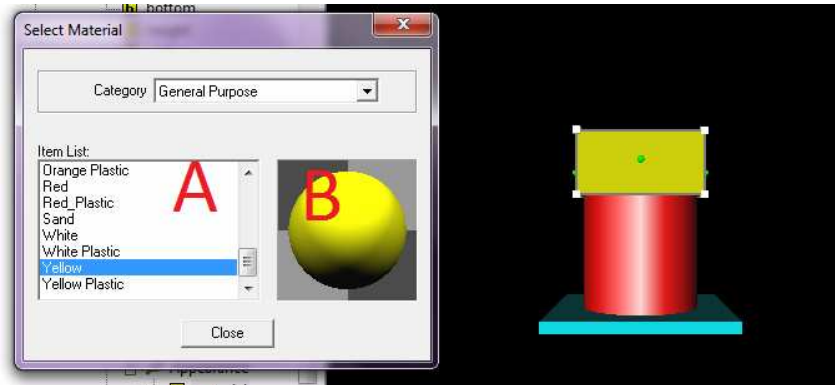
Σχήμα 2.8 Επεξεργασία κέντρου αντικειμένων στο V-Realm Builder

Τα αντικείμενα εμφανίζονται με μια απόχρωση του γκρι, αυτό συμβαίνει γιατί δεν έχουν ορισθεί ακόμα χρώματα για τα αντικείμενα. Για την εισαγωγή χρωμάτων, θα πρέπει στη Mode View ToolBar (Σχήμα 2.9, [A]), να επιλεγεί το πρώτο εικονίδιο που είναι η Βιβλιοθήκη Υλικών (Material Library), για την επιλογή ενός μονόχρωμου υλικού. Το τρίτο εικονίδιο είναι η Βιβλιοθήκη Υφής (Texture Library) και έχει τη δυνατότητα επιλογής έτοιμων χρωματιστών σχεδίων υφής (Texture).



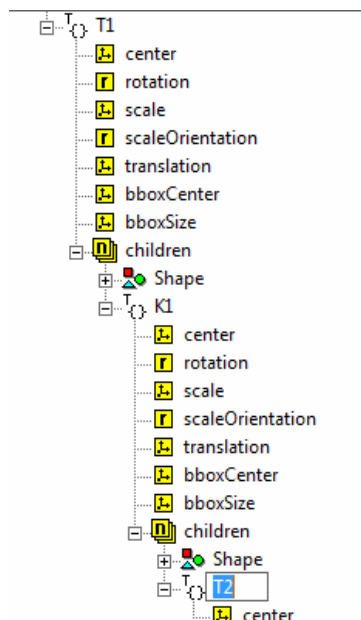
Σχήμα 3.9 Ορισμός χρωμάτων από το Mode View ToolBar του V-Realm Builder

Στο παράθυρο που θα εμφανιστεί πρέπει να επιλεγεί χρώμα από την “Item List:”(Σχήμα 2.10, [A]) και στη συνέχεια από την σφαίρα-δειγματολόγιο (Σχήμα 2.10, [B]),να τοποθετηθεί στο αντικείμενο (drag and drop).



Σχήμα 2.10 Εισαγωγή χρωμάτων αντικειμένων στο V-Realm Builder

Προκειμένου να ελεγχθούν τα αντικείμενα ενός εικονικού κόσμου, είναι αναγκαίο να χαρακτηρίζονται από κάποιο όνομα αναφοράς. Για να γίνει η μετονομασία των αντικειμένων αυτών, πρέπει στο δέντρο τους, να επιλεγεί το Transform και μετά να επιλεγεί ξανά, για να μπορεί να ονομαστεί από το πληκτρολόγιο. Θέτω T1 και T2 τα όνοματα του κύβων και K1 το όνομα του κυλίνδρου. Στο δέντρο δεν χρειάζεται να ονομαστεί και ο εικονικός κόσμος, δηλαδή το New World καθώς θα ονομαστεί αυτόματα μόλις αυτός αποθηκευτεί.



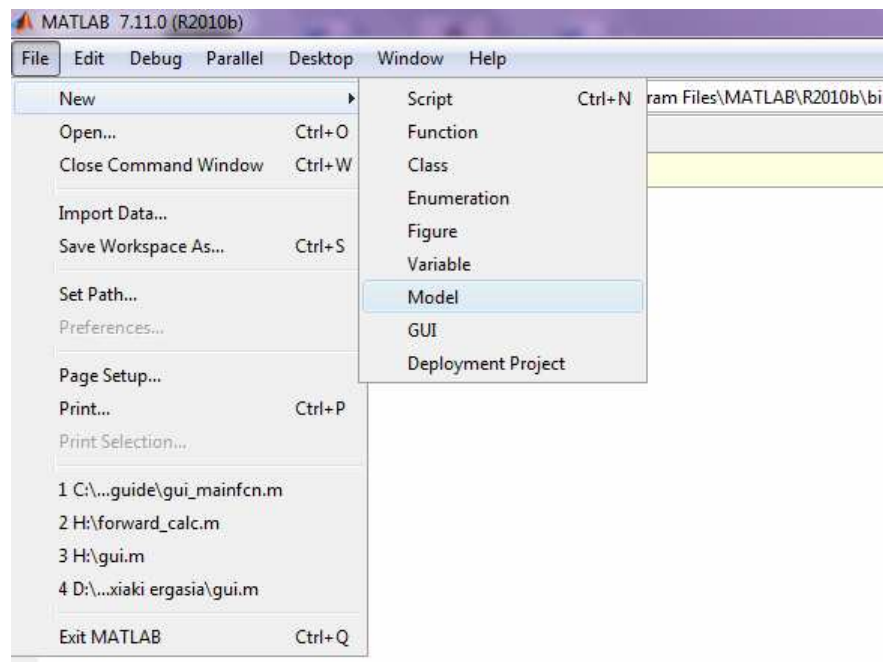
Σχήμα 2.11 Μετονομασία αντικειμένων στο V-Realm Builder

Για να αποθηκευτεί ο εικονικός κόσμος που έχει δημιουργηθεί, πρέπει στο μενού του V-Realm Builder, να επιλεγεί το File και μετά στο Save ή Ctrl+S ή να επιλεγεί το εικονίδιο Save (Σχήμα 2.9, [B]). Το όνομα μπορεί να αλλαχθεί ή να μείνει αυτό που ήδη υπάρχει ως προεπιλογή (vrml1). Έτσι, δημιουργείτε ένα αρχείο με κατάληξη .wrl(π.χ. vrml1.wrl) από τη λέξη world.

2.7 ΈΛΕΓΧΟΣ ΕΙΚΟΝΙΚΩΝ ΚΟΣΜΩΝ ΜΕΣΩ SIMULINK

Οι εικονικοί κόσμοι που δημιουργούνται με τη χρήση του V-Realm Builder, είναι δυνατόν να ελεγχθούν μέσω του λογισμικού Simulink. Το Simulink είναι ένα γραφικό περιβάλλον μοντελοποίησης και προσομοίωσης γραμμικών και μη γραμμικών συστημάτων. Αποτελεί τμήμα του λογισμικού MATLAB και διαφοροποιείται από αυτό, στο ότι διαθέτει, ένα παραθυρικό γραφικό περιβάλλον (GUI) που με την βοήθεια αυτού, ένας χρήστης μπορεί να δημιουργήσει δικά του μοντέλα. Για την αλληλεπίδρασή του με τους εικονικούς κόσμους, χρησιμοποιείται η βιβλιοθήκη εικονικής πραγματικότητας (Virtual Reality Toolbox library) του Simulink. Η βιβλιοθήκη εικονικής πραγματικότητας, παρέχει λογικά (block) διαγράμματα για απευθείας σύνδεση των σημάτων του λογισμικού προσομοίωσης με εικονικούς κόσμους.

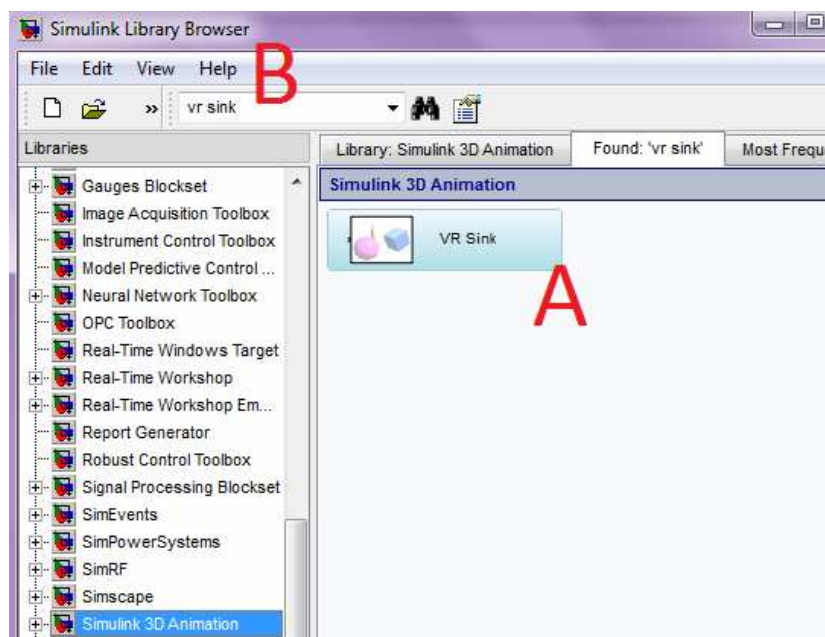
Για την δημιουργία ενός μοντέλου προσομοίωσης (Simulink Model) το οποίο έχει κατάληξη .mdl (από την λέξη model), θα πρέπει στο κεντρικό μενού του λογισμικού MATLAB να επιλεγεί File-New-Model και να αποθηκευτεί με ένα επιθυμητό όνομα. Συγκεκριμένα, “demo1.mdl”.



Σχήμα 2.12 Δημιουργία Μοντέλου προσομοίωσης στο λογισμικό MATLAB

Για να γίνει κατανοητός ο τρόπος σύνδεσης εικονικών κόσμων, με μοντέλα προσομοίωσης που έχουν αναπτυχθεί στο λογισμικό Simulink, θα περιγραφεί αναλυτικά ο τρόπος σύνδεσης του κόσμου που αναπτύχθηκε στις προηγούμενες παραγράφους, με ένα μοντέλο αυτού του τύπου.

Με την χρήση του διαγράμματος VR Sink (Σχήμα 2.13, [A]), είναι δυνατή η εγγραφή δεδομένων από το μοντέλο προσομοίωσης, στον εικονικό κόσμο. Για να εντοπιστεί το διάγραμμα αυτό, τοποθετείται στο φυλλομετρητή της Βιβλιοθήκης του λογισμικού Simulink (Simulink Library Browser), η λέξη κλειδί “VR Sink” (Σχήμα 2.13, [B]). Γίνεται η αναζήτηση του VR Sink και όταν αυτό βρεθεί, τοποθετείται στο μοντέλο προσομοίωσης.

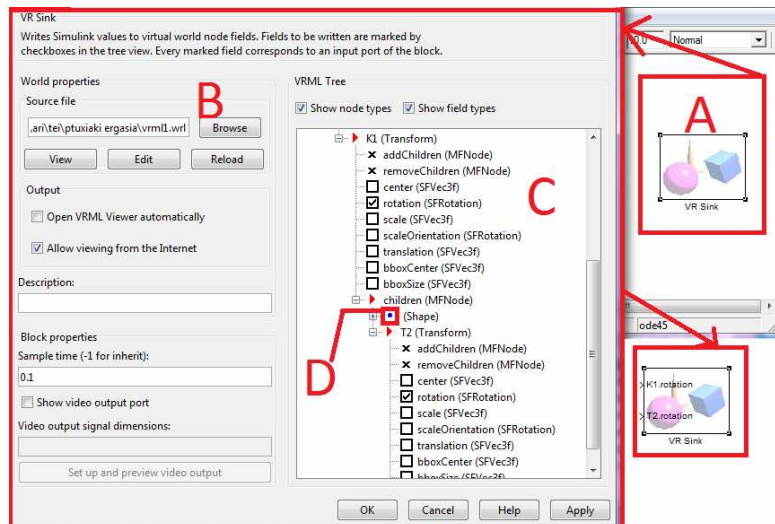


Σχήμα 2.13 Άνοιγμα του φυλλομετρητή της βιβλιοθήκης του λογισμικού Simulink από το λογισμικό MATLAB

Οι παράμετροι ελέγχου, μπορούν να επιλεγθούν, μέσα από το περιβάλλον προσομοίωσης. Μετά από την επιλογή των παραμέτρων, το λογισμικό Simulink ενημερώνει το διάγραμμα (VR Sink) με εισόδους και εξόδους, που αντιστοιχούν στους επιλεγμένους κόμβους του εικονικού κόσμου. Αφού συνδεθούν στις εισόδους τα κατάλληλα σήματα, ο χρήστης μπορεί να παρακολουθεί τη προσομοίωση με ένα λογισμικό αναπαραστάσης VRML (VRML viewer).

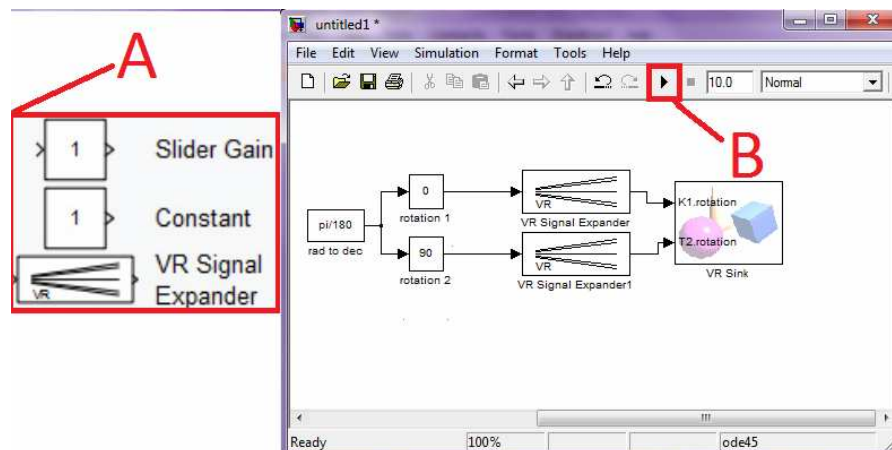
Όλες οι ιδιότητες των VRML κόμβων, παρατίθενται ιεραρχικά σε μορφή δέντρου. Όταν εισαχθεί στο VR Sink ένας εικονικός κόσμος, σαρώνεται αυτόματα για διαθέσιμους VRML κόμβους, που μπορούν να οριστούν ως είσοδοι του διαγράμματος και να οδηγηθούν σε αυτές, τα σήματα που παράγονται από το λογισμικό Simulink.

Συγκεκριμένα για τον έλεγχο των δυο εκ των τριών αντικειμένων που περιγράφονται στις προηγούμενες παραγράφους, τοποθετείται ένα VR Sink. Για να εμφανιστεί το παράθυρο ιδιοτήτων “Parameters: VR Sink”, θα πρέπει να ανοιχτεί το VR Sink (Σχήμα 2.14, [A]). Στα αριστερά του παραθύρου “Parameters: VR Sink”, υπάρχει ένα πλαίσιο (panel) με το όνομα “World properties”. Μέσα σε αυτό, υπάρχει άλλο ένα, με το όνομα “Source file” (Σχήμα 2.14, [B]). Στο Source file, πρέπει να επιλεγεί το “Browse”, για να αναζητηθεί και να βρεθεί, το αρχείο του εικονικού κόσμου που θα χρησιμοποιηθεί (vrml1.wrl). Στα δεξιά του παραθύρου “Parameters: VR Sink”, υπάρχει το πλαίσιο “VRML Tree”, στο οποίο εμφανίζεται, το VRML δέντρο του εικονικού κόσμου (Σχήμα 2.14, [C]). Για να επιλεχθούν οι παράμετροι ελέγχου του κόσμου, θα πρέπει να επεκταθούν (expand) τα Transform επιλέγοντας τα εικονίδια με το “+” (Σχήμα 2.14, [D]) και από εκεί να επιλεχθούν οι εισοδοί, συγκεκριμένα, τα “rotation (SFRotation)”.



Σχήμα 2.14 Παράθυρο ιδιοτήτων του VR Sink διαγράμματος

Για να επιτευχθεί η κίνηση των αντικειμένων, πρέπει να τοποθετηθούν δομικά τμήματα (Σχήμα 2.15, [A]), που να συνδέονται με τις εισόδους του VR Sink και δίνουν τις τιμές για την κίνηση των αντικειμένων.

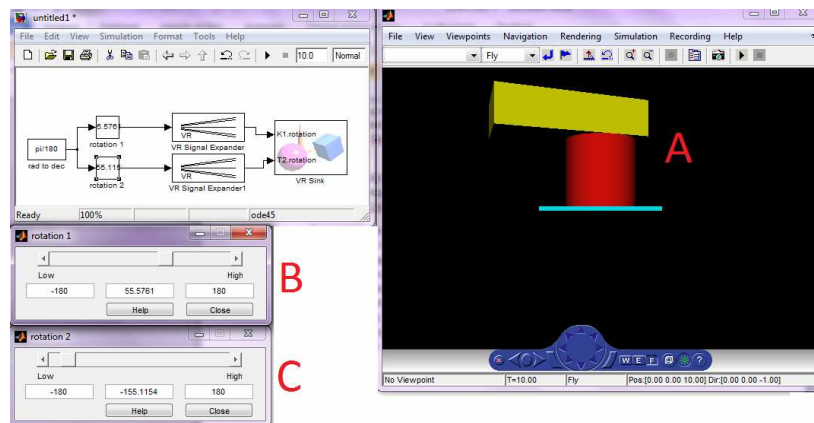


Σχήμα 2.15 Διαγράμματα και το Μοντέλο προσομοίωσης στο λογισμικό MATLAB

Για να κινηθεί περιστροφικά ολόκληρο το αντικείμενο ή μόνο το ορθογώνιο παραλληλόγραμμο, δίνονται τιμές από δύο “Slider Gain” διαγράμματα (ένα για την κάθε περιστροφική κίνηση), που έχουν τη δυνατότητα να πολλαπλασιάζουν το σήμα εισόδου, ανάλογα με την θέση που βρίσκεται η μπάρα ολίσθησης (Slider). Για τον έλεγχο της πλήρους περιστροφής (360°) του κάθε αντικειμένου, τα Slider Gain ρυθμίζονται με όριο από -180 έως 180 . Οι τιμές των Slider Gain είναι σε ακτίνια. Για την μετατροπή των ακτινίων σε μοίρες, τοποθετείται στην είσοδο των Slider Gain διαγραμμάτων, ένα “Constant” διάγραμμα ίσο με $\pi/180$, το οποίο έχει την ιδιότητα, να δίνει ένα σήμα εξόδου. Αυτό στην συνέχεια πολλαπλασιάζεται με το αποτέλεσμα των Slider Gain διαγραμμάτων και οδηγείται στην έξοδό τους. Πριν οδηγηθεί το σήμα στο VR Sink, τοποθετείται ανάμεσα στην έξοδο των Slider Gain διαγραμμάτων και την είσοδο του VR Sink διαγράμματος, ένα “VR Signal Expander” για την επέκταση του σήματος.

Για την έναρξη της προσομοίωσης, πρέπει στο μενού του Simulink Model (μοντέλου προσομοίωσης) του λογισμικού MATLAB, να επιλεγεί το Simulation και μετά το Start ή Ctrl+T ή να επιλεγεί το εικονίδιο Start simulation (Σχήμα 2.15, [B]).

Για να εμφανιστεί ο εικονικός κόσμος (Σχήμα 2.16, [A]), πρέπει να επιλεγεί το VR Sink. Για τον έλεγχο του εικονικού κόσμου υπάρχουν τα Slider Gain, που ανοίγοντάς τα εμφανίζονται οι μπάρες ολίσθησης. Η περιστροφή του κάθε αντικειμένου στον εικονικό κόσμο, είναι ανάλογη των τιμών της μπάρας ολίσθησης. Με την μπάρα ολίσθησης του παραθύρου που έχει ονομαστεί “rotation 1”, μπορεί να ελεγχθεί η περιστροφή ολόκληρου του αντικειμένου (Σχήμα 2.16, [B]). Με την μπάρα ολίσθησής του παραθύρου που έχει ονομαστεί “rotation 2”, μπορεί να ελεγχθεί η περιστροφή μόνο του ορθογώνιου παραλληλόγραμμου, του αντικειμένου (Σχήμα 2.16, [C]).



Σχήμα 2.16 Μπάρες ολίσθησης του Μοντέλου προσομοίωσης και προβολή από ειδικό λογισμικό προσομοίωσης του λογισμικού MATLAB

ΚΕΦΑΛΑΙΟ 3

Σχεδιασμός και Ανάπτυξη του Εικονικού Μοντέλου

3.1 ΕΙΣΑΓΩΓΗ

Στο κεφάλαιο αυτό θα παραθέσουμε αναλυτικά με χρήση εικόνων και σχολίων την κατασκευή στον εικονικό κόσμο ενός ρομποτικού βραχίονα τύπου scara. Η υλοποίηση έγινε όπως αναφέραμε παραπάνω μέσω matlab στο v-realm builder.

3.2 ΑΛΓΟΡΙΘΜΟΣ DENAVIT-HARTENBERG (DH)

Ένας ρομποτικός βραχίονας αποτελείται από άκαμπτα σώματα που αποκαλούνται σύνδεσμοι, συνδεδεμένοι μεταξύ τους με αρθρώσεις. Η συνδεσμολογία αυτή συνιστά μια κινηματική αλυσίδα. Οι συχνότερα χρησιμοποιούμενες αρθρώσεις είναι οι περιστροφικές (1 βαθμός ελευθερίας), οι πρισματικές (1 βαθμός ελευθερίας) και οι σφαιρικές (3 βαθμοί ελευθερίας). Ο αλγόριθμος των Denavit – Hartenberg, είναι μια διαδικασία σύμφωνα με την οποία τοποθετούνται ορθοκανονικά και δεξιόστροφα Συστήματα Συντεταγμένων (Σ.Σ) στους συνδέσμους του βραχίονα (στη βάση, στους συνδέσμους και στο άκρο εργασίας), με σκοπό την κατάστρωση και επίλυση του ευθέους και αντίστροφου κινηματικού προβλήματος. Το ευθύ κινηματικό πρόβλημα, συνίσταται στον υπολογισμό της θέσης και του προσανατολισμού του άκρου εργασίας, ως προς ένα σταθερό σύστημα αναφοράς με συγκεκριμένη τιμή. Η κίνηση και ο έλεγχος του βραχίονα και συγκεκριμένα του άκρου εργασίας, επιτυγχάνεται με την τοποθέτηση τοπικών συστημάτων συντεταγμένων σε κάθε άρθρωση του ρομπότ. Η αρχή κάθε τοπικού συστήματος συντεταγμένων, θεωρείται πακτωμένη στην αντίστοιχη άρθρωση και το σύστημα συντεταγμένων κινείται όπως κινείται ο επόμενος σύνδεσμος. Τα βήματα που του αλγόριθμου Denavit – Hartenberg (D-H) είναι τα εξής:

Βήμα 1

- Οι σύνδεσμοι και οι αρθρώσεις αριθμούνται διαδοχικά από τη βάση προς το άκρο εργασίας.
- Η βάση είναι ο σύνδεσμος 0 και το άκρο εργασίας είναι ο σύνδεσμος N.

- Η άρθρωση 1 συνδέει τη βάση (σύνδεσμος 0) με το σύνδεσμο 1 (ο σύνδεσμος που έπεται της άρθρωσης 1, είναι ο σύνδεσμος 1).
- Δεν υφίστανται σύνδεσμοι πέραν του οριακού σημείου του άκρου εργασίας.
- Το τελευταίο Σ.Σ. $\{N\}$ τοποθετείται στο άκρο εργασίας αυθαίρετα. Ο άξονας x_N πρέπει να είναι κάθετος στον άξονα z_{N-1} . Ο άξονας z_N τοποθετείται συνήθως παράλληλος με τον z_{N-1} .

Βήμα 2

- Τοποθέτηση των αξόνων z_i ($i = 0, 1, 2, \dots, N$):
- Ο άξονας z_0 τοποθετείται (οπουδήποτε) στη βάση, πρέπει όμως να βρίσκεται στη διεύθυνση κίνησης της άρθρωσης 1.
- Ο άξονας z_{i-1} τοποθετείται στη διεύθυνση κίνησης της άρθρωσης i .
- Αν η άρθρωση είναι περιστροφική, τότε η διεύθυνση του άξονα z είναι η διεύθυνση στροφής (η θετική φορά επιλέγεται αυθαίρετα). Αν είναι πρισματική, τότε η διεύθυνση του άξονα z , είναι η διεύθυνση της γραμμικής κίνησης (η θετική φορά είναι η φορά με την οποία γίνεται η επιμήκυνση).

Βήμα 3

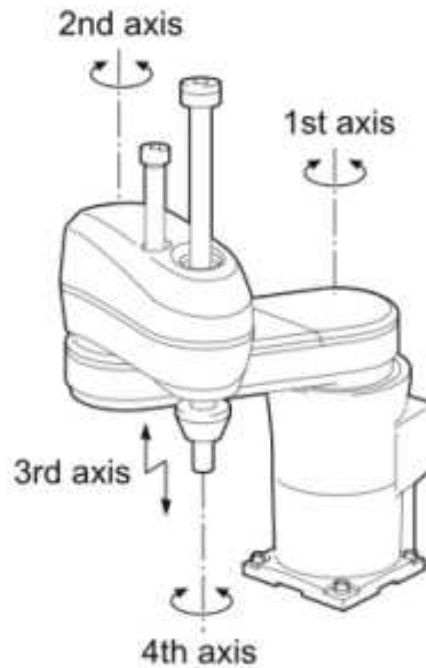
- Τοποθέτηση των αξόνων x_i, y_i ($i = 0, 1, 2, \dots, N$)
- Ο άξονας x_i τοποθετείται κάθετα στον z_{i-1} και στον z_i με φορά από τον z_{i-1} προς τον z_i . Βρίσκεται δηλαδή στην κοινή κάθετο των z_{i-1}, z_i .
- Αν οι άξονες z_{i-1}, z_i τέμνονται (όταν προεκταθούν), τότε η αρχή του Σ.Σ. $\{i\}$ είναι το σημείο τομής των αξόνων x_i, z_i .
- Αν οι άξονες z_{i-1}, z_i τέμνονται (όταν προεκταθούν), τότε η αρχή του Σ.Σ. $\{i\}$ είναι το σημείο τομής των αξόνων z_{i-1}, z_i . Η φορά του άξονα x_i , επιλέγεται (από τις δύο (πιθανές) αυθαίρετα. Στην περίπτωση αυτή, η παράμετρος d_i (βλέπε βήμα 4) είναι 0.
- Αν οι άξονες z_{i-1}, z_i είναι παράλληλοι, τότε δεν υπάρχει μονοσήμαντα ορισμένη κάθετο σε αυτούς. Στην περίπτωση αυτή, ο άξονας x_i τοποθετείται αυθαίρετα, αρκεί να είναι κάθετος στον z_i (με φορά από τον z_{i-1} προς τον z_i εφόσον είναι εφικτό).
- Αν οι άξονες z_{i-1}, z_i είναι συγγραμμικοί, τότε αρκεί ο άξονας x_i να είναι κάθετος στον z_i . Η αρχή του Σ.Σ. $\{i\}$ μπορεί να τοποθετηθεί οπουδήποτε πάνω στον άξονα z_i .
- Ο άξονας x_i πρέπει πάντα να τέμνει τον άξονα z_{i-1} όταν $i < N$.
- Ο άξονας y_i απλά συμπληρώνει το δεξιόστροφο Σ.Σ.

Βήμα 4

- Υπολογισμός των παραμέτρων θ_i , r_i , d_i , a_i και η συμπλήρωση του πίνακα παραμέτρων Denavit – Hartenberg.
- Με τη διαδικασία αυτή, το Σ.Σ. $\{i - 1\}$ (κινούμενο), ταυτίζεται με το Σ.Σ. $\{i\}$ (στιγμαιαία ακίνητο).
- θ_i : Είναι η γωνία στροφής ως προς τον άξονα z_{i-1} έτσι ώστε ο άξονας x_{i-1} να γίνει παράλληλος (και με την ίδια φορά) με τον άξονα x_i . Η στροφή γίνεται σύμφωνα με τον κανόνα του δεξιού χεριού (αν η στροφή είναι δεξιόστροφη, τότε λαμβάνεται ως θετική, αν είναι αριστερόστροφη, τότε λαμβάνεται ως αρνητική). Αν η άρθρωση i είναι περιστροφική, η γωνία θ_i είναι παράμετρος της άρθρωσης. Αν είναι πρισματική, είναι μια σταθερά ή 0.
- R_i : Είναι η ελάχιστη (κάθετη) απόσταση των αξόνων x_{i-1} και x_i κατά τον άξονα z_{i-1} . Αν η άρθρωση i είναι πρισματική, η απόσταση r_i είναι η παράμετρος της άρθρωσης. Αν είναι περιστροφική, είναι μια σταθερά, όχι απαραίτητα 0.
- D_i : Είναι η ελάχιστη (κάθετη) απόσταση των αξόνων z_{i-1} και z_i κατά μήκος του άξονα x_i . Υπολογίζεται επίσης ως η απόσταση μεταξύ των σημείων:
 - (α) τομής των αξόνων z_{i-1} , x_i
 - (β) της αρχής του Σ.Σ. $\{i\}$
- Για τεμνόμενους άξονες z_{i-1} , z_i ή για πρισματική άρθρωση, $d_i = 0$
- A_i : Είναι η γωνία στροφής ως προς τον άξονα x_i , έτσι ώστε ο άξονας z_{i-1} να γίνει παράλληλος (και με την ίδια φορά) με τον άξονα z_i . Η στροφή γίνεται σύμφωνα με τον κανόνα του δεξιού χεριού (αν η στροφή είναι δεξιόστροφη, τότε λαμβάνεται ως θετική, αν είναι αριστερόστροφη, τότε λαμβάνεται ως αρνητική) [20].

3.3 ΚΙΝΗΜΑΤΙΚΗ ΑΝΑΛΥΣΗ ΤΟΥ ΡΟΜΠΟΤΙΚΟΥ ΒΡΑΧΙΟΝΑ ΤΥΠΟΥ SCARA

Η διάταξη ενός τυπικού βραχίονα Scara παρουσιάζεται στο σχήμα 3.1



Σχήμα 3.1 Ο ρομποτικός βραχίονας τύπου Scara

Ο πίνακας D-H που προκύπτει για τον παραπάνω ρομποτικό βραχίονα είναι ο ακόλουθος:

i	a_i	d_i	α_i	θ_i
1	0	d_1	$\pi/2$	θ_1
2	a_2	0	0	θ_2
3	0	0	$-\pi/2$	θ_3

Κάθε κίνηση του ρομποτικού βραχίονα αναλύεται σε τέσσερις βασικές κινήσεις, οι οποίες περιγράφονται από τους αντίστοιχους πίνακες ως εξής [21]:

Περιστροφή του βραχίονα $i-1$ γύρω από τον άξονα z_{i-1} κατά γωνία θ_i

$$\text{Rot}_{z,\theta} = \begin{bmatrix} \cos \theta_1 & -\sin \theta_1 & 0 & 0 \\ \sin \theta_1 & \cos \theta_1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Μετατόπιση του βραχίονα $i-1$ κατά μήκος του άξονα z_{i-1} κατά απόσταση d_i

$$\text{Trans}_{z,d} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Μετατόπιση του ήδη στρεφόμενου άξονα x_{i-1} (συμπίπτει με τον x_i) κατά μήκος της κοινής καθέτου a_i

$$\text{Trans}_{x,a} = \begin{bmatrix} 1 & 0 & 0 & a_i \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Περιστροφή του βραχίονα $i-1$ γύρω από τον άξονα x_i κατά γωνία a_i

$$\text{Rot}_{x,a} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos a_i & -\sin a_i & 0 \\ 0 & \sin a_i & \cos a_i & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Οι πίνακες που προκύπτουν είναι οι ακόλουθοι:

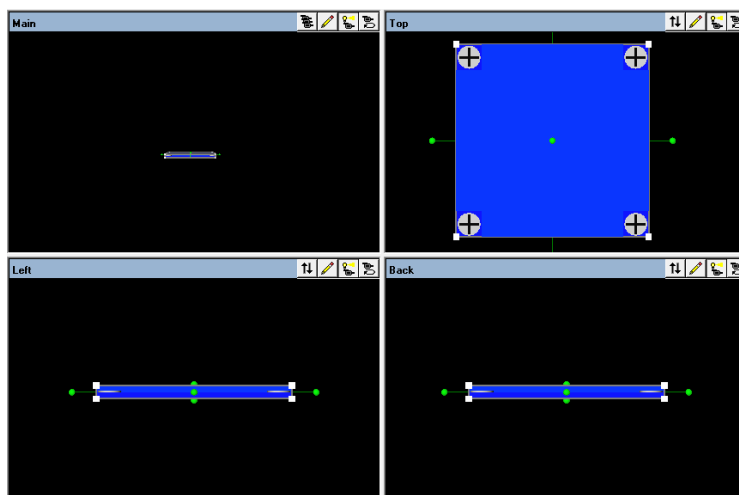
$$\begin{aligned}
 T_{01} &= \begin{bmatrix} \cos(\theta_1) & 0 & \sin(\theta_1) & 0 \\ 0 & 1 & 0 & L_1 \\ -\sin(\theta_1) & 0 & \cos(\theta_1) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \\
 T_{1A} &= \begin{bmatrix} \cos(\theta_2) & 0 & \sin(\theta_2) & L_2 \\ 0 & 1 & 0 & 0 \\ -\sin(\theta_2) & 0 & \cos(\theta_2) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \\
 T_{A2} &= \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & x \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \\
 T_{23} &= \begin{bmatrix} \cos(\theta_3) & 0 & \sin(\theta_3) & L_3 \\ 0 & 1 & 0 & 0 \\ -\sin(\theta_3) & 0 & \cos(\theta_3) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \\
 T_{34} &= \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & d \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}
 \end{aligned}$$

Για τον υπολογισμό του συνολικού πίνακα ομογενούς μετασχηματισμού έχουμε:

$${}^0T_n = {}^0T_1 {}^1T_2 \dots {}^{n-1}T_n$$

3.4 ΣΧΕΔΙΑΣΜΟΣ ΒΑΣΗΣ

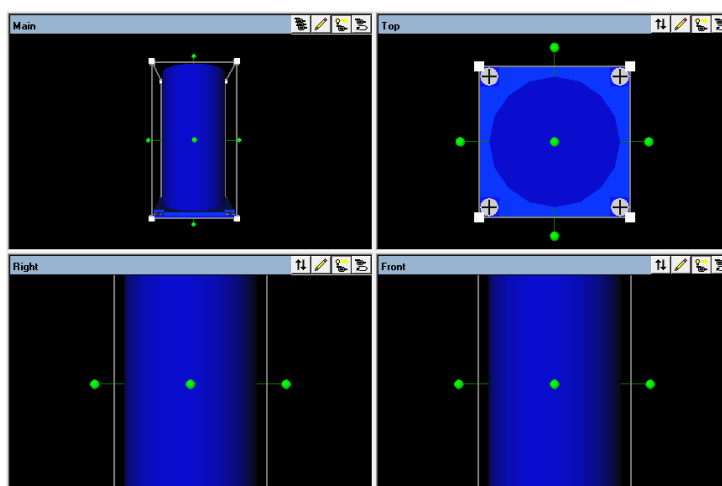
Για τον σχεδιασμό της βάσης επιλέξαμε απο το v-realm builder δυο τετράγωνα όπου τοποθετήθηκαν έτσι ώστε να υπάρχει κενό για να μπορούν να μπουν οι βίδες στήριξης της. Ο σχεδιασμός επιλέχθηκε έτσι ώστε οι βίδες να αντιπροσωπεύουν την πραγματική στήριξη που θα χρειαζόταν το ρομπότ για να έχει υπόσταση στον πραγματικό κόσμο.)



Σχήμα 3.2 Σχεδιασμός βάσης

3.5 ΣΧΕΔΙΑΣΜΟΣ ΒΑΣΗΣ 2

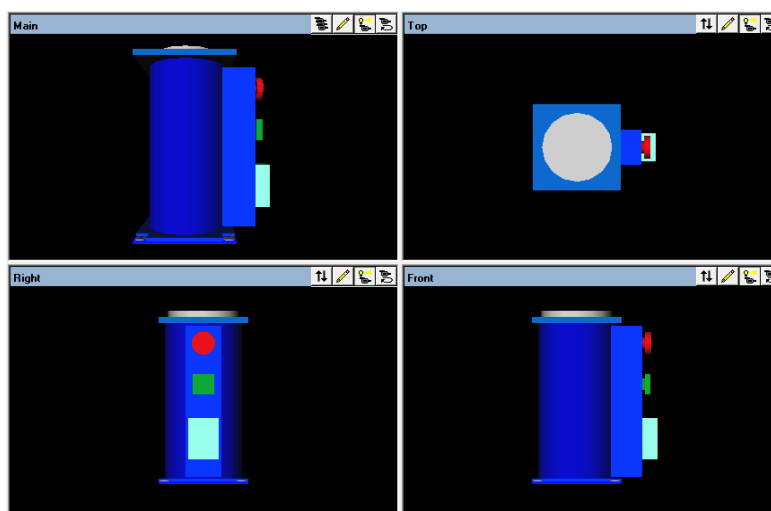
Αφού φτιάξαμε την βάση επικεντρωθήκαμε στην συνέχεια αυτής όπου θα ορθώνεται κάθετα στην προηγούμενη, κάθετα στον βραχίονα και παράλληλα στο άκρο εργασίας. Έτσι πήραμε έναν κύλινδρο και τον τοποθετήσαμε πάνω στο τετράγωνο που φτιάξαμε προηγουμένως. (Σχήμα 3.3)



Σχήμα 3.3 Σχεδιασμός βάσης 2

3.6 ΣΧΕΔΙΑΣΜΟΣ ΚΟΥΜΠΙΩΝ

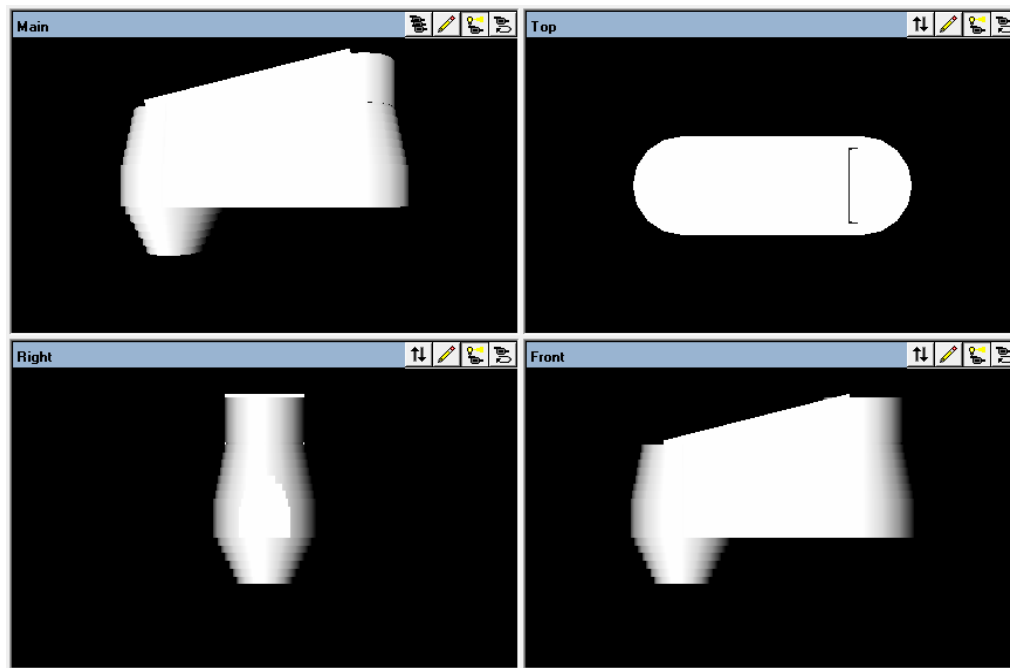
Για την πραγματική αναπαράσταση όπως αναφέραμε και παραπάνω θα πρέπει να υπάρχουν και κουμπιά σε κάποιο σημείο που θα μπορεί ο χειριστής του βραχίονα εύκολα να δίνει εντολές. Έτσι πάνω στο προηγούμενο σχέδιο του κυλίνδρου βάλουμε 2 τετράγωνα και έναν κύλινδρο που τα χρωμάτισαμε αναλόγως έτσι ώστε να φαίνεται η διαφορά καθώς και υπήρξε η ανάλογη ρύθμιση του μεγέθους. (Σχήμα 3.4)



Σχήμα 3.4 Σχεδιασμός βάσης

3.7 ΣΧΕΔΙΑΣΜΟΣ ΒΡΑΧΙΟΝΑ

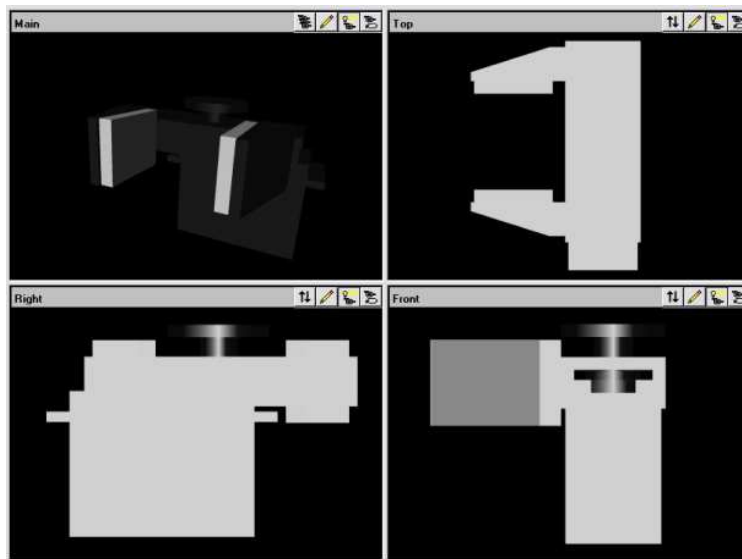
Ο βραχίονας είναι ο συνδετικός κρίκος μεταξύ της βάσης και του άκρου εργασίας. Με την σειρά του ο βραχίονας μπορεί και δίνει κατεύθυνση στον χώρο και τοποθετεί περιστροφικά το άκρο εργασίας σε όποιο σημείο του χώρου εργασίας επιθυμούμε. Για την δημιουργία του βάλαμε 2 κυλίνδρους παράλληλα όπου κάθετα τους και ανάμεσα τους μίγχε ένα τετράγωνο όπου έδωσε ομογένεια στο σχήμα μας. Είναι ένας τρόπος μίξης αντικειμένων που επιτρέπει το λογισμικό για να μας διευκολύνει στην απλούστευση σχημάτων. (Σχήμα 3.5)



Σχήμα 3.5 Σχεδιασμός βραχίονα

3.8 ΣΧΕΔΙΑΣΜΟΣ ΑΚΡΟΥ ΕΡΓΑΣΙΑΣ

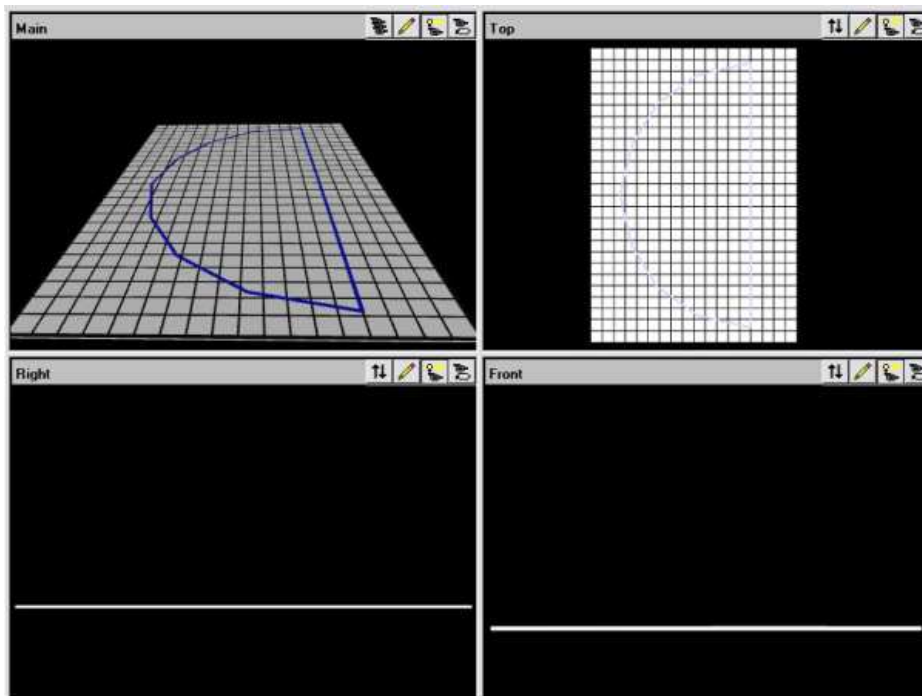
Το άκρο εργασίας αποτελούμενο από μια αρπάγη όπου μπορεί να ανοιγοκλείνει είναι το τελευταίο κομμάτι του ρομποτικού βραχίονα. Για την κατασκευή του χρειάστηκε μια πληθώρα διάφορων σχημάτων από το λογισμικό όπου τοποθετήθηκαν κατάλληλα έτσι ώστε να έρθει στην τελική του μορφή. (Σχήμα 3.6)



Σχήμα 3.6 Σχεδιασμός άκρου εργασίας

3.9 ΣΧΕΔΙΑΣΜΟΣ ΔΑΠΕΔΟΥ

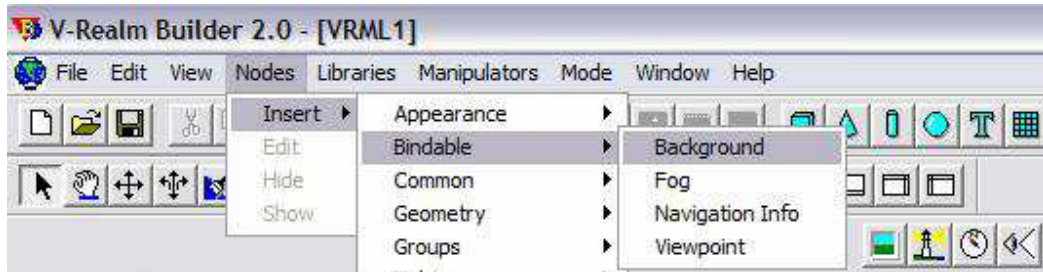
Για το σχεδιασμό του δαπέδου του βραχίονα, δημιουργείται μια επιφάνεια που χωρίζεται σε τετράγωνα πέντε εκατοστών για να αναπαριστάται με μεγαλύτερη ακρίβεια, η κίνηση του βραχίονα πάνω στο δάπεδο. Στο δάπεδο τοποθετήθηκε και ένα ημικύκλιο που δείχνει τα όρια της έκτασης του βραχίονα. (Σχήμα 3.7)



Σχήμα 3.7 Σχεδιασμός δαπέδου

3.10 ΣΧΕΔΙΑΣΜΟΣ ΦΟΝΤΟΥ

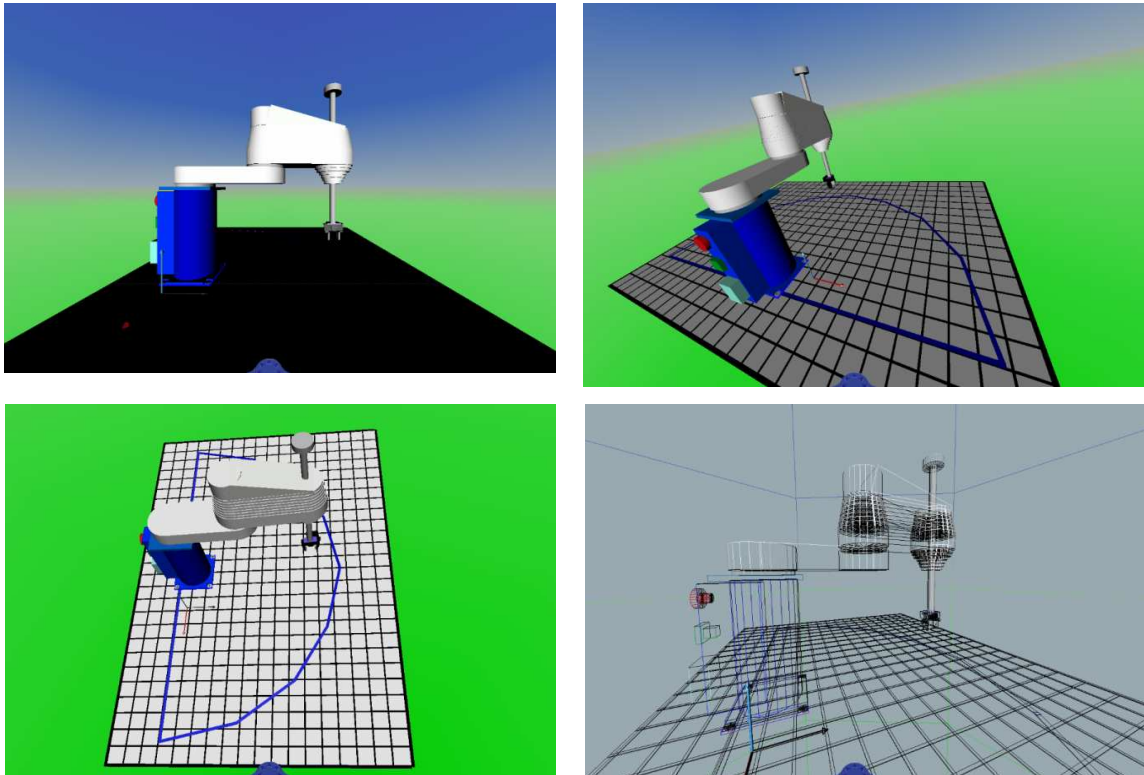
Για τον ορισμό φόντου του βραχίονα, επιλέγεται από την γενική εργαλειομπάρα (Common Node ToolBar) (Σχήμα 3.8), το εικονίδιο “Insert Background” (εισαγωγή φόντου) και αυτόματα εισάγεται φόντο, με αποχρώσεις ουρανού και Γής (μπλε και πράσινο).



Σχήμα 3.8 Σχεδιασμός δαπέδου

3.11 ΣΥΝΟΛΙΚΗ ΕΙΚΟΝΑ ΡΟΜΠΟΤΙΚΟΥ ΒΡΑΧΙΟΝΑ ΤΥΠΟΥ SCAR

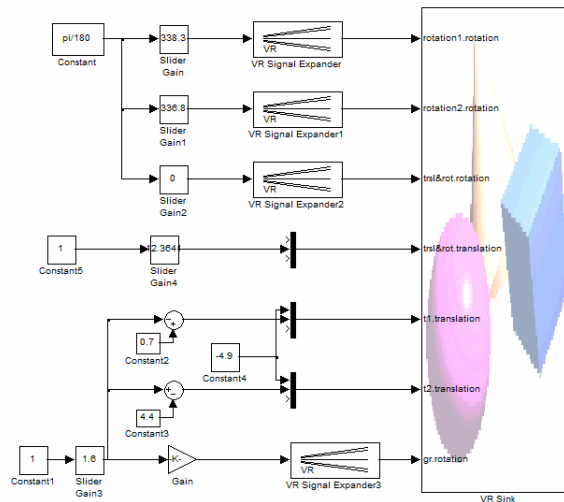
Αφού κατασκευάσαμε ένα ένα τα κομμάτια για την κατασκευή του βραχίονα και αφού συνδέθηκαν σωστά μπορούμε να δούμε την τελική μορφή όπως παρουσιάζεται παρακάτω σχήματα Σχήμα 3.9.



Σχήμα 3.9 Το εικονικό μοντέλο του Ρομποτικού Βραχίονα

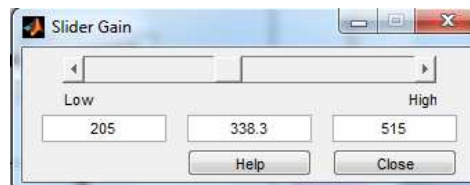
3.12 ΔΙΑΣΥΝΔΕΣΗ ΤΟΥ ΕΙΚΟΝΙΚΟΥ ΜΟΝΤΕΛΟΥ ΜΕ SIMULINK

Προκειμένου να κινήσουμε το εικονικό μοντέλο θα πρέπει να διασυνδεθεί με το λογισμικό simulink (όπως αναφέραμε και σε προηγούμενο παράδειγμα στο κεφάλαιο Παράγραφος 2.7). Έτσι θα μπορεί να ελεγχθεί κάθε κίνηση του ρομπότ είτε περιστροφική είτε πρισματική ακόμα και το άνοιγμα και κλείσιμο της αρπάγης. Το μοντέλο που αναπτύχθηκε σε περιβάλλον Simulink, παρουσιάζεται στο σχήμα 3.10.



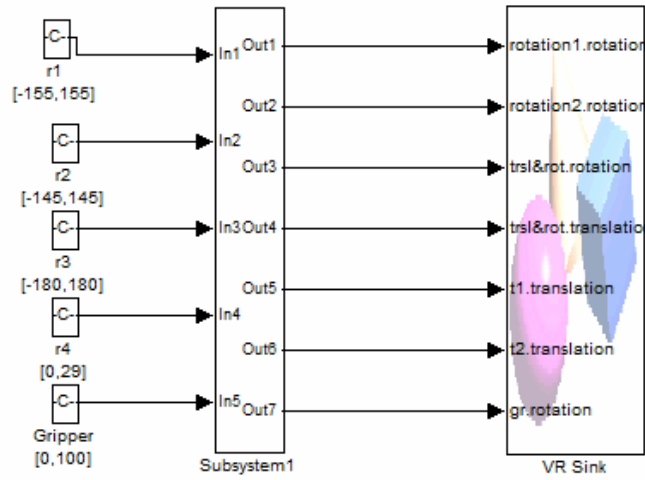
Σχήμα 3.10 Μοντέλο Simulink για τον έλεγχο του ρομποτικού βραχίονα

Πατώντας με διπλό κλικ πάνω σε κάθε slider gain ανοίγει ένα παράθυρο με μια μπάρα ολίσθησης (σχήμα 3.12) όπου ολισθαίνοντας την παίρνει διάφορες τιμές της περιστροφής η της πρισματικής ακόμα και την κίνηση της αρπάγης. Στο σχήμα παρακάτω φαίνεται ένα τέτοιο παράθυρο.

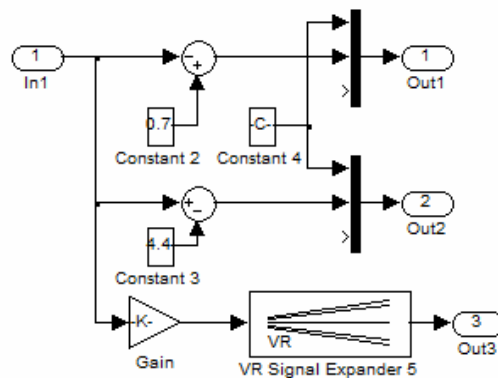


Σχήμα 3.11 Μπάρα ολίσθησης

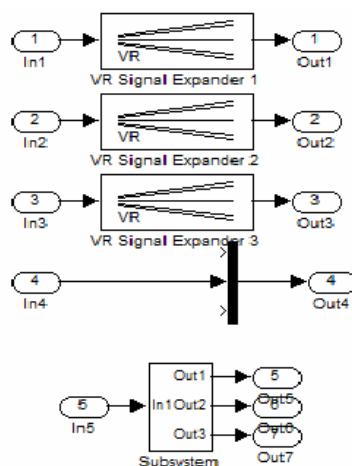
Για την απλοποίηση των συστημάτων μας το Simulink μας δίνει την δυνατότητα να συμπύξουμε το μοντέλο μέσω υποσυστημάτων για να γίνει ευκολότερη η μελέτη του μοντέλου. Ακόμη αυτο εξυπηρετεί στην διευκόλυνση μας στην κατασκευή γραφικού περιβάλλοντος διέπαφης που θα μελετήσουμε παρακάτω. Έτσι όπως βλέπουμε στα παρακάτω σχήματα 3.12-3.13-3.14α απλοποιούμε τις μορφές σε 3 υποσυστήματα που συπύσσονται σε ένα υποσύστημα σχήμα 3.14β.



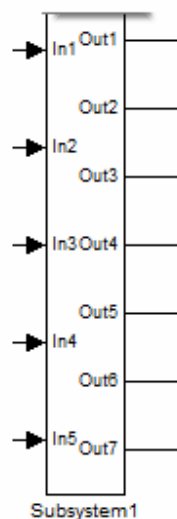
Σχήμα 3.12-Υποσύστημα 1



Σχήμα 3.13-Υποσύστημα 2



Σχήμα 3.14α-Υποσύστημα 3

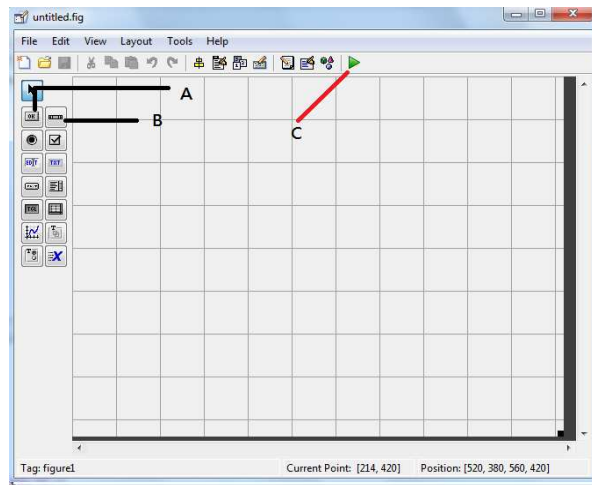


Σχήμα 3.14β Υποσύστημα 1 (απομονωμένο)

Προκειμένου η χρήση του βραχίονα να είναι φιλικότερη στο χρήστη και να μπορεί να χρησιμοποιηθεί ως μέρος εργαστηριακών ασκήσεων αναπτύχθηκε ένα γραφικό περιβάλλον διεπαφής (Graphical User Interface - GUI). Οι δυνατότητες του καθώς και ο τρόπος σχεδιασμού και ανάπτυξης θα περιγραφούν αναλυτικά στη συνέχεια.

3.13 ΑΝΑΠΤΥΞΗ ΓΡΑΦΙΚΟΥ ΠΕΡΙΒΑΛΛΟΝΤΟΣ ΔΙΕΠΑΦΗΣ

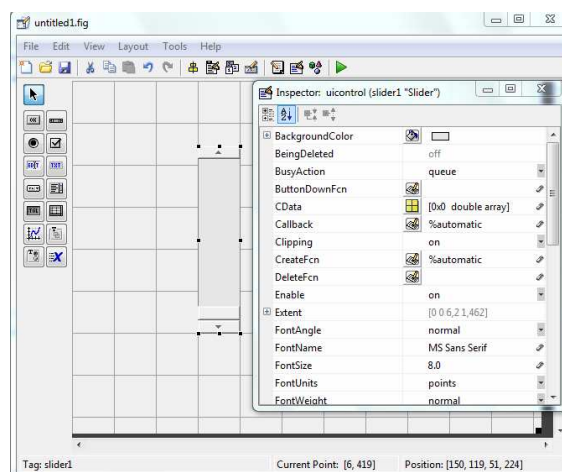
Προκειμένου να αναπτύξουμε το γραφικό περιβάλλον διεπαφής για τον έλεγχο του ρομποτικού βραχίονα, χρησιμοποιήσαμε το GUI builder του λογισμικού MATLAB. Το



Σχήμα 3.16 Εργαλειοθήκη του GUI

Ενδεικτικά παρουσιάζονται (σχήμα 3.16) το κουμπί A (προσαρμόζεται ανάλογα με τις λειτουργίες που είναι επιθυμητό να πραγματοποιηθούν). Το B είναι μια μπάρα ολίσθησης που παίρνει μια μέγιστη και μια ελάχιστη τιμή και αναλόγως το βήμα που θα δώσουμε κινείται πάνω η κάτω. Ανάλογες λειτουργίες έχουν και τα υπόλοιπα κουμπιά.

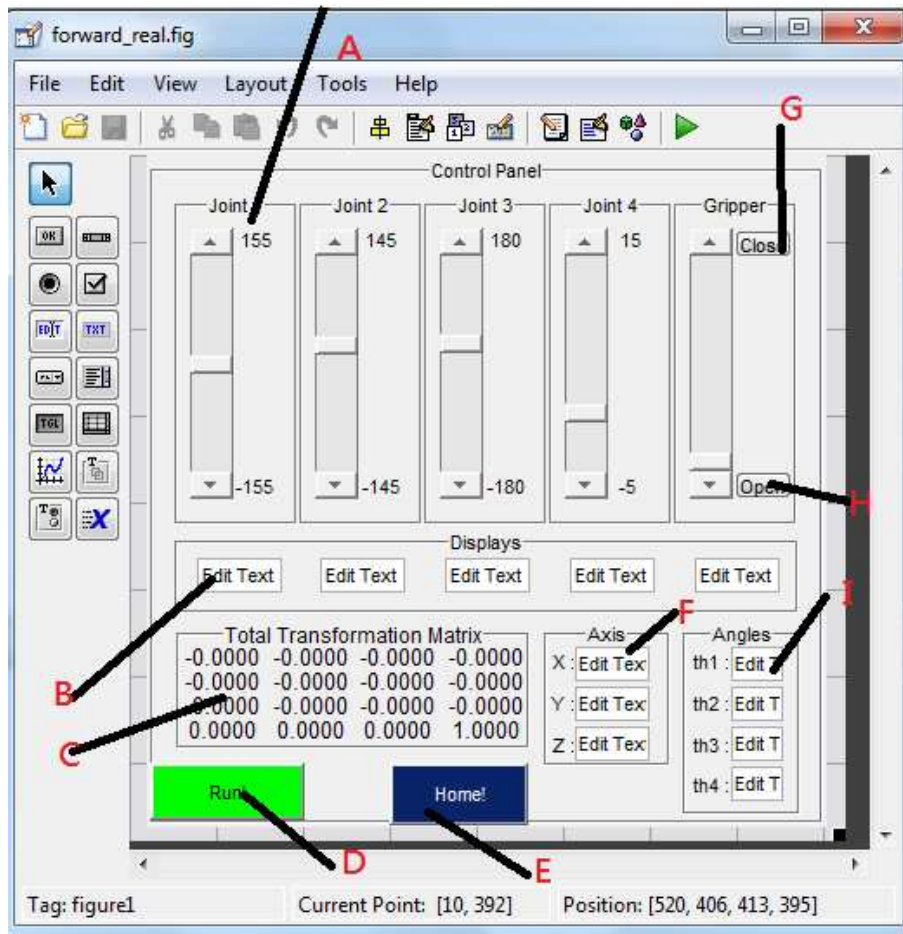
Όταν ολοκληρώσουμε το σχεδιασμό, πατάμε το C και γίνεται έλεγχος λειτουργίας και ο αντίστοιχος κώδικας που θα κάνει την λειτουργία αποθηκεύεται σε ένα αρχείο τύπου *.m*. Προκειμένου να λειτουργήσει σωστά το GUI, θα πρέπει να έχουν γίνει όλες οι αντιστοιχίες δηλώσεις των μεταβλητών του Simulink μέσα στο αρχείο τύπου *.m*. Στο σχήμα 3.17 παρουσιάζεται ενδεικτικά το menu μέσω του οποίου γίνεται επεξεργασία της μπάρας ολίσθησης. Αντίστοιχα menu υπάρχουν και για όλα τα κουμπιά της εργαλειοθήκης του GUI. Στο σχήμα 3.18 παρουσιάζεται, ενδεικτικά μέρος του κώδικα που αναπτύσσεται αυτόματα και συνοδεύει το GUI.



Σχήμα 3.17 Επεξεργασία μπάρας ολίσθησης

3.14 ΓΡΑΦΙΚΟ ΠΕΡΙΒΑΛΛΟΝ ΔΙΕΠΑΦΗΣ ΓΙΑ ΤΟΝ ΕΛΕΓΧΟ ΤΟΥ ΡΟΜΠΟΤΙΚΟΥ ΒΡΑΧΙΟΝΑ ΤΥΠΟΥ SCARA

Για τον έλεγχο του βραχίονα που αναπτύχθηκε στα πλαίσια της παρούσας εργασίας αναπτύχθηκε το γραφικό περιβάλλον διεπαφής που παρουσιάζεται στο σχήμα 3.19.



Σχήμα 3.19 Ολοκληρωμένη εικόνα του GUI

Οι λειτουργίες του γραφικού περιβάλλοντος αναλύονται συνοπτικά στη συνέχεια. Συγκεκριμένα στο σχήμα 3.19 με:

A: Περιγράφονται οι μπάρες ολίσθησης, που είναι υπεύθυνες για την κίνηση του βραχίονα στο χώρο.

B: Στα σημεία αυτά εμφανίζονται οι τιμές που παίρνουν οι μπάρες ολίσθησης.

C: Στο σημείο αυτό εμφανίζεται ο ο τελικός πίνακας ομογενούς μετασχηματισμού, όπως αυτός υπολογίζεται από τον αντίστοιχο κώδικα.

D: RUN! είναι υπεύθυνο για την εκτέλεση της κίνησης του βραχίονα και την τοποθέτηση του στην αντίστοιχη θέση.

E: Το πλήκτρο HOME! είναι υπεύθυνο για την τοποθέτηση του βραχίονα στην αντίστοιχη θέση.

F: Στο σημείο αυτό εμφανίζει η τρέχουσα τιμή του άκρου εργασίας.

G-H: Τα δύο πλήκτρα αυτά θέτουν την αρπάγη σε ανοιχτή ή κλειστή θέση

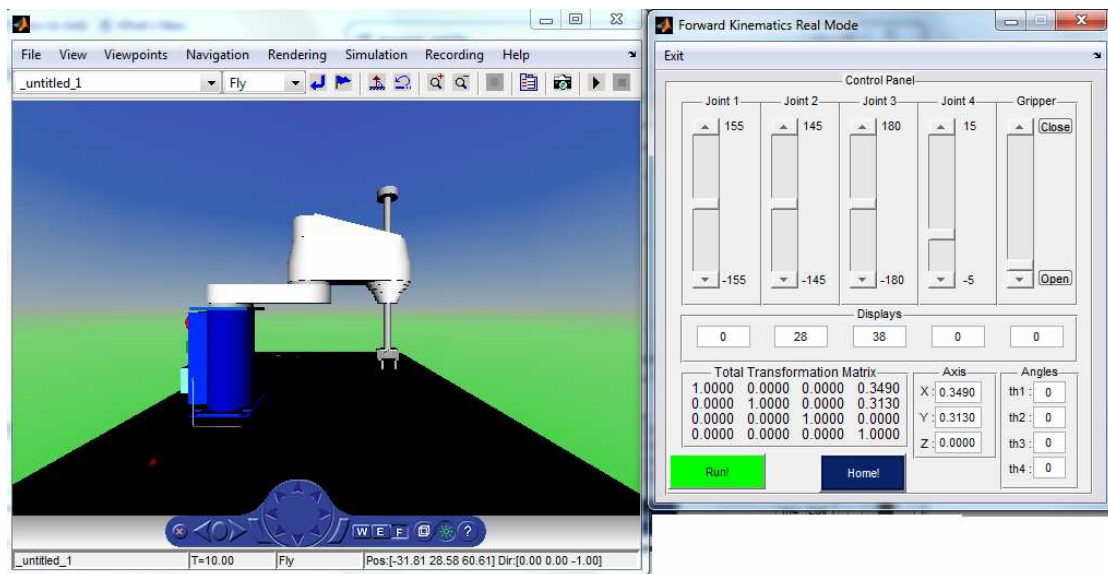
I: Εμφανίζει τις τρέχουσες τιμές των αρθρώσεων.

ΚΕΦΑΛΑΙΟ 4

Ενδεικτικά Σενάρια λειτουργίας

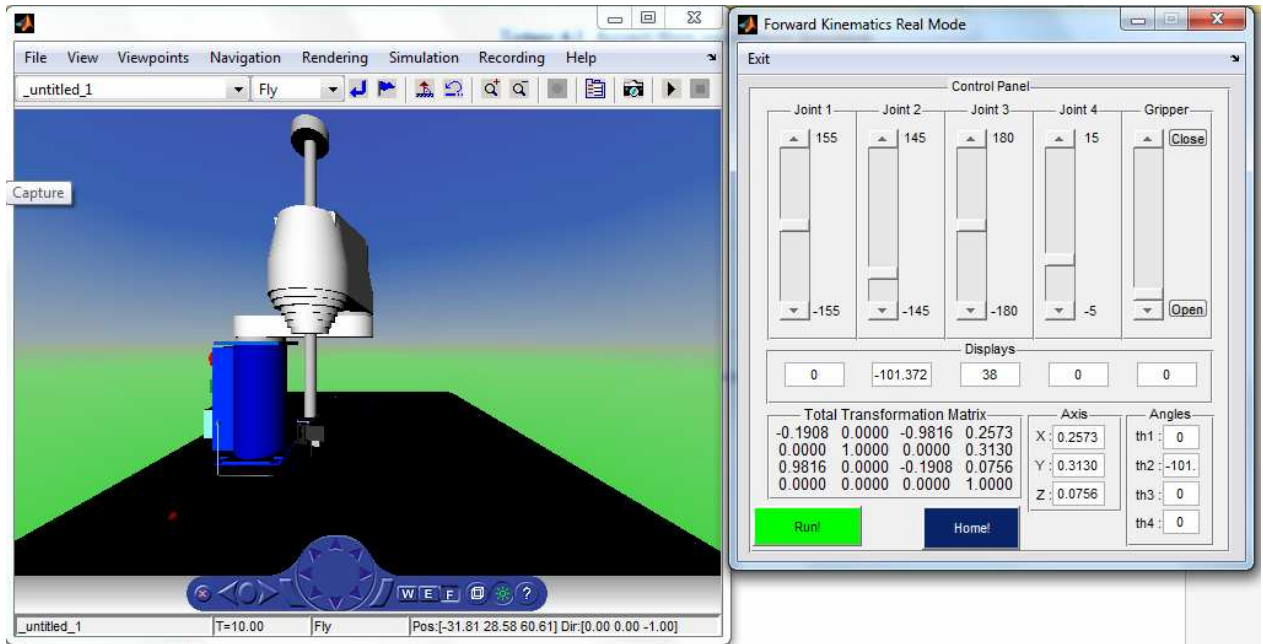
4.1 ΣΕΝΑΡΙΑ ΛΕΙΤΟΥΡΓΙΑΣ

Στο κεφάλαιο αυτό θα δούμε μερικά σενάρια λειτουργίας που παρουσιάζονται με την βοήθεια του εικονικού ρομποτικού βραχίονα που δημιουργήσαμε. Αρχικά θέτουμε το ρομπότ στην θέση HOME! πιέζοντας το πλήκτρο που βρίσκεται πάνω στο GUI στα δεξιά του σχήματος 4.1



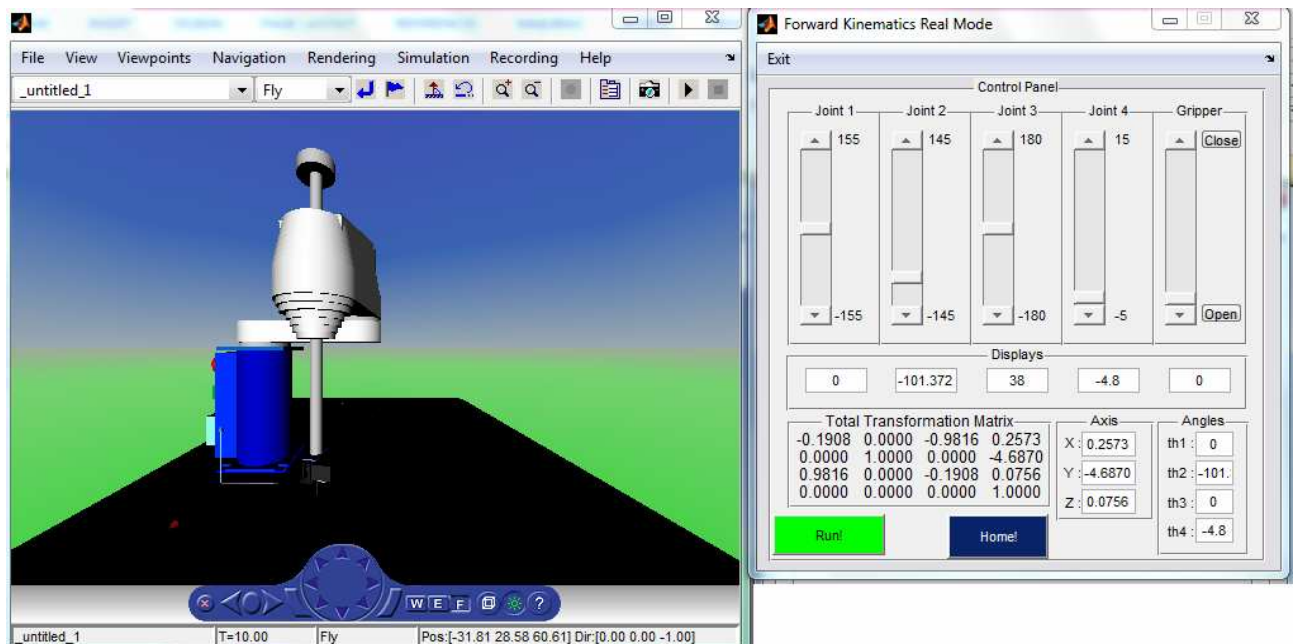
Σχήμα 4.1 Αρχική θέση ρομποτικού βραχίονα (θέση Home)

Εάν τώρα θέλουμε να δώσουμε κίνηση στο ρομποτικό βραχίονα μπορούμε να κινήσουμε μια μπάρα ολίσθησης (joint 2). Θα δούμε τώρα στο σχήμα 4.2 πως κινήθηκε ο βραχίονας στον χώρο. Κάτω απο το αντίστοιχο joint που κινήσαμε βλέπουμε και σε μορφή αριθμού τις μοίρες τις οποίες κινήθηκε.



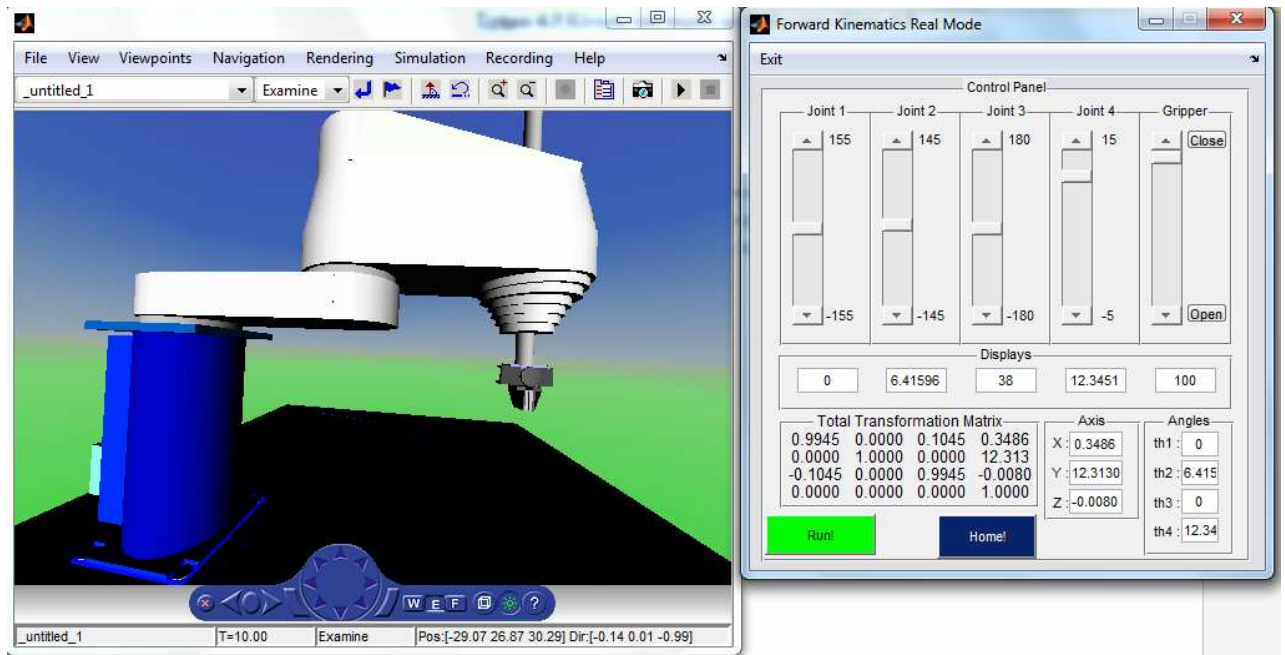
Σχήμα 4.2 Κίνηση της άρθρωσης 2 κατα -101.372 μοίρες

Στην συνέχεια εάν πάμε στην άρθρωση 4 θα κινηθεί το άκρο εργασίας κατακόρυφα. Το άκρο εργασίας έχει κινηθεί προς τα κάτω (δάπεδο) κίνηση που επιτρέπει να προσανατολιστεί θεωρητικά σε κάποιο αντικείμενο που θέλουμε να προσεγγίσουμε.



Σχήμα 4.3 Κίνηση του άκρου εργασίας στην θέση -4.8

Τέλος θα δούμε πώς η αρπάγη του βραχίονα έχει την δυνατότητα να κλείνει και να ανοίγει (Σχήμα 4.4).



Σχήμα 4.4 Σύμπτυξη του άκρου εργασίας και της αρπάγης

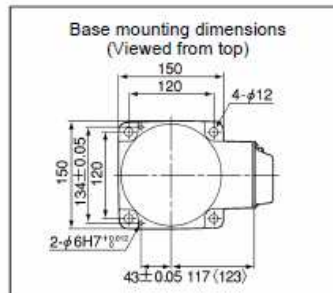
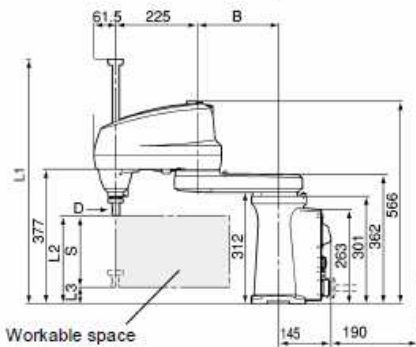
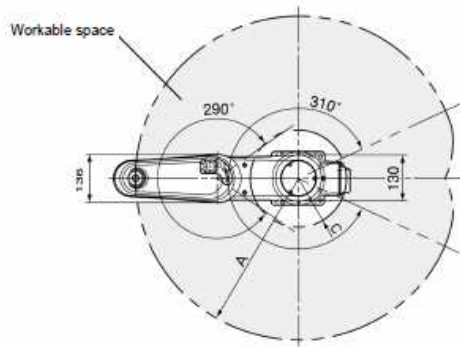
BIBΛΙΟΓΡΑΦΙΑ

- [1] Wikipediabots(url)<http://el.wikipedia.org/wiki/%CE%A1%CE%BF%CE%BC%CF%80%CF%8C%CF%84>
- [2] Εισαγωγικές σημειώσεις ρομποτικής
(courseware.mech.ntua.gr/ml23419/robotics_pdf/intro.pdf)
- [3] Γκανάς Γιάννης, Βαρτζής Τριαντάφυλλος, “Ανάπτυξη ρομποτικών συστημάτων με το πακέτο εφαρμογής Webot”, Πτυχιακή εργασία, Α.Τ.Ε.Ι. Κρήτης, 2009
- [4] Robotics & Automation Magazine, IEEE, “Review of the Robotica software package for robotic manipulators”, 1994
- [5] Robotica Control Software,
([url:http://www.robotica.co.uk/robotica/ramc/products/servo_control/servo_software.htm](http://www.robotica.co.uk/robotica/ramc/products/servo_control/servo_software.htm))
- [6] Robotics Studio, από Wikipedia, ([url: http://en.wikipedia.org/wiki/Robotics_Studio](http://en.wikipedia.org/wiki/Robotics_Studio))
- [7] The Player Project, ([url:http://playerstage.sourceforge.net/](http://playerstage.sourceforge.net/))
- [8] Virtual Reality Modelling Language (VRML),
([url:http://www.it.uom.gr/project/MultimediaTechnologyNotes/extra/append9_3.htm](http://www.it.uom.gr/project/MultimediaTechnologyNotes/extra/append9_3.htm))
- [9] Φοίβος-Απόστολος Μυλωνάς, “Σχεδιασμός & Υλοποίηση VRML Browser σε Java”, Διπλωματική εργασία, Εθνικό και Καποδιστριακό Πανεπιστήμιο Αθηνών, 2003
- [10] Εισαγωγή στην Virtual Reality Modelling Language (VRML),
([url:http://alexandra.di.uoa.gr/mmtech/VirtualReality/eisagvgh.pdf](http://alexandra.di.uoa.gr/mmtech/VirtualReality/eisagvgh.pdf))
- [11] Μαυραντζάς Νικόλαος, “Οι τεχνολογίες 3D στην τάξη και παραδείγματα ενσωμάτωσης στη διδασκαλία χρησιμοποιώντας την γλώσσα VRML”, 4ο Συνέδριο στη Σύρο - ΤΠΕ στην Εκπαίδευση, 2007.
- [12] VRML, από Wikipedia, ([url:http://en.wikipedia.org/wiki/VRML](http://en.wikipedia.org/wiki/VRML))
- [13] 3DMLW, από Wikipedia, ([url:http://en.wikipedia.org/wiki/3DMLW](http://en.wikipedia.org/wiki/3DMLW))
- [14] COLLADA, από Wikipedia, ([url:http://en.wikipedia.org/wiki/COLLADA](http://en.wikipedia.org/wiki/COLLADA))
- [15] O3D, από Wikipedia, ([url:http://en.wikipedia.org/wiki/O3D](http://en.wikipedia.org/wiki/O3D))
- [16] Universal 3D, από Wikipedia, ([url:http://en.wikipedia.org/wiki/Universal_3D](http://en.wikipedia.org/wiki/Universal_3D))
- [17] X3D, από Wikipedia, ([url:http://en.wikipedia.org/wiki/X3D](http://en.wikipedia.org/wiki/X3D))
- [18] Introduction to V-Realm Builder Concepts, από το μενού help του V-Realm Builder 2.0
- [19] Installing the VRML Editor on the Host Computer: Installation (Virtual Reality Toolbox™), από το μενού help του λογισμικού MATLAB
- [20] Δ.Μ. Εμίρης, Δ.Ε. Κουλουριώτης, Ρομποτική 3η έκδοση, Αθήνα 2006.

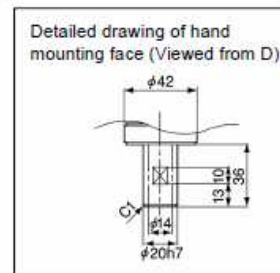
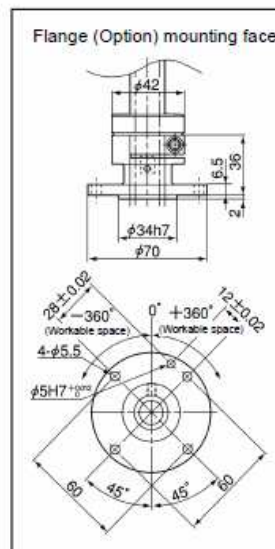
- [21] Καραστεργίου Βασιλικής του Γεωργίου- Τσίλομήτρου Ουρανίας του Αποστόλου «Ανάπτυξη εφαρμογών σε όλα τα δυνατά περιβάλλοντα προγραμματισμού του ρομπότ Katana-Neuronics», Διπλωματική Εργασία Ηλεκτρολόγων Μηχανικών και Τεχνολογίας Υπολογιστών στις Πάτρα, Ιούλιος 2010
- [22] Θωμάς Σακάρος, “Κατασκευή και έλεγχος κατακόρυφου αρθρωτού ρομποτικού βραχίονα τεσσάρων βαθμών ελευθερίας”, Πτυχιακή εργασία, Α.Τ.Ε.Ι Χανίων 2009.
- [23] Δημήτριος Τσόντος “Ανάπτυξη εκπαιδευτικού λογισμικού για προσομοίωση και έλεγχο πραγματικού ρομποτικού βραχίονα τεσσάρων βαθμών ελευθερίας”, Πτυχιακή εργασία, Α.Τ.Ε.Ι Χανίων 2009

ΠΑΡΑΡΤΗΜΑ

Σχηματικό διαστάσεων του ρομποτικού βραχίονα



Unit : mm



Model	A	B	C
HS-4545*G	450	225	136

Z axis stroke : S(mm)	Type	L1	L2	L3
*=2 : 200	standard	697	246	46
*=3 : 320	standard	817	246	-74 (note1)

Note1 : In case of the Z axis 320mm stroke type, the Z axis in the lowermost position may reach lower than the base mounting face