



**ΤΕΧΝΟΛΟΓΙΚΟ ΕΚΠΑΙΔΕΥΤΙΚΟ  
ΙΔΡΥΜΑ ΚΡΗΤΗΣ  
ΤΜΗΜΑ ΗΛΕΚΤΡΟΝΙΚΗΣ**

**ΕΚΠΑΙΔΕΥΤΙΚΗ ΡΟΜΠΟΤΙΚΗ  
(EDUCATIONAL ROBOTICS)**

Τριμελής Επιτροπή:

Ε. Δοϊτσίδης

Επίκουρος Καθηγητής

Α. Κωνσταντάρας

Επίκουρος Καθηγητής

Ν. Φραγκιαδάκης

Καθηγητής Εφαρμογών

υπό

Μυλωνά Σταύρο Α.Μ. 3311

Χανιά, 2012

## **ΕΥΧΑΡΙΣΤΙΕΣ**

Ευχαριστώ τους αληθινούς μου φίλους.-

## **ΠΕΡΙΛΗΨΗ ΠΤΥΧΙΑΚΗΣ**

Στην παρούσα εργασία γίνεται εκτενή επισκόπηση διαφόρων διαθέσιμων εργαλείων τόσο σε επίπεδο λογισμικού όσο και σε επίπεδο πραγματικών διατάξεων που δύναται να χρησιμοποιηθούν σαν επικουρικά μέσα για την διδασκαλία της ρομποτικής. Παρουσιάζονται αναλυτικά διάφοροι τύποι ρομποτικών προσομοιωτών καθώς και έτοιμα πακέτα ρομποτικής, πάντα με έμφαση στην εκπαιδευτική διαδικασία.

Γίνεται εκτενή αναφορά στον ρομποτικό προσομοιωτή Microsoft Robotics Developer Studio (MRDS) καθώς και στη δυνατότητα διασύνδεσης του με το ρομποτικό πακέτο LEGO MINDSTORMS, ενώ τέλος παρουσιάζονται ορισμένοι παραδείγματα εφαρμογών του στο MRDS.

## **ABSTRACT**

The present work is an extensive overview of various tools available both in software and in terms of actual devices that can be used as subsidiary means for the teaching of robotics. There are presented in detail, various types of robotic simulators and ready robotics kits, always emphasizing the educational process. There is an extensive reference to the robotic simulator named Microsoft Robotics Developer Studio (MRDS) and its possibility of interconnection with robotic pack LEGO MINDSTORMS, finally there are present some examples of LEGO applications in MRDS.



---

## ΠΕΡΙΕΧΟΜΕΝΑ

ΚΕΦΑΛΑΙΟ 1 .....	5
ΕΙΣΑΓΩΓΗ .....	5
1.1 ΑΝΤΙΚΕΙΜΕΝΟ ΤΗΣ ΕΡΓΑΣΙΑΣ .....	5
1.2 ΔΥΣΚΟΛΙΕΣ ΠΟΥ ΥΠΑΡΧΟΥΝ ΣΤΗΝ ΑΝΑΠΤΥΞΗ ΡΟΜΠΟΤΙΚΩΝ ΕΦΑΡΜΟΓΩΝ .....	5
ΚΕΦΑΛΑΙΟ 2 .....	7
ΡΟΜΠΟΤΙΚΟ ΛΟΓΙΣΜΙΚΟ .....	7
2.1 ΡΟΜΠΟΤΙΚΟ ΛΟΓΙΣΜΙΚΟ (ROBOTICS SOFTWARE) .....	7
2.2 ΡΟΜΠΟΤΙΚΗ ΠΡΟΣΟΜΟΙΩΣΗ (ROBOTICS SIMULATION) .....	7
2.2.1 ΠΛΕΟΝΕΚΤΗΜΑΤΑ ΤΗΣ ΡΟΜΠΟΤΙΚΗΣ ΠΡΟΣΟΜΟΙΩΣΗΣ.....	8
2.2.2 ΠΕΡΙΟΡΙΣΜΟΙ ΚΑΙ ΜΕΙΟΝΕΚΤΗΜΑΤΑ ΣΤΗΝ ΠΡΟΣΟΜΟΙΩΣΗ.....	9
2.3 ΡΟΜΠΟΤΙΚΟΣ ΠΡΟΣΟΜΟΙΩΤΗΣ (ROBOTICS SIMULATOR) .....	10
2.4 ΜΗ ΕΚΠΑΙΔΕΥΤΙΚΕΣ ΕΦΑΡΜΟΓΕΣ ΡΟΜΠΟΤΙΚΗΣ ΠΡΟΣΟΜΟΙΩΣΗΣ....	11
2.4.1 ΠΡΟΣΟΜΟΙΩΣΗ ΣΤΗ ΓΡΑΜΜΗ ΠΑΡΑΓΩΓΗΣ ΡΟΜΠΟΤ.....	11
2.4.2 ΔΙΑΣΤΗΜΙΚΗ ΠΡΟΣΟΜΟΙΩΣΗ .....	12
2.4.3 ROBOT OPERATING SYSTEM (ROS) .....	12
2.4.4 URBI.....	14
2.4.4.1 Η ΓΛΩΣΣΑ URBIScript.....	14
2.4.4.2 ΚΥΡΙΟΤΕΡΑ ΧΑΡΑΚΤΗΡΙΣΤΙΚΑ ΤΗΣ URBI [15].....	15
ΚΕΦΑΛΑΙΟ 3 .....	17
ΕΚΠΑΙΔΕΥΤΙΚΟΙ ΡΟΜΠΟΤΙΚΟΙ ΠΡΟΣΟΜΟΙΩΤΕΣ (ROBOTICS SUITES).....	17
3.1 ΕΙΣΑΓΩΓΗ .....	17
3.2 MICROSOFT ROBOTICS DEVELOPER STUDIO .....	18
3.3 WEBOTS .....	22
3.4 ANYCODE MARILOU .....	27
3.5 OPENRAVE .....	30
3.6 PLAYER PROJECT .....	32
3.6.1 PLAYER.....	33
3.6.2 STAGE.....	33

---

3.6.3 GAZEBO .....	33
3.7 NI LABVIEW ROBOTICS MODULE .....	35
3.8 CAMELOT ROPSIM.....	37
3.9 CARMEN TOOLKIT .....	38
3.10 SIMBAD .....	40
3.7 DIALOGOS .....	43
ΚΕΦΑΛΑΙΟ 4 .....	44
ΕΚΠΑΙΔΕΥΤΙΚΑ ΡΟΜΠΟΤ (ROBOTICS KITS) .....	44
4.1 ΕΙΣΑΓΩΓΗ .....	44
4.2 LEGO MINDSTORMS .....	45
4.2.1 ΕΙΣΑΓΩΓΗ .....	45
4.2.2 ΙΣΤΟΡΙΚΑ ΣΤΟΙΧΕΙΑ.....	46
4.2.3 LEGO MINDSTORMS NXT .....	47
4.2.4 LEGO MINDSTORMS NXT 2.0 .....	49
4.2.5 ΥΠΟΣΤΗΡΙΖΟΜΕΝΕΣ ΠΡΟΓΡΑΜΜΑΤΙΣΤΙΚΕΣ ΓΛΩΣΣΕΣ .....	50
4.3 MOBILE ROBOT PROGRAMMING TOOLKIT .....	51
4.4 ROBOPLUS BIOLOID .....	52
4.5 VEX ROBOTICS DESIGN SYSTEM .....	54
ΚΕΦΑΛΑΙΟ 5.....	55
MICROSOFT ROBOTICS DEVELOPER STUDIO (MRDS).....	55
5.1 ΕΙΣΑΓΩΓΗ .....	55
5.2 ΠΑΡΟΥΣΙΑΣΗ ΤΗΣ VPL (Visual Programming Language) .....	56
5.3 ΠΑΡΟΥΣΙΑΣΗ ΤΟΥ VSE (VISUAL SIMULATION ENVIROMENT).....	59
5.4 ΧΡΗΣΙΜΟΠΟΙΩΝΤΑΣ ΤΟ RDS .....	69
5.4.1 ΠΡΩΤΑ ΒΗΜΑΤΑ.....	69
5.4.2 ΒΗΜΑΤΑ ΕΚΙΝΗΣΗΣ ΕΝΟΣ ΚΟΜΒΟΥ DDS .....	69
5.4.3 ΧΡΗΣΗ BROWSER ΓΙΑ ΕΠΙΒΛΕΨΗ ΕΝΟΣ ΚΟΜΒΟΥ DSS .....	71
5.5 ΠΡΟΓΡΑΜΜΑΤΙΣΤΙΚΕΣ ΡΟΥΤΙΝΕΣ .....	73
5.5.1 ΑΝΑΛΟΓΙΚΟ ΑΙΣΘΗΤΗΡΙΟ.....	74
5.5.2 ΟΜΑΔΑ ΑΝΑΛΟΓΙΚΩΝ ΑΙΣΘΗΤΗΡΙΩΝ .....	75

---

5.5.3 ΑΡΘΡΩΤΟΣ ΒΡΑΧΙΟΝΑΣ.....	76
5.5.4 ΜΠΑΤΑΡΙΑ .....	77
5.5.5 ΑΙΣΘΗΤΗΡΙΑ ΕΠΑΦΗΣ.....	77
5.5.6 ΚΑΜΕΡΑ ΒΑΘΟΥΣ (Kinect).....	78
5.5.7 ΔΙΑΦΟΡΙΚΟΣ ΟΔΗΓΟΣ .....	80
5.5.7.1 <i>ΕΝΤΟΛΕΣ GENERIC DIFFERENTIAL DRIVE</i> .....	81
5.5.7.1.1 <i>ALL STOP</i> .....	81
5.5.7.1.2 <i>ENABLE DRIVE</i> .....	81
5.5.7.1.3 <i>SETDRIVEPOWER</i> .....	81
5.5.7.1.4 <i>SETDRIVESPEED</i> .....	82
5.5.7.1.5 <i>DRIVEDISTANCE</i> .....	82
5.5.7.1.6 <i>ROTATEDEGREES</i> .....	83
5.5.7.2 <i>ΕΙΔΙΟΠΟΙΗΣΕΙΣ ΤΗΣ ΥΠΗΡΕΣΙΑΣ GDD</i> .....	83
<i>ΕΛΕΓΧΟΜΕΝΕΣ ΚΙΝΗΣΕΙΣ</i> .....	83
<i>UPDATE</i> .....	85
<i>DRIVE DISTANCE/ROTATE DEGREES</i> .....	85
5.5.8 ΚΩΔΙΚΟΠΟΙΗΤΗΣ (ENCODER).....	86
5.5.9 ΚΙΝΗΤΗΡΑΣ.....	87
5.5.10 SONAR.....	89
5.5.11 STREAM .....	90
5.5.12 ΚΑΜΕΡΑ.....	90
5.5.13 ΑΙΣΘΗΤΗΡΙΟ ΚΑΜΕΡΑΣ.....	92
5.5.14 SICK LASER RANGE FINDER .....	93
5.6 ΧΡΗΣΙΜΕΣ ΔΡΑΣΤΗΡΙΟΤΗΤΕΣ ΤΗΣ ΓΛΩΣΣΑΣ VPL .....	94
5.6.1 ACTIVITY.....	94
5.6.2 CALCULATE.....	94
5.6.3 COMMENT .....	95
5.6.4 DATA .....	95
5.6.5 IF 96	
5.6.6 JOIN.....	96

---

5.6.7 LIST .....	97
5.6.8 LIST FUNCTIONS.....	97
5.6.9 MERGE.....	97
5.6.10 SWITCH .....	97
5.6.11 VARIABLE .....	98
5.6.12 ΤΥΠΟΙ ΔΕΔΟΜΕΝΩΝ.....	99
ΚΕΦΑΛΑΙΟ 6 .....	100
ΕΦΑΡΜΟΓΗ ΤΟΥ ΛΟΓΙΣΜΙΚΟΥ MRDS ΓΙΑ ΕΛΕΓΧΟ ΚΑΙ ΠΡΟΣΟΜΟΙΩΣΗ ΤΟΥ LEGO MINDSTORMS NXT.....	100
6.1 ΕΙΣΑΓΩΓΗ .....	100
6.2 ΧΡΗΣΙΜΟΠΟΙΩΝΤΑΣ ΤΟ MRDS ΚΑΙ ΤΑ LEGO MINDSTORMS.....	100
6.3 ΠΑΡΟΥΣΙΑΣΗ ΤΩΝ LEGO ΥΠΗΡΕΣΙΩΝ .....	101
6.4 ΡΥΘΜΙΣΗ ΤΗΣ ΥΠΗΡΕΣΙΑΣ ΤΟΥ ΚΟΥΤΙΟΥ NXT .....	104
6.5 ΡΥΘΜΙΣΕΙΣ ΔΙΑΜΟΡΦΩΣΗΣ ΣΥΣΚΕΥΩΝ .....	105
6.6 ΚΥΡΙΕΣ ΡΥΘΜΙΣΕΙΣ ΣΤΟ LEGO MINDSTORMS NXT .....	107
6.6.1 ΡΥΘΜΙΣΕΙΣ ΥΠΗΡΕΣΙΑΣ ΚΙΝΗΤΗΡΩΝ.....	107
6.6.2 ΡΥΘΜΙΣΕΙΣ ΥΠΗΡΕΣΙΑΣ ΟΔΗΓΩΝ .....	108
6.6.3 ΡΥΘΜΙΣΕΙΣ ΥΠΗΡΕΣΙΑΣ ΑΙΣΘΗΤΗΡΙΟΥ ΦΩΤΟΣ .....	110
6.6.4 ΡΥΘΜΙΣΕΙΣ ΥΠΗΡΕΣΙΑΣ ΑΙΣΘΗΤΗΡΙΟΥ ΧΡΩΜΑΤΟΣ.....	111
ΚΕΦΑΛΑΙΟ 7 .....	112
TUTORIALS .....	112
7.1 ΕΙΣΑΓΩΓΗ .....	113
7.2 ΕΚΠΑΙΔΕΥΤΙΚΟ ΠΑΡΑΔΕΙΓΜΑ ΗΛΕΚΤΡΙΚΟΥ ΦΑΚΟΥ.....	113
7.3 ΕΚΠΑΙΔΕΥΤΙΚΟ ΠΑΡΑΔΕΙΓΜΑ ΑΙΣΘΗΤΗΡΙΟΥ ΥΠΕΡΗΧΩΝ.....	121
ΒΙΒΛΙΟΓΡΑΦΙΑ .....	133
ΠΑΡΑΡΤΗΜΑ.....	137
Α. ΠΛΗΡΗΣ ΚΑΤΑΛΟΓΟΣ ΤΟΥ ΥΠΟΣΤΗΡΙΖΟΜΕΝΟΥ ΛΟΓΙΣΜΙΚΟΥ ΓΙΑ ΤΟ LEGO MINDSTORMS NXT 2.0.....	137

---

# ΚΕΦΑΛΑΙΟ 1

## ΕΙΣΑΓΩΓΗ

### 1.1 ΑΝΤΙΚΕΙΜΕΝΟ ΤΗΣ ΕΡΓΑΣΙΑΣ

Η παρούσα πτυχιακή εργασία αποτελεί εκτενή επισκόπηση διαφόρων διαθέσιμων λογισμικών ρομποτικής με έμφαση στην εκπαίδευση. Παρουσιάζονται αναλυτικά διάφοροι τύποι λογισμικού και συγκεκριμένα ρομποτικοί προσομοιωτές και έτοιμα πακέτα ρομπότ, πάντα με έμφαση στην εκπαιδευτική διαδικασία. Ε μεγαλύτερο βάθος παρουσιάζεται ο ρομποτικός προσομοιωτής Microsoft Robotics Developer Studio (MRDS) και η εφαρμογή του ρομποτικού πακέτου LEGO MINDSTORMS, ενώ τέλος παρουσιάζονται ορισμένοι οδηγοί εφαρμογών του στο MRDS.

### 1.2 ΔΥΣΚΟΛΙΕΣ ΠΟΥ ΥΠΑΡΧΟΥΝ ΣΤΗΝ ΑΝΑΠΤΥΞΗ ΡΟΜΠΟΤΙΚΩΝ ΕΦΑΡΜΟΓΩΝ

Στη βιβλιογραφία παρουσιάζονται ποικίλες δυσκολίες σχετικά με την ανάπτυξη ρομποτικών εφαρμογών [1]:

- Το κόστος και η δυσκολία εύρεσης ρομποτικού υλισμικού (Hardware)

Τα ρομποτικά κιτ, όπως το LEGO MINDSTORMS, έχουν κάνει προσβάσιμη την ρομποτική σε ένα ευρύ καταναλωτικό κοινό. Ενώ όμως αυτά τα κιτ είναι άριστα για την εισαγωγή στη ρομποτική, αν ο προγραμματιστής θελήσει να εντρυφήσει σε υψηλότερες βαθμίδες πολυπλοκότητας των ρομποτικών εφαρμογών ή στον αριθμό των ρομπότ που συμπεριλαμβάνονται σε μια εφαρμογή, το κόστος είναι απαγορευτικό.

- Η δυσκολία επίλυσης σφαλμάτων του υλισμικού

Η επίλυση σφαλμάτων υλισμικού (hardware troubleshooting), ακόμα και για συσκευές ευρείας καταναλώσεως, είναι δύσκολη. Ενώ όμως τέτοιου είδους συσκευές είναι πολύ αξιόπιστες και οι καταναλωτές δεν χρειάζεται να ανησυχούν για σφάλματα, η κατασκευή ενός ρομπότ, ειδικότερα ενός παραμετροποιημένου ρομπότ βασισμένου σε διάφορες πλατφόρμες με ειδικευμένα εξαρτήματα, χρειάζεται σημαντική δεξιότητα, χρόνο και προσπάθεια ώστε να επιλυθούν ακόμα και τα πιο απλά σφάλματα.

---

- Η δυσκολία της ταυτόχρονης χρήσης

Η ανάπτυξη ρομπότ από ομάδες ανθρώπων, όπως σε διαγωνισμούς της DARPA, έχουν γίνει συχνό φαινόμενο και μια από τις μεγαλύτερες προκλήσεις είναι ότι το ρομπότ υπό κατασκευή είναι πολύ ακριβό και μοναδικό. Αυτά τα δύο θέματα κάνουν δύσκολη την δοκιμή με άλλους χωρίς τον κίνδυνο καταστροφής του ρομπότ. Έτσι αυτές οι ομάδες δουλεύουν ταυτόχρονα απομονωμένες αναπτύσσοντας τα στοιχεία ξεχωριστά, κάνοντας δυσκολότερη την ολοκλήρωση του έργου, ενώ δημιουργούν νέα σφάλματα που είναι δύσκολο να αντιμετωπισθούν.

---

## ΚΕΦΑΛΑΙΟ 2

# ΡΟΜΠΟΤΙΚΟ ΛΟΓΙΣΜΙΚΟ

### 2.1 ΡΟΜΠΟΤΙΚΟ ΛΟΓΙΣΜΙΚΟ (ROBOTICS SOFTWARE)

Ρομποτικό λογισμικό είναι οι προγραμματισμένες εντολές που ορίζουν σε μια μηχανική συσκευή (γνωστή ως ρομπότ) τι είδους εργασίες να εκτελέσει και ελέγχουν τις πράξεις της [2]. Καθώς ο απευθείας προγραμματισμός των ρομπότ είναι περίπλοκος, έχουν αναπτυχθεί πολλά συστήματα λογισμικού με στόχο την απλοποίηση της διαδικασίας. Συνήθως χρησιμοποιούνται τεχνικές προγραμματισμού ροής δεδομένων από τους περισσότερους κατασκευαστές.

Δυστυχώς η ανάπτυξη ρομποτικού λογισμικού είναι μια επίπονη και περίπλοκη διαδικασία κυρίως διότι υπάρχει έλλειψη ενοποιημένων προτύπων στην βιομηχανία παραγωγής τέτοιου είδους λογισμικού.

### 2.2 ΡΟΜΠΟΤΙΚΗ ΠΡΟΣΟΜΟΙΩΣΗ (ROBOTICS SIMULATION)

Η προσομοίωση έχει σημαντικό ρόλο στον τομέα της ρομποτικής, καθώς επιτρέπει πειραματισμούς οι οποίοι θα ήταν διαφορετικά εξαιρετικά ακριβοί και χρονοβόροι [3]. Επιτρέπει την δοκιμή μεθοδολογιών σε μεταβαλλόμενα, δυναμικά περιβάλλοντα ενώ διευκολύνει την περισυλλογή αναδραστικών δεδομένων για τον καθορισμό της ποιότητας του συστήματος ελέγχου. Επιτρέπει επίσης την ανάπτυξη των ρομποτικών συστημάτων ελέγχου τα οποία εξαρτώνται από τυχαίες αλλαγές μέσω πολλών επαναλήψεων (πχ γενετικοί αλγόριθμοι).

Από τεχνικής απόψεως, η ακρίβεια των εφαρμογών της σύγχρονης προσομοίωσης σε επίπεδο του πραγματικού κόσμου, επιτρέπει την ανάπτυξη ενός προγράμματος το οποίο περιέχει πολύ περισσότερα από την απλή πορεία κίνησης ενός ρομπότ. Η ρομποτική προσομοίωση είναι πλέον χαμηλού κόστους και ευρέως διαδεδομένη και αυτό οφείλεται σε ένα βαθμό στα βιντεοπαιχνίδια υπολογιστή και κονσόλων που παρείχαν τα εργαλεία όσον αφορά τη φωτορεαλιστική αναπαράσταση και την προηγμένη προσομοίωση φυσικών ιδιοτήτων του πραγματικού κόσμου.

---

### 2.2.1 ΠΛΕΟΝΕΚΤΗΜΑΤΑ ΤΗΣ ΡΟΜΠΟΤΙΚΗΣ ΠΡΟΣΟΜΟΙΩΣΗΣ

Τα κυριότερα πλεονεκτήματα της ρομποτικής προσομοίωσης είναι τα εξής [4]:

- Η προσομοίωση επιτρέπει την ανάπτυξη ρομπότ ή ρομποτικών εφαρμογών, χρησιμοποιώντας έναν υπολογιστή και έτσι το κόστος μειώνεται σε τέτοια επίπεδα, ώστε ο μοναδικός παράγοντας περιορισμού είναι ο διαθέσιμος χρόνος και η φαντασία του χρήστη.
- Η προσομοίωση επιτρέπει την προσέγγιση της ανάπτυξης ενός έργου βήμα-βήμα, επιτρέποντας τους προγραμματιστές να ασχολούνται με την πολυπλοκότητα του έργου εκεί που πραγματικά χρειάζεται. Αυτό σημαίνει ότι ο προγραμματιστής μπορεί να ελέγχει για σφάλματα το ρομπότ έχοντας μόνο τις ουσιαστικές βασικές γνώσεις για το σφάλμα, κάνοντας τον έλεγχο μέσω της προσομοίωσης μια απλούστατη διαδικασία.
- Μπορούν να δημιουργηθούν πλατφόρμες νέων φυσικών μοντέλων ρομπότ μέσω της προσομοίωσης, από πολλούς μεμονωμένους προγραμματιστές ή ακόμα και κοινότητες ανάπτυξης λογισμικού όπου πολλοί χρήστες θα μπορούν να τις χρησιμοποιήσουν και να τις μορφοποιήσουν χωρίς να ανησυχούν μήπως καταστρέψουν πανάκριβα πρωτότυπα.
- Η προσομοίωση μπορεί να γίνει ένα πανίσχυρο εκπαιδευτικό εργαλείο. Ο προγραμματιστής μπορεί να επιλέξει το που θα επικεντρωθεί ώστε να αναπτύξει την απαιτούμενη πολυπλοκότητα και να ελέγξει το περιβάλλον για να μάθει δοκιμάζοντας. Μπορεί επίσης να εισάγει τελείως εικονικά στοιχεία και σενάρια τα οποία αν και δεν είναι εύκολο να πραγματοποιηθούν, όμως είναι πολύ χρήσιμα για εκμάθηση διάφορων καταστάσεων.
- Ένα ακόμα σημαντικό κομμάτι της προσομοίωσης είναι το ότι μπορεί να χρησιμοποιηθεί ως ένα εργαλείο πρόβλεψης ή ως ένα κομμάτι της μεθόδου επιβλεπόμενης μάθησης, καθώς λειτουργεί το ρομπότ. Εδώ και αρκετό καιρό, οι προγραμματιστές χρησιμοποιούν μαζί με το ρομπότ σε λειτουργία και την αντίστοιχη προσομοίωσή του, ώστε να δοκιμάσουν πράγματα στον εικονικό κόσμο που ανανεώνεται συνεχώς από τα δεδομένα που λαμβάνει από τα αισθητήρια. Με αυτόν τον τρόπο η προσομοίωση μπορεί να τους βοηθήσει να αποφασίσουν αν κάτι είναι καλή ιδέα ή όχι.

Τέλος ένα από τα μεγαλύτερα πλεονεκτήματα της προσομοίωσης το συναντάμε στις πολύ-ρομποτικές προσομοιώσεις. Μια δημοφιλής εφαρμογή αυτών των προσομοιώσεων είναι το Robot Soccer, όπου είτε μέσω προσομοίωσης, είτε μέσω πραγματικών ρομπότ, μια ομάδα συναγωνίζεται μια άλλη στο ποδόσφαιρο (ιδανικό για διεθνείς διαγωνισμούς). Τα ρομπότ της ίδιας ομάδας πρέπει να συνεργαστούν (πιθανώς με κάποιο είδος επικοινωνίας) και ταυτόχρονα να συναγωνισθούν ενάντια στα ρομπότ της αντίπαλης

---

ομάδας, ενέργειες που έχουν σαν αποτέλεσμα μια αρκετά δύσκολη δοκιμή ρομποτικής συμπεριφοράς.

### **2.2.2 ΠΕΡΙΟΡΙΣΜΟΙ ΚΑΙ ΜΕΙΟΝΕΚΤΗΜΑΤΑ ΣΤΗΝ ΠΡΟΣΟΜΟΙΩΣΗ**

Καθώς στην κατάσταση της προσομοίωσης το πρόβλημα μεταφέρεται από την κατάσταση υλισμικού στον χώρο του λογισμικού, προκύπτουν και τα ανάλογα προβλήματα και περιορισμοί της ανάπτυξης λογισμικού. Έτσι έχουμε εφαρμογές όπου η ενδείκνυται η προσομοίωσή τους και εφαρμογές που η προσομοίωσή τους βρίσκεται ακόμα στην ανάπτυξη και η χρήση πραγματικού ρομπότ είναι ευκολότερη ή αναγκαία. Όσο όμως αναπτύσσονται τα περιβάλλοντα προσομοίωσης και αυξάνεται η διαθέσιμη επεξεργαστική ισχύς, τόσο αυξάνεται και το εύρος των εφαρμογών που μπορούν να προσομοιωθούν με επιτυχία.

Οι κυριότεροι περιορισμοί που συναντούνται στην προσομοίωση είναι οι παρακάτω [5]:

- Σύμφωνα με τους ανθρώπους που ασχολούνται σε μεγάλες ρομποτικές εφαρμογές, οι προγραμματιστές πρέπει να περάσουν αρκετό χρόνο δουλεύοντας σε ένα πραγματικό ρομπότ, άσχετα με το πόσο καλή προσομοίωση χρησιμοποιούν, λόγω του ότι υπάρχει πολύς ακόμα χώρος για βελτίωση των προσομοιώσεων αλλά κυρίως γιατί ο πραγματικός κόσμος είναι απρόβλεπτος και περίπλοκος, με τα αισθητήρια να λαμβάνουν πολύ θορυβώδη σήματα.
- Ένας πολύ μεγάλος αριθμός φυσικών φαινομένων του πραγματικού κόσμου είναι ανεξήγητος ή πολύ δύσκολο στο να μοντελοποιηθεί, κάνοντας έτσι αδύνατη την ακριβή μοντελοποίηση, ειδικά σε πραγματικό χρόνο. Για κάποιες εφαρμογές, όπως τα οχήματα, η κίνηση σε χαμηλή ταχύτητα και η μοντελοποίηση του σόναρ είναι από τα πιο μεγάλα προβλήματα στις προσομοιώσεις.
- Ενώ είναι αρκετά εύκολο να τοποθετηθεί ένα ρομπότ σε έναν εικονικό κόσμο και να αλληλεπιδρά με άλλα αντικείμενα μέσα σε ένα περιβάλλον προσομοίωσης, χρειάζεται σημαντική προσπάθεια να παραμετροποιηθεί το προσομοιωμένο υλισμικό (οι λεγόμενες 'entities') ώστε να συμπεριφέρεται όπως θα συμπεριφερόταν στην πραγματικότητα, ειδικά όταν πρόκειται για δημιουργία αυτοματοποιημένων εργαλείων που ρυθμίζουν τέτοιες παραμέτρους προσομοίωσης.

Παρά τα μειονεκτήματα, το κέρδος της εκμάθησης και εξαγωγής συμπερασμάτων από την ρομποτική προσομοίωση σε δυναμικά περιβάλλοντα είναι αρκετά μεγάλο.

---

## 2.3 ΡΟΜΠΟΤΙΚΟΣ ΠΡΟΣΟΜΟΙΩΤΗΣ (ROBOTICS SIMULATOR)

Ένας προσομοιωτής ρομποτικής χρησιμοποιείται για δημιουργία ενσωματωμένων εφαρμογών στα ρομπότ, χωρίς φυσική εξάρτηση από το πραγματικό μηχάνημα, εξοικονομώντας έτσι χρόνο και χρήμα. Σε ορισμένες περιπτώσεις, οι εφαρμογές μπορούν να μεταφερθούν (ή να ανακατασκευαστούν) στο πραγματικό ρομπότ χωρίς τροποποιήσεις [6].

Ο όρος ‘προσομοιωτής ρομποτικής’ αντιστοιχεί σε αρκετές διαφορετικές ρομποτικές εφαρμογές λογισμικού. Για παράδειγμα, σε εφαρμογές κινούμενων ρομπότ, οι προσομοιωτές συμπεριφοράς επιτρέπουν στους χρήστες να δημιουργούν απλούς κόσμους στέρεων αντικειμένων, και πηγών φωτός και να τα προγραμματίζουν κατάλληλα, ώστε να αλληλεπιδρούν με αυτούς τους κόσμους. Η προσομοίωση βασισμένη στη συμπεριφορά, επιτρέπει ενέργειες οι οποίες είναι περισσότερο φυσιολογικές, σε σχέση με προσομοιωτές οι οποίοι βασίζονται σε άλλους τρόπους υλοποίησης ενεργειών. Επιπροσθέτως τέτοιου είδους προσομοιωτές μπορούν να ‘μάθουν’ από λάθη και είναι ικανοί να επιδείξουν ανθρωπομορφική συνεκτικότητα.

Μια από τις πιο δημοφιλείς εφαρμογές των προσομοιωτών ρομποτικής είναι το 3D modeling και rendering ενός ρομπότ και του περιβάλλοντος του. Αυτός ο τύπος ρομποτικού λογισμικού, έχει έναν προσομοιωτή απεικόνισης ενός εικονικού ρομπότ που είναι ικανός να προσομοιώσει ένα κανονικό ρομπότ σε πραγματικές συνθήκες εργασίας. Ορισμένοι προσομοιωτές ρομποτικής, όπως πχ το RoboLogix , χρησιμοποιούν μηχανές φυσικών ιδιοτήτων (Physics Engines), για πιο ρεαλιστική απεικόνιση κίνησης του ρομπότ.

Η χρήση ενός προσομοιωτή ρομποτικής για ανάπτυξη ρομποτικών προβλημάτων ελέγχου συνίσταται ανεξαρτήτως της ύπαρξης πραγματικού ρομπότ για εφαρμογή. Ο προσομοιωτής επιτρέπει το εύκολο γράψιμο και έλεγχο σφαλμάτων των ρομποτικών προγραμμάτων εκτός σύνδεσης (Off-line) έτσι ώστε η τελική μορφή του προγράμματος να εφαρμόζεται στο πραγματικό ρομπότ.

Φυσικά τέτοιου είδους προσομοίωση έχει ευρεία χρήση κυρίως στις βιομηχανικές εφαρμογές των ρομπότ, καθώς η επιτυχία του προγραμματισμού εκτός σύνδεσης εξαρτάται από το πόσο πιστά απεικονίζεται το πραγματικό περιβάλλον του ρομπότ στο εικονικό.

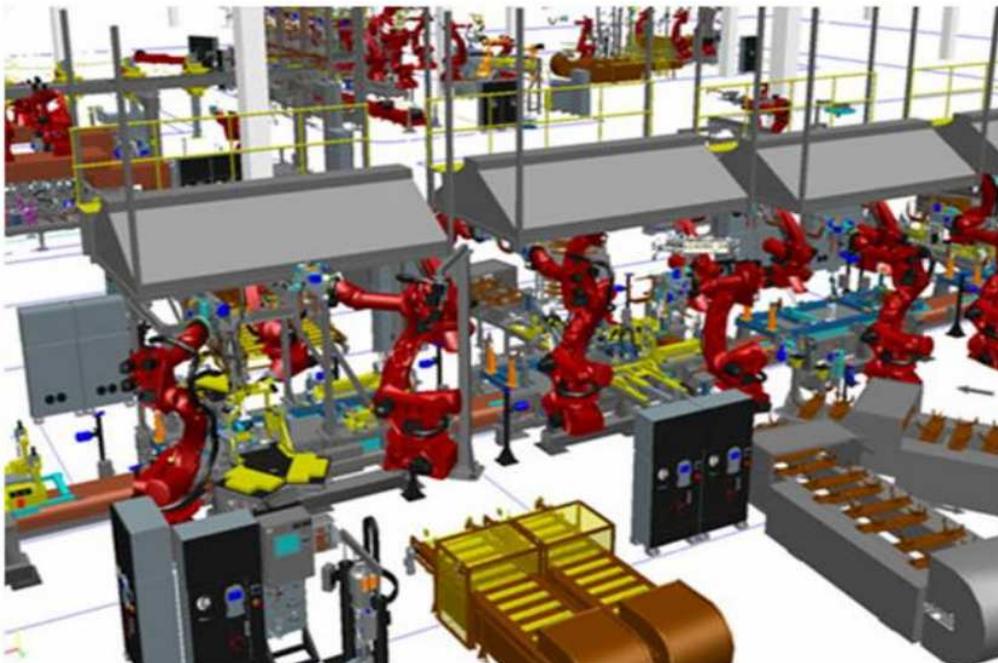
Οι, εξαρτώμενες από τα αισθητήρια, πράξεις του ρομπότ, είναι πολύ δύσκολο να προσομοιωθούν και να προγραμματιστούν εκτός σύνδεσης καθώς η κίνηση του ρομπότ εξαρτάται από τα στιγμιαία ερεθίσματα του πραγματικού κόσμου.

---

## 2.4 ΜΗ ΕΚΠΑΙΔΕΥΤΙΚΕΣ ΕΦΑΡΜΟΓΕΣ ΡΟΜΠΟΤΙΚΗΣ ΠΡΟΣΟΜΟΙΩΣΗΣ

### 2.4.1 ΠΡΟΣΟΜΟΙΩΣΗ ΣΤΗ ΓΡΑΜΜΗ ΠΑΡΑΓΩΓΗΣ ΡΟΜΠΟΤ

Το περιβάλλον υλοποίησης και η διαδικασία παραγωγής των τελικών προϊόντων των εργοστασίων είναι περίπλοκο και αυτό αποδίδεται στο επίπεδο της αυτοματοποίησης που εφαρμόζεται στη γραμμή παραγωγής, αλλά και σε παράγοντες όπως διαφορετικοί τύποι ρομπότ, ελεγκτών και αυτοματιστικού εξοπλισμού. Αυτό έχει οδηγήσει στο να γίνει σχεδόν απαραίτητη η ρομποτική προσομοίωση και ο off-line προγραμματισμός για την διαχείριση και υλοποίηση των γραμμών παραγωγής, διασφαλίζοντας έτσι την ακρίβεια και την ποιότητα των προϊόντων. Η συμμετοχή της εικονικής προσομοίωσης στη γραμμή παραγωγής ρομπότ, αυξάνεται όσο μεγαλώνει η ακρίβεια των εργαλείων προσομοίωσης. Για να υπάρξουν πραγματικά οφέλη στην προσομοίωση μιας εργοστασιακής ρομποτικής γραμμής παραγωγής, τα εργαλεία προσομοίωσης θα πρέπει να μπορούν να αναπαραστήσουν πιστά τουλάχιστον το 95% της πραγματικής διαδικασίας. Επιπροσθέτως η ακριβής σχεδίαση και προσομοίωση της συμπεριφοράς μιας συσκευής πρέπει να παρέχει λογικό έλεγχο και προφίλ μηχανικής κίνησης, λεπτομερή ρομποτικό προγραμματισμό, ακριβή χρονικό κύκλο (cycle time) και ανάλυση των διαφόρων παρεμβολών.[7]



*Εικόνα 2.1 Η πολυπλοκότητα των ρομποτικών εφαρμογών στην βιομηχανία χρειάζεται ανεπτυγμένη προσομοίωση [7]*

---

### 2.4.2 ΔΙΑΣΤΗΜΙΚΗ ΠΡΟΣΟΜΟΙΩΣΗ

Ένας εξελιγμένος τύπος ρομποτικού προσομοιωτή είναι αυτός που χρησιμοποιείται από την NASA για την προετοιμασία των αστροναυτών.

Ο προσομοιωτής υπάρχει στο εσωτερικό του περιβάλλοντος προσομοίωσης πραγματικού κόσμου (CAE) που έχει αποδειχθεί ότι παρέχει ντετερμινιστικές και συνεπείς προσομοιώσεις πτήσης παγκοσμίως. Έχει χρησιμοποιηθεί από αστροναύτες και έχει κερδίσει την αποδοχή τους, όμως υπολείπονται χαρακτηριστικά τα οποία θα μπορούσαν να προστεθούν για να εκμεταλλεύονται πλήρως την σύγχρονη τεχνολογία. Επιπροσθέτως, αυτή η προσομοίωση είναι βασισμένη πάνω σε συμπαγή τεχνικά και αρχιτεκτονικά θεμέλια και έχει αποδειχθεί ικανή για να υποστηρίζει κάθε ανάγκες χρηστών του είδους. [8]



Εικόνα 2.2 Προσομοίωση του Διεθνούς Διαστημικού Σταθμού [8]

### 2.4.3 ROBOT OPERATING SYSTEM (ROS)

Το Robot Operating System (ROS), είναι ένα λογισμικό πλαίσιο (software framework) ανάπτυξης ρομποτικού λογισμικού, που παρέχει λειτουργίες OS (operating system - λειτουργικού συστήματος), σε ετερογενή ρομποτικά δίκτυα.[9]

Αναπτύχθηκε αρχικά υπό το όνομα switchyard από το εργαστήριο τεχνητής νοημοσύνης του Stanford για την υποστήριξη του Stanford Air Robot (STAIR) το 2007. Από το 2008 και έπειτα η ανάπτυξη συνεχίζεται κυρίως στο ερευνητικό ινστιτούτο

---

Willow Garage, ενώ συνεργάζονται για την ανάπτυξή του περισσότερα από 20 ακόμα ιδρύματα. [10, 11]

Το ROS παρέχει υπηρεσίες λειτουργικού συστήματος όπως έλεγχος συσκευών χαμηλού επιπέδου, ενσωμάτωση των πιο συχνά χρησιμοποιούμενων λειτουργιών, μεταφορά μηνυμάτων μεταξύ διεργασιών και διαχείριση πακέτων δεδομένων. Βασιζόμενο σε μια αρχιτεκτονική γραφημάτων όπου η επεξεργασία γίνεται μέσω κόμβων οι οποίοι λαμβάνουν, τοποθετούν και πολυπλέκουν μηνύματα αισθητηρίων, ενεργοποιητών, σχεδιασμού, ελέγχου, κατάστασης κλπ.

Η βιβλιοθήκη υποστηρίζει περισσότερο ένα σύστημα του UNIX (πχ το Ubuntu Linux χαρακτηρίζεται υποστηριζόμενο, ενώ και άλλα συστήματα όπως το Fedora και το Mac OS X, ως πειραματικά)

Χωρίζεται σε δύο μέρη: Το μέρος του λειτουργικού συστήματος το οποίο ονομάζεται *ros* το οποίο περιγράφει παραπάνω και το *ros-pkg*, μια σουίτα από πακέτα δημιουργημένα από χρήστες (ομαδοποιημένα σε επονομαζόμενα 'Stacks') τα οποία εφαρμόζουν διάφορες λειτουργικές εργασίες, όπως ταυτόχρονη απεικόνιση θέσης, αντίληψη και σχεδίαση δράσεων, προσομοιώσεις κλπ

Το ROS μπορεί να υποστηρίζει τομείς όπως [10] :

- Ένα κύριο κόμβο συντονισμού
- Δημοσίευση ή εγγραφή σε συρμούς δεδομένων (data streams): εικόνες, laser, ενεργοποιητές, διάφοροι έλεγχοι κλπ
- Πολύπλεξη πληροφορίας
- Δημιουργία και διαγραφή κόμβων
- Η καταναεμημένη λειτουργία μέσω πολύ-επεξεργαστών, πολύ-πυρήνων CPU's, GPU's και σμήνη είναι εφικτή μέσω ομαλής κατανομής των κόμβων
- Τήρηση αρχείων ημερολογίου
- Διάφορες παραμέτρους διακομιστών (servers)
- Δοκιμές συστημάτων

Ενώ οι περιοχές που εφαρμόζεται συμπεριλαμβάνουν [10]:

- Αντίληψη (perception)
- Αναγνώριση αντικειμένων
- Κατάτμηση και αναγνώριση δεδομένων
- Αναγνώριση προσώπων
- Αναγνώριση χειρονομιών
- Αυτοκίνηση
- Κατανόηση κίνησης
- Structure from motion (SFM)

- 
- Στερεοσκοπική όραση: αντίληψη βάθους μέσω δύο καμερών
  - Κίνηση
  - Κινούμενα ρομπότ
  - Έλεγχος
  - Σχεδίαση (planning)
  - Πιάσιμο (Grasping)

Μεταφορές του λογισμικού σε ρομπότ συμπεριλαμβάνουν [10]:

- Το PR2, ένα ρομπότ για προσωπική χρήση (Personal Robot) ανεπτυγμένο από το Willow Garage
- Το PR1, ένα ρομπότ για προσωπική χρήση ανεπτυγμένο στο εργαστήριο του Ken Salisbury στο Stanford
- Το HERB, ανεπτυγμένο στο CMU, το πρόγραμμα προσωπικής ρομποτικής της Intel
- Τα STAIR I και II, ρομπότ που αναπτύχθηκαν από τον Anderw Ng στο Stanford
- Το πανεπιστήμιο του Βελγίου Katholieke Universiteit Leuven, μετέφερε το ROS στο Beagleboard
- Το εργαστήριο ανθρωποειδούς ρομποτικής του University of Freiburg έχει αναπτύξει μια έκδοση του ROS για το ανθρωποειδές Nao
- Το ρομπότ Husky A200 έχει αναπτυχθεί μέσω ROS από την Clearpath Robotics

#### **2.4.4 URBI**

Η URBI είναι μια πλατφόρμα ανοικτού λογισμικού γραμμένη σε C++ , όπου χρησιμοποιείται στην ανάπτυξη εφαρμογών ρομποτικής και πολύπλοκων συστημάτων. Βασίζεται στο UObject, μια αρχιτεκτονική βασισμένη σε στοιχεία C++ ενώ περιλαμβάνει την γλώσσα παράλληλου προγραμματισμού urbiscript. Τα στοιχεία του UObject μπορούν να χρησιμοποιηθούν από την urbiscript και να ενσωματωθούν ως δικά της, ενώ μπορούν και να ενωθούν μέσω του διερμηνέα της urbiscript και να εκτελεστούν ως ανεξάρτητες διεργασίες. [12]

##### **2.4.4.1 Η ΓΛΩΣΣΑ URBIScript**

Η γλώσσα urbiscript αναπτύχθηκε από το 2003 υπό τον Jean-Christophe Baillie στο ρομποτικό εργαστήριο της ENSTA στο Παρίσι. Από το 2006 και έπειτα αναπτύσσεται από την εταιρεία Gostai.

Η συγκεκριμένη γλώσσα θα μπορούσε να περιγραφεί ως μια γλώσσα συμπληρωματικών προγραμμάτων (scripting language). Η urbiscript μπορεί να ενώσει

---

κομμάτια γραμμένα σε C++ και κομμάτια λειτουργικής συμπεριφοράς. Έτσι το αλγοριθμικό κομμάτι που επιβαρύνει τον επεξεργαστή αναλαμβάνεται από την C++, ενώ το προγραμματιστικό κομμάτι της συμπεριφοράς αφήνεται στην urbiscript, που είναι πιο ευέλικτη, ευκολότερη στην συντήρηση και επιτρέπει δυναμική αλληλεπίδραση κατά την εκτέλεση του προγράμματος.

Ως γλώσσα συμπληρωματικών προγραμμάτων η urbiscript, έχοντας βάση της τον παράλληλο και αντικειμενοστραφή προγραμματισμό, προσφέρει την πολύ χρήσιμη ανεξαρτησία των εφαρμογών της στους προγραμματιστές. Ο προγραμματισμός παράλληλων συμπεριφορών λειτουργίας και αντιδράσεων σε γεγονότα είναι ο πυρήνας των απαιτήσεων των περισσότερων εφαρμογών της τεχνητής νοημοσύνης και ρομποτικής, κάνοντας έτσι την urbiscript ιδανική για τέτοιες εφαρμογές.

Χαρακτηριστικά της γλώσσας [12]:

- Αντικειμενοστραφής προγραμματισμός
- Προγραμματισμός βασισμένος σε πρωτοτύπηση
- Συντακτικό όμοιο με της C++
- Αρχιτεκτονική όμοια με της C++
- Αρχιτεκτονική βασισμένη σε δομές client/server [13]
- Διεπαφές του client μέσω Java και MATLAB
- Υποστηριζόμενες πλατφόρμες Linux, Mac, Windows κλπ
- Εντολές για έλεγχο ροής προγραμμάτων [14]

Πλεονεκτήματα της γλώσσας [12]:

- Είναι ανοικτό λογισμικό
- Υποστηρίζει το Robot Operating System (ROS)
- Παρέχεται τεχνική υποστήριξη, συχνές νέες εκδόσεις και καθορισμένη προγραμματιστική ομάδα ανάπτυξής της

#### **2.4.4.2 ΚΥΡΙΟΤΕΡΑ ΧΑΡΑΚΤΗΡΙΣΤΙΚΑ ΤΗΣ URBI [15]**

- Αναγνώριση φωνής
- Σύνθεση φωνής
- Αναγνώριση προσώπου
- SLAM
- Αναγνώριση κηλίδων φωτός
- Αναγνώριση αντικειμένων βασισμένη στο SIFT

Υποστηριζόμενα ρομπότ και προσομοιωτές μέσω URBI:

- Webots [16]
- Gostai Open Jazz [16]

- 
- Segway
  - Aibo ERS7/ERS2xx
  - iRobot Create
  - Lego Mindstorms NXT
  - HRP-2
  - Nao [16]
  - Robotics Bioloid [16]
  - Mobile Robotics Pioneer [16]
  - Khepera III
  - e-puck
  - Surveyor

---

## **ΚΕΦΑΛΑΙΟ 3**

# **ΕΚΠΑΙΔΕΥΤΙΚΟΙ ΡΟΜΠΟΤΙΚΟΙ ΠΡΟΣΟΜΟΙΩΤΕΣ (ROBOTICS SUITES)**

### **3.1 ΕΙΣΑΓΩΓΗ**

Εκπαιδευτικό ρομποτικό προσομοιωτή ονομάζουμε ένα γραφικό περιβάλλον για έλεγχο και προσομοίωση ρομπότ. Συνήθως είναι ολοκληρωμένες πλατφόρμες ανάπτυξης ρομπότ συμπεριλαμβάνοντας εργαλεία οπτικού προγραμματισμού για δημιουργία και έλεγχο καλής λειτουργίας ρομποτικών εφαρμογών. Μέσω διεπαφών βασιζόμενων στο διαδίκτυο (web-based) ή μέσω οπτικών διεπαφών (interfaces), οι προγραμματιστές συχνά μπορούν να αλληλεπιδρούν με τα ρομπότ.

Κύριο χαρακτηριστικό σε έναν εκπαιδευτικό ρομποτικό προσομοιωτή, είναι η υποστήριξη πολλών διαφορετικών ρομποτικών πλατφόρμων, μέσω μιας κοινής προγραμματιστικής διεπαφής (interface). Το κύριο πλεονέκτημα ενός εκπαιδευτικού ρομποτικού προσομοιωτή είναι ότι θα τρέξει ο ίδιος κώδικας είτε σε ένα προσομοιωτή ρομπότ είτε στο πραγματικό μοντέλο δίχως τροποποιήσεις. [17]

### 3.2 MICROSOFT ROBOTICS DEVELOPER STUDIO



*Εικόνα 3.1 Προσομοίωση του Mars Rover σε περιβάλλον του πλανήτη Άρη μέσω MRDS [18]*

Το Microsoft Robotics Developer Studio (MRDS), είναι ένας εκπαιδευτικός ρομποτικός προσομοιωτής ρομποτικού ελέγχου, σε περιβάλλον βασισμένο στα Windows και απευθύνεται κυρίως σε προγραμματιστές του ακαδημαϊκού και του εμπορικού χώρου, όπως και σε χομπίστες, υποστηρίζοντας μεγάλη ποικιλία ρομποτικού υλισμικού (hardware). [18]

Το MRDS στηρίζεται στο CCR (Concurrency and Coordination Runtime): Μια βιβλιοθήκη ταυτόχρονα εκτελούμενου κώδικα, όπου διαχειρίζεται ασύγχρονες παράλληλες εργασίες. Αυτή η τεχνική εμπεριέχει τη χρήση του DSS (Decentralized Software services) ένα εκτελέσιμο που προσανατολίζεται στην ολοκλήρωση των ελαφρών εργασιών, επιτυγχάνοντας έτσι την δημιουργία πολλαπλών συμπληρωματικών προγραμμάτων, για εκπλήρωση περίπλοκων συμπεριφορών (behaviors).

Ένα σημαντικό χαρακτηριστικό του, είναι η χρήση της Microsoft Visual Programming Language (VPL), ένα ‘εργαλείο’ οπτικού προγραμματισμού, για δημιουργία και έλεγχο σφαλμάτων διαφόρων ρομποτικών εφαρμογών, διεπαφών (interfaces) (είτε web-based είτε windows-based), 3D προσομοιώσεων (με ενσωματωμένη επιτάχυνση υλισμικού) και εύκολης πρόσβασης σε αισθητήρια και ενεργοποιητές (actuators) ενός ρομπότ. Η κύρια γλώσσα προγραμματισμού είναι η C#.

Επίσης το MRDS υποστηρίζει διάφορα πακέτα, τα οποία προσθέτουν κι άλλες εφαρμογές αυτού της σουίτας. Από αυτές που είναι διαθέσιμες αυτήν την στιγμή, αξίζει να αναφερθούν το Soccer Simulation και το Sumo Competition της Microsoft, όπως και

---

το Maze Simulator, ένα πρόγραμμα ανεπτυγμένο από την κοινότητα του MRDS, στο οποίο δημιουργούνται κόσμοι με τοίχους όπου μπορούν να εξερευνηθούν από εικονικά ρομπότ, ενώ υπάρχει και ένα σετ υπηρεσιών για το OpenCV.

Περισσότερα επιπλέον πακέτα υπάρχουν διαθέσιμα στο CodePlex, όπως και υλικό εκμάθησης της σουίτας.

Τα εργαλεία ανάπτυξης μιας MRDS εφαρμογής περιλαμβάνουν ένα γραφικό περιβάλλον (VPL), εργαλεία γραμμής εντολών όπου επιτρέπεται η εμπλοκή άλλων projects από το Visual Studio σε C#, όπως και εργαλεία προσομοίωσης 3D.

Η VPL είναι ένα γραφικό αναπτυξιακό περιβάλλον το οποίο χρησιμοποιεί καταλόγους Services και activities.

- Υπάρχει γραφική αλληλεπίδραση, όπου μια υπηρεσία ή μια δραστηριότητα αναπαρίστανται από ένα μπλοκ, που έχει εισόδους και εξόδους, οι οποίες χρειάζεται απλά να συρθούν από τους καταλόγους στο διάγραμμα εργασίας.
- Η σύνδεση μεταξύ τους μπορεί να γίνει με το ποντίκι, επιτρέπει τον ορισμό του αν θα είναι τα σήματα σύγχρονα ή ασύγχρονα, επιτρέπει επεμβάσεις στις μεταδιδόμενες τιμές κλπ.
- Επίσης είναι δυνατόν από τα διαγράμματα που δημιουργούνται να παράγεται κώδικας νέων υπηρεσιών
- Όπως και να τροποποιηθούν ήδη υπάρχουσες υπηρεσίες σε διαφορετικό υλισμικό.

Το περιβάλλον προσομοίωσης 3D του MRDS επιτρέπει την προσομοίωση της συμπεριφοράς ρομπότ σε εικονικό κόσμο χρησιμοποιώντας την τεχνολογία PhysX της NVIDIA, όπου περιλαμβάνει ανεπτυγμένη μηχανή φυσικών ιδιοτήτων.

Υπάρχουν αρκετά περιβάλλοντα προσομοίωσης στο MRDS τα οποία έχουν αναπτυχθεί από την SimplySim:

- ο Apartment (Διαμέρισμα)
  - ο Factory (Εργοστάσιο)
  - ο Modern House (Σύγχρονο σπίτι)
  - ο Outdoor (Εξοχή)
  - ο Urban (Πόλης)
- Για τα διάφορα εργαλεία είναι διαθέσιμα αρκετά παραδείγματα και οδηγοί για γρήγορη κατανόηση του MRDS.

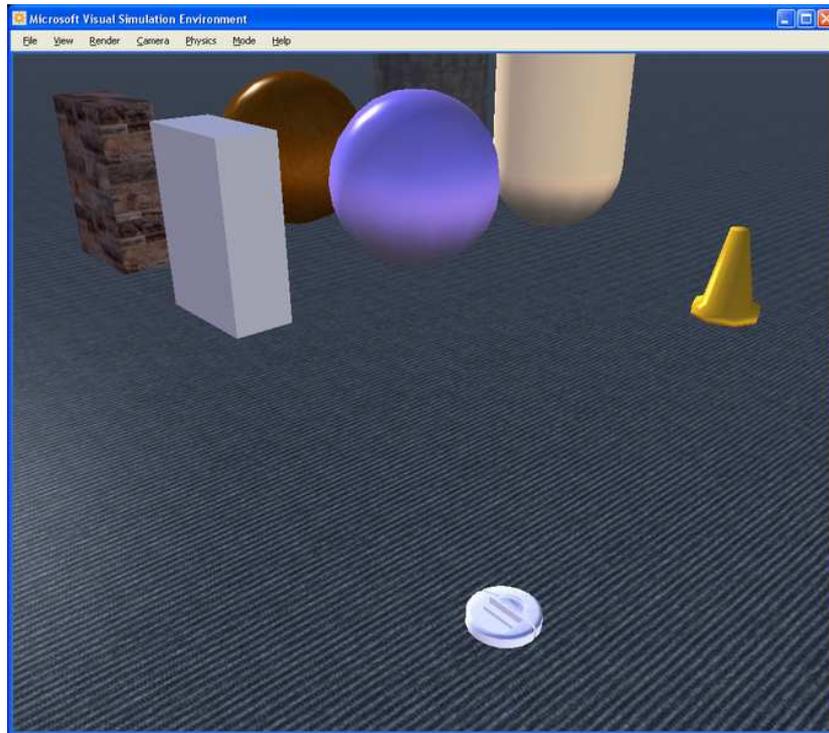
Από τα 4 βασικά στοιχεία του MRDS που αναφέρθηκαν προηγουμένως, τα CCR και DSS είναι διαθέσιμα και για ξεχωριστή χρήση εμπορικών εφαρμογών, οι οποίες απαιτούν μεγάλο επίπεδο συγχρονισμού ή/και πρέπει να διανέμονται μέσω πολλαπλών κόμβων σε ένα δίκτυο. Αυτό το πακέτο ονομάζεται CCR and DSS Toolkit.

Το MRDS έχει κάποιες αξιοσημείωτες εφαρμογές, όπως:

- 
- Η συμμετοχή του Princeton University στο διαγωνισμό DARPA URBAN GRAND CHALLENGE, είχε προγραμματιστεί σε MRDS [19]
  - Η γνωστή ιστοσελίδα MySpace, χρησιμοποιεί τις βιβλιοθήκες CCR και DSS, για μη-ρομποτικές εφαρμογές στον πυρήνα της ιστοσελίδας του [20]
  - Το Indiana University χρησιμοποιεί το MRDS σε μία μη-ρομποτική εφαρμογή όπου διαχειρίζεται ένα υπολογιστικό δίκτυο υψηλής απόδοσης. [20]

Μερικά από τα υποστηριζόμενα ρομπότ είναι τα εξής [18]:

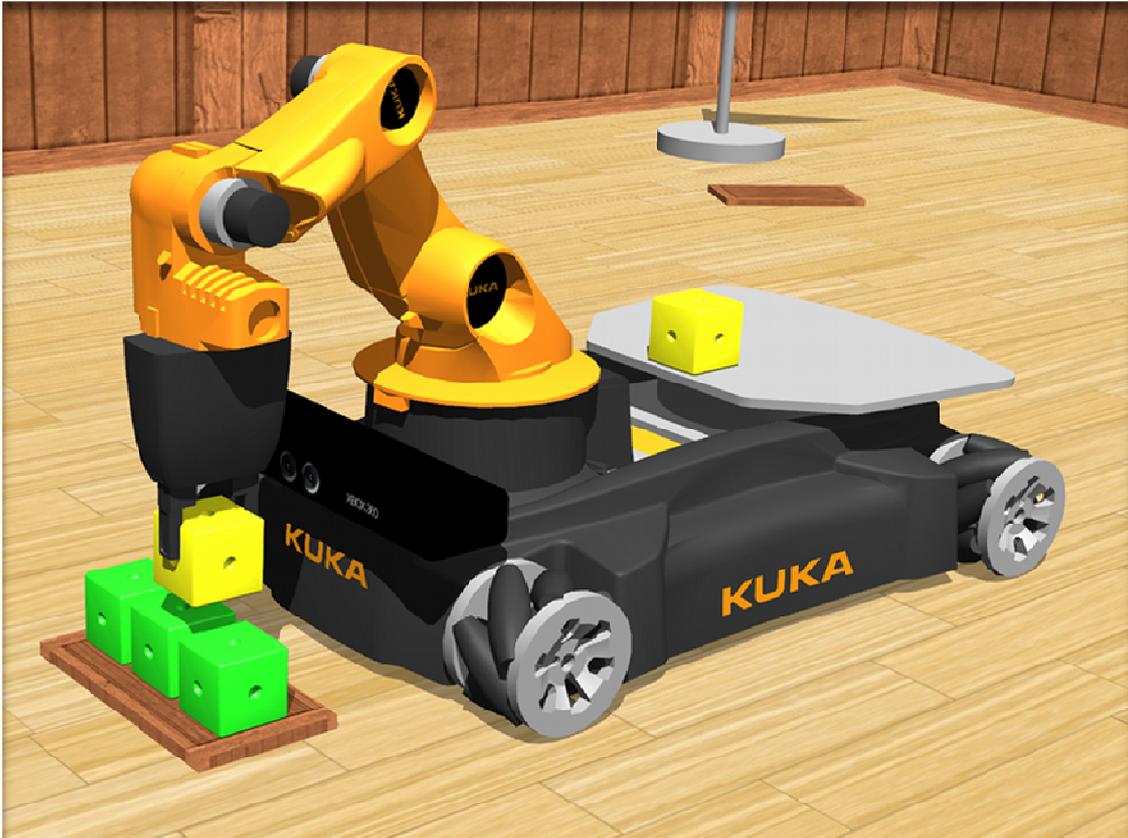
- ABB Group Robotics - ABB Connect for Microsoft Robotics Developer Studio
- Adept MobileRobots Pioneers (το πρώτο ρομπότ που υποστήριξε το MRDS)
- ArieH Robotics Project Junior (Physical computing PC)
- CoroWare CoroBot and Explorer
- Lego Mindstorms NXT Official Website
- iRobot Create
- KUKA Robotics Educational Framework
- Parallax Boe-Bot
- Parallax Scribbler.
- Parallax Eddie robot.
- fischertechnik FT16
- Robosoft's robots
- Kondo KHR-1
- Segway RMP
- Surveyor SRV-1
- RoboticsConnection Traxster.
- RoombaDevTools by RoboDynamics
- uBot-5 University of Massachusetts Amherst
- Vex Robotics Design System
- WowWee RoboSapien,
- ZMP INC. e-nuvo
- CNRobot Co. Ltd
- Robotino



*Εικόνα 3.2 Ένα iRobot Create ρομπότ στο Visual Simulation Environment του MRDS[18]*

Σε επόμενη ενότητα παρουσιάζεται εκτενέστερα ο εκπαιδευτικός ρομποτικός προσομοιωτής MRDS.

### 3.3 WEBOTS



Εικόνα 3.3 Προσομοίωση ενός KUKA youBot πάνω σε μια συσκευή Kinect, σε περιβάλλον WEBOTS [21]

Το WEBOTS είναι ένας εκπαιδευτικός ρομποτικός προσομοιωτής που χρησιμοποιείται κυρίως για εκπαιδευτικούς σκοπούς. Η ανάπτυξή του ξεκίνησε το 1996 από τον Dr. Olivier Michel στο Swiss Federal Institute of Technology (EPFL) στην Λωζάννη της Ελβετίας.

Χρησιμοποιεί την βιβλιοθήκη ODE (Open Dynamics Engine) για ανίχνευση συγκρούσεων και για προσομοίωση ιδιοτήτων στερεών σωμάτων, η συγκεκριμένη βιβλιοθήκη επιτρέπει την προσομοίωση με ακρίβεια τις φυσικές ιδιότητες αντικειμένων, όπως ταχύτητα, αδράνεια και τριβή.

Το WEBOTS υποστηρίζει μια μεγάλη συλλογή μοντέλων ρομπότ, τα οποία είναι τροποποιήσιμα, ενώ υπάρχει η δυνατότητα της δημιουργίας νέων μοντέλων από την αρχή. Όταν σχεδιάζεται ένα ρομποτικό μοντέλο, ο χρήστης πρέπει να καθορίσει μαζί με τις φυσικές και τις γραφικές ιδιότητες των εμπλεκόμενων αντικειμένων.

---

Οι γραφικές ιδιότητες συμπεριλαμβάνουν έννοιες όπως το σχήμα, τις διαστάσεις, την θέση και τον προσανατολισμό, τα χρώματα και την υφή του αντικειμένου. Οι φυσικές ιδιότητες μπορεί να συμπεριλαμβάνουν τη μάζα, τον συντελεστή τριβής και τις σταθερές ελατηρίου και χρόνου απόσβεσης.

Επίσης συμπεριλαμβάνεται ομάδα αισθητηρίων και ενεργοποιητών, που χρησιμοποιούνται ευρύτατα σε ρομποτικά πειράματα, όπως πχ αισθητήρες απόστασης, αισθητήρες φωτός, αισθητήρες αφής, GPS, επιταχυνσιόμετρα, κάμερες, πομποί και δέκτες, σερβοκινητήρες (γραμμικοί και περιστροφικοί), αισθητήρες θέσης και δύναμης, LEDS, λαβίδες, γυροσκόπια, πυξίδες, κλπ. [21]

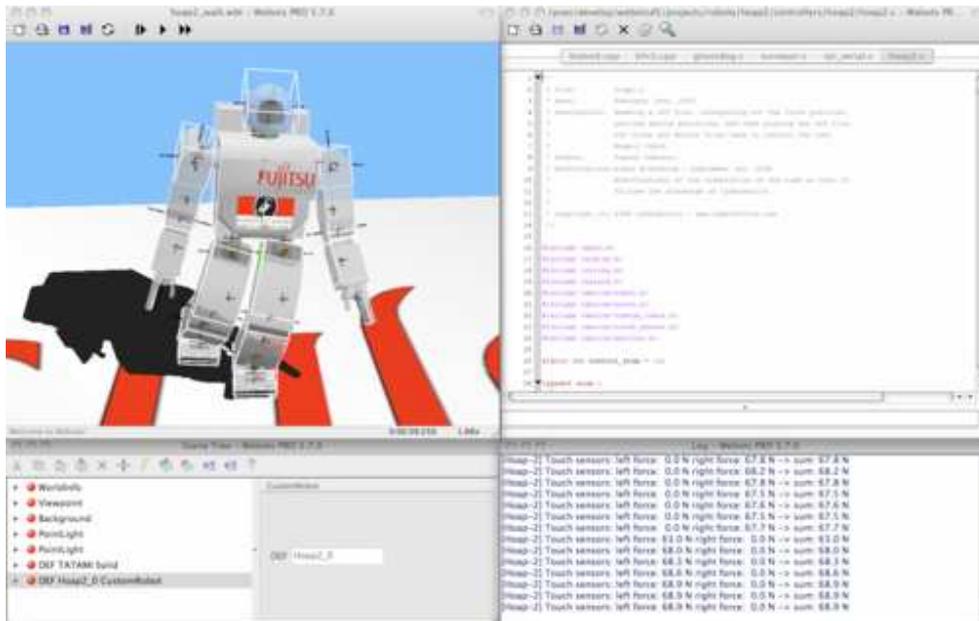
Μερικά από τα υποστηριζόμενα ρομπότ είναι τα εξής [21]:

- AIBO ERS7 and ERS210, Sony Corporation [22]
- Bioloids (dog), Robotis
- Boe-Bot
- DARwIn-OP, Robotis
- E-puck
- Hemisson
- HOAP-2, Fujitsu Limited
- iCub, RobotCub Consortium
- iRobot Create, iRobot
- Katana IPR, Neuronics AG
- Khepera mobile robot I, II, III, K-Team Corporation
- KHR-2HV, KHR-3HV, Kondo
- Koala, K-Team Corporation
- Lego Mindstorms (RCX Rover model)
- Magellan
- Nao V2, V3, Aldebaran Robotics
- MobileRobots Inc Pioneer 2
- Puma 560, Unimate
- Scout 2
- Shrimp III, BlueBotics SA
- Surveyor SRV-1, Surveyor Corporation
- youBot, KUKA

Τα προγράμματα των ρομποτικών ελεγκτών, μπορούν να γραφούν σε C, C++, Java, Python καθώς και σε Matlab. Πιο συγκεκριμένα ρομποτικά μοντέλα όπως το AIBO, το NAO και το E-Puck, μπορούν να προγραμματιστούν επιπροσθέτως στην γλώσσα URBI.

Όσον αφορά την αποθήκευση των δεδομένων, το WEBOTS παρέχει την δυνατότητα για απόκτηση PNG εικόνων και καταγραφή βίντεο σε MPEG (Mac/Linux) και AVI (Windows) αρχεία. Τα αρχεία του WEBOTS αποθηκεύονται στην cross-platform μορφή

.wbt η οποία βασίζεται στην γλώσσα VRML. Είναι επίσης δυνατόν τα αρχεία του WEBOTS να εισαχθούν/εξαχθούν σε καθαρή VRML μορφή. Ένα επίσης χρήσιμο χαρακτηριστικό είναι η δυνατότητα του χρήστη να αλληλεπιδράσει με την προσομοίωση ενώ αυτή εκτελείται (πχ να κουνήσει το ρομπότ ή άλλα αντικείμενα με το ποντίκι).



Εικόνα 3.4 Προσομοίωση ενός HOAP-2 σε βηματισμό [21]

Παρακάτω παρουσιάζεται ένα απλό παράδειγμα γραμμένο σε C/C++, ενός ελεγκτή στο WEBOTS: μια ρουτίνα αποφυγής συγκρούσεων. Αρχικά το ρομπότ πηγαίνει μπροστά, έπειτα αν ανιχνευτεί εμπόδιο περιστρέφεται γύρω από τον εαυτό του για λίγο και ύστερα συνεχίζει την κίνησή του προς τα εμπρός. [21]

---

```
#include <webots/robot.h>
#include <webots/differential_wheels.h>
#include <webots/distance_sensor.h>
#define TIME_STEP 64

int main() {
    // initialize Webots
    wb_robot_init();

    // get handle and enable distance sensor
    WbDeviceTag ds = wb_robot_get_device("ds");
    wb_distance_sensor_enable(ds, TIME_STEP);

    // control loop
    while (1) {
        // read sensors
        double v = wb_distance_sensor_get_value(ds);

        // if obstacle detected
        if (v > 512) {
            // turn around
            wb_differential_wheels_set_speed(-600, 600);
        }
        else {
            // go straight
            wb_differential_wheels_set_speed(600, 600);
        }

        // run a simulation step
        wb_robot_step(TIME_STEP);
    }
    return 0;
}
```

Τα κύρια πεδία που εφαρμόζεται το WEBOTS είναι [21]:

- 
- Προτυποποίηση ρομπότ με ρόδες και πόδια
  - Έρευνα ρομποτικής μετακίνησης [23]
  - Swarm intelligence (πολύ-ρομποτικές προσομοιώσεις) [24] [25]
  - Τεχνητή ζωή και εξελικτική ρομποτική
  - Προσομοίωση προσαρμοστικής συμπεριφοράς [26] [27]
  - Αυτό-διευθετούμενη αρθρωτή ρομποτική [28]
  - Πειραματικά μοντέλα υπολογιστικής όρασης και
  - Διαγωνισμοί διδασκαλίας και ρομποτικού προγραμματισμού



Εικόνα 3.5 Διαγωνισμός ρομποτικού πρωταθλήματος με ρομπότ NAO [21]

Το WEBOTS χρησιμοποιείται σε αρκετούς online διαγωνισμούς ρομποτικού προγραμματισμού, όπως ο διαγωνισμός Robostadium, μια προσομοίωση για το ρομποτικό πρωτάθλημα Robocup. Σε αυτήν την προσομοίωση, δύο ομάδες αποτελούμενες από ρομπότ NAO παίζουν ποδόσφαιρο με κανονισμούς παρόμοιους με αυτούς του κανονικού ποδοσφαίρου. Τα ρομπότ εκεί χρησιμοποιούν προσομοιωτές καμερών και αισθητήρων υπερήχων και πίεσης.

---

Σε διαφορετικό διαγωνισμό, όπως το Rat's Life, δύο E-ruck ρομπότ σε προσομοίωση, διαγωνίζονται με ενεργειακά κριτήρια σε έναν λαβύρινθο φτιαγμένο από LEGO. Αυτοί οι αγώνες διεξάγονται καθημερινά και τα αποτελέσματα μπορούν να παρακολουθηθούν σε βίντεο από το διαδίκτυο.

### 3.4 ANYKODE MARILOU



Εικόνα 3.6 Εξωτερικός χώρος προσομοιωμένος στο Anykode Marilou [29]

Το AnyKode Marilou είναι ένας εκπαιδευτικός ρομποτικός προσομοιωτής για κινούμενα ρομπότ, ανθρωποειδή, αρθρωτούς βραχίονες και παραλληλισμένα ρομπότ τα οποία λειτουργούν σε συνθήκες πραγματικού κόσμου και υπόκεινται στους νόμους της φυσικής. Αυτός ο εκπαιδευτικός ρομποτικός προσομοιωτής χρησιμοποιείται σε κέντρα ερευνών, αλλά και στην βιομηχανία για διάφορα έργα, όπως αρχιτεκτονική σχεδίαση ανθρωποειδών, οχήματα με ρόδες ή 'πολύποδα' καθώς και για πολύ-ρομποτικά συστήματα (Multi-agents).

Η βιβλιοθήκη πραγματικού χρόνου που χρησιμοποιεί είναι η ODE (Open Dynamics Engine) όπως και το WEBOTS, για ανίχνευση συγκρούσεων και δυναμική διαχείριση. Διάφορες παράμετροι του πραγματικού κόσμου, όπως δυνάμεις, ροπές, μάζες, χρόνοι απόσβεσης, τριβές κλπ μπορούν να εφαρμοστούν κατευθείαν πάνω στις επιφάνειες των αντικειμένων.

---

Το Marilou χρησιμοποιεί τα ονομαζόμενα ‘entities’ για τα στοιχεία που υπάρχουν σε ένα περιβάλλον προσομοίωσης και τα λεγόμενα ‘scenes’ για το ίδιο το περιβάλλον. Ο editor των entities, επιτρέπει τον σχεδιασμό ρομποτικών μοντέλων συγκρούσεων χρησιμοποιώντας οποιαδήποτε αντικείμενα του προσομοιωμένου κόσμου, είτε στατικά είτε δυναμικά (static/dynamic). Τα εργαλεία επεξεργασίας έχουν πολλές ομοιότητες με προγράμματα CAD και λειτουργούν σε γραφικό περιβάλλον, που διευκολύνει τη χρήση του και ταυτόχρονα επιτρέπονται γρήγορες αλλαγές και άμεση δημιουργία νέων καταστάσεων προσομοίωσης.

Μέσω ενός μενού, μπορούν να αλλαχθούν οι ιδιότητες των ρομπότ, των dynamics και των scenes. Ο editor επίσης, μπορεί να επεξεργαστεί επαναχρησιμοποιήσιμες entities φυσικών ιδιοτήτων, όπως και καθαρά μοντέλα 3D.

Λόγω των παραπάνω, ακόμα και τα πιο πολύπλοκα αντικείμενα μπορούν εύκολα να κατασκευασθούν, συναρμολογώντας πολλά μικρά κομμάτια entities. Το Marilou χρησιμοποιεί ένα ιεραρχικό σύστημα εμφάνισης ολοκληρωμένων αντικειμένων στο τελικό υψηλότερο επίπεδο (του κόσμου). Με αυτήν την προσέγγιση, επιτρέπεται η επαναχρησιμοποίηση κάποιων μελών ενός πολύπλοκου αντικειμένου ως κομμάτια ενός άλλου.

Στο Marilou χρησιμοποιείται μια ομάδα εικονικών συσκευών που είναι πλήρως επεξεργάσιμες από τον χρήστη. Η συμπεριφορά αυτών των συσκευών μπορεί να ανανεωθεί από ιδιότητες πραγματικών συσκευών που είναι διαθέσιμες στη ρομποτική. Αυτό το χαρακτηριστικό επιτρέπει στον χρήστη να χρησιμοποιήσει τις παραμέτρους μιας ήδη γνωστής συσκευής απευθείας στην προσομοίωσή του. [29]

Παρακάτω δίδεται μια λίστα του τι περιλαμβάνουν οι υποστηριζόμενες συσκευές του Marilou [29]:

- Ενσωματωμένα ρομποτικά εξαρτήματα
- Πυξίδα
- Επιταχυνσιόμετρα
- Γυροσκόπια
- Εναέριες δυνάμεις πίεσης
- Προφυλακτήρες
- Αισθητήρια απόστασης (Υπερηχητικά, Υπέρυθρα και LASER)
- Κινητήρες και σερβο-κινητήρες
- Πομπές και δέκτες
- Αισθητήρια Δύναμης και Ροπής
- GPS
- Αισθητήρια εύρους LASER
- LED
- LCD οθόνες

- 
- Πηγές φωτός
  - LIDAR
  - Οδόμετρα
  - Τυπικές και πανοραμικές σφαιρικές κάμερες
  - Περιοχές αφής

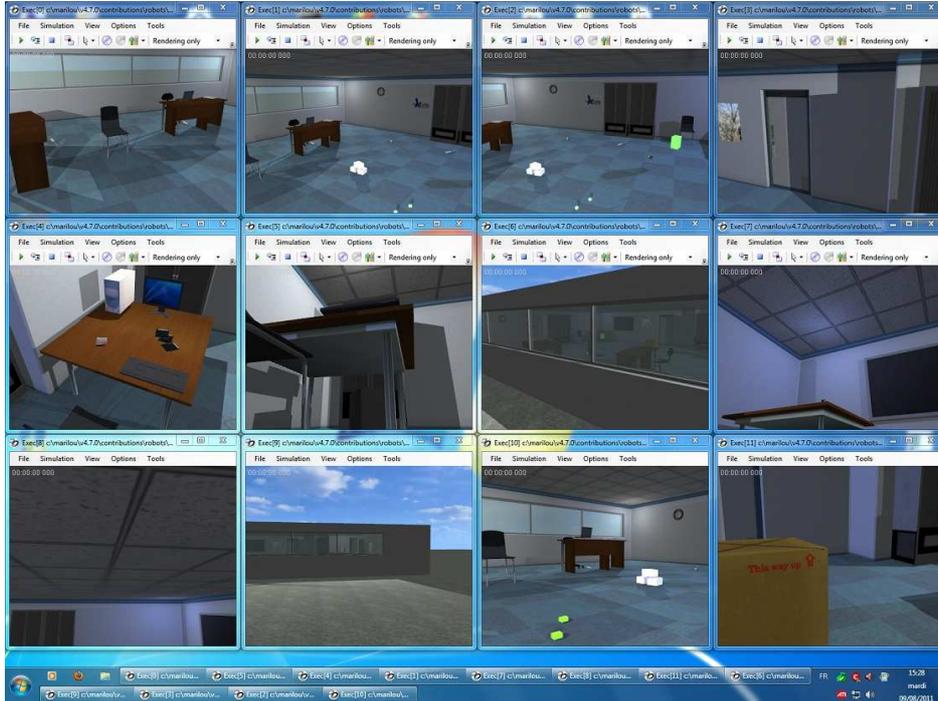
Για την διαχείριση των προσομοιωμένων ρομπότ και των ενσωματωμένων συσκευών τους, όπως αισθητήρια και ενεργοποιητές, το Marilou χρησιμοποιεί το MODA (Marilou Open Devices Access) ένα SDK γενικού περιεχομένου, το οποίο ανάλογα με την επιλεγμένη γλώσσα προγραμματισμού, παρέχει βιβλιοθήκες (.lib, .a) ή .NET στοιχεία (.dll) για την πρόσβαση στην προσομοίωση μέσω δικτύου. Συγχρονισμένοι σε ένα προσομοιωμένο ρολόι, οι αλγόριθμοι μπορούν να τρέξουν σε οποιοδήποτε υπολογιστή του δικτύου, ενώ μεμονωμένα ρομπότ μπορούν να τρέχουν πολλά τέτοια διαφορετικά προγράμματα. Επιπροσθέτως, ένα πρόγραμμα MODA μπορεί να ελέγχει αρκετά ρομπότ, είτε προσομοιώνονται στον ίδιο κόσμο, είτε όχι. Ο εξυπηρετητής MODA TCP μπορεί να εισαχθεί σε πραγματικό ρομπότ χωρίς κανένα πρόβλημα.

Παρακάτω παραθέτονται οι υποστηριζόμενες γλώσσες προγραμματισμού στο AnyKode Marilou [29]:

- Γλώσσες: C/C++, C++ CLI, C#, J#, VB#
- Compilers: Microsoft Visual Studio suites, DevC++, Borland C++ RAD Studio, G++ για Linux, CodeBlocks
- Επίσης είναι συμβατό με το Linux

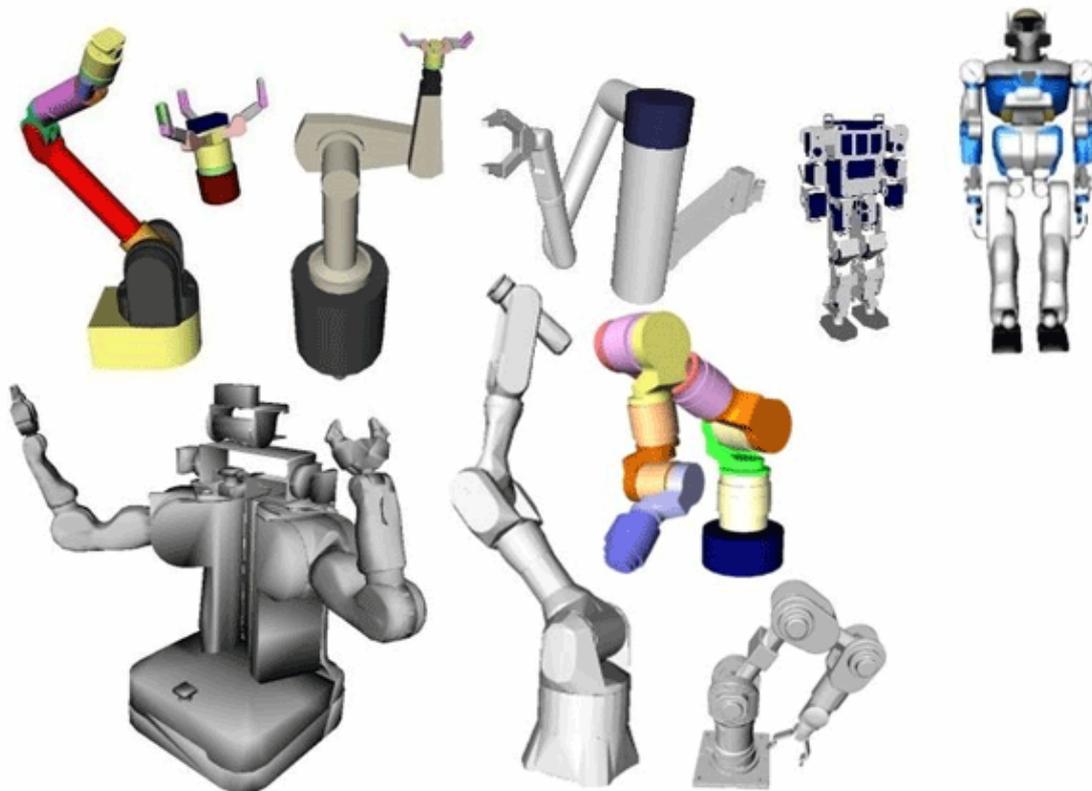
Οι εφαρμογές του AnyKode Marilou περιλαμβάνουν πολλά διαφορετικά χαρακτηριστικά, μεταξύ άλλων οι κυριότερες είναι οι [29]:

- Πλήρης γραφική διαχείριση ρομποτικών μοντέλων και των περιβαλλόντων τους (3D μοντέλα και φυσικές ιδιότητες)
- Βοηθήματα μοντελοποίησης, εργαλεία τροποποίησης κώδικα.
- Στερεά σώματα, περιορισμούς στους άξονες
- Μηχανικούς περιορισμούς
- Ιδιότητες επιφανειών (αντικατοπτρισμοί, δονήσεις, τριβές, προσπτώσεις, αναπηδήσεις, υπέρυθρες και υπερηχητικές αντιδράσεις κλπ)
- Ιεραρχικές και πολύπλοκες συναρμοολογήσεις
- Προσομοιώσεις πραγματικού χρόνου η επιταχυνόμενες
- Πολύ-ρομποτικά μοντέλα, πολλαπλές ενσωματωμένες εφαρμογές, συγκεντρωμένες η διάσπαρτες
- Απόκτηση/μέτρηση χρονικών κύκλων μέχρι και 1ms
- Αλληλεπιδράσεις με ήδη εκτελούμενες προσομοιώσεις
- Υψηλής ποιότητας 3D rendering χρησιμοποιώντας pixel και vertex shaders
- Φωτισμό διαφόρων ειδών και σκιές



Εικόνα 3.7 Πολλαπλές περιβάλλοντος προσομοίωσης στο Marilou [29]

### 3.5 OPENRAVE



---

*Εικόνα 3.8 Ρομποτικά μοντέλα στο OpenRAVE [30]*

Το Open Robotics Automation Virtual Environment (Open RAVE), είναι ένα περιβάλλον ανάπτυξης, εφαρμογής και δοκιμής αλγορίθμων σχεδίασης κίνησης σε πραγματικές ρομποτικές εφαρμογές. Η κύρια χρήση του είναι στην προσομοίωση και ανάλυση των κινηματικών και γεωμετρικών πληροφοριών που αφορούν την σχεδίαση της κίνησης. [31]

Αναπτύχθηκε από τον Rosen Diankon στο Quality of Life Technology Center του Carnegie Mellon University. Εμπνευσμένο από τον προσομοιωτή RAVE του James Kuffner του 1995 όπου τον χρησιμοποιούσε σε πολλά από τα πειράματά του. Το έργο ξεκίνησε το 2006 ως επανεγγραφή του RAVE, ώστε να υποστηρίζει plug-ins και γρήγορα αναπτύχθηκε δική του αρχιτεκτονική και πλέον χρησιμοποιείται από πολλούς ερευνητές ανά τον κόσμο. [32]

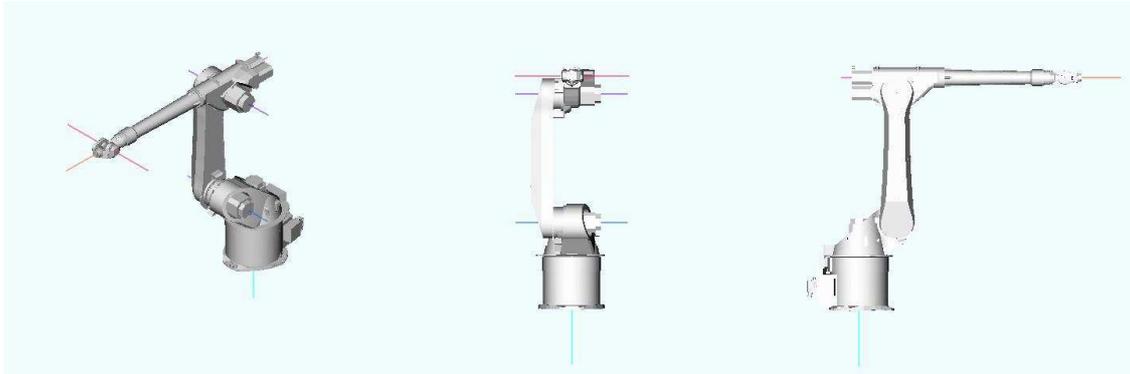
Επιτρέπει την εύκολη ενσωμάτωσή του σε ήδη υπάρχοντα ρομποτικά συστήματα, ενώ επίσης προσφέρει πολλά εργαλεία εκτέλεσης μέσω της γραμμής εντολών για εργασία με ρομπότ και έχει αρκετά μικρό εκτελέσιμο πυρήνα με αποτέλεσμα την εύκολη χρήση του μέσα σε μικρό-ελεγκτές αλλά και μεγαλύτερα frameworks. Η σημαντικότερη τεχνολογία που διαθέτει το OpenRAVE είναι ένα εργαλείο ονομαζόμενο IKFAST, όπου είναι ο κινηματικός μεταγλωττιστής για τα ρομπότ. Αντίθετα με τα περισσότερα εργαλεία λύσης κινηματικών προβλημάτων, το IKFAST μπορεί να λύσει αναλυτικά κινηματικές εξισώσεις κάθε πολύπλοκης κινηματικής αλυσίδας και να παράγει συγκεκριμένα αρχεία προγραμματιστικών γλωσσών (πχ C++) για μεταγενέστερη χρήση. Το αποτέλεσμα είναι εξαιρετικά σταθερές επιλύσεις, οι οποίες εκτελούνται μέχρι και σε 5 μικρό-δευτερόλεπτα σε σύγχρονους επεξεργαστές. [33]

Υποστηρίζεται το είδος αρχείων COLLADA (COLLABorative Design Activity) 1.5 για αποθήκευση της περιγραφής των ρομπότ, ενώ υπάρχει και δική του ομάδα ονομασίας ρομποτικών αρχείων. Τα συμπεριλαμβανόμενα αρχεία είναι:

- Αισθητήρια
- Βραχίονες
- Ειδικές παράμετροι σχεδιασμού

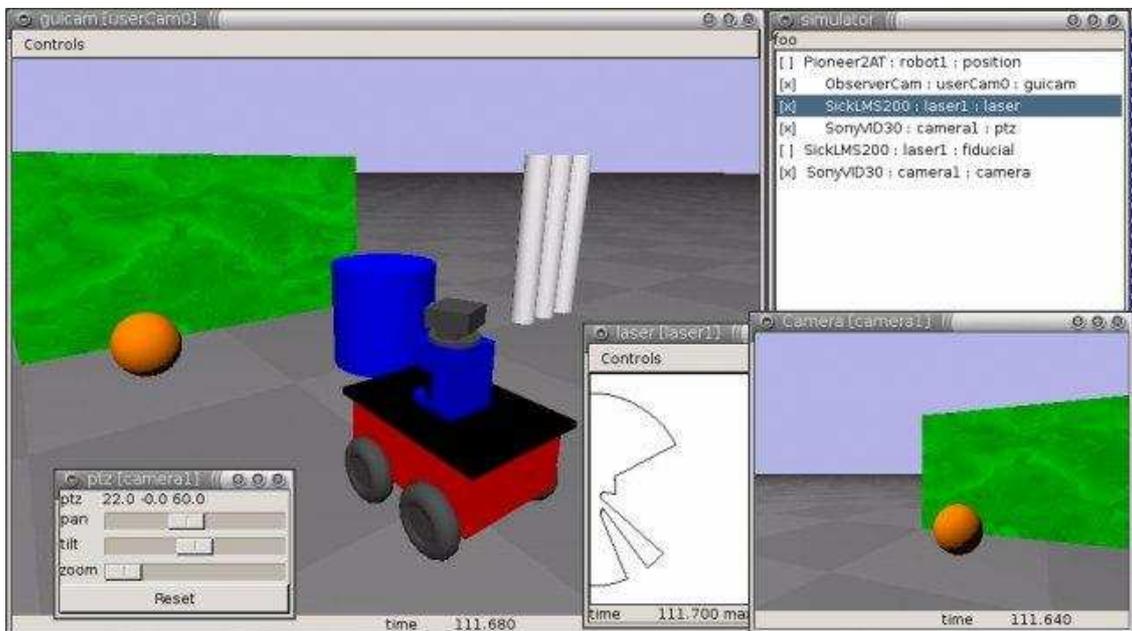
Ο πυρήνας του σχεδιασμού στο OpenRAVE επικεντρώνεται στην παροχή διεπαφών (interfaces) και εφαρμογές αλγορίθμων σχεδίασης κίνησης. Οι περισσότερες εφαρμογές είναι για ρομποτικούς βραχίονες και χρησιμοποιούν δειγματοληψία για να εφαρμόσουν το έργο της διαμόρφωσης χώρου.

Τέλος ίσως ο σημαντικότερος τομέας εφαρμογής του OpenRAVE είναι στους βιομηχανικούς ρομποτικούς αυτοματισμούς.



Εικόνα 3.9 Ρομποτικός βραχίονας KUKA-KR0116 προσομοιωμένος στο OpenRAVE [33]

### 3.6 PLAYER PROJECT



Εικόνα 3.10 Ένα Pioneer 3-AT σε περιβάλλον προσομοίωσης Gazebo 3D

Το Player Project (παλαιότερα γνωστό ως Player/Stage/Gazebo Project) είναι ένα ελεύθερο ερευνητικό λογισμικό για ρομπότ και συστήματα αισθητήριων [34]. Περιλαμβάνει τρία ταυτόχρονα αναπτυσσόμενα έργα ρομποτικής, όπου αυτά είναι ένας δικτυακός ρομποτικός εξυπηρετητής (Player), ένα δυσδιάστατο περιβάλλον ρομποτικής προσομοίωσης (Stage) και ένα τρισδιάστατο περιβάλλον ρομποτικής προσομοίωσης (Gazebo), ενώ το λογισμικό διακινείται ελεύθερα μέσω της GPL (General Public License).

Αν και είναι δύσκολο να συλλεχθούν ακριβή στατιστικά στοιχεία, το Player είναι ένα από τα πιο δημοφιλή open-source interfaces στο πεδίο της έρευνας και της εκπαίδευσης

---

[35]. Τα περισσότερα περιοδικά και συνέδρια ρομποτικής δημοσιεύουν συχνά papers χρησιμοποιώντας τα Player, Stage και Gazebo. Ξεκίνησε το 2000 από τους Brian Gerkey, Richard Vaughn και Andrew Howard στο University of Southern California του Los Angeles.[36]

### 3.6.1 PLAYER

Το λογισμικό του Player εκτελείται σε λειτουργικά συστήματα συμβατά με POSIX (Portable Operating System Interface for Unix), όπως Linux, MacOS X, Solaris, BSD και Windows. Υποστηρίζει μια ευρεία γκάμα hardware (ρομποτικές πλατφόρμες, συσκευές αισθητηρίων κλπ), ενώ συμπεριλαμβάνει υποστήριξη σε βιβλιοθήκες ενός μεγάλου αριθμού προγραμματιστικών γλωσσών, όπως C, C++, Python και Ruby. Επίσης υπάρχουν βιβλιοθήκες και τρίτων εταιρειών σε γλώσσες τύπου Java και TCL. Άλλα χαρακτηριστικά του είναι το μινιμαλιστικό και ευέλικτο περιβάλλον του, η υποστήριξη interfacing σε ταυτόχρονες σε πολλαπλές συσκευές ταυτόχρονα και on-the-fly διαμόρφωση ενός εξυπηρετητή.

### 3.6.2 STAGE

Το Stage είναι ένα 2D περιβάλλον προσομοίωσης σχεδιασμένο πάνω στο FLTK (.Fast, Light Toolkit). Παρέχει το περιβάλλον για βασική προσομοίωση μοντελοποίησης ενός έως και εκατοντάδων ρομπότ κάθε φορά. Το Stage μπορεί να χρησιμοποιηθεί μόνο του για να προσομοιώσει ρομποτικές συμπεριφορές μέσω προγραμμάτων ελεγχόμενων από το χρήστη. Επίσης μπορεί να αλληλεπιδράσει με τα interfaces του Player επιτρέποντας πρόσβαση στους χρήστες σε προσομοιώσεις αισθητηρίων και άλλων συσκευών.

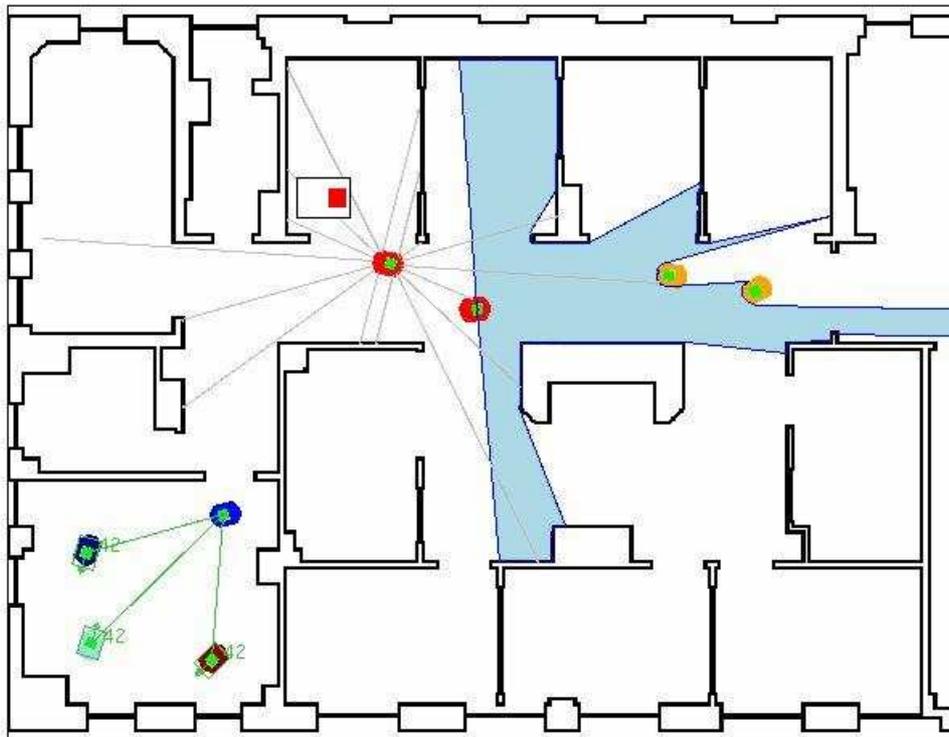
### 3.6.3 GAZEBO

Το Gazebo είναι ένα 3D περιβάλλον προσομοίωσης σχεδιασμένο πάνω στο OGRE (Object-Oriented Graphics Rendering Engine). Όπως και το Stage, μπορεί να χρησιμοποιηθεί μόνο του ή μέσω του Player server.

Μερικά από τα υποστηριζόμενα ρομπότ είναι τα εξής [37]:

- Acroname's Garcia
- Botrics's Obot d100
- CoroWare Inc. Corobot and Explorer
- Evolution Robotics' ER1 and ERSK robots

- 
- iRobot's Roomba vacuuming robot
  - K-Team's Robotics Extension Board (REB) attached to Kameleon 376BC
  - K-Team's Khepera
  - MobileRobots' (formerly ActivMedia) PSOS/P2OS/AROS-based robots
  - Nomadics' NOMAD200 (and possibly related) mobile robots
  - RWI/iRobot's RFLEX-based robots (e.g., B21r, ATRV Jr)
  - Segway's Robotic Mobility Platform (RMP)
  - UPenn GRASP's Clodbuster
  - Videre Design's ERRATIC mobile robot platform
  - White Box Robotics' 914 PC-BOT



*Εικόνα 3.11 Δυσδιάστατη προσομοίωση στο Stage*

### 3.7 NI LABVIEW ROBOTICS MODULE



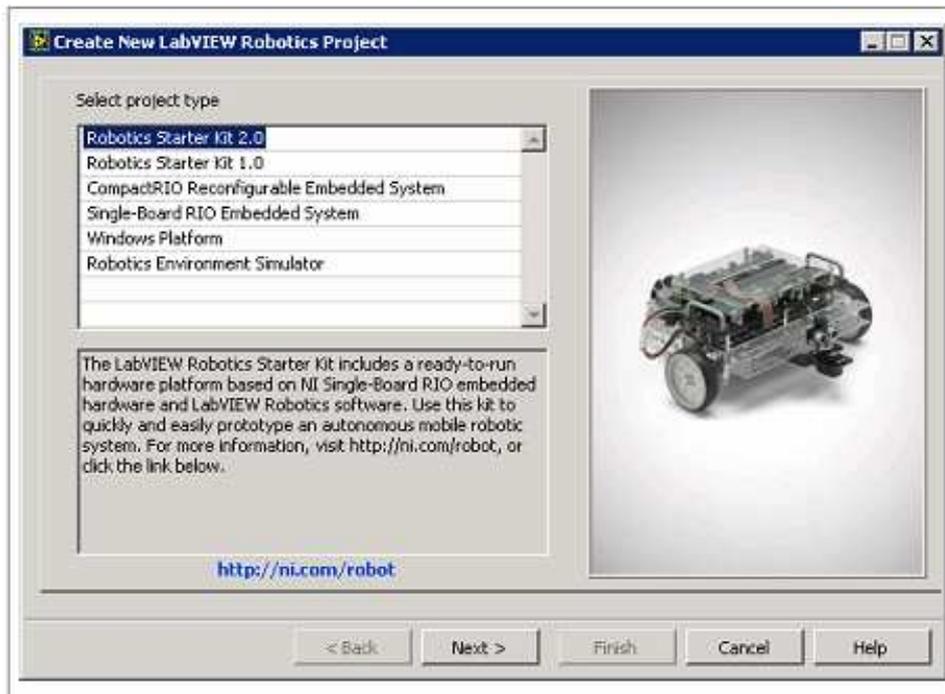
Εικόνα 3.12 Παράθυρο εκκίνησης στο LabVIEW [38]

Το NI LabVIEW Robotics module είναι ένας εκπαιδευτικός ρομποτικός προσομοιωτής ελέγχου. Το συγκεκριμένο module ενσωματώνεται στο αναπτυξιακό περιβάλλον LabVIEW, εμπλουτίζοντας το με βιβλιοθήκες ρομποτικής που προσφέρουν συνδεσιμότητα και ρομποτικά αισθητήρια/actuators, θεμελιώδεις αλγόριθμους για λειτουργίες τεχνητής νοημοσύνης και ισχυρές αντίληψης, όπως επίσης και ρουτίνες κίνησης.

Το LabVIEW Robotics απευθύνεται κυρίως σε μηχανικούς και επιστήμονες που σχεδιάζουν και ασχολούνται με τα παρακάτω [38]

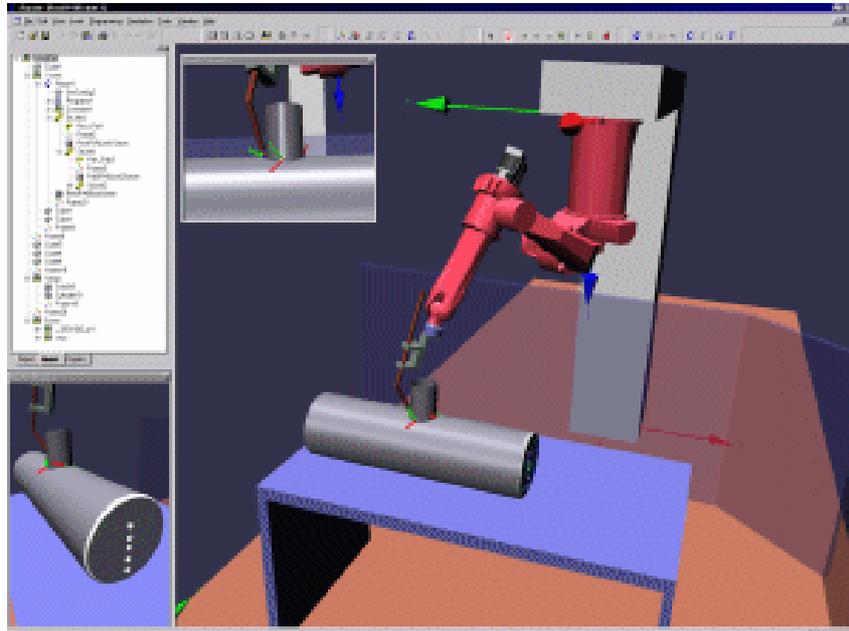
- Αυτόνομα και ημιαυτόνομα ρομποτικά οχήματα (συμπεριλαμβανομένων και αγροτικών/στρατιωτικών συστημάτων)
- Πλατφόρμες ρομπότ διάσωσης
- Υποβρύχια και ιπτάμενα ρομποτικά οχήματα
- Ρομπότ προσωπικών υπηρεσιών
- Συσκευές ιατρικών ρομπότ
- Ακαδημαϊκά και ερευνητικά ρομπότ

Παρέχει υψηλού επιπέδου αυτονομία, οπότε η διαδικασία προγραμματισμού πολύπλοκων ρομποτικών εφαρμογών απλοποιείται σημαντικά ενώ επίσης υπάρχει η δυνατότητα εισαγωγής κώδικα από άλλες γλώσσες (όπως C και HDL) και αλληλεπίδρασης με μεγάλο αριθμό προ-εγκατεστημένων αισθητηρίων, μειώνοντας έτσι τον χρόνο ανάπτυξης. Επίσης υποστηρίζονται αρκετοί επεξεργαστές και μικροεπεξεργαστές.



Εικόνα 3.13 Δημιουργία νέου project στο LabVIEW [38]

### 3.8 CAMELOT ROPSIM

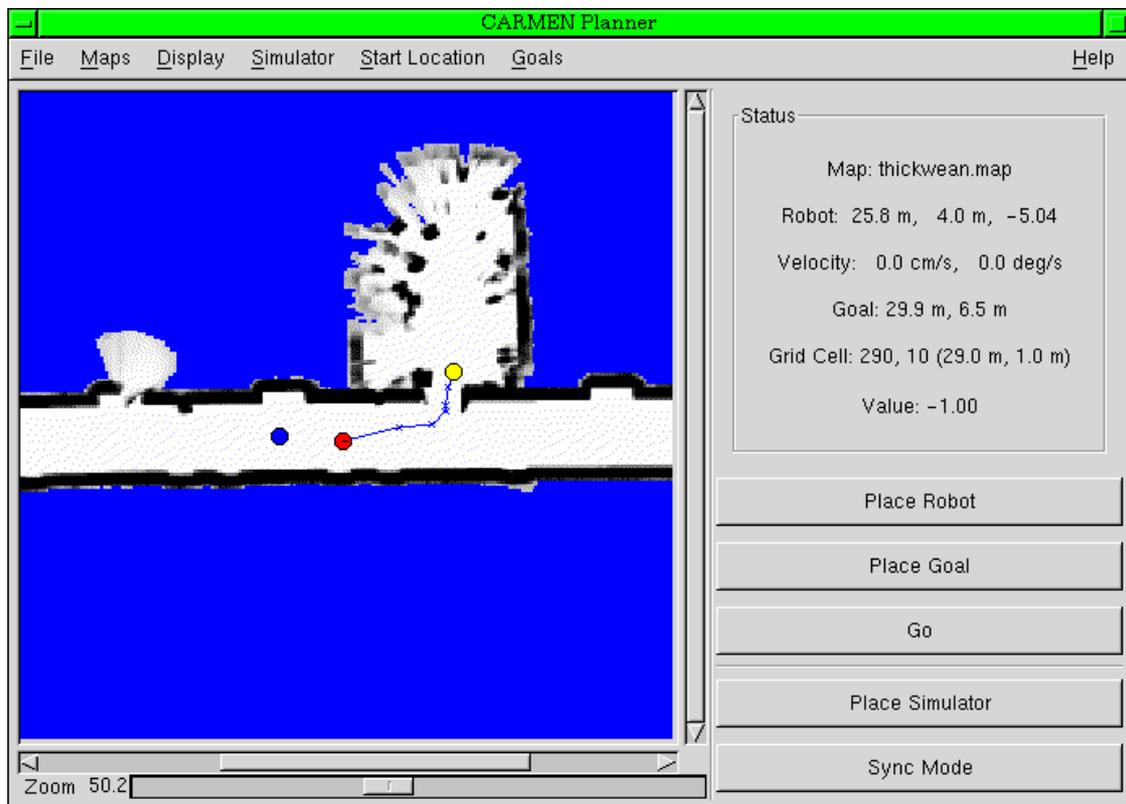


Εικόνα 3.14 Προσομοίωση στο περιβάλλον ROPSIM [39]

Η εταιρεία Camelot έχει αναπτύξει, σε συνεργασία με το Danish Technical University και το Institute for Construction and Controlling, ένα σύστημα προγραμματισμού μοντέλων ρομπότ για χρήση σε συστήματα ολοκληρωμένης παραγωγής, το οποίο ονομάζεται ROPSIM. Το ROPSIM είναι μια suite προσομοίωσης μοντέλων ρομπότ, το οποίο είναι βασισμένο πάνω σε χρήση PC και υποστηρίζει 3D οπτικοποίηση. Η εικονική προσομοίωση επιτρέπει την παρακολούθηση της γραμμής παραγωγής σε γραφικό περιβάλλον.

Στο εικονικό σύστημα παραγωγής είναι τρισδιάστατη προσομοίωση σε πραγματικό χρόνο, έτσι επιτρέπει πειραματισμούς και αναλύσεις χωρίς ακριβό εξοπλισμό. Επίσης προσφέρει πολύ ακριβείς προσομοιωτές που πλησιάζουν πιστότητα 100% με το πραγματικό περιβάλλον εργασίας. [39]

### 3.9 CARMEN TOOLKIT



Εικόνα 3.15 Εμφάνιση προσομοίωσης ρομπότ για αυτόνομη πλοήγηση [40]

Το CARMEN (Carnegie Mellon Robot Navigation Toolkit) είναι ένας εκπαιδευτικός ρομποτικός προσομοιωτής ανοικτού λογισμικού, για έλεγχο φορητών ρομποτικών συσκευών. Σχεδιάστηκε ως κομμάτι λογισμικού για παροχή βασικής λειτουργικότητας πλοήγησης, όπως: Βασικό έλεγχο, έλεγχο αισθητήρων, καταγραφή γεγονότων, αποφυγή εμποδίων, βασική τοποθέτηση στο χώρο, σχεδίαση διαδρομών και χαρτογραφήσεων.

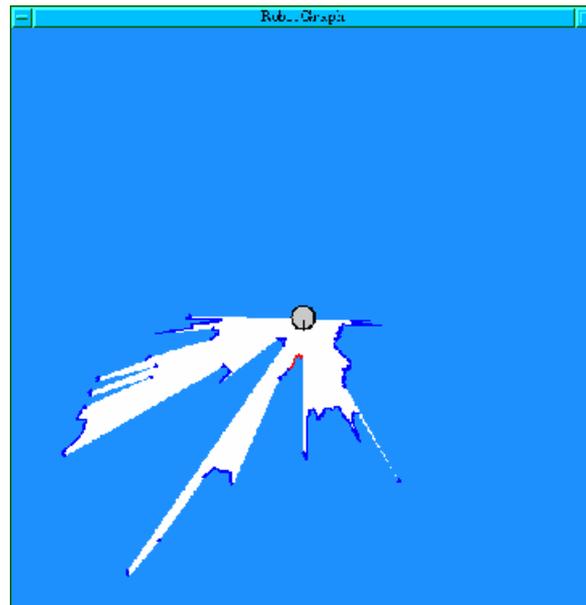
Μερικά από τα υποστηριζόμενα ρομπότ είναι τα εξής: [40]

- iRobot ATRV
- iRobot
- iRobot B21R
- ActivMedia Pioneer I
- ActivMedia Pioneer II
- Nomadic Technologies Scout
- Nomadic Technologies XR4000
- OrcBoard
- Segway

---

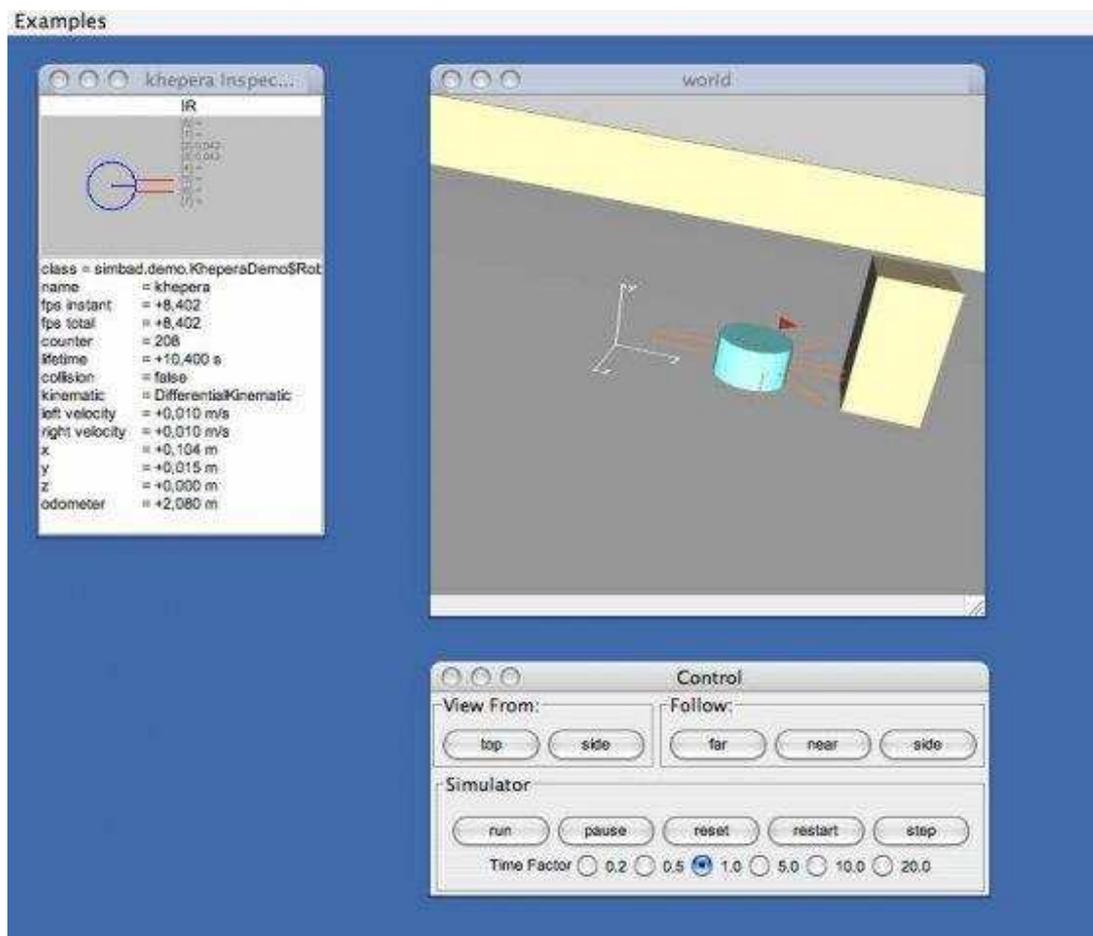
Επίσης παρακάτω είναι μια λίστα με τα υποστηριζόμενα μοντέλα αισθητηρίων: [40]

- SICK LMS laser σύστημα μέτρησης
- SICK PLS laser scanner απόστασης
- Συσκευή GPS χρησιμοποιώντας το πρωτόκολλο NMEA
- Sonar
- Hokuyo IR (PB9)



Εικόνα 3.16 Απεικόνιση δεδομένων αισθητηρίου ενός SICK PLS laser scanner [40]

### 3.10 SIMBAD



Εικόνα 3.17 Προσομοίωση του ρομπότ Khepera στο Simbad [41]

Το Simbad είναι ένας, ελεύθερος στη χρήση, προσομοιωτής ρομπότ για ερευνητικούς και εκπαιδευτικούς σκοπούς. Παρέχει μια απλή βάση για μελέτη τομέων όπως Situated AI, Machine Learning κλπ, ενώ υποστηρίζει ανάπτυξη πιο γενικευμένων αλγορίθμων AI, πάνω στο πεδίο της αυτόνομης Ρομποτικής και αυτόνομων Agents.

Ο εκπαιδευτικός ρομποτικός προσομοιωτής Simbad, περιλαμβάνει τα εξής μέρη [42]:

*Simbad*: Ο Ρομποτικός προσομοιωτής που μπορεί να διαχειριστεί περίπλοκα τρισδιάστατα περιβάλλοντα και αλληλεπιδράσεις βασισμένες στη φυσική.

*PicoNode*: Βιβλιοθήκη ελεγκτών βασισμένων σε γραφήματα, όπως τα εμπροσθοτροφοδοτούμενα και τα επαναλαμβανόμενα νευρωνικά δίκτυα.

*PicoEvo*: Βιβλιοθήκη εξελικτικών αλγορίθμων, όπου συμπεριλαμβάνει Γενετικούς αλγόριθμους, γενετικό προγραμματισμό (γραφήματα) και άλλες δημοφιλείς προσεγγίσεις στον τομέα της βελτιστοποίησης των στοχαστικών αλγορίθμων βασιζομένων στον πληθυσμό.

---

Ο χρήστης μπορεί να επιλέξει περιβάλλον από τον κατάλογο του EnvironmentDescription και ρομποτικό ελεγκτή από τον κατάλογο Robot. Ο ελεγκτής περιλαμβάνει μια κατάσταση έναρξης (initBehavior) και καταστάσεις οι οποίες χρησιμοποιούνται σε κάθε βήμα της προσομοίωσης (performBehavior). Έπειτα ο προσομοιωτής εκτελεί τις εντολές των κινητήρων με τρόπο παρόμοιο ενός πραγματικού ρομποτικού ελεγκτή (πχ επαναλαμβανόμενες κλήσεις στον μικροελεγκτή).

Ο παρακάτω κώδικας δείχνει την δημιουργία του προσομοιωτή, ενός περιβάλλοντος με ένα κουτί και ενός ρομπότ μέσα σε αυτό: [42]

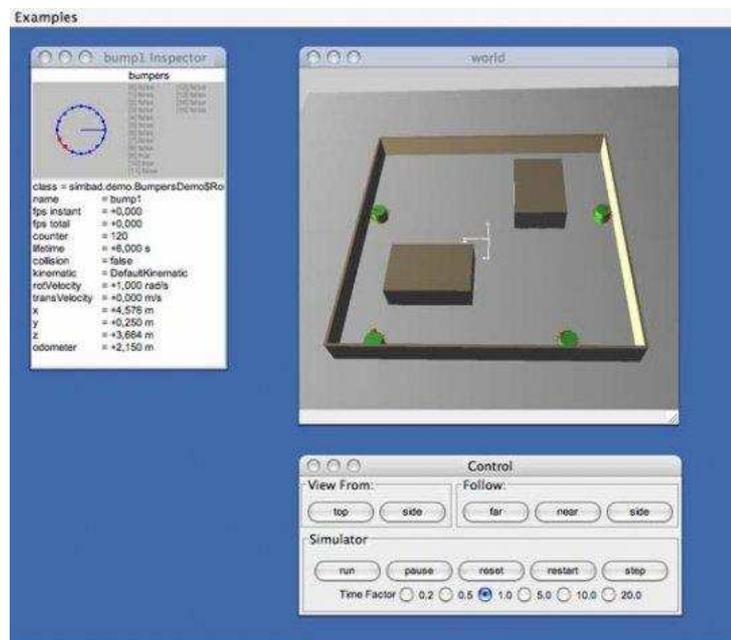
```
import simbad.sim.*; import simbad.gui.*; import javax.vecmath.*;
public class MyEnv extends EnvironmentDescription {
public MyEnv() {
/* create four walls and the robot */
Wall w1 = new Wall( new Vector3d(9, 0, 0), 19, 1, this);
w1.rotate90(1); add(w1);
Wall w2 = new Wall( new Vector3d(-9, 0, 0), 19, 2, this);
w2.rotate90(1); add(w2);
Wall w3 = new Wall( new Vector3d(0, 0, 9), 19, 1, this);
add(w3);
Wall w4 = new Wall( new Vector3d(0, 0, -9), 19, 2, this);
add(w4);
add(new MyRobot( new Vector3d(0, 0, 0), "my robot"));
}
public class MyRobot extends Robot {
MyRobot (Vector3d _position, String _name )
{ super (_position, _name); }
public void initBehavior()
{
κώδικας αρχικοποίησης
}
public void performBehavior()
{
κώδικας που εκτελείται σε κάθε βήμα
}
}
public static void main(String[] args)
{
Simbad frame = new Simbad( new MyEnv(), false);
}
}
```

Μέχρι τώρα έχουν επιτευχθεί αρκετές μελέτες πάνω στον τομέα της εξελικτικής ρομποτικής χρησιμοποιώντας το Simbad. Επίσης το Simbad χρησιμοποιείται στο χώρο της εκπαίδευσης, όπως στο τελευταίο έτος σπουδών του IFIPS (Institut de Formation d'Ingénieurs de Paris-Sud' σχολή μηχανικών του Université Paris-Sud) και στο Ecole Polytechnique, για εκμάθηση Εξελικτικής ρομποτικής πάνω σε προσομοιωμένα και πραγματικά ρομπότ Khepera.

Οι συνεχιζόμενες εργασίες ανάπτυξης της σουίτας Simbad περιλαμβάνουν προεκτάσεις του προσομοιωτή για εύκολη αλλαγή από κατάσταση προσομοίωσης σε πραγματικό περιβάλλον και προεκτάσεις επίσης των PicoNode και PicoEvo βιβλιοθηκών, ώστε να συμπεριλάβουν και άλλους αλγόριθμους μάθησης (Ενισχυμένης εκμάθησης, εκμάθησης τοπολογίας σε μαγαίσιανά δίκτυα ρομποτικού ελέγχου, σε μοντελοποίηση δυναμικών συστημάτων κλπ).

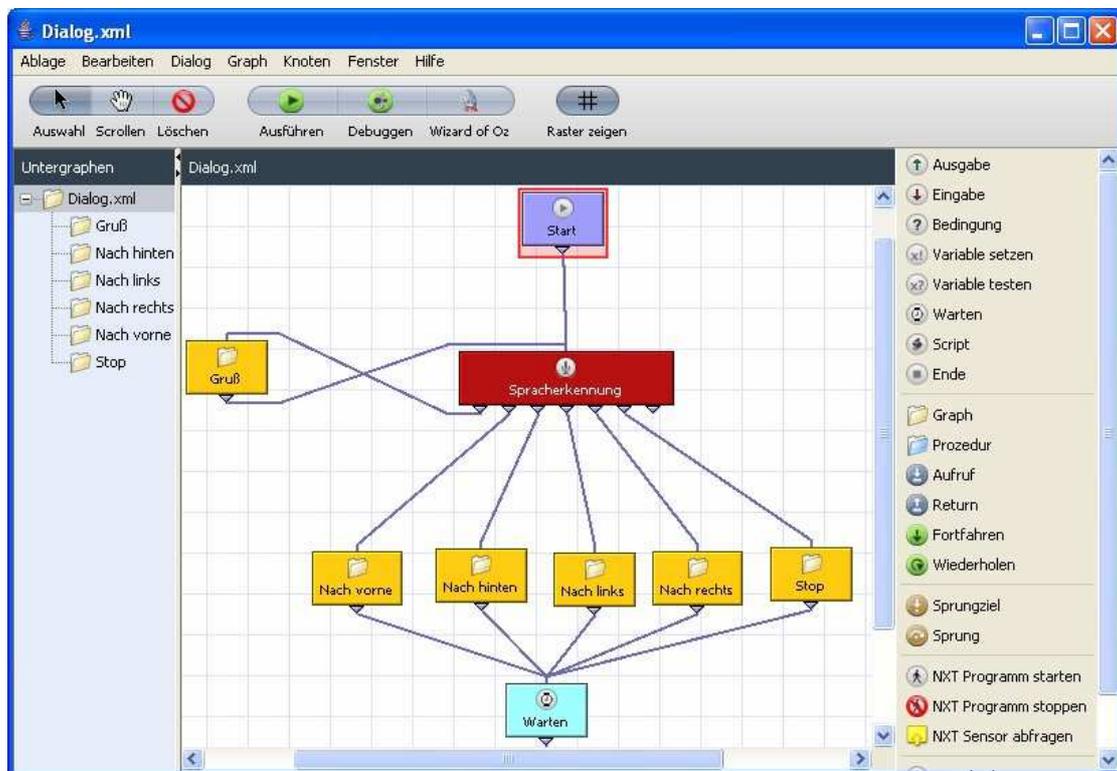
Ο προσομοιωτής παρέχει τις εξής λειτουργίες: [42]

- Προσομοίωση ενός ή πολλών ρομπότ
- Έγχρωμες μονό-σκοπικές κάμερες
- Αισθητήρια επαφής και απόστασης
- Προσομοίωση μέσω διαδικτύου ή συνολικά
- Προγραμματισμός και σε Python, μέσω της γλώσσας jython.
- Επεκτάσιμες διεπαφές χρηστών
- Απλοποιημένη μηχανή περιβάλλοντος



Εικόνα 3.18 Προσομοίωση τεσσάρων ρομπότ [41]

### 3.7 DIALOGOS



Εικόνα 3.19 Το προγραμματιστικό περιβάλλον του DialogOS 74

Το DialogOS είναι ένα γραφικό προγραμματιστικό περιβάλλον για σχεδιασμό υπολογιστικών συστημάτων τα οποία αλληλεπιδρούν μέσω φωνής με τον χρήστη. Οι λεγόμενοι “Dialogs” συνδέονται μεταξύ τους μέσω διαγραμμάτων ροής. Το πακέτο περιλαμβάνει συντομεύσεις ελέγχου των LEGO MINDSTORMS με φωνή. [43]

Χρησιμοποιείται για μαθήματα πληροφορικής σε σχολεία και πανεπιστήμια με σκοπό την εκμάθηση προγραμματισμού και για να εισάγει τους αρχαίους στις βασικές αρχές της αλληλεπίδρασης ανθρώπου/υπολογιστή και στο σχεδιασμό ‘διαλόγων’.

---

## **ΚΕΦΑΛΑΙΟ 4**

### **ΕΚΠΑΙΔΕΥΤΙΚΑ ΡΟΜΠΟΤ (ROBOTICS KITS)**

#### **4.1 ΕΙΣΑΓΩΓΗ**

Τα παρακάτω εκπαιδευτικά συστήματα είναι ειδικά κατασκευαστικά κιτ, με ειδίκευση στη δημιουργία κινητών και αυτόνομων ρομπότ. Υπάρχει πληθώρα διαθέσιμων εκπαιδευτικών ρομποτικών συστημάτων στην αγορά. Ενδεικτικά αναφέρω το LEGO Mindstorms και το Robotics Bioloid.

Αυτά τα συστήματα αποτελούνται από διάφορα δομικά στοιχεία, όπως κινητήρες (ή άλλου είδους ενεργοποιητές), αισθητήρια και ελεγκτικές πλακέτες που ελέγχουν τις εισόδους/εξόδους του ρομπότ, ενώ σε ορισμένες περιπτώσεις είναι διαθέσιμα χωρίς ηλεκτρονικά στοιχεία, δίνοντας στον χρήστη την ευκαιρία να χρησιμοποιήσει τα δικά του.

---

## 4.2 LEGO MINDSTORMS



*Εικόνα 4.1 Ένα ρομπότ φτιαγμένο με το LEGO MINDSTORMS NXT [44]*

### 4.2.1 ΕΙΣΑΓΩΓΗ

Η σειρά κιτ LEGO MINDSTORMS, περιέχει λογισμικό και υλισμικό (hardware) για δημιουργία μικρών, παραμετροποιήσιμων προγραμματιζόμενων ρομπότ. Αποτελείται από έναν προγραμματιζόμενο υπολογιστή “BRICK” όπου ελέγχει το σύστημα, μια ομάδα από αρθρωτά αισθητήρια και κινητήρες, όπως και κομμάτια LEGO από την σειρά Technics για τη δημιουργία των μηχανικών μερών.

Αυτά τα κιτ πωλούνται και χρησιμοποιούνται ως ένα εκπαιδευτικό εργαλείο, αρχικά μέσω μιας συνεργασίας μεταξύ της LEGO και του MIT Media Laboratory. [45]

Η εκπαιδευτική έκδοση των προϊόντων ονομάζεται “LEGO MINDSTORMS for Schools” και συμπεριλαμβάνει το ROBOlab, ένα προγραμματιζόμενο λογισμικό βασισμένο σε γραφική διεπαφή χρηστών (GUI) και ανεπτυγμένο από το Tufts University, χρησιμοποιώντας την μηχανή του National Instruments LabVIEW. Επιπροσθέτως, το υπάρχον λογισμικό μπορεί να αντικατασταθεί με υλικολογισμικό (firmware) και προγραμματιστικές γλώσσες τρίτων, όπως είναι μερικές από τις πιο δημοφιλείς επαγγελματικά χρησιμοποιούμενες στη βιομηχανία ενσωματωμένων συστημάτων υπολογιστών (embedded systems), Java και C. Η μόνη διαφορά μεταξύ των εκπαιδευτικών σειρών (“Challenge Set”) και των εμπορικών σειρών (“Inventor Set”)

---

είναι στο ότι στην τελευταία συμπεριλαμβάνεται ακόμα ένα αισθητήριο φωτός και μερικές ακόμα επιλογές μετάδοσης κίνησης.

#### 4.2.2 ΙΣΤΟΡΙΚΑ ΣΤΟΙΧΕΙΑ

Οι ρίζες του υλισμικού/λογισμικού (hardware/software) βρίσκονται στο πρώτο προγραμματιζόμενο ‘τουβλάκι’ όπου κατασκευάστηκε στο MIT Lab, με προγραμματισμό BRICK LOGO. Το πρώτο οπτικό προγραμματιζόμενο περιβάλλον ονομαζόταν LEGOSheets και δημιουργήθηκε στο University of Colorado το 1994 βασιζόμενο στο AgentSheets. [46]

Η πρώτη γενιά του LEGO MINDSTORMS βασίστηκε γύρω από ένα ‘τουβλάκι’, γνωστό ως RCX (Robotics Command Explorer). Σαν εσωτερικό επεξεργαστή περιέχει τον οκτάμπιτο μικροελεγκτή H8/300. Περιέχει επίσης 32K μνήμης RAM όπου αποθηκεύεται το υλικολογισμικό καθώς και άλλα προγράμματα. Το ‘τουβλάκι’ προγραμματίζεται, κάνοντας upload λογισμικό γραμμένο σε μια από τις πολλές υποστηριζόμενες γλώσσες προγραμματισμού, μέσω ενός PC ή MAC στη RAM του μηχανήματος χρησιμοποιώντας μια ειδική υπέρυθρη διεπαφή (interface).

Με το που γίνεται εκκίνηση του προγράμματος από το χρήστη, το ρομπότ έχει τη δυνατότητα να είναι αυτόνομο, αντιδρώντας στα εσωτερικά και εξωτερικά ερεθίσματα σύμφωνα με τις οδηγίες με τις οποίες προγραμματίστηκε. Επίσης δύο διαφορετικά μηχανήματα μπορούν να επικοινωνήσουν μέσω αυτής της υπέρυθρης διεπαφής, επιτρέποντας έτσι συνεργασίες ή συναγωνισμούς μεταξύ τους. Υπάρχουν τρεις είσοδοι αισθητηρίων και τρεις έξοδοι κινητήρων καθώς και μία LCD οθόνη που δείχνει το επίπεδο μπαταρίας, την κατάσταση των εισόδων/εξόδων, τι πρόγραμμα τρέχει κλπ.

Η πρώτη έκδοση του RCX διαθέτει μια υποδοχή ρεύματος, επιτρέποντας έτσι τη συνεχή λειτουργία, αντί του περιορισμού που υπήρχε με την μπαταρία. Στην δεύτερη έκδοσή του, η υποδοχή αφαιρέθηκε. Τα μηχανήματα της πρώτης έκδοσης είναι δημοφιλή για σχεδιασμούς σταθερών ρομπότ (όπως ρομποτικοί βραχίονες).

Η υπέρυθρη διεπαφή του RCX μπορεί να επικοινωνήσει με τα SpyBots, ScoutBricks, LEGOtrain και NXT. Η πρώτη έκδοσή του έχει συχνότητα δέκτη 38.5 Khz, ενώ η δεύτερη 76Khz (η συχνότητα πομπού μπορεί να είναι οιαδήποτε). Το σήμα αναπαράγεται από έναν εσωτερικό μετρητή του RCX, ενώ επικοινωνεί με τον υπολογιστή μέσω σειριακής ή θύρας USB υπέρυθρων. Λειτουργεί κανονικά στα Win98, ME και 32bit XP, ενώ υπάρχει διόρθωση για πολυπύρηνους επεξεργαστές. Δεν υπάρχει επίσημη υποστήριξη για Win7/Vista, όμως υπάρχουν αναφορές ομαλής λειτουργίας.

---

Τέλος, όλες οι εκδόσεις του RCX έχουν ένα μοναδικό αριθμό τυπωμένο πάνω τους, απαραίτητο για τεχνική υποστήριξη. Το πρώτο RCX που παρήχθη έχει τον αριθμό ‘000001’. [47]

Το πρώτο LEGO MINDSTORMS, περιελάμβανε δύο κινητήρες, δύο αισθητήρια αφής και ένα αισθητήριο φωτός. Το LEGO MINDSTORMS μπορεί να χρησιμοποιηθεί για να χτισθεί μοντέλο ενός ενσωματωμένου συστήματος υπολογιστών με ηλεκτρομηχανικά μέρη, ελεγχόμενα από τον υπολογιστή. Πολλά είδη πραγματικών ενσωματωμένων συστημάτων υπολογιστών, από ελεγκτές ανελκυστήρων μέχρι βιομηχανικά ρομπότ, μπορούν να μοντελοποιηθούν χρησιμοποιώντας το MINDSTORMS.

#### **4.2.3 LEGO MINDSTORMS NXT**

Το LEGO MINDSTORMS NXT κυκλοφόρησε στα τέλη Ιουνίου του 2006 και αντικατέστησε το RCX. Διατίθεται σε δύο εκδόσεις, την εμπορική και την εκπαιδευτική έκδοση, συμπεριλαμβανομένου και του προγραμματιστικού λογισμικού NXT-G.

Το κύριο στοιχείο του είναι το ‘NXT Intelligent Brick’, ένας υπολογιστής σε σχήμα κουτιού που μπορεί να έχει ως είσοδο μέχρι τέσσερις αισθητήρες και να ελέγξει έως τρεις κινητήρες μέσω καλωδίων RJ12. Επίσης έχει μια μονόχρωμη οθόνη LCD 100x64 εικονοστοιχείων και τέσσερα κουμπιά τα οποία μπορούν να χρησιμοποιηθούν για χειρισμούς της διεπαφής του χρήστη μέσω ιεραρχικών μενού. Συμπεριλαμβάνει και ένα μεγάφωνο το οποίο αναπαράγει αρχεία ήχου μέχρι 8 KHz. Η τροφοδοσία παρέχεται από έξι AA μπαταρίες 1,5 V η καθεμία για την εμπορική έκδοση του προϊόντος, ενώ για την εκπαιδευτική έκδοσή του, η τροφοδοσία παρέχεται από μια επαναφορτιζόμενη μπαταρία Li-Ion.



*Εικόνα 4.2 Το NXT Intelligent Brick [48]*

---

Το πακέτο LEGO MINDSTORMS NXT περιλαμβάνει αναλυτικά τα παρακάτω εξαρτήματα [49]:

- Φώτο-αισθητήριο όπου ανιχνεύει το επίπεδο φωτός προς μια κατεύθυνση και συμπεριλαμβάνει ένα LED για φωταγωγήση αντικειμένων.  
Το φώτο-αισθητήριο μπορεί να ανιχνεύσει τιμές ανακλώμενου η φυσικού φωτός. Στο λογισμικό NXT-G το φως ανιχνεύεται σε κλίμακα τοις εκατό, όπου το 100 είναι πολύ φωτεινό ενώ το 0 το απόλυτο σκοτάδι. Αν ρυθμιστεί ανάλογα μπορεί να ορισθεί και σαν αισθητήριο απόστασης.
- Αισθητήριο αφής, το οποίο ανιχνεύει πίεση, χτύπημα ή απελευθέρωση. Τα κουμπιά του NXT μπορούν να προγραμματιστούν ανάλογα, ώστε να λειτουργούν σαν αισθητήρια αφής. Μια μηδενική τιμή δίνεται όταν δεν είναι πατημένα και μια μοναδιαία τιμή δίνεται όταν πατηθούν.
- Αισθητήριο ήχου, το οποίο μετρά το επίπεδο του ήχου σε κλίμακα από 0 έως 100, με το 100 να είναι πολύ δυνατός και το 0 να είναι εντελώς αθόρυβο.
- Αισθητήριο υπερήχων, που μετρά την απόσταση του στόχου προς μέτρηση από το αισθητήριο και μπορεί να ανιχνεύσει κίνηση. Επίσης μπορεί να εμφανίσει την απόσταση σε εκατοστά ή ίντσες. Η μέγιστη απόσταση που μπορεί να μετρήσει είναι 233cm με ακρίβεια 3 εκατοστών. Το αισθητήριο υπερήχων λειτουργεί στέλνοντας υπερηχητικά κύματα τα οποία ανακλώνται σε ένα αντικείμενο και επιστρέφουν πίσω, καταγράφοντας τον χρόνο που χρειάστηκε για να ολοκληρωθεί η διαδικασία.
- Αισθητήριο θερμοκρασίας το οποίο μετρά θερμοκρασίες σε βαθμούς κελσίου η Φαρενάιτ.

---

#### 4.2.4 LEGO MINDSTORMS NXT 2.0



*Εικόνα 4.3 Το LEGO MINDSTORMS NXT 2.0*

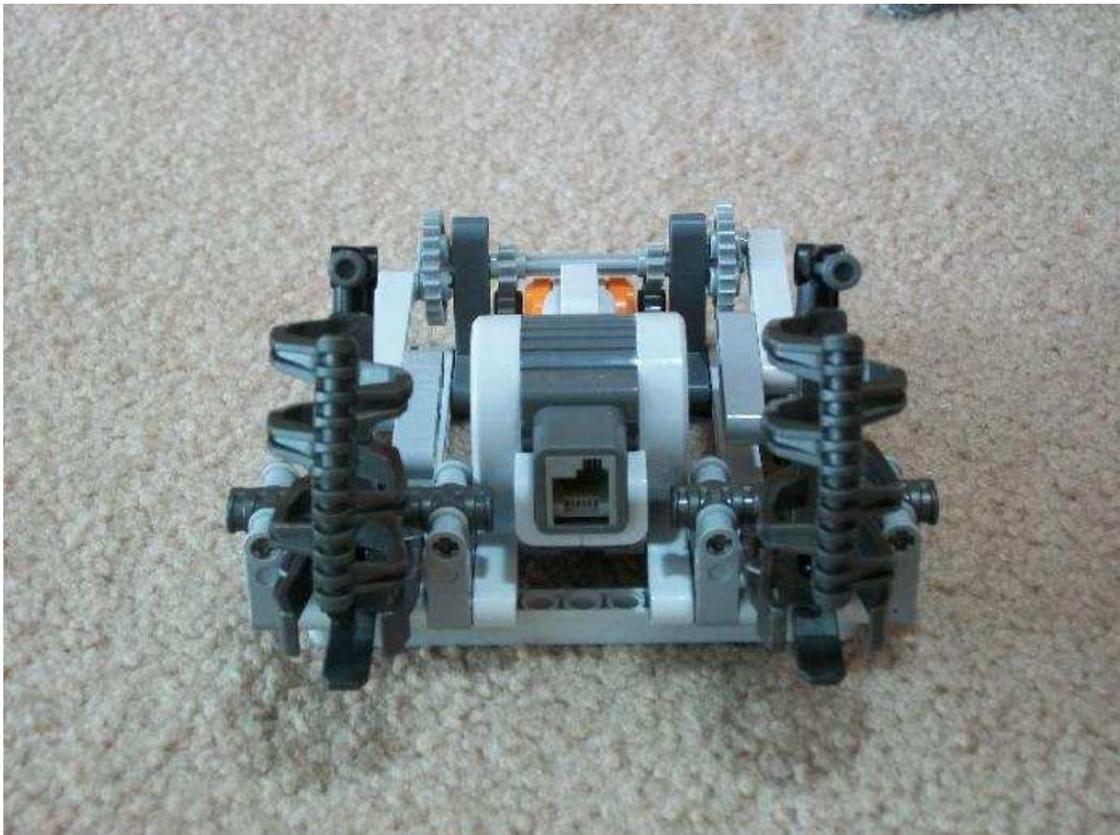
Είναι το νεότερο πακέτο της σειράς, όπου κυκλοφόρησε στις 5 Αυγούστου 2009 και περιέχει, εκτός από τα προαναφερθέντα κομμάτια του NXT, τα εξής παρακάτω χαρακτηριστικά [49]:

- Επεξεργασία ήχου. Μπορεί να ηχογραφήσει κάθε ήχο και έπειτα να προγραμματιστεί ώστε να αναπαράγει το ήδη ηχογραφημένο αρχείο.
- Εμφάνιση εικόνας. Μπορεί να γίνει upload μια εικόνα και να εμφανιστεί στην οθόνη του NXT.
- Αισθητήριο χρώματος RGB. Επιτρέπει στο ρομπότ να αντιληφθεί τα έξι βασικά χρώματα: λευκό, μαύρο, κίτρινο, κόκκινο, πράσινο και μπλε.
- 619 κομμάτια (συμπεριλαμβανομένων και των αισθητηρίων/κινητήρων)
- Το ‘Zamor Sphere Launcher’, ένα εξάρτημα εκτόξευσης σφαιρών με 12 σφαίρες τεσσάρων χρωμάτων (κίτρινο, κόκκινο, πράσινο και μπλε)
- 7 καλώδια σύνδεσης των κινητήρων/αισθητηρίων με το NXT
- Το κουτί NXT. Είναι ο προγραμματιστικός πυρήνας του πακέτου, όπου δέχεται δεδομένα από τα αισθητήρια και ελέγχει τους κινητήρες, ενώ είναι ικανό για

---

συνδέσεις με υπολογιστή μέσω Bluetooth και μαζί με άλλα τρία ξεχωριστά κουτιά NXT.

Επίσης εξαρτήματα όπως πυξίδες, γυροσκόπια, RFID readers και επιταχυνσιόμετρα κατασκευάζονται από τρίτες εταιρείες και πωλούνται από τη LEGO.



*Εικόνα 4.4 To Zamor Sphere Launcher [49]*

Τέλος να σημειωθεί ότι τα αισθητήρια έρχονται προκατασκευασμένα και προγραμματισμένα, ενώ μέσω του ανάλογου λογισμικού μπορεί να αποφασισθεί πως θα γίνει η επεξεργασία των πληροφοριών που λαμβάνονται. Τα αισθητήρια συνδέονται στο NXT μέσω ενός αρθρωτού βύσματος 6 θέσεων όπου υποστηρίζει αναλογικές και ψηφιακές διεπαφές (interfaces), ενώ είναι συμβατό με το RCX.

#### **4.2.5 ΥΠΟΣΤΗΡΙΖΟΜΕΝΕΣ ΠΡΟΓΡΑΜΜΑΤΙΣΤΙΚΕΣ ΓΛΩΣΣΕΣ**

Ο προγραμματισμός των γλωσσών που παρέχει η LEGO, είναι μέσω οθόνης καταχώρησης εντολών (command box), έναντι του κλασικού προγραμματισμού μέσω κώδικα. Αυτές είναι δύο: η RCX Code που συμπεριλαμβάνεται με την εμπορική σειρά που πωλείται στα καταστήματα παιχνιδιών και η ROBO LAB που βασίζεται στο LabVIEW και αναπτύχθηκε από το Tufts University. [49]

---

Από γλώσσες τρίτων εταιρειών έχουμε τις [44]:

- GNAT GPL, επιτρέπει προγραμματισμό σε γλώσσα Ada, για ενσωματώσιμο προγραμματισμό πραγματικού χρόνου
- Ch interpreter, ένας διερμηνέας σε C/C++ για πολλές πλατφόρμες
- Ch Mindstorms NXT Control Package
- C και C++, μέσω του brickOS (παλαιότερα γνωστό ως LegOS)
- C και Assembly, υπό το υλικολογισμικό kit NXTGCC
- Java, υπό το leJOS και το TinyVM
- Not eXactly C, μια προγραμματιστική γλώσσα υψηλού επιπέδου βασισμένη στη c
- Not Quite C
- pbFORTH, επέκταση της FORTH
- pbLua, έκδοση της Lua
- Visual Basic, μέσω της διεπαφής COM+ που παρέχεται με το CD

Πλήρης κατάλογος παρατίθεται στο παράρτημα. [44]

### 4.3 MOBILE ROBOT PROGRAMMING TOOLKIT



Εικόνα 4.5 Αποφυγή εμποδίων σε ρομπότ πάνω σε αναπηρική καρέκλα βασισμένο στο MRPT [50]

Το MRPT είναι μια βιβλιοθήκη C++ ελεύθερου λογισμικού που βασικός της σκοπός είναι να βοηθήσει τους ρομποτικούς ερευνητές να σχεδιάσουν και να εφαρμόσουν αλγορίθμους σχετιζόμενους με το SLAM (Simultaneous localization and mapping), την

---

όραση υπολογιστή (computer vision) και την σχεδίαση κίνησης για αποφυγή εμποδίων. Έχει εφαρμοστεί από διάφορες ερευνητικές ομάδες σε προγράμματα δημοσιευμένα στα μεγαλύτερα επιστημονικά περιοδικά και συνέδρια ρομποτικής [51-55]

Τα κύρια χαρακτηριστικά του είναι[56]:

- Οπτικοποίηση και διαχείριση μεγάλων ομάδων δεδομένων
- Off-line αλγόριθμοι για SLAM μέσω ICP (Αλγόριθμος Iterative closest point), Extended Kalman Filtering ή Rao-Blackwellizer particle filters
- Απόκτηση δεδομένων μέσω ρομποτικών αισθητηρίων

Μέσω επιλογής διάφορων βιβλιοθηκών του MRPT, οι χρήστες μπορούν να αναπτύξουν νέες εφαρμογές.

#### 4.4 ROBOPLUS BIOLOID



*Εικόνα 4.6 Ένα ανθρωποειδές κατασκευασμένο με το kit Bioloid [57]*

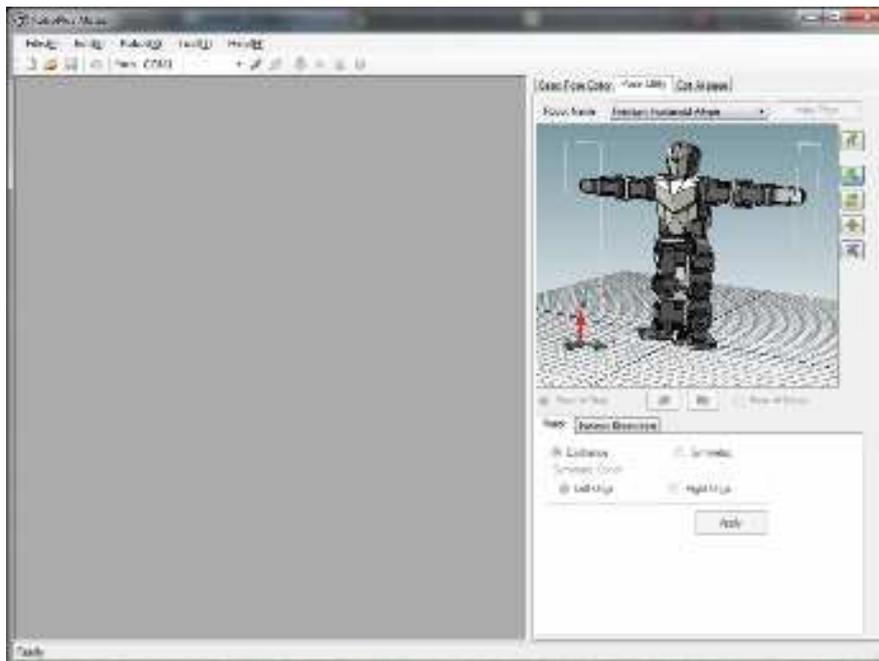
Το RoboPlus Bioloid είναι ένα ρομποτικό kit για χομπίστες και εκπαιδευτικούς σκοπούς που παράγεται από την κορεάτικη εταιρεία ROBOTIS. Η πλατφόρμα αποτελείται από βασικά στοιχεία και από μικρούς σερβομηχανισμούς επωνομαζόμενοι AX-12 Dynamixels και χρησιμοποιούνται με κυκλικό σχηματισμό για την κατασκευή διάφορων ρομπότ, όπως με ρόδες, πόδια ή ανθρωπόμορφα ρομπότ. [57]

---

Προγραμματίζεται με την Roborplus, μια γλώσσα βασισμένη στη C. Είναι παρόμοιο με το kit των LEGO MINDSTORMS και του VEXplorer, χρησιμοποιείται από την Αμερικανική Ναυτική Ακαδημία για εκμάθηση Μηχανολογικών μαθημάτων [58] ενώ έχει μεγάλη παρουσία σε διαγωνισμούς όπως το ROBOCUP.

Υπάρχουν τέσσερις παραλλαγές προς πώληση του Bioloid kit [57]:

- Το Bioloid Beginner Kit που περιέχει κομμάτια και σχέδια για 14 είδη ρομπότ
- Το Bioloid Comprehensive Kit που περιέχει κομμάτια και σχέδια για 26 διαφορετικά είδη ρομπότ
- Το Bioloid Expert Kit που έχει σχεδιαστεί για εκπαιδευτική ή ερευνητική χρήση
- Το Bioloid Premium Kit που είναι αναβαθμισμένη έκδοση του Comprehensive Kit και περιέχει 29 διαφορετικές διαμορφώσεις
- Το Bioloid GP Kit που στοχεύει σε ρομποτικούς διαγωνισμούς



Εικόνα 4.7 Το περιβάλλον ανάπτυξης Roborplus [57]

---

#### 4.5 VEX ROBOTICS DESIGN SYSTEM



*Εικόνα 4.8 Εξαρτήματα του VEX Robotics Design System [59]*

Το VEX Robotics Design System, είναι ένα ρομποτικό κιτ που έχει σκοπό να εισάγει μαθητές και ενήλικες στον κόσμο της ρομποτικής. [59]

Επικεντρωμένο γύρο από το VEX Protobot Kit, προσφέρει δύο αισθητήρια επαφής, δύο διακόπτες, τέσσερις ηλεκτροκινητήρες και ένα σερβο-κινητήρα, από τέσσερα μικρά, μεσαία και μεγάλα λάστιχα καθώς και τα βασικά κομμάτια στήριξης της κατασκευής.

Επίσης διαθέτει στην αγορά πολλά διαφορετικά μοντέλα αισθητηρίων που πωλούνται ξεχωριστά.

---

## ΚΕΦΑΛΑΙΟ 5

# MICROSOFT ROBOTICS DEVELOPER STUDIO (MRDS)

### 5.1 ΕΙΣΑΓΩΓΗ

Το Microsoft Robotics Developer Studio (MRDS) παρέχει ένα περιβάλλον ανάπτυξης για σχεδίαση, εκτέλεση και έλεγχο σφαλμάτων, εξαιρετικά επεκτάσιμων ταυτόχρονα διανεμημένων ρομποτικών εφαρμογών και ενσωματωμένων στο πλαίσιο .NET. Τα πεδία στα οποία επικεντρώνεται το RDS είναι ο συντονισμός, η παρατηρησιμότητα, η διαμόρφωση, η ανάπτυξη και η επαναχρησιμοποίηση των εφαρμογών.

Μαζί με την εγκατάσταση παρέχεται ένας μεγάλος αριθμός οδηγιών και δειγμάτων κώδικα, τόσο για αρχαίους όσο και για προχωρημένους στον τομέα της ρομποτικής.

Τα κυριότερα στοιχεία του MRDS, περιλαμβάνουν [60]:

- Το CCR (Concurrency and Coordination Runtime), όπου είναι ένα προγραμματιστικό μοντέλο το οποίο παρέχει ισχυρά βασικά στοιχεία για ενορχήστρωση εφαρμογών, επιτρέπει τον συντονισμό των μηνυμάτων χωρίς χρήση σηματοδοτών κλπ. Με αυτού του είδους τη λειτουργία, το CCR διευκολύνει την διαχείριση ασύγχρονων λειτουργιών που ασχολούνται με τον συγχρονισμό, εκμεταλλεύεται το παράλληλο υλισμικό και χειρίζεται πιθανά αποσπασματικά σφάλματα.
- Το DSS (Decentralized Software Services) είναι ένα μοντέλο εφαρμογών το οποίο ενσωματώνει κύρια σημεία αρχιτεκτονικής βασισμένης στο διαδίκτυο (γνωστή ως REST- Representational state transfer) με μια προσέγγιση ώστε να δημιουργούνται επεκτάσιμες και μεγάλης απόδοσης εφαρμογές. Οι υπηρεσίες δημιουργούνται ως έγγραφα XML τα οποία είναι προσβάσιμα και προγραμματιζόμενα μέσω ενός φυλλομετρητή. Μέσω της επέκτασης το διαδικτυακού μοντέλου με χειρισμό δομημένων δεδομένων, το DSS καλύπτει την ανάγκη για εφαρμογές οι οποίες να είναι ιδιαίτερα ευανάγνωστες και αραιά συνδεδεμένες μεταξύ τους μέσω ενός κόμβου ή ενός δικτύου.

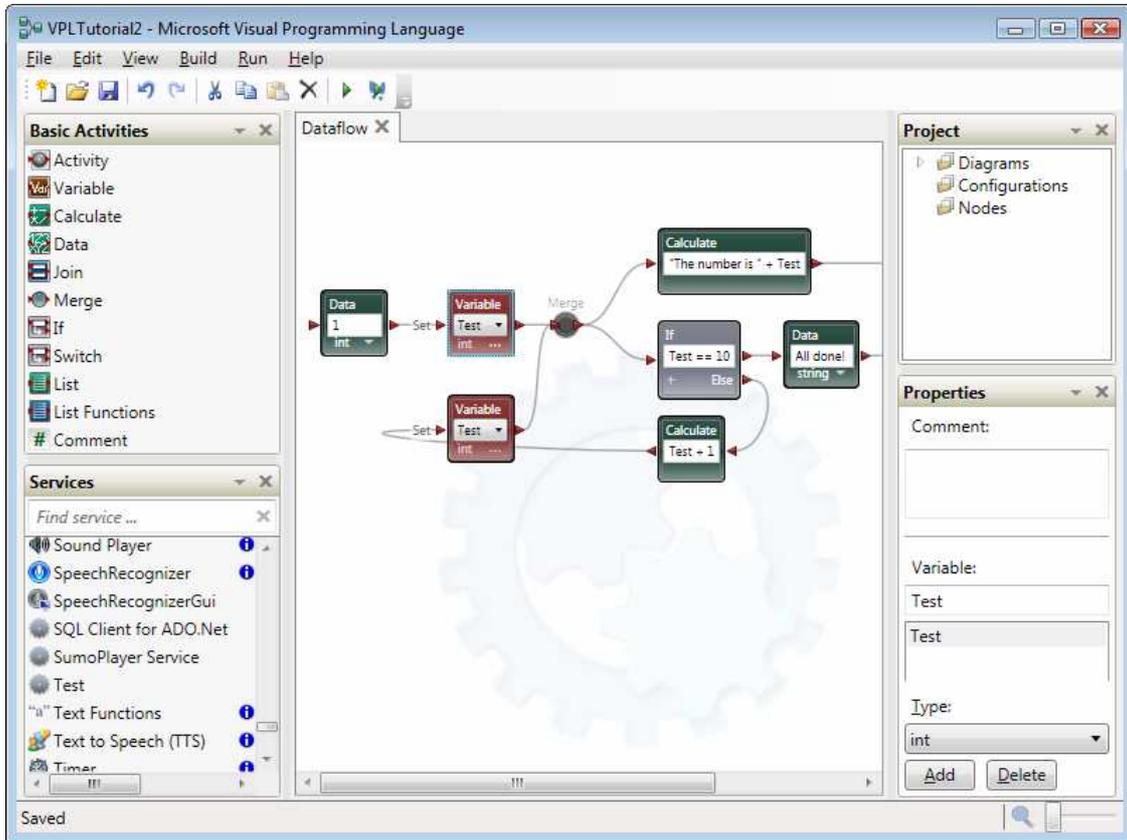
- 
- Η VPL (Visual Programming Language) είναι ένα γραφικό αναπτυξιακό περιβάλλον εφαρμογών CCR/DSS. Αντί του τυπικού προγραμματισμού, στην VPL γίνεται έλεγχος στο ποια στοιχεία θα εκτελεστούν βάσει άφιξης δεδομένων. Οι εφαρμογές είναι συνθέσεις υπηρεσιών, οι οποίες δημιουργούνται απλά μέσω στοιχείων τοποθετούμενων στην επιφάνεια εργασίας της VPL και συνδεδεμένων μεταξύ τους βάσει των εξαρτήσεών τους. Επιπροσθέτως η VPL μπορεί να παράγει υπηρεσίες σε κώδικα C# και έτσι μπορεί να μειωθεί δραματικά ο χρόνος ανάπτυξης του προγράμματος
  - Το DSSME (Decentralized Software Services Manifest Editor) είναι ένα γραφικό περιβάλλον για σύνδεση, επεξεργασία και εκτέλεση εφαρμογών ενός κόμβου ή δικτύου, που αναπτύχθηκαν είτε χρησιμοποιώντας την VPL, είτε γράφτηκαν απευθείας στο Visual Studio. Το DSSME επιτρέπει στο χρήστη να καθορίσει ποιες υπηρεσίες είναι μέρος ποιάς εφαρμογής, πότε πρέπει να εκτελεστούν και ποια πρέπει να είναι η αρχική τους κατάσταση. Απευθύνεται σε επαγγελματίες προγραμματιστές που ασχολούνται σε περιοχές όπως ο βιομηχανικός αυτοματισμός, η οικονομική ανάλυση, τα συστήματα ελέγχου και παρόμοιους τομείς που είναι αναγκαίος ο συντονισμός.
  - Το VSE (Visual Simulation Environment) είναι ένα περιβάλλον δημιουργίας και εκτέλεσης προσομοιώσεων ρομπότ, κόσμων και άλλων οντοτήτων χρησιμοποιώντας τρισδιάστατο rendering υψηλής πιστότητας. Επιπροσθέτως εκτός από την μηχανή για rendering όπου βασίζεται στο πλαίσιο του Microsoft XNA, το VSE παρέχει και μια μηχανή παραγωγής φυσικών ιδιοτήτων επιπέδου πραγματικού κόσμου από την NVIDIA (PhysX). Αυτή η μηχανή επιτρέπει στις οντότητες να αλληλεπιδρούν με το περιβάλλον με έναν πολύ ρεαλιστικό τρόπο. Επίσης μαζί με τις λειτουργίες επεξεργασίας και εκτέλεσης η VSE προσφέρει και χαρακτηριστικά record/playback, τη δημιουργία ενός δυσδιάστατου διαμορφωτή κατόψεων, δυσδιάστατους πίνακες και πινακίδες και πολλά άλλα.

## 5.2 ΠΑΡΟΥΣΙΑΣΗ ΤΗΣ VPL (VISUAL PROGRAMMING LANGUAGE)

Η VPL είναι ένα περιβάλλον ανάπτυξης εφαρμογών βασισμένο σε ένα μοντέλο γραφικού προγραμματισμού ροής δεδομένων. Αντί για την εκτέλεση μιας σειράς εντολών μια προς μια, στον προγραμματισμό ροής δεδομένων έχουμε μια παράσταση ανάλογη με γραμμή παραγωγής σε εργοστάσιο, όπου οι εργάτες εκτελούν μια συγκεκριμένη εργασία με το που φτάσουν τα υλικά. Έτσι έχουμε ως αποτέλεσμα η VPL να είναι κατάλληλη για τον προγραμματισμό ταυτόχρονων εφαρμογών.

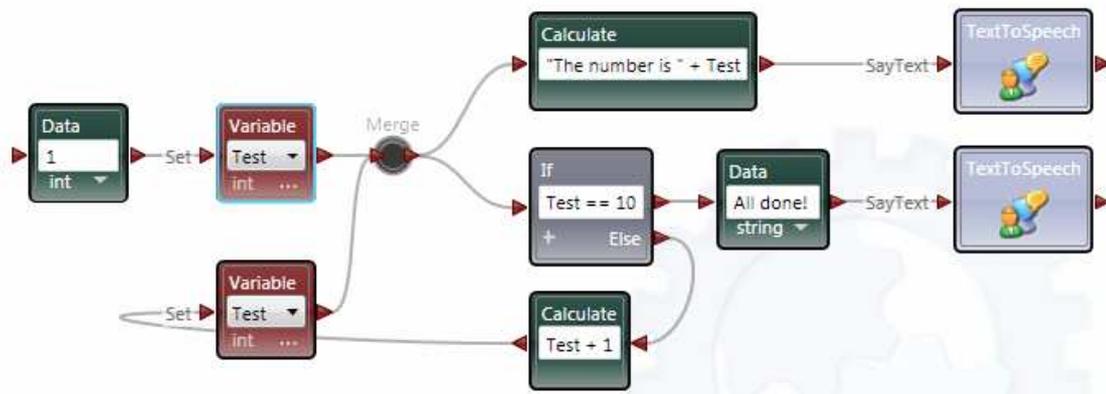
Στοχεύει σε αρχάριους προγραμματιστές με βασικές γνώσεις σε έννοιες όπως μεταβλητές και λογική, χωρίς όμως να περιορίζεται εκεί. Μπορεί να χρησιμοποιηθεί και

από πιο προχωρημένους προγραμματιστές για γρήγορες δοκιμές ή ανάπτυξη κώδικα. Ως αποτέλεσμα η VPL είναι προσιτή σε ένα ευρύ κοινό χρηστών, όπως φοιτητές, χομπίστες, προγραμματιστές εφαρμογών διαδικτύου και επαγγελματίες προγραμματιστές.



Εικόνα 5.1 – Απλό διάγραμμα της VPL [61]

Ένα πρόγραμμα της VPL αποτελείται από μια αλληλουχία από ‘activities’ που αναπαριστώνται ως κουτιά με εισόδους και εξόδους συνδεδεμένες μεταξύ τους. [61]



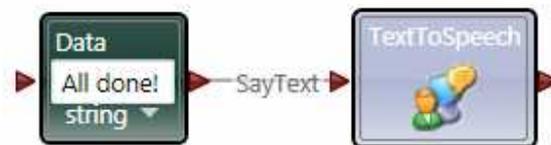
Εικόνα 5.2 Τα κουτιά των ‘activities’ έχουν συνδέσεις που αναπαριστούν μηνύματα από την μια στην άλλη [61]

---

Μπορούν να αναπαραστήσουν ρουτίνες επεξεργασίας ή ελέγχου οι οποίες είναι έτοιμες υπηρεσίες DSS ή ‘activities’ δημιουργημένες στην VPL και να συνδεθούν μεταξύ τους για να μεταφέρουν δεδομένα με την μορφή μηνυμάτων. Η εφαρμογή που δημιουργείται αναφέρεται και ως συμπληρωματικό πρόγραμμα ροής δεδομένων, όπου συντονίζει συνδεδεμένες διεργασίες.

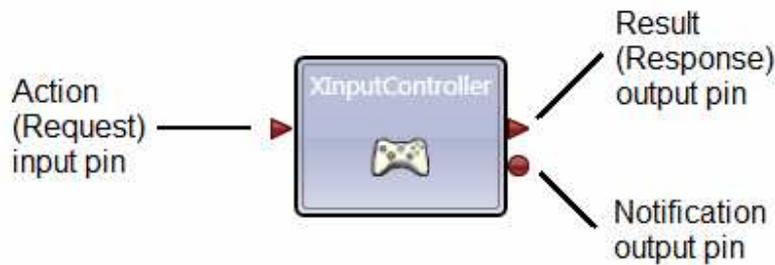
Όπως αναφέρθηκε, οι ‘activities’ συμπεριλαμβάνουν λειτουργίες δημιουργίας άλλων ‘activities’, κάνοντας δυνατή την δημιουργία νέων συνδυάζοντας ήδη υπάρχουσες και χρησιμοποιώντας την νέα σύνθεση ως ξεχωριστό κουτί. Με αυτήν την έννοια μια εφαρμογή της VPL είναι ταυτόχρονα και ‘activity’.

Το εικονίδιο μιας ‘activity’ συνήθως έχει το όνομά της, τα μέρη σύνδεσής της με άλλες, γραφικά που δείχνουν τον σκοπό της και μπορεί να έχει διάφορα στοιχεία που επιτρέπουν στο χρήστη να εισάγει τιμές, να εκχωρήσει ή να μετασχηματίσει δεδομένα που χρησιμοποιούνται σε αυτήν.



Εικόνα 5.3 Οι συνδέσεις αναπαριστούν πράξεις, όπως το SayText [61]

Συνδέονται μέσω pins, όπου το αριστερό αναπαριστά σημείο σύνδεσης για μηνύματα εισόδου που μεταφέρουν δεδομένα, ενώ το δεξί αναπαριστά σημείο σύνδεσης για μηνύματα εξόδου. Τα pins εισόδου είναι σημεία σύνδεσης προς προκαθορισμένες εσωτερικές ρουτίνες, γνωστές σαν ‘actions’, όπου μπορούν να λειτουργούν ως ρουτίνες υπηρεσιών ή εμφωλιασμένες ροές δεδομένων. Με το που λαμβάνεται ένα έγκυρο εισερχόμενο μήνυμα μια ‘activity’ ενεργοποιείται και επεξεργάζεται το μήνυμα, απορροφώντας τα δεδομένα, ενώ για να αναμεταδώσει τα δεδομένα στην έξοδό της, η ‘activity’ πρέπει να αναπαράγει και να τα μεταδώσει μέσω του εξωτερικού της σημείου σύνδεσης. Μπορεί να έχει πολλαπλά pins εισόδου, το καθένα με αντίστοιχο pin εξόδου, όπου αναπαρίστανται ως βέλη που δείχνουν προς το κουτί, ενώ τα pins εξόδου μπορεί να είναι δύο ειδών, καθορίζονται από την ‘activity’ που θα χρησιμοποιήσουν και είναι : είτε έξοδος αποτελέσματος, είτε έξοδος ειδοποίησης, με τα πρώτα να αναπαρίστανται από βέλη που δείχνουν προς τα έξω και τα δεύτερα ως κυκλικά pins σύνδεσης.



Εικόνα 5.4 Είδη pins σύνδεσης [61]

Ένα pin αποτελέσματος χρησιμοποιείται σε περιπτώσεις όπου το μήνυμα εξόδου αποστέλλεται ως αποτέλεσμα κάποιας πράξης ή αιτήματος και το pin ειδοποίησης αποστέλλει πληροφορία ενός εισερχόμενου μηνύματος ή πιο συχνά στέλνει ένα μήνυμα αλλαγής της εσωτερικής κατάστασης του κουτιού. Τα δύο είδη pin εξόδου διαφέρουν και στο πως μεταδίδουν τα μηνύματα δεδομένων. Μια σύνδεση αποτελέσματος σε ένα εισερχόμενο μήνυμα, με το που δεχθεί μια αίτηση, την επεξεργάζεται, αποστέλλει το μήνυμα αποτελέσματος και τελειώνει. Όμως μια σύνδεση ειδοποίησης μπορεί να αναπαραστήσει το ίδιο μήνυμα πολλές φορές, έχοντας ως αποτέλεσμα τη χρήση των συνδέσεων αποτελέσματος για ερώτηση της παρούσας κατάστασης μιας 'activity' και οι συνδέσεις ειδοποίησης να χρησιμοποιούνται για την παροχή συνεχών ενημερώσεων της κατάστασης της 'activity', εκμηδενίζοντας την ανάγκη για συνεχή αποστολή μηνυμάτων κατάστασης της 'activity'.

### 5.3 ΠΑΡΟΥΣΙΑΣΗ ΤΟΥ VSE (VISUAL SIMULATION ENVIRONMENT)

Το VSE σχεδιάστηκε για χρήση σε ανεπτυγμένες εφαρμογές που έχουν μεγάλες απαιτήσεις πιστότητας και οπτικοποίησης. Ταυτόχρονα όμως ένας αρχάριος χρήστης με καθόλου ή πολύ λίγη εμπειρία στον προγραμματισμό, μπορεί να χρησιμοποιήσει τον προσομοιωτή του VSE, για ανάπτυξη εφαρμογών σε ένα περιβάλλον όμοιο με βιντεοπαιχνίδι. Με την ενσωμάτωση του NVIDIA PhysX έχουμε έναν ώριμο και συνεχώς αναπτυσσόμενο προσομοιωτή φυσικών ιδιοτήτων, ενώ η μηχανή rendering του βασίζεται στο πλαίσιο του Microsoft XNA.

Αποτελείται από τα παρακάτω στοιχεία [62]:

- Το Simulation Engine Service: Μια υπηρεσία υπεύθυνη για το rendering των οντοτήτων και την εξέλιξη του χρόνου προσομοίωσης της μηχανής φυσικών ιδιοτήτων. Παρακολουθεί την κατάσταση όλου το προσομοιωμένου κόσμου και παρέχει την εμφάνιση στο χρήστη των υπηρεσιών της προσομοίωσης.

- 
- **Managed Physics Engine Wrapper:** Μια διεπαφή όπου δεν φορτώνει τον χρήστη με την χαμηλού επιπέδου μηχανή φυσικών ιδιοτήτων αλλά προσφέρει περισσότερο συνοπτικό περιβάλλον προσομοίωσης.
  - **Native Physics Engine Library:** Μια βιβλιοθήκη που επιτρέπει την επιτάχυνση του υλισμικού μέσω της τεχνολογίας NVIDIA PhysX και την υποστήριξη της προσομοίωσης στον επεξεργαστή του NVIDIA PhysX.
  - **Entities:** Οι 'entities' αναπαριστούν υλισμικό και φυσικά αντικείμενα στον προσομοιωμένο κόσμο. Υπάρχει ένας προκαθορισμένος αριθμός entities στο RDS που επιτρέπουν στο χρήστη την γρήγορη αξιοποίηση του για την δημιουργία ρομποτικών εφαρμογών σε διάφορα εικονικά περιβάλλοντα.

Αν ο προγραμματιστής δεν θελήσει οπτικοποίηση της προσομοίωσης, μπορεί να επιλέξει την αλληλεπίδραση μόνο με το Managed Physics Engine, όμως συνίσταται αντί αυτού να χρησιμοποιείται πάντα η υπηρεσία της Simulation Engine και να γίνεται ορισμός τροποποιημένων οντοτήτων που απενεργοποιούν το rendering. Αυτός ο τρόπος απλοποιεί την εμμόνη κατάστασης, την αποσφαλμάτωση και την εποπτεία του κώδικα προσομοίωσης σε μεγάλο βαθμό.

Για την προσομοίωση ενός φυσικού αντικειμένου και της υπηρεσίας του συνήθως χρησιμοποιούνται δύο στοιχεία λογισμικού:

**Το Entity:** είναι το στοιχείο λογισμικού που αλληλεπιδρά με τη μηχανή γραφικών και τη μηχανή rendering. Αναδεικνύει το απαραίτητο interface υψηλού επιπέδου προσομοίωσης υλισμικού και αποκρύπτει τις συγκεκριμένες ενέργειες των API φυσικών ιδιοτήτων.

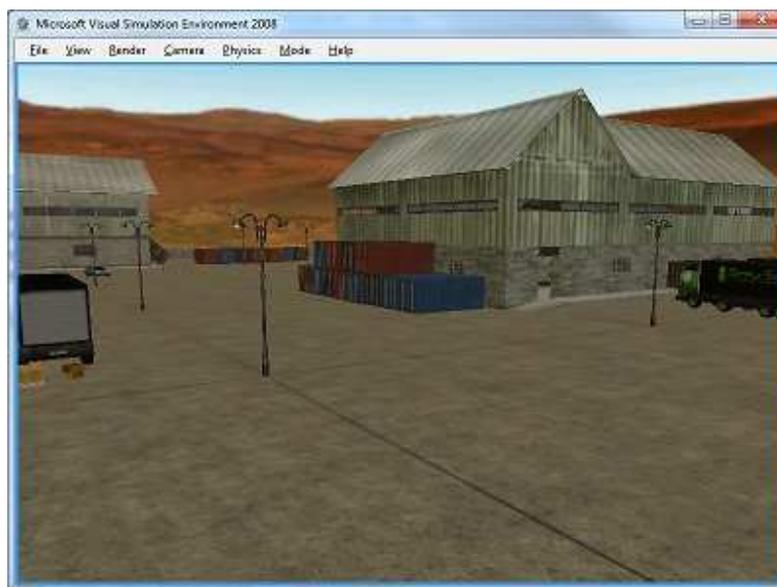
**Service:** Είναι μια υπηρεσία που προσφέρει την απαραίτητη εμφάνιση στην entity και εκτελεί τις ίδιες λειτουργίες που θα εκτελούσε η υπηρεσία που προσομοιώνει.

Υπάρχουν πέντε απλά περιβάλλοντα που προσφέρονται ως δείγματα προσομοίωσης στο VSE. Αυτά είναι [62]:

- Το μοντέλο διαμερίσματος (*Εικόνα 5.5*)
- Το μοντέλο εργοστασίου (*Εικόνα 5.6*)
- Το μοντέλο σύγχρονου σπιτιού (*Εικόνα 5.7*)
- Το μοντέλο εξοχής (*Εικόνα 5.8*)
- Το μοντέλο πόλης (*Εικόνα 5.9*)



*Εικόνα 5.5. Το μοντέλο διαμερίσματος*



*Εικόνα 5.6 Το μοντέλο εργοστασίου*



*Εικόνα 5.7 Το μοντέλο σύγχρονου σπιτιού*

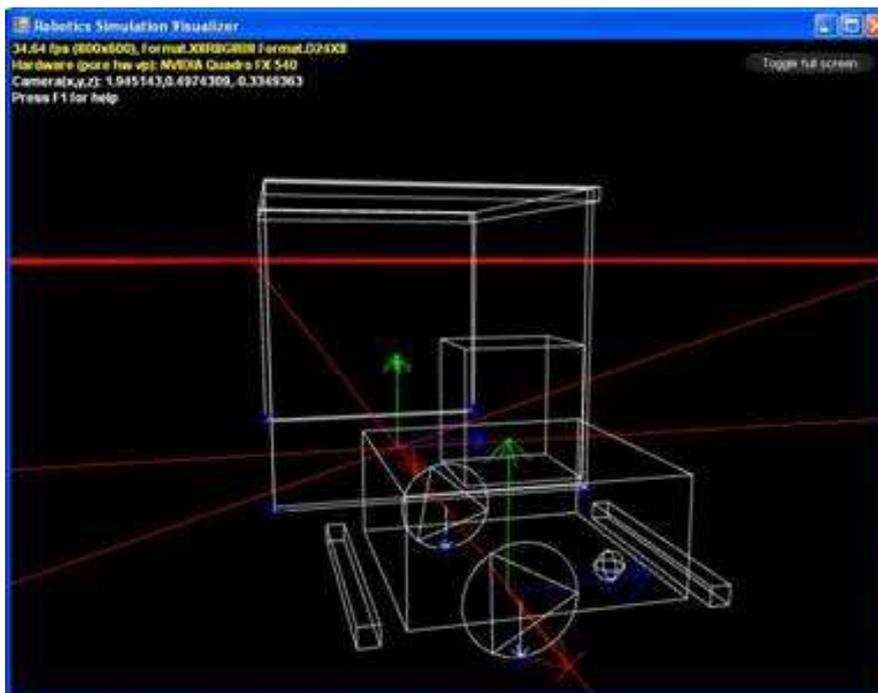
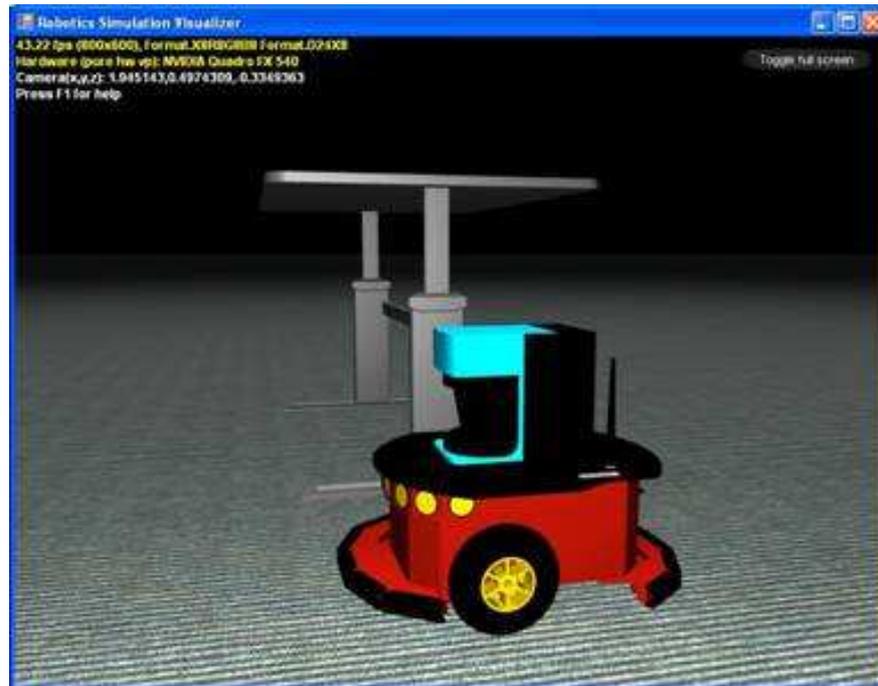


*Εικόνα 5.8 Το μοντέλο εξοχής*

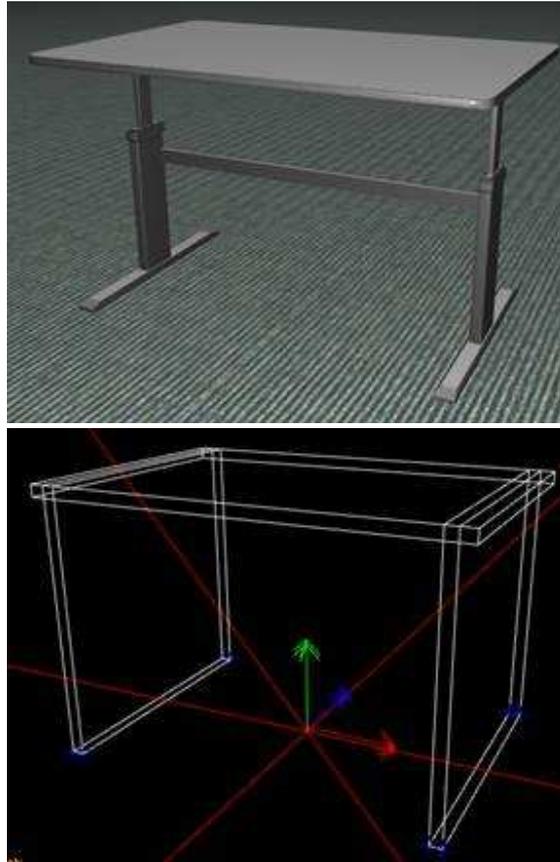


*Εικόνα 5.9 Το μοντέλο πόλης*

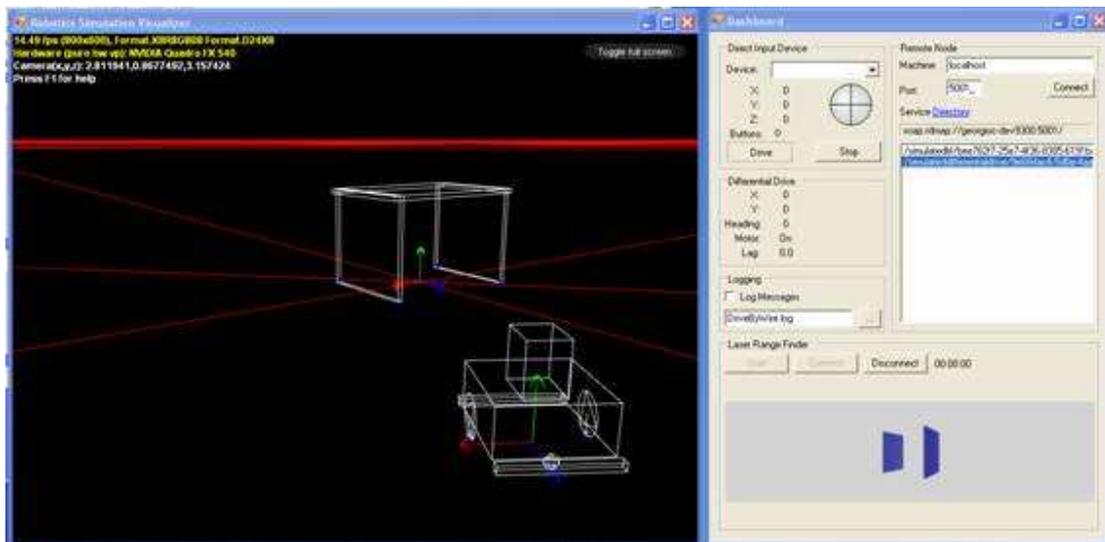
Αυτά τα μοντέλα δείχνουν το μέγεθος της λεπτομέρειας που μπορεί να υλοποιηθεί μέσω του προσομοιωτή. Παρακάτω παρουσιάζονται διάφορες εικόνες του προσομοιωτή, όπως αρθρωτών ρομπότ, περιβαλλόντων πολλών σχημάτων, προσομοιώσεις φυσικών μοντέλων laser, πλεγμάτων αντικειμένων, πολλών ρομπότ μαζί, καθώς και σε περιβάλλοντα ανωμάλων επιφανειών (εικόνες 5.10 έως 5.15) [62]



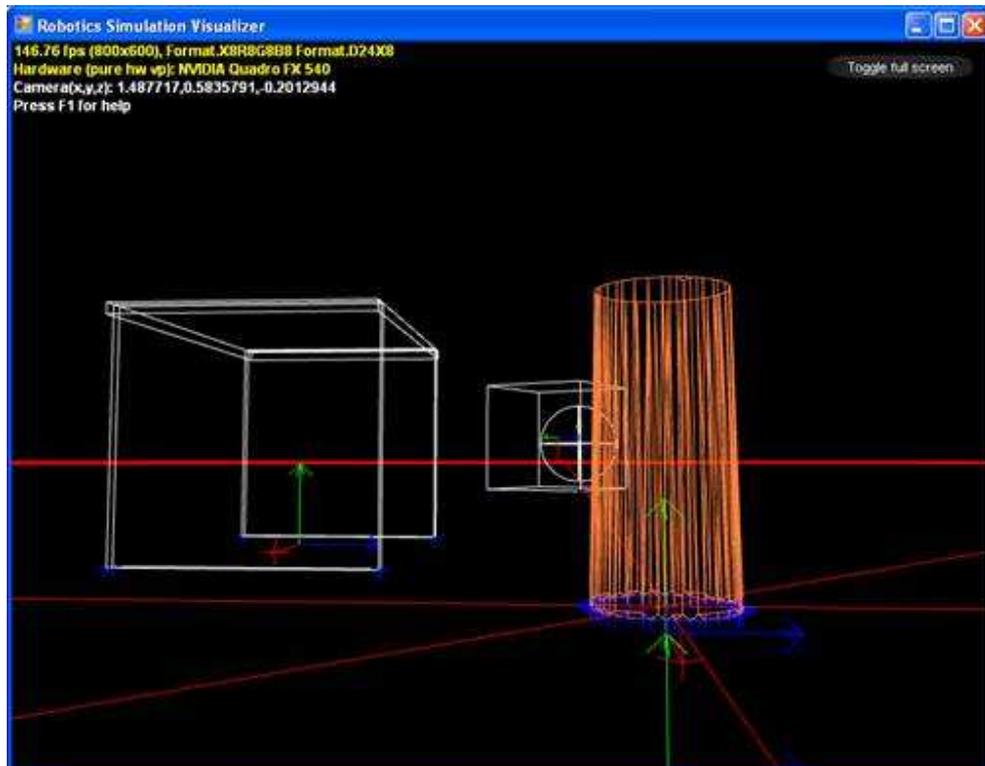
*Εικόνα 5.10 Αρθρωτό ρομπότ με διαφορική κίνηση, laser range finder και bumper array. Η δεύτερη εικόνα είναι από τη βασική εικόνα φυσικών ιδιοτήτων, όπου δείχνει πως ο πίνακας και το ρομπότ προσεγγίζονται ως συμπαγή σχήματα.*



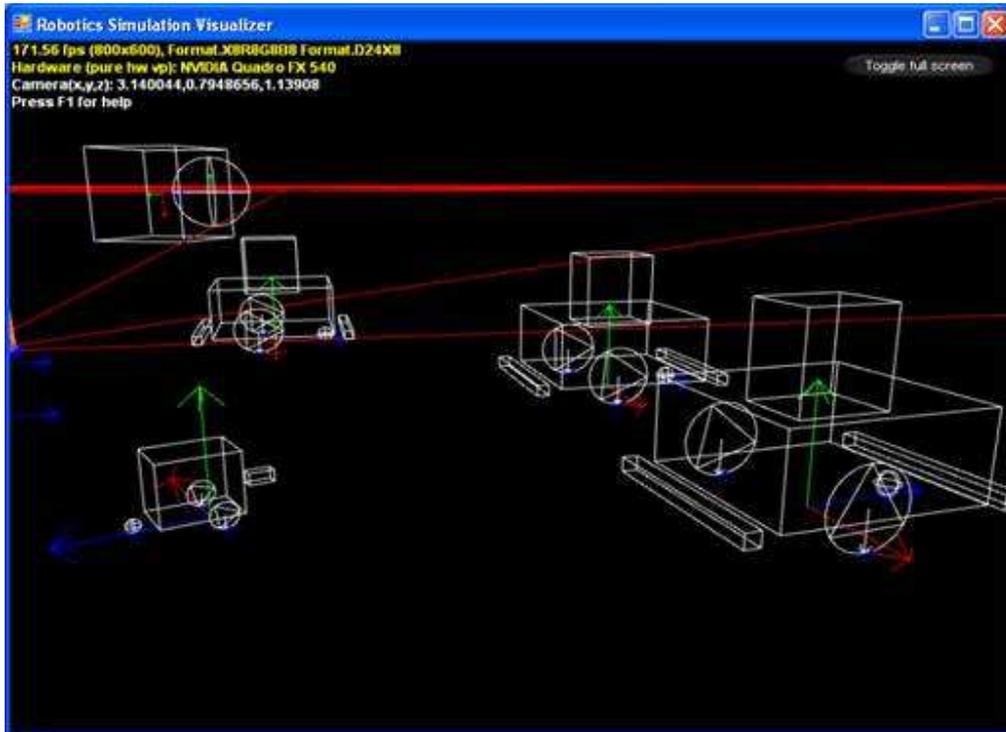
Εικόνα 5.11 Κοντινό πλάνο ενός entity πολύμορφου περιβάλλοντος και του φυσικού του μοντέλου



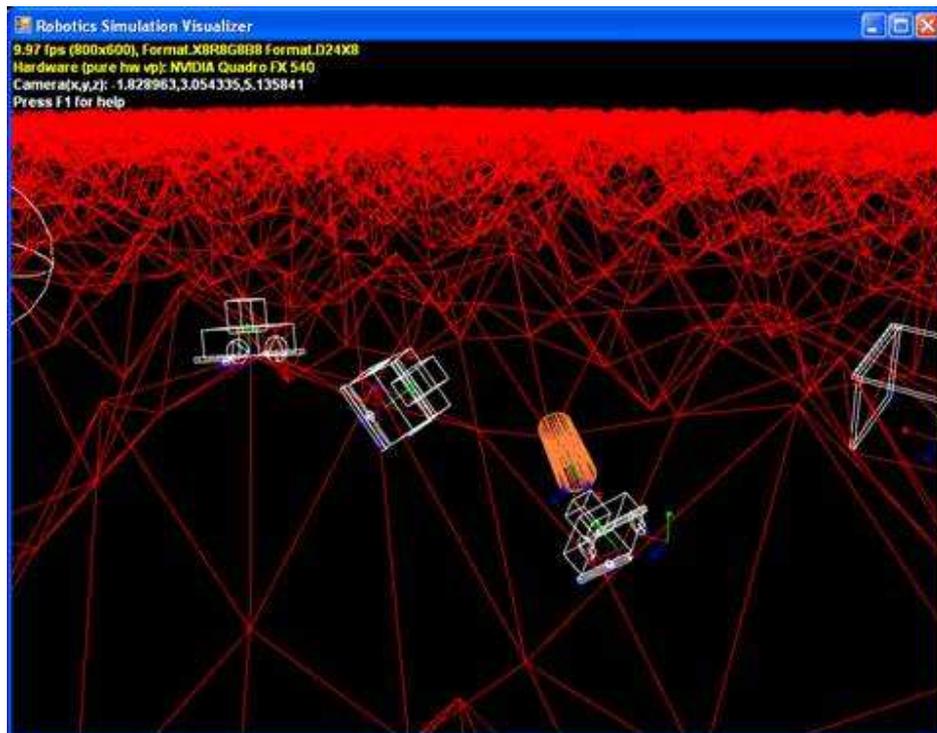
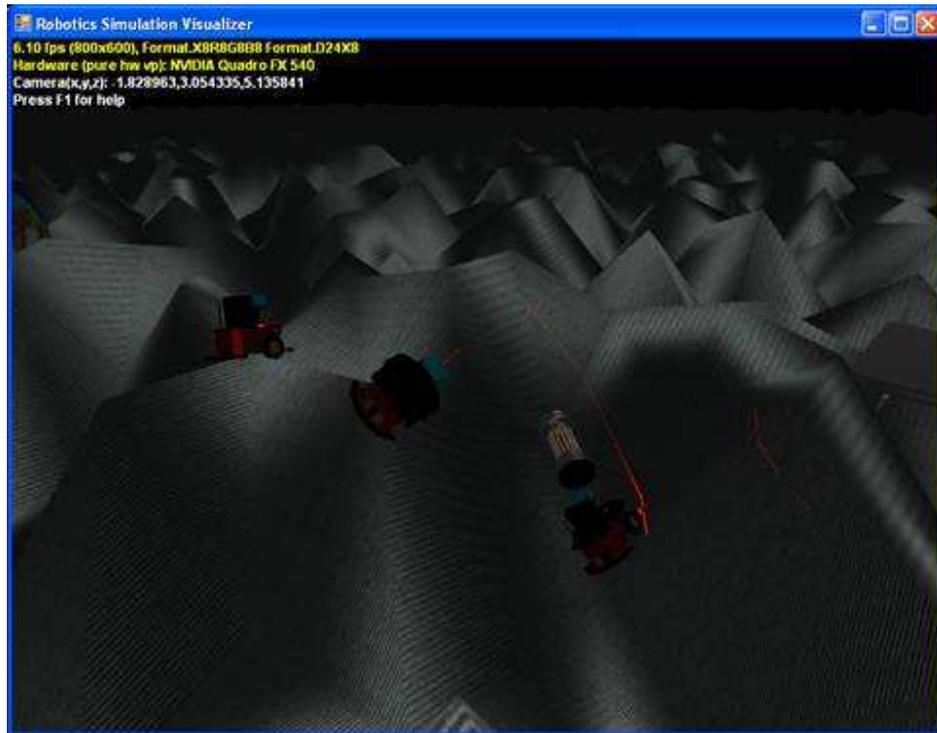
Εικόνα 5.12 Απλό ταμπλό που αναπαριστά προσομοίωση laser σε φυσικό μοντέλο



Εικόνα 5.13 Πολύπλοκο entity πλέγματος, με το απλοποιημένο μοντέλο κυρτού πλέγματος



Εικόνα 5.14 'Φίλοι' -Τρία MobileRobots και ένα LEGO MINDSTORMS.



*Εικόνα 5.15 Ανώμαλη επιφάνεια εδάφους. Παράδειγμα χρήσης ενός entity υψωμένου πεδίου για αναπαράσταση του εδάφους, με τυχαία δείγματα ύψους, κάθε ένα μέτρο. Μπορεί επίσης να εισαχθεί διαφορετικό υλικό για κάθε δείγμα σε προχωρημένη προσομοίωση στην εξοχή. Οι κόκκινες κουκίδες είναι η οπτικοποίηση των σημείων του laser από τις entities του laser range finder.*

---

## 5.4 ΧΡΗΣΙΜΟΠΟΙΩΝΤΑΣ ΤΟ RDS

Προκειμένου να γίνει κατανοητός ο τρόπος λειτουργίας του RDS, θα περιγραφεί συνοπτικά το πώς είναι δυνατόν να γίνει η εγκατάσταση του και η εκτέλεση βασικών λειτουργιών του. Πριν την εγκατάσταση του RDS, πρέπει να βεβαιωθεί ότι είναι εγκατεστημένη η Visual C# που βρίσκεται σε όλες τις εκδόσεις του Visual Studio 2010 καθώς χωρίς αυτή δεν θα μπορέσει να εγκατασταθεί ο βοηθός δημιουργίας νέων υπηρεσιών.

### 5.4.1 ΠΡΩΤΑ ΒΗΜΑΤΑ

Η διαδικασία εγκατάστασης του RDS εξαρτάται από διάφορους παράγοντες: το λειτουργικό σύστημα, την ασφάλεια του συστήματος, τα δικαιώματα διαχειριστή, αν έχει εγκατασταθεί ξανά κλπ. Καθώς δεν είναι δυνατόν να δοθεί οδηγός βήμα προς βήμα, αυτό το κεφάλαιο δίνει τις βασικές οδηγίες που επιλύουν τα περισσότερα προβλήματα εγκατάστασης. Το RDS βασίζεται στον κόμβο DSS για την εκτέλεση και τη διαχείριση υπηρεσιών. Χρησιμοποιώντας είτε το DSShost.exe είτε το DSShost32.exe γίνεται η εκκίνηση του κόμβου και έτσι είναι δυνατή η χρήση του RDS.

Υπάρχουν δύο μέρη για την ενεργοποίηση του DSS. Το ένα είναι η εκτέλεση του κόμβου DSS ώστε να γίνει επιβεβαίωση σύνδεσης στο δίκτυο και το άλλο είναι η εκτέλεση ενός φυλλομετρητή για να γίνει επιβεβαίωση της πρόσβασης σε υπηρεσίες μέσω του κόμβου DSS.

### 5.4.2 ΒΗΜΑΤΑ ΕΚΙΝΗΣΗΣ ΕΝΟΣ ΚΟΜΒΟΥ DSS

Μετά την εγκατάσταση του RDS, για να γίνει εκκίνηση του κόμβου DSS, χρειάζεται να γίνουν τα ακόλουθα βήματα:

Από το 'Start menu', επιλέξτε 'Run DSS Node'. Αυτό θα εμφανίσει ένα παράθυρο γραμμής εντολών και ένα παράθυρο φυλλομετρητή. Κανονικά δεν θα πρέπει να εμφανιστεί κανένα μήνυμα σφάλματος και το DSS να λειτουργήσει κανονικά στον υπολογιστή που εγκαταστάθηκε. Σε αυτήν την περίπτωση παραλείπουμε την υπόλοιπη ενότητα. Αν δεν λειτουργήσει κανονικά, θα αναβοσβήσει γρήγορα ένα παράθυρο, όπου χρειάζεται περαιτέρω διάγνωση. Η συντόμευση του 'Run DSS Node' περιέχεται στην γραμμή εντολής όμοια με το:

```
C:\Users\xxxx\Microsoft Robotics Dev Studio 4\bin\DssHost32.exe"  
/p:50000 /t:50001 /m:"samples\config\UriLauncher.manifest.xml"
```

---

Αυτή η εντολή χρησιμοποιεί την θύρα 50000 για επικοινωνία μέσω HTTP (XML) και την 50001 για απευθείας (δυναδική) επικοινωνία TCP. Σε ορισμένους υπολογιστές η μια ή και οι δύο θύρες είναι ήδη σε χρήση από άλλες υπηρεσίες, οπότε χρειάζεται έλεγχος για επιβεβαίωση. Ανοίξτε ένα παράθυρο γραμμής εντολών DSS από το RDS 'Start menu' και όταν εμφανιστεί εισάγετε την ακόλουθη εντολή για την χειροκίνητη εκκίνηση ενός κόμβου DSS:

```
dsshost /p:50000 /t:50001
```

Αν έχει γίνει λάθος στις παραμέτρους, το DSShost θα εμφανίσει μήνυμα λάθους δείχνοντας το σφάλμα ενώ αν δεν εισαχθούν καθόλου παράμετροι θα εμφανιστεί το :

```
*** Initialization failure: A valid transport port was not specified
```

Αν συνεχίζεται η μη ομαλή λειτουργία, υπάρχει η περίπτωση που στα Windows 7 χρειάζεται επιβεβαίωση του USER ACCOUNT CONTROL (UAC) και ίσως κληθείτε από το Windows Firewall να επιτρέψετε στο πρόγραμμα να ρυθμίσει μια TCP/IP θύρα. Σε αυτήν την περίπτωση επιλέξτε 'unblock'.

```
*** Initialization failure: Could not create TCP Listener.
```

**The two most common causes for this are:**

- 1) You already have another program listening on the specified port**
- 2) You don't have permissions to listen to TCP requests.**

**Exception message: Only one usage of each socket address (protocol/network address/port) is normally permitted**

Το παραπάνω μήνυμα σφάλματος μπορεί να εμφανιστεί αν τρέχει ήδη το DSShost ή η VPL. Σε αυτήν την περίπτωση κλείστε το άλλο πρόγραμμα και ξαναδοκιμάστε. Είναι πιθανό οι αριθμοί από αυτές τις θύρες να χρησιμοποιούνται ήδη από άλλα προγράμματα. Για να το επιβεβαιώσετε, εισάγετε την ακόλουθη εντολή:

```
netstat -a -p tcp
```

---

Κοιτάξτε προσεκτικά στη λίστα των χρησιμοποιούμενων θυρών. Στο Local Address αναζητείστε διευθύνσεις που τελειώνουν σε ':50000' ή σε ':50001'. Αν υπάρχουν κοιτάξτε για άλλους αριθμούς που δεν είναι σε χρήση. Πολλοί χρήστες επιλέγουν τις θύρες 40000 και 40001 για παράδειγμα. Έπειτα δοκιμάστε με τους νέους αριθμούς και αν λειτουργήσει περάστε στην επόμενη ενότητα. Αν όχι η επόμενη πιθανότητα σφάλματος είναι η εμφάνιση ενός μηνύματος στα Windows 7 όπου αναφέρει ότι δεν έχετε δικαίωμα χρήσης των θυρών:

**\*\*\* Initialization failure: Could not start HTTP Listener.**

**The two most common causes for this are:**

- 1) You already have another program listening on the specified port**
- 2) You don't have permissions to listen to http requests.**

**Use the httpreserve command line utility to run using a non-administrator account.**

**Exception message: Access is denied**

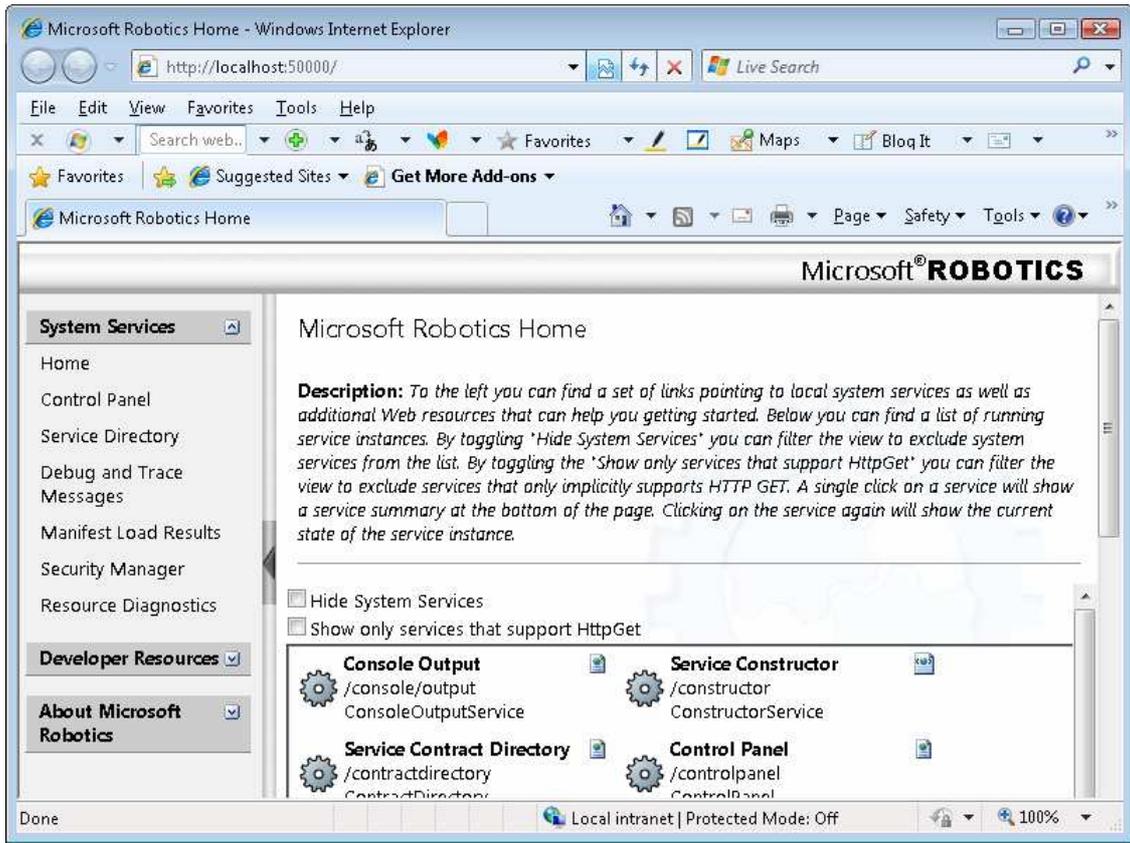
Αυτό γίνεται καθώς δεν έχετε δικαιώματα administrator και θα πρέπει να αποκτήσετε πρόσβαση σε τέτοιου είδους δικαιώματα για να ανοίξετε τις θύρες. Όταν ανοίξετε τις θύρες το DSShost θα τρέχει χωρίς προβλήματα (Οι δεσμεύσεις των θυρών είναι μόνιμες εκτός κι αν αφαιρεθούν, έτσι δεν χρειάζεται η ίδια διαδικασία κάθε φορά που ανοίγετε το RDS)

### **5.4.3 ΧΡΗΣΗ BROWSER ΓΙΑ ΕΠΙΒΛΕΨΗ ΕΝΟΣ ΚΟΜΒΟΥ DSS**

Το DSS έχει σχεδιαστεί έτσι ώστε να μπορεί να χρησιμοποιηθεί από ένα φυλλομετρητή για την εξέταση των υπηρεσιών που τρέχουν σε έναν κόμβο. Το τελευταίο βήμα της εγκατάστασης είναι για την επιβεβαίωση της πρόσβασης σε έναν κόμβο DSS μέσω υπολογιστή.

Υποθέτοντας ότι η θύρα που χρησιμοποιεί ο κόμβος είναι η 50000, πρέπει να γίνει εκκίνηση του φυλλομετρητή και να εισαχθεί η ακόλουθη διεύθυνση: `http://localhost:50000`

Αν έχει γίνει σωστή εκκίνηση του κόμβου DSS, τότε στον φυλλομετρητή θα εμφανιστεί μια σελίδα όπως αυτή:



Εικόνα 5.16 Αν βλέπετε αυτήν την εικόνα το RDS είναι έτοιμο για χρήση

Αν δεν εμφανιστεί η παραπάνω σελίδα και δεν χρησιμοποιείτε τον Internet Explorer, ίσως σας ζητηθεί η εισαγωγή των πιστοποιητικών σύνδεσης, ενώ ακόμα και αν συμβεί και στον IE η διαδικασία είναι η ίδια. Αν γνωρίζετε τον κωδικό εισόδου, μπορείτε να εισαγάγετε τις πληροφορίες και να δείτε την σελίδα. Αν χρησιμοποιείτε έναν υπολογιστή που δεν απαιτείται η σύνδεση, μπορείτε να την προμηθευτείτε από την εντολή **Set** στην γραμμή εντολών. Αυτή η εντολή θα εμφανίζει μια λίστα μεταβλητών όπου θα πρέπει να κοιτάξετε για τις:

```
USERDOMAIN=xxx
USERNAME=xxx
```

Αν επιτύχει, μπορείτε να αλλάξετε τις ρυθμίσεις στο Security Manager από το αριστερό μέρος της σελίδας, αν όμως δεν μπορείτε να συνδεθείτε θα πρέπει να ρυθμίσετε χειροκίνητα τις ρυθμίσεις ασφαλείας.

Μπορείτε να αλλάξετε τις ρυθμίσεις ασφαλείας του DSS από ένα αρχείο ρύθμισης XML το οποίο ονομάζεται SecuritySettings.XML και βρίσκεται στο φάκελο Store.

---

**ΠΡΟΣΟΧΗ:** Οι παρακάτω οδηγίες απενεργοποιούν την ασφάλεια του DSS στον υπολογιστή.

Χρησιμοποιήστε έναν επεξεργαστή κειμένου για να δημιουργήσετε ένα αρχείο κειμένου που να περιέχει τον ακόλουθο κώδικα:

***XML***

```
<?xml version="1.0"?>
<SecuritySettings
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns="http://schemas.microsoft.com/robotics/2008/02/security.html"
>
<AuthenticationRequired>false</AuthenticationRequired>
<OnlySignedAssemblies>false</OnlySignedAssemblies>
<Users />
</SecuritySettings>
```

Αποθηκεύστε το στο store/SecuritySettings.xml και έπειτα εκκινήστε ξανά τον κόμβο DSS. Αυτήν την φορά θα πρέπει να μπορείτε να συνδεθείτε μέσω του browser.

## **5.5 ΠΡΟΓΡΑΜΜΑΤΙΣΤΙΚΕΣ ΡΟΥΤΙΝΕΣ**

Στην ενότητα αυτή θα παρουσιαστεί η Robotics Common. Πρόκειται για μια Dynamic Link Library (DLL) που περιέχει τις σημαντικότερες προγραμματιστικές υπηρεσίες (ρουτίνες) στο MRDS ενώ ο πηγαίος κώδικάς της μπορεί να βρεθεί στο samples/Common. Αν και δεν είναι δυνατή η επεξεργασία του DLL καθώς είναι κύριο συστατικό του MRDS, ο χρήστης μπορεί να δει πως λειτουργούν οι διάφορες υπηρεσίες ώστε να έχει τον βέλτιστο προγραμματιστικό κώδικα χρησιμοποιώντας τις ή να χρησιμοποιήσει τον κώδικα για να δημιουργήσει νέες που να ταιριάζουν απόλυτα στην εφαρμογή που θέλει να υλοποιήσει. Παρακάτω είναι οι υπηρεσίες που περιέχονται σε αυτή την DLL [63]:

### 5.5.1 ΑΝΑΛΟΓΙΚΟ ΑΙΣΘΗΤΗΡΙΟ



Η υπηρεσία ‘Generic Analog Sensor’ υποστηρίζει αναλογικά αισθητήρια τα οποία μετρούν δεδομένα συνεχούς πληροφορίας. Ένα παράδειγμα τέτοιου αισθητηρίου είναι το αισθητήριο φωτός το οποίο αποθηκεύει την ποσότητα φωτός ανά μονάδες χρόνου. Τα αναλογικά αισθητήρια διαχωρίζονται από τα ψηφιακά, τα οποία χρησιμοποιούν διακριτές τιμές για την αναπαράσταση της πληροφορίας εισόδου. Συχνά βέβαια χρησιμοποιείται οποιαδήποτε προσέγγιση για όμοιο τύπο πληροφορίας. Παραδείγματος χάριν οι φωτογραφικές συσκευές με φιλμ είναι αναλογικές συσκευές ενώ οι Web-cams είναι ψηφιακές.

Η συγκεκριμένη υπηρεσία υποστηρίζει δύο εντολές:

Λειτουργία	Περιγραφή
Get	Επιστρέφει την παρούσα κατάσταση
AnalogSensorUpdate	Αλλάζει την κατάσταση του αισθητηρίου ή ειδοποιεί τότε άλλαξε η κατάστασή του

Ενώ η κατάσταση ενός ‘Generic Analog Sensor’ περιέχει τις παρακάτω μεταβλητές:

Όνομα	Τύπος	Περιγραφή
HardwareIdentifier	int	Η θύρα υλισμικού που χρησιμοποιεί το αισθητήριο
NormalizedMeasurement	double	Η κανονικοποιημένη τιμή της εισόδου
Pose	Pose	Η θέση και ο προσανατολισμός του αισθητηρίου
RawMeasurement	double	Η ανεπεξέργαστη τιμή του αισθητηρίου
RawMeasurementRange	double	Το άνω όριο της ανεπεξέργαστης τιμής

Το ‘AnalogSensorUpdate’ μπορεί να χρησιμοποιηθεί ως εντολή αλλαγής της μεταβλητής κατάστασης του αισθητηρίου, όπως της HardwareIdentifier του ή της Pose του. Μπορεί επίσης να χρησιμοποιηθεί ως ειδοποίηση όπου θα παρέχει ενημερώσεις όταν αλλάζει η κατάσταση του αισθητηρίου, παρέχοντας μια ένδειξη της ανεπεξέργαστης τιμής του αισθητηρίου (RawMeasurement) και της κανονικοποιημένης τιμής του (δηλαδή εύρος τιμών 0-1/NormalizedMeasurement), όπου είναι υπολογιζόμενη διαιρώντας την ανεπεξέργαστη τιμή (RawMeasurement) με το άνω όριο της

(RawMeasurementRange). Επίσης με την εντολή Get, επιστρέφεται η τρέχουσα τιμή του αισθητηρίου.

### 5.5.2 ΟΜΑΔΑ ΑΝΑΛΟΓΙΚΩΝ ΑΙΣΘΗΤΗΡΙΩΝ



Η υπηρεσία ‘Generic Analog Sensor Array’ είναι όμοια με την ‘Generic Analog Sensor’ όσον αφορά την δυνατότητα πρόσβασης δεδομένων από αναλογικά αισθητήρια. Όμως με τη χρήση αυτής της υπηρεσίας, αντί να γίνεται η πρόσβαση σε κάθε ένα αισθητήριο ξεχωριστά, γίνεται ως ομάδα. Αυτό είναι πιο αποτελεσματικό όταν υπάρχει μια ομάδα πολλών ίδιων αισθητηρίων.

Η συγκεκριμένη υπηρεσία υποστηρίζει δύο εντολές:

Λειτουργία	Περιγραφή
Get	Επιστρέφει την παρούσα κατάσταση
AnalogSensorReplace	Αλλάζει την κατάσταση του αισθητηρίου στην ομάδα ή ειδοποιεί τότε άλλαξε η κατάστασή του

Η κατάσταση ενός ‘Generic Analog Sensor Array’ είναι μια λίστα καταστάσεων των αισθητηρίων στην ομάδα. Η πρόσβαση στην κατάσταση ενός συγκεκριμένου αισθητηρίου στην ομάδα μπορεί να γίνει με την αναζήτηση του εν λόγω αισθητηρίου από τον πίνακα καταστάσεων. Ενώ η κατάσταση ενός συγκεκριμένου αισθητηρίου στην ‘Generic Analog Sensor Array’ περιέχει τις παρακάτω μεταβλητές:

Όνομα	Τύπος	Περιγραφή
HardwareIdentifier	int	Η θύρα υλισμικού που χρησιμοποιεί το αισθητήριο
NormalizedMeasurement	double	Η κανονικοποιημένη τιμή της εισόδου
Pose	Pose	Η θέση και ο προσανατολισμός του αισθητηρίου
RawMeasurement	double	Η ανεπεξέργαστη τιμή του αισθητηρίου
RawMeasurementRange	double	Το άνω όριο της ανεπεξέργαστης τιμής

Το ‘AnalogSensorUpdate’ μπορεί να χρησιμοποιηθεί ως εντολή αλλαγής της μεταβλητής κατάστασης του αισθητηρίου, όπως της HardwareIdentifier του ή της Pose του. Μπορεί επίσης να χρησιμοποιηθεί ως ειδοποίηση όπου θα παρέχει ενημερώσεις όταν αλλάζει η κατάσταση του αισθητηρίου, παρέχοντας μια ένδειξη της ανεπεξέργαστης τιμής του αισθητηρίου (RawMeasurement) και της κανονικοποιημένης τιμής του

(δηλαδή εύρος τιμών 0-1/NormalizedMeasurement), όπου είναι υπολογιζόμενη διαιρώντας την ανεπεξέργαστη τιμή (RawMeasurement) με το άνω όριο της (RawMeasurementRange). Επίσης με την εντολή Get, επιστρέφεται η τρέχουσα τιμή του αισθητηρίου.

### 5.5.3 ΑΡΘΡΩΤΟΣ ΒΡΑΧΙΟΝΑΣ



Η υπηρεσία ‘Generic Articulated Arm’ υποστηρίζει εφαρμογές για τον έλεγχο ενός αρθρωτού βραχίονα, συμπεριλαμβανομένης της θέσης και του προσανατολισμού των αρθρώσεων και του τελευταίου μέρους του βραχίονα (συνήθως ένα εργαλείο ή μια λαβή). Η συγκεκριμένη υπηρεσία υποστηρίζει τις εξής εντολές:

Λειτουργία	Περιγραφή
ArticulatedArmReplace	Αλλάζει την κατάσταση του βραχίονα ή ειδοποιεί για αλλαγή κατάστασης
Get	Επιστρέφει την παρούσα κατάσταση του βραχίονα
GetEndEffectorPose	Επιστρέφει την θέση και τον προσανατολισμό της απόληξης
SetEndEffectorPose	Θέτει την θέση και τον προσανατολισμό της απόληξης (αλλάζοντας τις κλειδώσεις αν χρειαστεί)
SetJointTargetPose	Θέτει την τελική θέση και τον προσανατολισμό μιας κλειδώσης
SetJointTargetVelocity	Θέτει την ταχύτητα μιας κλειδώσης

Τα ArticulatedArmReplace, SetEndEffectorPose, SetJointTargetPose, SetJointTargetVelocity, παρέχονται ως αιτήματα και έτσι χρησιμοποιούνται για να θέσουν τη θέση και τον προσανατολισμό των μερών του βραχίονα. Επίσης είναι διαθέσιμες και με τη μορφή ειδοποιήσεων όπου παρέχουν ενημερώσεις όταν αλλάξει η κατάσταση μιας κλειδώσης ή απόληξης. Το ArticulatedArmReplace επιστρέφει το σύνολο των κλειδώσεων του βραχίονα (Joints) και τον προσανατολισμό της απόληξης (EndEffectorPose). Η Pose περιγράφεται από την θέση και τον προσανατολισμό: Η θέση καθορίζεται από τις συντεταγμένες των τριών διαστάσεων x,y,z και ο προσανατολισμός είναι η κατεύθυνση που αντικρίζει η κλειδώση ή η απόληξη και καθορίζεται από τέσσερις διαστάσεις x,y,z,w.

## 5.5.4 ΜΠΑΤΑΡΙΑ



Η υπηρεσία ‘Generic Battery’ περιέχει πληροφορίες για την κατάσταση της μπαταρίας και ενδείξεις όταν πέσει κάτω από ένα καθορισμένο επίπεδο.

Η συγκεκριμένη υπηρεσία υποστηρίζει τις εξής εντολές:

Λειτουργία	Περιγραφή
Get	Επιστρέφει την παρούσα κατάσταση της μπαταρίας
BatteryUpdate	Επιστρέφει την παρούσα κατάσταση της μπαταρίας ή υποδεικνύει αλλαγή κατάστασης
CriticalLevelUpdate	Θέτει το επίπεδο προειδοποίησης της μπαταρίας ή υποδεικνύει όταν η ενέργεια της μπαταρίας έπεσε κάτω από το επίπεδο προειδοποίησης

Ενώ η κατάσταση ενός ‘Generic Battery’ περιέχει τις παρακάτω μεταβλητές:

Όνομα	Τύπος	Περιγραφή
MaxBatteryPower	double	είναι πλήρως φορτισμένη (προεπιλεγμένη τιμή 1)
PercentBatteryPower	double	Ποσοστό υπολείπουσας ενέργειας της μπαταρίας
PercentCriticalBattery	double	Ποσοστό ενέργειας μπαταρίας όπου η κανονική λειτουργία του συστήματος μπορεί να προκαλέσει προβλήματα

## 5.5.5 ΑΙΣΘΗΤΗΡΙΑ ΕΠΑΦΗΣ



Η υπηρεσία ‘Generic Contact Sensors’ υποστηρίζει λειτουργίες αισθητηρίων που έχουν πιεζόμενες και μη πιεζόμενες καταστάσεις, όπως προφυλακτήρες ή κουμπιά πίεσης.

Η συγκεκριμένη υπηρεσία υποστηρίζει τις εξής εντολές:

Λειτουργία	Περιγραφή
Get	Επιστρέφει την κατάσταση της υπηρεσίας του αισθητηρίου
ContactSensorReplace	Αλλάζει την κατάσταση μιας ομάδας αισθητηρίων επαφής ή υποδεικνύει πότε άλλαξε η κατάσταση
ContactSensorUpdate	Ανανεώνει ή ειδοποιεί για ανανεώσεις σε ένα αισθητήριο επαφής

Το ContactSensorReplace επιστρέφει μια λίστα με τα αισθητήρια επαφής. Το ContactSensorUpdate παρέχει την κατάσταση ενός μεμονωμένου αισθητηρίου.

Ενώ η κατάσταση ενός 'Generic Contact Sensor' περιέχει τις παρακάτω μεταβλητές:

Όνομα	Τύπος	Περιγραφή
HardwareIdentifier	int	Η θύρα υλισμικού που χρησιμοποιεί το αισθητήριο
Name	string	Περιγραφή του αισθητηρίου
Pose	Pose	Η θέση και ο προσανατολισμός του αισθητηρίου
Pressed	Boolean	Δείχνει αν το αισθητήριο είναι πιεσμένο (T) ή όχι (F)
TimeStamp	DateTime	Η ημερομηνία της ένδειξης

### 5.5.6 ΚΑΜΕΡΑ ΒΑΘΟΥΣ (Kinect)

Το αισθητήριο της 'Depth Camera' παρέχει πληροφορίες για την κάμερα βάθους. Προς το παρόν το μόνο υποστηριζόμενο εξάρτημα κάμερας βάθους είναι το Kinect, κάνοντας αυτή την υπηρεσία όχι και τόσο 'γενικής χρήσεως'.

Η συγκεκριμένη υπηρεσία υποστηρίζει τις εξής εντολές:

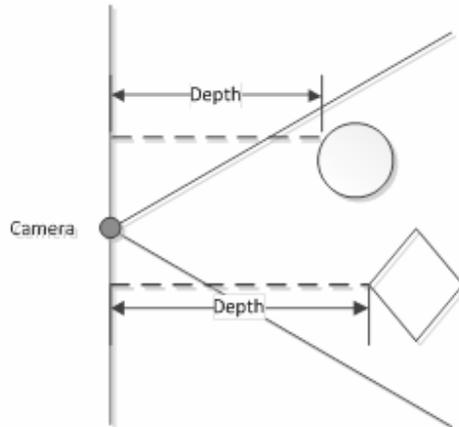
Λειτουργία	Περιγραφή
Get	Επιστρέφει την κατάσταση της υπηρεσίας
HttpGet	Επιστρέφει την κατάσταση της υπηρεσίας με εντολή μέσω Http, συνήθως από φυλλομετρητή
HttpQuery	Χρησιμοποιείται για να ζητήσει εικόνες μέσω φυλλομετρητή στο δίκτυο
Replace	Κατοχυρωμένη. Χρησιμοποιείται εσωτερικά για ειδοποιήσεις
Subscribe	Με αυτήν την εντολή γίνεται εγγραφή στις ειδοποιήσεις που στέλνονται μέσω των μηνυμάτων του Replace

Μπορεί να δοθεί η εντολή Subscribe στην υπηρεσία και θα αποσταλούν εικόνες και ειδοποιήσεις των μηνυμάτων Replace. Ένας άλλος τρόπος είναι η λειτουργία Get.

Η κατάσταση της 'Depth Camera' περιέχει τις παρακάτω μεταβλητές:

Όνομα	Τύπος	Περιγραφή
DepthImage	short[]	Δεδομένα βάθους κατανεμημένα σε σειρά από πάνω προς τα κάτω και από αριστερά προς τα δεξιά ως ένας μονοδιάστατος πίνακας. Οι τιμές είναι σε χιλιοστά
VisibleImage	byte[]	Η ορατή εικόνα σε τύπο αρχείου BGR24. Δεν είναι RGB
DepthSizeImage	size	Οι διαστάσεις βάθους της εικόνας
Pose	pose	Καθορίζει τη θέση και τον προσανατολισμό της κάμερας. Αυτό πρέπει να αρχικοποιηθεί σε ένα αρχείο ρύθμισης
ProjectionMatrix	matrix	Πίνακας ο οποίος καθορίζει την μετατροπή του πραγματικού τρισδιάστατου κόσμου σε χώρο της κάμερας βάθους
InverseProjectionMatrix	matrix	Το αντίστροφο του ProjectionMatrix
Timestamp	Datetime	Πότε κατεγράφη η εικόνα
FieldOfView	double	Οριζόντιο οπτικό πεδίο της κάμερας (σε rad)
MaximumRange	double	Μέγιστη τιμή βάθους (σε μέτρα)
MinimumRange	double	Ελάχιστη τιμή βάθους (σε μέτρα)
FurtherThanMaxDepthValue	short	Τιμή ενός εικονοστοιχείου που είναι μακρύτερα της μέγιστης τιμής βάθους
NearerThanMinDepthValue	short	Τιμή ενός εικονοστοιχείου που είναι κοντύτερα από την ελάχιστη τιμή βάθους.
ImageMode	DepthCamSensor ImageMode	Κατοχυρωμένο για μελλοντική χρήση

Η κατάσταση της υπηρεσίας περιλαμβάνει εκτός από τα δεδομένα βάθους (που είναι σε μορφή πίνακα) και δεδομένα RGB. Τα δεδομένα βάθους είναι 11 bit και αναπαριστώνται σε χιλιοστά μετρούμενα από την απόσταση της κάμερας, όπως φαίνεται στην παρακάτω εικόνα:



Εικόνα 5.17 Μέτρηση δεδομένων βάθους

Αν γίνει εκκίνηση της υπηρεσίας με ενεργοποιημένο την δυνατότητα ανίχνευσης του σκελετού του αντικειμένου ('Skeleton Tracking'), στα τελευταία bits περιλαμβάνεται και ένας δείκτης στα δεδομένα βάθους. Έτσι πρέπει να ληφθεί υπόψη όταν είναι ενεργοποιημένο το 'Skeleton Tracking' αυτά τα δεδομένα μετακινούνται προς τα αριστερά κατά τρία bit.

Ο χώρος του βάθους της εικόνας αποτελείται από συντεταγμένες εικονοστοιχείων x & y (κανονικοποιημένες στο χώρο του 0 έως 1) και τιμή βάθους z σε χιλιοστά. Οι τιμές των εικονοστοιχείων δεν αντιπροσωπεύουν κάτι στον πραγματικό κόσμο, ενώ οι τιμές του βάθους των εικονοστοιχείων για διάφορους λόγους δεν συμβαδίζουν τελείως με τα εικονοστοιχεία RGB αλλά η υπηρεσία παρέχει έναν τρόπο αντιστοίχισης των εικονοστοιχείων βάθους εικόνας με αυτά του τύπου RGB.

### 5.5.7 ΔΙΑΦΟΡΙΚΟΣ ΟΔΗΓΟΣ

Η υπηρεσία 'Generic Differential Drive' (GDD) καθορίζει τον έλεγχο ρομπότ διαφορικής οδήγησης. Ένας διαφορικός οδηγός αποτελείται συνήθως από δύο τροχούς οι οποίοι μπορούν να λάβουν εντολές ανεξάρτητα, αλλά αυτή η έννοια μπορεί να εφαρμοστεί και σε άλλες μορφές κίνησης όπως ρομπότ με πόδια, πχ ένα εξάποδο με σειρές τριών ποδιών σε κάθε πλευρά. Χαρακτηριστικό του διαφορικού οδηγού είναι ότι δίνοντας εντολές αντίθετης κατεύθυνσης σε κάθε μέλος του, το ρομπότ μπορεί να περιστραφεί επί τόπου.

Το GDD αποτελείται από την κατάσταση της υπηρεσίας και από τις ιδιότητες της λειτουργίας του, δηλαδή τα DriveState.cs και το DriveTypes.cs. Όλα τα ρομπότ που συνεργάζονται με το GDD, μπορούν απευθείας να ελεγχθούν μέσω εφαρμογών που εκτελούνται κάτω από το MRDS, όπως το 'simple dashboard' λόγω του ότι το GDD παρέχει μια κοινή προδιαγραφή για την προγραμματιστική διεπαφή του.

---

Το MRDS παρέχει ένα εργαλείο γραμμής εντολών επονομαζόμενο DssNewService και έναν οδηγό του Visual Studio τα οποία μπορούν να χρησιμοποιηθούν για δημιουργία υπηρεσιών που ενσωματώνονται σε μια ήδη υπάρχουσα υπηρεσία.

#### **5.5.7.1 ENTOΛΕΣ GENERIC DIFFERENTIAL DRIVE**

Παρακάτω παραθέτονται οι εντολές της υπηρεσίας GDD:

##### **5.5.7.1.1 ALL STOP**

Ακυρώνει κάθε εκτελούμενη εργασία και θέτει την ισχύ στο μηδέν και στους δύο τροχούς. Επίσης για λόγους ασφαλείας ο οδηγός απενεργοποιείται, αν χρειαστεί να ξαναεκτελεστεί χρειάζεται να ενεργοποιηθεί με την εκτέλεση της εντολής EnableDrive και να τεθεί σε 'αληθή' κατάσταση. Μέχρι να ενεργοποιηθεί ο οδηγός, όλες οι εντολές θα επιστρέφουν σφάλμα. Αυτή η λειτουργία πρέπει να καλείται μόνο σε περιπτώσεις ανάγκης.

##### **5.5.7.1.2 ENABLE DRIVE**

Η εντολή 'Enable Drive' χρησιμοποιείται για να ενεργοποιήσει (true) ή να απενεργοποιήσει (false) τον οδηγό. Αρχικά πρέπει να ενεργοποιείται ο οδηγός με μια παράμετρο της άλγεβρας Boole. Όταν απενεργοποιείται όλες οι εντολές κίνησης αποτυγχάνουν και επιστρέφουν σφάλμα.

##### **5.5.7.1.3 SETDRIVEPOWER**

Η εντολή 'SetDrivePower' δέχεται ρυθμίσεις ισχύος για τους αριστερούς και δεξιούς τροχούς ως τιμές μεταξύ του -1.0 και 1.0. Οι αρνητικές τιμές υποδεικνύουν ότι ο τροχός πρέπει να περιστραφεί προς τα πίσω και οι θετικές τιμές ότι πρέπει να περιστραφεί μπροστά. Για την επίτευξη περίπλοκων ρουτινών οδήγησης, μπορούν να σταλούν επαναλαμβανόμενες εντολές του 'SetDrivePower' για να διαφοροποιηθεί η ισχύς του κινητήρα. Κάθε νέα εντολή αλλάζει απλά την ισχύ που πηγαίνει στους κινητήρες και αντικαθιστά την προηγούμενη. Επίσης βάζοντας μηδενική τιμή και στις δύο ρόδες σημαίνει ότι το ρομπότ πρέπει να σταματήσει, όμως κάποια ρομπότ που είναι σε υψηλή ταχύτητα, μπορεί να προχωρήσουν για λίγο πριν συμβεί αυτό. Επιπροσθέτως, δίνοντας ισοδύναμη ισχύ αλλά προς διαφορετική κατεύθυνση στους δύο τροχούς, έχει ως αποτέλεσμα την περιστροφή των ρομπότ.

Δεν υπάρχει κάποια γραμμικότητα στη σχέση τιμής της ισχύος και της ταχύτητας του ρομπότ και η μόνη σχέση είναι ότι η μεγαλύτερη απόλυτη τιμή κάνει το ρομπότ να τρέχει γρηγορότερα. Υπάρχει η δυνατότητα να χρησιμοποιείται ένα σημείο κάτω από τη μονάδα όπου να αναπαριστά τη μέγιστη ταχύτητα του ρομπότ και μερικά ρομπότ μπορεί να υποστηρίξουν διακριτές τιμές ισχύος, ενώ τέλος είναι δυνατό το ρομπότ να έχει μόνο δύο διακριτές τιμές 'ανοικτό' (προχωρά) και 'κλειστό' (σταματά).

---

Λόγω των ιδιοτήτων των πραγματικών κινητήρων, είναι πιθανό ότι για πολύ μικρές τιμές, παραδείγματος χάριν 0,1, το ρομπότ να μην έχει αρκετή ισχύ για να κινηθεί, όπως και ότι η τιμή της μονάδος μπορεί να κάνει το ρομπότ να τρέξει υπερβολικά γρήγορα. Μόνο μέσω δοκιμών μπορεί να επιτευχθεί η ιδανική τιμή για κάθε εφαρμογή κινητήρα.

Τέλος για τα ρομπότ που δεν έχουν οδηγό δύο τροχών, οι ρυθμίσεις μπορούν να αναπαριστούν την αριστερή πλευρά και την δεξιά πλευρά της κίνησης.

#### **5.5.7.1.4 SETDRIVESPEED**

Η εντολή SetDriveSpeed θέτει την ταχύτητα του οδηγού σε συγκεκριμένη τιμή μέτρων ανά δευτερόλεπτο (m/s), καθώς όμως δεν υπάρχει ακριβής τρόπος μέτρησης της ταχύτητας για το ρομπότ, αρκετά ρομπότ δεν υποστηρίζουν αυτήν την εντολή.

Η ταχύτητα του ρομπότ εξαρτάται από τη συνολική γεωμετρία των τροχών του, τη σχέση της ισχύος με την περιστροφική ταχύτητα των κινητήρων και πιθανώς το επίπεδο ενέργειας της μπαταρίας. Έτσι ακριβής τρόπος μέτρησης γίνεται στα πολύ πιο ακριβά μοντέλα. Για την ακριβή μέτρηση της ταχύτητας χρειάζεται επίσης ανάδραση από τους τροχούς. Αυτό παρέχεται από τους κωδικοποιητές του τροχού και ο έλεγχος έπειτα γίνεται από έναν ελεγκτή (πχ P.I.D.) όπου για λόγους απόδοσης πρέπει να είναι ενσωματωμένα στο ρομπότ.

#### **5.5.7.1.5 DRIVEDISTANCE**

Η εντολή DriveDistance χρησιμοποιείται για να μετακινήσει το ρομπότ μπροστά ή πίσω σε μια απόσταση μετρούμενη σε μέτρα. Υπάρχουν τρεις παράμετροι: Distance, Power και DriveDistanceStage.

Η παράμετρος Distance πρέπει πάντα να έχει θετική τιμή σε μέτρα. Η παράμετρος Power καθορίζεται με τον ίδιο τρόπο με την SetDrivePower. Η DriveDistanceStage πρέπει να αρχικοποιηθεί πριν δοθεί εντολή εκτέλεσης, μέσω της τιμής DriveStage.InitialRequest. Αυτή η τιμή ανανεώνεται κάθε φορά που στέλνονται μηνύματα ειδοποίησης στην υπηρεσία GDD και αν δεν γίνει αρχικοποίηση, τότε τίθεται η προεπιλεγμένη τιμή της. Καλή πρακτική θα ήταν πάντα να τίθεται τιμή, καθώς αν αποτύχει να τεθεί η προεπιλεγμένη τιμή για οποιοδήποτε λόγο, η εντολή δεν γίνεται δεκτή και επιστρέφει σφάλμα.

Αυτή η εντολή δεν ενσωματώνεται σε όλες τις υπηρεσίες οδηγών διότι απαιτεί κωδικοποιητές στους τροχούς. Κάποιες διαφορετικές υπηρεσίες μπορούν ίσως να ενσωματώσουν τέτοιου είδους λειτουργίες χρησιμοποιώντας ένα χρονιστή για να εκτιμήσει την κίνηση, χωρίς όμως να είναι αξιόπιστη μέθοδος μέτρησης.

---

#### **5.5.7.1.6 ROTATEDEGREES**

Ο σκοπός της εντολής RotateDegrees είναι να περιστρέψει το ρομπότ στο ίδιο σημείο κατά μια καθορισμένη γωνία. Οι γωνίες μετρώνται σε μοίρες με θετικές τιμές να αντιστοιχούν σε στροφή αριστερά καθώς βλέπουμε το ρομπότ από πάνω. Υπάρχουν τρεις παράμετροι: Degrees, Power και RotateDegreesStage. Οι δύο πρώτες λειτουργούν όπως και οι παράμετροι της DriveDistance, ενώ η RotateDegreesStage πρέπει να θέσει την DriveStage.InitialRequest όταν χρησιμοποιείται.

Δεν μπορεί να είναι σίγουρο ότι το ρομπότ θα περιστραφεί ακριβώς στη γωνία που του δόθηκε η εντολή, ενώ δεν χρειάζεται να υποστηρίζεται τιμή μεγαλύτερη των 360 μοιρών. Ανάλογα με το υλισμικό η ακρίβεια των περιστροφών ποικίλει από ρομπότ σε ρομπότ. Χαμηλότερα επίπεδα ισχύος, έχουν ως αποτέλεσμα ακριβέστερες περιστροφές, όμως μερικά ρομπότ ίσως σταματήσουν αν το επίπεδο είναι πολύ χαμηλό, με αποτέλεσμα να μην ολοκληρώσουν την περιστροφή. Δίνοντας οποιαδήποτε εντολή κίνησης (ακόμα και μια νέα RotateDegrees) θα ακυρώσει την ήδη εκτελούμενη. Βέβαια εξαρτάται από το υλισμικό και μερικά ρομπότ δεν θα μπορέσουν να ακυρώσουν την εντολή.

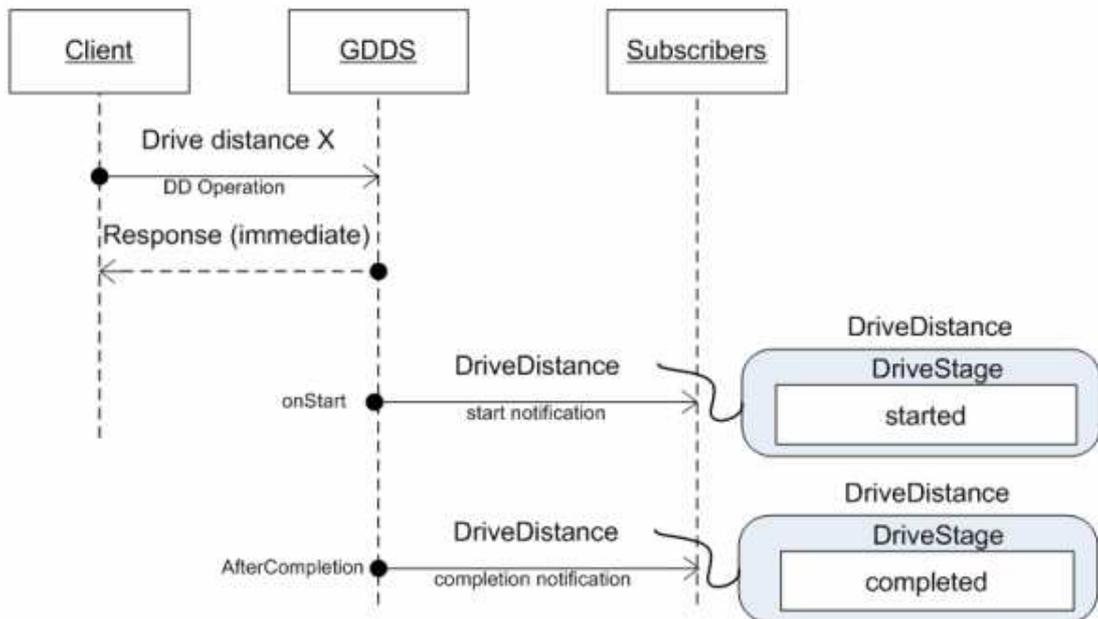
Τέλος να σημειωθεί ότι ούτε αυτή η εντολή δεν ενσωματώνεται σε όλες τις υπηρεσίες οδηγών για τον ίδιο λόγο με την DriveDistance.

#### **5.5.7.2 ΕΙΔΟΠΟΙΗΣΕΙΣ ΤΗΣ ΥΠΗΡΕΣΙΑΣ GDD**

Υπάρχει επιλογή καταγραφής μηνυμάτων του ‘Generic Differential Drive’.

#### **ΕΛΕΓΧΟΜΕΝΕΣ ΚΙΝΗΣΕΙΣ**

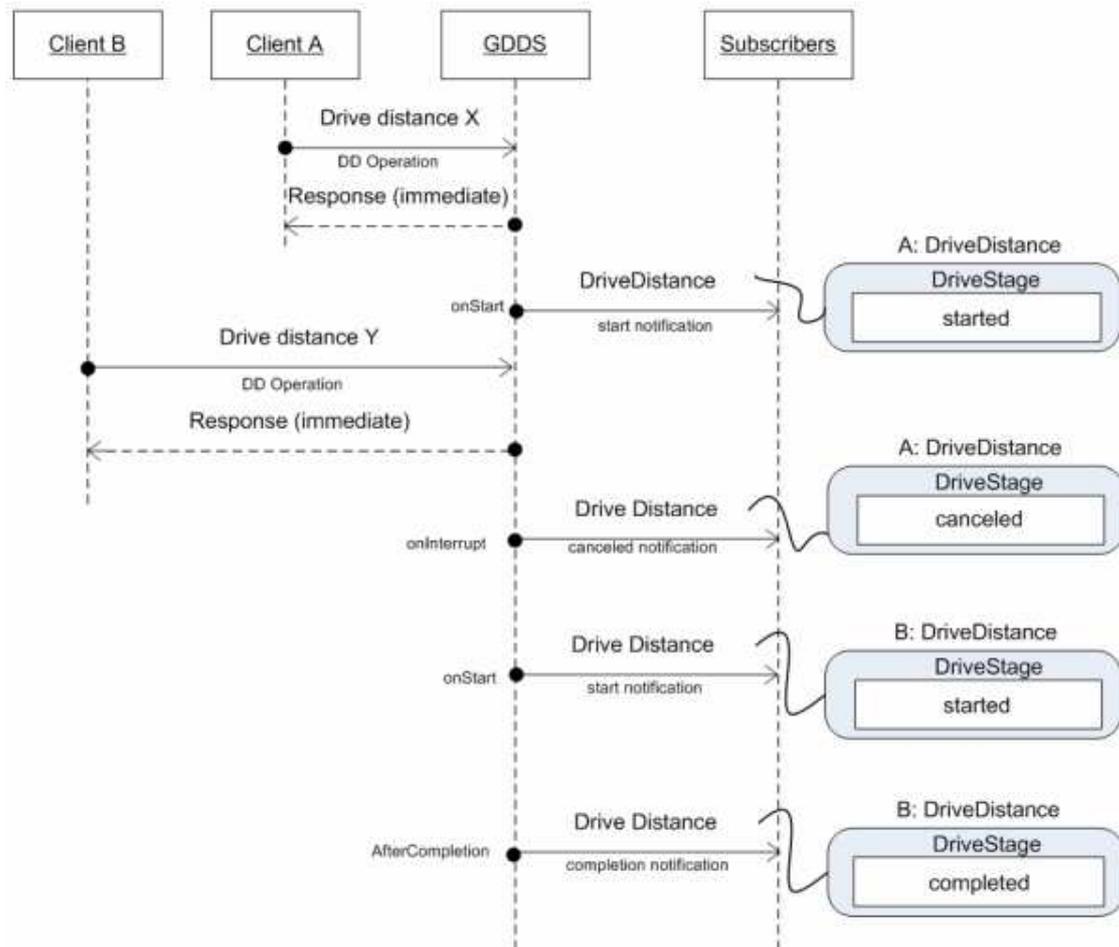
Οι δύο επόμενες ειδοποιήσεις, η DriveDistance και η RotateDegrees, λειτουργούν με παρόμοιο τρόπο μεταξύ τους. Και οι δύο εντολές επιστρέφουν αμέσως απόκριση και έπειτα ακολουθούν οι ειδοποιήσεις. Τα μηνύματα των ειδοποιήσεων είναι ίδιου τύπου δεδομένων με τα μηνύματα εντολών, με τη σημαντική διαφορά ότι τα πεδία των DriveDistanceStage ή RotateDegreesStage ανανεώνονται ώστε να δείχνουν την πορεία της λειτουργίας. Στην απλούστερη περίπτωση, ο οδηγός ξεκινά την επιθυμητή κίνηση και την ολοκληρώνει κάποια στιγμή όπως αυτό φαίνεται διαγραμματικά για την εντολή DriveDistance στην παρακάτω εικόνα:



Εικόνα 5.18 Απεικόνιση ολοκλήρωσης ειδοποιήσεων [63]

Αν το 'DriveDistance' ή το 'RotateDegrees' ακυρωθεί λόγω κάποιας άλλης εντολής οδηγού πριν ολοκληρωθεί η διαδικασία, τότε αλλάζει και η σειρά των ειδοποιήσεων. Σε αυτήν την περίπτωση η πρώτη λειτουργία ακυρώνεται και η δεύτερη εκτελείται κανονικά. Να σημειωθεί ότι οποιαδήποτε εντολή κίνησης θα προκαλέσει ακύρωση της τρέχουσας λειτουργίας.

Η παρακάτω εικόνα δείχνει μια εντολή 'DriveDistance' να διακόπτει μια άλλη. Αν και δείχνει δυο υπηρεσίες A και B, θα μπορούσε να είναι η ίδια υπηρεσία δίνοντας εντολές βασισμένες σε εισόδους αισθητηρίων:



Εικόνα 5.19 Απεικόνιση ακύρωσης ειδοποίησης από άλλη [63]

## UPDATE

Η ειδοποίηση 'Update' περιέχει την καινούργια κατάσταση του 'Generic Differential Drive', η οποία περιλαμβάνει πληροφορίες για τους κινητήρες. Από αυτήν την ειδοποίηση μπορεί να διευκρινιστεί αν ο οδηγός κινείται ή όχι.

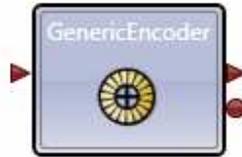
## DRIVE DISTANCE/ROTATE DEGREES

Όταν γίνει μια αίτηση για την DriveDistance ή την RotateDegrees, το GDD επιστρέφει ένα μήνυμα απόκρισης άμεσα (εκτός αν υπάρξει σφάλμα). Έπειτα ξεκινά μια ειδοποίηση εν ονόματι 'Started' όταν ξεκινήσει να εκτελείται η εντολή στο ρομπότ. Όταν τελειώσει η περιστροφή, στέλνεται μια ειδοποίηση εν ονόματι 'Completed'. Αν η DriveDistance ή η RotateDegrees διακοπεί από άλλη εντολή κίνησης, επιστρέφεται μια ειδοποίηση εν ονόματι 'Cancelled'.

---

Ο τρόπος με τον οποίο χειρίζονται οι εντολές ποικίλει από ρομπότ σε ρομπότ και εξαρτάται στον τρόπο εφαρμογής των υπηρεσιών. Ως αποτέλεσμα είναι πιθανό μια ειδοποίηση ‘Started’ να σταλεί πριν από μια νέα ειδοποίηση ‘Cancelled’.

### 5.5.8 ΚΩΔΙΚΟΠΟΙΗΤΗΣ (ENCODER)



Η υπηρεσία ‘Generic Encoder’ επιτρέπει την πρόσβαση σε πληροφορίες κατάστασης των κωδικοποιητών. Η πιο συνήθης χρήση των κωδικοποιητών είναι η μέτρηση της περιστροφής μια περιστροφικής συσκευής, όπως ένας κινητήρας, η μέτρηση της διανυθείσας απόστασης και από αυτό την σχετική τοποθεσία του ρομπότ. Επίσης μπορεί να χρησιμοποιηθεί για έλεγχο της ταχύτητας. Αν και η τοποθεσία και η απόσταση μπορούν να εκτιμηθούν με βάση το χρόνο και την ισχύ των κινητήρων, τέτοιες εναλλακτικές δεν είναι τόσο ακριβείς καθώς η απόσταση που διανύθηκε μπορεί να επηρεαστεί από παράγοντες όπως η επιφάνεια στην οποία ταξιδεύει ο τροχός που είναι συνδεδεμένος με τον κινητήρα. Βέβαια ούτε και οι κωδικοποιητές μπορούν να είναι πάντα τελείως ακριβείς, καθώς μπορεί ένας τροχός να γλιστρήσει ή το ρομπότ μπορεί να μετακινηθεί από κάποιους άλλους παράγοντες.

Η συγκεκριμένη υπηρεσία υποστηρίζει τις εξής εντολές οι οποίες εκτός από την Get, είναι επίσης διαθέσιμες ως ειδοποιήσεις:

Λειτουργία	Περιγραφή
Get	Επιστρέφει την κατάσταση της υπηρεσίας
EncoderReplace	Αλλάζει η υποδεικνύει μια αλλαγή στην κατάσταση της υπηρεσίας
ResetTickCounter	Επαναφέρει το μετρητή του κωδικοποιητή η υποδεικνύει μια αλλαγή στον μετρητή
UpdateTickCount	Ανανεώνει η υποδεικνύει μια ανανέωση στο μετρητή

Ενώ η κατάσταση ενός 'Generic Encoder' περιέχει τις παρακάτω μεταβλητές:

Όνομα	Τύπος	Περιγραφή
CurrentAngle	double	Υποδεικνύει την παρούσα μέτρηση γωνίας σε rad
CurrentReading	int	Υποδεικνύει την παρούσα μέτρηση του κωδικοποιητή σε σημεία
HardwareIdentifier	int	Αναγνωρίζει την θύρα υλισμικού που χρησιμοποιεί ο κωδικοποιητής
TicksPerRevolution	int	Υποδεικνύει τον αριθμό των σημείων που χρειάζονται για μια περιστροφή του άξονα
TickSinceReset	int	Υποδεικνύει τον αριθμό των σημείων
TimeStamp	DateTime	Η ημερομηνία της ένδειξης

Η υπηρεσία του κωδικοποιητή γενικής χρήσης αναπαριστά τις ενδείξεις της περιστροφής με μορφή σημείων. Ένα σημείο αναπαριστά ένα μικρό μέρος της περιστροφής του άξονα (ενός κινητήρα η ενός τροχού). Επειδή οι κωδικοποιητές μπορεί να διαφοροποιούνται στον αριθμό των σημείων ανά περιστροφή, αυτή η επιλογή είναι διαμορφώσιμη. Έτσι χρησιμοποιείται η εντολή EncoderReplace για να ρυθμίσει το TicksPerRevolution. Η ειδοποίηση του UpdateTickCount παρέχει ενημερώσεις καθώς αλλάζει το πλήθος των σημείων. Μπορεί να ελεγχθεί ο αριθμός των σημείων (CurrentReading) με την εντολή Get. Αυτή η εντολή θα δώσει την ένδειξη ως γωνία σε rads (αν είναι ενεργό το TicksPerRevolution) και το χρόνο. Επίσης επιστρέφει και τον αριθμό των σημείων από το τελευταίο μηδενισμό του μετρητή.

### 5.5.9 ΚΙΝΗΤΗΡΑΣ



Η υπηρεσία 'Generic Motor' επιτρέπει τον έλεγχο και την πρόσβαση στην κατάσταση ενός κινητήρα. Διαφέρει από την υπηρεσία 'Generic Differential Drive' που συντονίζει δύο υπηρεσίες κινητήρων.

Έτσι παρέχει τις παρακάτω λειτουργίες, αλλά μόνο ως εντολές. Οι ειδοποιήσεις ενός κινητήρα γίνονται από την υπηρεσία 'Generic Encoder':

<b>Λειτουργία</b>	<b>Περιγραφή</b>
Get	Επιστρέφει την κατάσταση της υπηρεσίας
MotorReplace	Αλλάζει την κατάσταση του κινητήρα
SetMotorPower	Θέτει την ισχύ ενός κινητήρα. Αυτή η εντολή καθορίζει την ισχύ, χρησιμοποιώντας την με τιμές -1.0 έως 1.0 (αρνητικές τιμές σημαίνουν όπισθεν)

Καθώς δεν είναι κάθε κινητήρας μέρος του συστήματος κίνησης του ρομπότ οι κατάσταση ενός κινητήρα γενικής χρήσης περιέχει τις παρακάτω μεταβλητές:

<b>Όνομα</b>	<b>Τύπος</b>	<b>Περιγραφή</b>
CurrentPower	double	Υποδεικνύει την παρούσα ισχύ του κινητήρα, με τιμές από -1.0 έως 1.0 (αρνητικές τιμές σημαίνουν όπισθεν)
HardwareIdentifier	int	Αναγνωρίζει την θύρα υλισμικού που χρησιμοποιεί ο κωδικοποιητής
Name	string	Καθορίζει ένα ξεχωριστό όνομα για τον κινητήρα
Pose	Pose	Υποδεικνύει τον προσανατολισμό και την θέση του κινητήρα
PowerScalingFactor	double	Υποδεικνύει τον πολλαπλασιαστή που εφαρμόζεται στο CurrentPower για να τεθεί η ισχύς στον κινητήρα
ReversePolarity	Boolean	Υποδεικνύει την κατεύθυνση (πόλωση) της ισχύος του κινητήρα. Αληθής τιμή αντιστρέφει την κίνηση του κινητήρα

---

Επίσης υποστηρίζεται η έννοια των κινητήρων στους τροχούς και συμπεριλαμβάνει τις ακόλουθες μεταβλητές:

Όνομα	Τύπος	Περιγραφή
WheelSpeed	double	Υποδεικνύει την παρούσα ρύθμιση ταχύτητας των τροχών σε m/s
Name	string	Καθορίζει ένα ξεχωριστό όνομα για τον κινητήρα
MotorState	MotorState	Υποδεικνύει τις βασικές ιδιότητες ενός κινητήρα
Radius	double	Καθορίζει την ακτίνα του τροχού του κινητήρα
GearRatio	double	Καθορίζει την αναλογία του κινητήρα έναντι του τροχού. Π.χ. 1/5 σημαίνει ότι ο τροχός περιστρέφεται πέντε φορές πιο αργά από τον κινητήρα
EncoderState	EncoderState	Εδώ προαιρετικά μπορεί να συμπεριληφθεί η κατάσταση του κωδικοποιητή του κινητήρα

### 5.5.10 SONAR



Η υπηρεσία 'Generic Sonar' επιτρέπει την πρόσβαση σε δεδομένα ενός αισθητηρίου sonar, συμπεριλαμβανομένων πληροφοριών όπως η μέτρηση της διανυθείσας απόστασης και η ανάλυση και το η μέτρηση του τριγωνικού φάσματος. Τα αισθητήρια sonar χρησιμοποιούν ηχητικά κύματα (συνήθως υπερηχητικά) για να καθορίσουν την απόσταση ενός αντικειμένου. Καθώς όμως κάποια υλικά μπορούν να απορροφήσουν ήχους ή μπορεί το κύμα να εκτραπεί έξω από την εμβέλεια του ανιχνευτή, το αισθητήριο μπορεί να είναι αναξιόπιστο αν χρησιμοποιείται μόνο του. Έτσι χρησιμοποιούνται πολλαπλά αισθητήρια sonar σε μια κυκλική διάταξη για να υπερκεράσουν κάποιους από αυτούς τους περιορισμούς.

Η υπηρεσία 'Generic Sonar' παρέχει τις παρακάτω εντολές:

Λειτουργία	Περιγραφή
Get	Επιστρέφει την κατάσταση της υπηρεσίας
SonarUpdate	Αλλάζει η ειδοποιεί για αλλαγή της κατάστασης του σοναρ

Η εντολή SonarUpdate είναι διαθέσιμη και ως ειδοποίηση.

Η κατάσταση του sonar αναπαριστάται από τις παρακάτω μεταβλητές:

Όνομα	Τύπος	Περιγραφή
AngularRange	double	Καθορίζει την γωνία σάρωσης του sonar
AngularResolution	double	Καθορίζει το μικρότερο ανιχνεύσιμο χαρακτηριστικό σε rad
DistanceMeasurement	double	Καθορίζει τις ενδείξεις απόστασης του sonar
DistanceMeasurements	array ή double	Ομάδα ενδείξεων απόστασης του sonar
HardwareIdentifier	int	Αναγνωρίζει την θύρα υλισμικού που χρησιμοποιεί το sonar
MaxDistance	int	Καθορίζει την μέγιστη απόσταση που μπορεί να ανιχνεύσει το sonar
Pose	Pose	Υποδεικνύει την θέση και το προσανατολισμό του sonar
TimeStamp	DateTime	Υποδεικνύει την ημερομηνία της ένδειξης

### 5.5.11 STREAM

Η υπηρεσία ‘Generic Stream’ παρέχει πρόσβαση διπλής κατεύθυνσης στη ροή δεδομένων. Χρησιμοποιώντας λειτουργίες ανάγνωσης και εγγραφής, επιτρέπει ροή κειμένου ή δυαδική ροή (σε byte) επικοινωνίας μεταξύ υπηρεσιών. Η ροή έπειτα μετατρέπεται σε μια ακολουθία δυαδικών πακέτων ή πακέτων κειμένου.

### 5.5.12 KAMERA



Η υπηρεσία ‘Generic Webcam’ επιτρέπει την απόκτηση δεδομένων από μια συμβατική ψηφιακή κάμερα, η οποία συνδέεται μέσω USB ή IEEE 1394 (Firewire).

Η υπηρεσία παρέχει τις παρακάτω εντολές. Όλες εκτός της Get είναι διαθέσιμες και για ειδοποιήσεις:

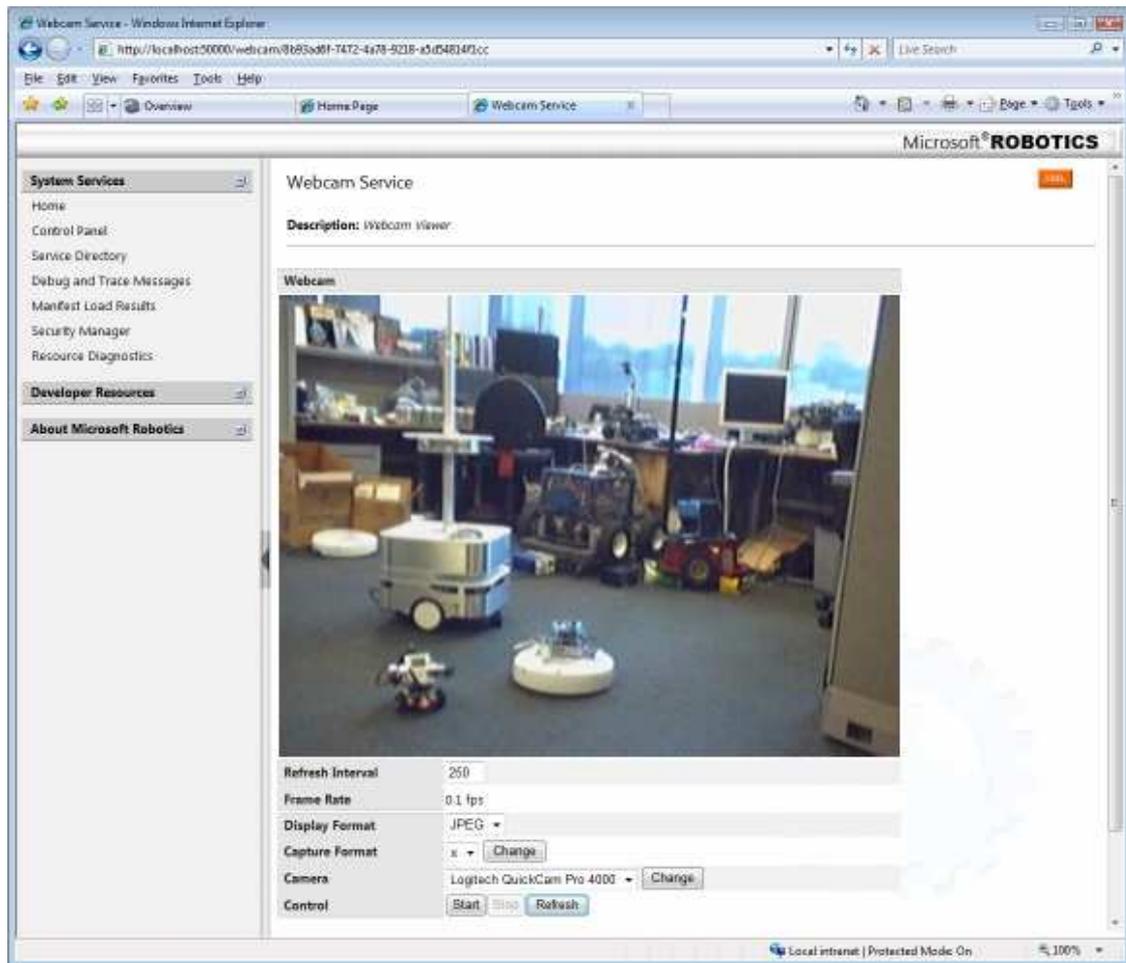
<b>Λειτουργία</b>	<b>Περιγραφή</b>
Get	Επιστρέφει την κατάσταση της υπηρεσίας
QueryFrame	Επιστρέφει την πιο πρόσφατη εικόνα με καταγεγραμμένη ημερομηνία. Συμπεριλαμβάνει και τα δεδομένα της εικόνας
UpdateFrame	Υποδεικνύει πότε κατεγράφη νέα εικόνα. Δεν περιέχει δεδομένα εικόνας, ώστε τέτοιου είδους μηνύματα να παραμένουν μικρά
WebCamReplace	Ανανεώνει η υποδεικνύει ανανέωση στην κατάσταση της υπηρεσίας

Η κατάσταση της κάμερας αναπαριστάται από τις παρακάτω μεταβλητές:

<b>Όνομα</b>	<b>Τύπος</b>	<b>Περιγραφή</b>
CameraName	string	Καθορίζει το όνομα της κάμερας
Pose	Pose	Υποδεικνύει την θέση και τον προσανατολισμό της κάμερας
ImageSize	Vector2	Υποδεικνύει το μέγεθος της εικόνας της κάμερας
ViewAngle	float	Καθορίζει την γωνία της άποψης της κάμερας
Quality	double	Καθορίζει το μέγεθος συμπίεσης στην εικόνα
LastFrameUpdate	DateTime	Ημερομηνία της τελευταίας εικόνας

Εκτός από αυτήν την υπηρεσία, υπάρχει και μια υλοποίηση κάμερας, όπου δουλεύει με κάμερες που έχουν οδηγούς DirectShow με την τελευταία έκδοση του DirectX ή του πολυπλοκότερου μοντέλου VideoForWindows.

Καθώς δεν υπάρχει αναγνώστης για την υπηρεσία κάμερας στην VPL, είναι αναγκαία η χρήση ενός φυλλομετρητή για την εξέταση της κατάστασης της υπηρεσίας καθώς εκτελείται. Στην εκκίνηση της εφαρμογής, ανοίξτε έναν φυλλομετρητή και πλοηγηθείτε στη διεύθυνση <http://localhost:50000/webcam>. Θα εμφανιστεί μια ιστοσελίδα με την εικόνα της κάμερας όπως φαίνεται παρακάτω. Χρησιμοποιώντας τα κουμπιά ελέγχου στην ιστοσελίδα, μπορείτε να επιλέξετε μια διαφορετική κάμερα αν υπάρχει και τον τύπο αρχείου αποθήκευσης. Υπάρχει η δυνατότητα για αυτόματη ανανέωση της σελίδας κάθε κάποιο καθορισμένο διάστημα ή ένδειξη μιας ανανεωμένης εικόνας κάθε φορά που πατιέται το 'refresh'.



Εικόνα 5.20 Απεικόνιση της εικόνας της webcam

Να σημειωθεί ότι η επεξεργασία εικόνας είναι ένα πολύ περίπλοκο έργο και το προγραμματιστικό περιβάλλον της VPL δεν είναι το κατάλληλο για αυτό, μπορεί ωστόσο να γίνει εισαγωγή υπηρεσιών που κάνουν την επεξεργασία εικόνας. Για παράδειγμα υπάρχουν υπηρεσίες διαχώρισης χρώματος και αναγνώρισης διαφορών, σαν δείγματα κώδικα C#.

### 5.5.13 ΑΙΣΘΗΤΗΡΙΟ ΚΑΜΕΡΑΣ

Η υπηρεσία 'Webcam Sensor' παρέχει πρόσβαση στην RGB βιντεοκάμερα του Kinect, ενώ να σημειωθεί ότι είναι διαφορετικού είδους υπηρεσία από την 'Generic Webcam'. Επίσης δεν χρειάζεται να ενεργοποιηθεί η υπηρεσία 'Webcam Sensor' αν ήδη χρησιμοποιείται η υπηρεσία 'DepthCam' καθώς η τελευταία συμπεριλαμβάνει δεδομένα βάθους και δεδομένα RGB. Αν υπάρχει ανάγκη για δύο διαδοχικές εικόνες τραβηγμένες όσο το δυνατόν κοντύτερα στο χρόνο μεταξύ τους, τότε συνίσταται η υπηρεσία 'DepthCam'. Η υπηρεσία 'Webcam Sensor' στοχεύει κυρίως σε εφαρμογές οι οποίες δεν χρησιμοποιούν δεδομένα βάθους.

Η συγκεκριμένη υπηρεσία παρέχει τις παρακάτω εντολές. Η εντολή Replace είναι διαθέσιμη για ειδοποιήσεις:

Λειτουργία	Περιγραφή
Get	Επιστρέφει την κατάσταση της υπηρεσίας
HttpGet	Επιστρέφει την κατάσταση της υπηρεσίας με εντολή μέσω http (συνήθως με φυλλομετρητή)
HttpQuery	Αίτηση εικόνας από ένα φυλλομετρητή
Replace	Κατοχυρωμένη. Μην χρησιμοποιηθεί
Subscribe	Εγγραφή σε ειδοποιήσεις. Μονο μηνύματα από την Replace αποστέλλονται

Αντίθετα με την ‘Generic Webcam’ μπορεί να γίνει αίτηση για ενημερώσεις από το αισθητήριο και θα αποστέλλονται καρέ ως ειδοποιήσεις (μηνύματα Replace). Επίσης μπορεί να χρησιμοποιηθεί η εντολή Get.

Η κατάσταση του αισθητηρίου κάμερας αναπαριστάται από τις παρακάτω μεταβλητές:

Όνομα	Τύπος	Περιγραφή
DeviceName	string	Καθορίζει το όνομα της κάμερας
TimeStamp	DateTime	Ημερομηνία εικόνας
Width	int	Πλάτος εικόνας σε εικονοστοιχεία
Height	int	Υψος εικόνας σε εικονοστοιχεία
Stride	int	Αριθμός bytes σε κάθε σειρά
Data	byte[]	Δεδομένα εικόνας σε BGR24 τύπο αρχείου. Δεν είναι RGB
HorizontalFieldOfView	double	Οριζόντια άποψη της κάμερας σε rad
ProjectionMatrix	Matrix	Πίνακας που καθορίζει την μετατροπή πραγματικής τρισδιάστατης εικόνας σε εικόνα κάμερας (εικονοστοιχεία επί βάθος)
InverseProjectionMatrix	Matrix	Το αντίστροφο του ProjectionMatrix (πχ μετατρέπει στην αντίθετη κατεύθυνση)

#### 5.5.14 SICK LASER RANGE FINDER

Η υπηρεσία ‘SICK LASER Ranger Finder’ παρέχει περιβάλλον διεπαφής για το τηλεμετρικό LASER τύπου SICK LMS200 και είναι σε μορφή C#. Τα αρχεία του έργου είναι στην παρακάτω διεύθυνση της εγκατάστασης του προγράμματος:

*Samples\Sensors\SickLRF*

Χρειάζεται για την λειτουργία της ένα τηλεμετρικό LASER (LRF), συνδεδεμένο σε υπολογιστή μέσω σειριακής θύρας και ένα φυλλομετρητή. Η υπηρεσία ρυθμίζει το LRF για εκτέλεση σε κατάσταση συνεχούς παρακολούθησης κάνοντας περάσματα 180

---

μοιρών και δειγματολογώντας δεδομένα κάθε 0,5 μοίρες σε μέγιστο ρυθμό δεδομένων. Κάθε φορά που το LRF αναφέρει ένα πλήρες πέρασμα στην υπηρεσία, τότε δημιουργείται μια ειδοποίηση 'Replace' η οποία περιέχει τα δεδομένα του.

## 5.6 ΧΡΗΣΙΜΕΣ ΔΡΑΣΤΗΡΙΟΤΗΤΕΣ ΤΗΣ ΓΛΩΣΣΑΣ VPL

Η VPL συμπεριλαμβάνει μια ομάδα βασικών δραστηριοτήτων εμφανιζόμενων ως κουτιά οι οποίες χρησιμοποιούνται ως βοήθεια στη δημιουργία προγραμμάτων ροής. Αυτά τα κουτιά συνήθως χρησιμοποιούνται για σύνδεση κουτιών υπηρεσιών, αλλά μπορούν επίσης να συνδεθούν και μεταξύ τους.

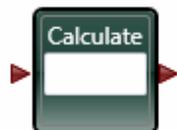
Παρακάτω παρουσιάζονται με σειρά προτεραιότητας [64]:

### 5.6.1 ACTIVITY



Το κουτί Activity, μερικές φορές αναφερόμενο ως 'Custom Activity', επιτρέπει τη δημιουργία δραστηριοτήτων οι οποίες έχουν ξεχωριστές ομάδες εσωτερικών διαγραμμάτων ροής. Χρησιμοποιούνται για δημιουργία διαγραμμάτων τα οποία αναπαριστώνται ως μεμονωμένα κουτιά σε άλλα διαγράμματα για χρήση, όπως οι προκατασκευασμένες δραστηριότητες. Μπορούν επίσης να μεταγλωττιστούν σε υπηρεσίες οι οποίες μπορούν να συνδυαστούν με άλλες ή με άλλα διαγράμματα VPL.

### 5.6.2 CALCULATE



Η δραστηριότητα Calculate εκτελεί απλές αριθμητικές ή λογικές πράξεις, όπου η έκφραση εισάγεται στο κουτί κειμένου. Η έκφραση μπορεί να περιλαμβάνει αριθμητικές τιμές, την τιμή ενός εισερχόμενου μηνύματος ή προκαθορισμένες τιμές που παρέχονται από άλλες υπηρεσίες στο διάγραμμα.

Για αριθμητικά δεδομένα χρησιμοποιούνται τα εξής:

+	Πρόσθεση
-	Αφαίρεση
*	Πολ/σμός
/	Διαίρεση
%	Το μέτρο της τιμής

---

Ο τελεστής '+' μπορεί επίσης να χρησιμοποιηθεί για δημιουργία αλληλουχιών, συνδυασμών και στοιχειοσειρών. Επίσης για συνδυασμό κειμένου, στοιχειοσειρών και αριθμητικών δεδομένων με τη χρήση διπλών εισαγωγικών.

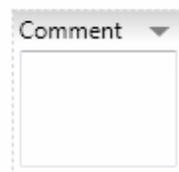
Για λογικούς τελεστές χρησιμοποιούνται τα εξής:

&&	AND
	OR
!	NOT

Επίσης μπορούν να χρησιμοποιηθούν παρενθέσεις για να τονιστεί η σειρά των πράξεων.

Πατώντας στο κουτί κειμένου της δραστηριότητας εμφανίζεται μια λίστα που περιέχει την τιμή του εισερχόμενου μηνύματος, διάφορα δεδομένα του μηνύματος, οι μεταβλητές κατάστασης και οι προκαθορισμένες τιμές οι οποίες μπορεί να έχουν ορισθεί από άλλες υπηρεσίες.

### 5.6.3 COMMENT



Η δραστηριότητα Comment επιτρέπει την εισαγωγή ενός κουτιού κειμένου στο διάγραμμα για σκοπούς τεκμηρίωσης, καθώς είναι καλή συνήθεια η προσθήκη σχολίων στον κώδικα διότι βοηθά στην κατανόηση του προγράμματος και επίσης βοηθά σε μεταγενέστερες βελτιώσεις του προγράμματος. Το κουτί Comment δεν υποστηρίζει καμία σύνδεση με άλλα και μπορεί να τοποθετηθεί οπουδήποτε στο διάγραμμα. Επίσης μπορεί να κρυφθεί αν πατηθεί το βέλος στην δεξιά πλευρά του τίτλου του.

### 5.6.4 DATA



Η δραστηριότητα Data χρησιμοποιείται για να τροφοδοτήσει μια άλλη δραστηριότητα ή υπηρεσία με απλές τιμές δεδομένων. Για να ορισθεί ένα συγκεκριμένο είδος δεδομένων, από το μενού επιλέγεται το είδος δεδομένων και έπειτα εισάγεται η τιμή. Αν είναι τύπου string η τιμή, δεν χρειάζονται εισαγωγικά.

### 5.6.5 IF



Η δραστηριότητα If παρέχει επιλογή εξόδου για την προώθηση του εισερχόμενου μηνύματος βασισμένη σε μια συνθήκη που έχει εισαχθεί. Αν η συνθήκη είναι αληθής τα εισαγόμενα δεδομένα προωθούνται στο πάνω βέλος. Αν είναι ψευδής τότε χρησιμοποιείται η έξοδος Else. Οι συνθήκες μπορούν να χρησιμοποιούν τους ακόλουθους τελεστές για αξιολόγηση:

Τελεστής = ή ==	Ισοδύναμο
!= ή <>	Μη ισοδύναμο
<	Μικρότερο από
>	Μεγαλύτερο από
<=	Μικρότερο ή ίσο από
>=	Μεγαλύτερο ή ίσο από

Οι ίδιοι τελεστές μπορούν να χρησιμοποιηθούν για την δραστηριότητα Calculate, με την προϋπόθεση ότι το αποτέλεσμα θα έχει τιμές αληθές ή ψευδές. Επίσης μπορούν να προστεθούν περισσότερες από μια προϋποθέσεις στην δραστηριότητα If, πατώντας το κουμπί '+', ενώ κάθε νέα προϋπόθεση έχει το δικό της βέλος εξόδου και εξετάζονται από πάνω προς τα κάτω και χρησιμοποιείται το πρώτο αποτέλεσμα που έχει αληθή τιμή.

### 5.6.6 JOIN



Η δραστηριότητα Join συνδυάζει δύο ροές δεδομένων και διαφέρει από την Merge στο ότι τα εισερχόμενα δεδομένα των μηνυμάτων συνδυάζονται και τα μηνύματα πρέπει να παραληφθούν σε όλες τις εισερχόμενες συνδέσεις πριν η δραστηριότητα τα μεταδώσει. Το κείμενο που εισάγεται είναι ονόματα μεταβλητών που αντιπροσωπεύουν τα μηνύματα. Αυτές οι μεταβλητές χρησιμοποιούνται είτε άμεσα, είτε με τη χρήση συμβολισμών για αναφορά ενός μέλους δεδομένων. Π.χ. το 'value.field' αναφέρεται στο μέλος δεδομένων εν ονόματι 'field' της μεταβλητής 'value'.

Μια χρήση της Join είναι η κατασκευή ενός πολύπλοκου τύπου δεδομένων που χρησιμοποιείται ως είσοδος σε μια υπηρεσία.

---

### 5.6.7 LIST



Η δραστηριότητα List δημιουργεί μια άδεια λίστα δεδομένων. Για να δημιουργηθεί μια λίστα, επιλέγεται ο τύπος δεδομένων από τη λίστα επιλογών στο κουτί της δραστηριότητας ή στο κομμάτι του μενού 'Properties'. Για να προστεθούν είσοδοι σε μια λίστα χρησιμοποιείται η δραστηριότητα List Functions. Για αποθήκευση μιας λίστας για χρήση κάπου αλλού στο διάγραμμα, δημιουργείται μια λίστα μεταβλητών χρησιμοποιώντας την δραστηριότητα Variable.

### 5.6.8 LIST FUNCTIONS



Η δραστηριότητα List Functions επιτρέπει την τροποποίηση μιας ήδη υπάρχουσας λίστας. Μέσω του μενού στο κουτί, επιλέγεται τι είδος ρουτίνας θα εισαχθεί στη λίστα.

### 5.6.9 MERGE



Η δραστηριότητα Merge συνδυάζει τη ροή δύο διαγραμμάτων. Δεν υπάρχει κάποια συνθήκη ή εξάρτηση σε άλλους κλάδους όταν γίνεται μετάδοση μηνυμάτων. Ο ρόλος της δραστηριότητας είναι απλά να περνά τα μηνύματα στην επόμενη δραστηριότητα.

### 5.6.10 SWITCH



Η δραστηριότητα Switch μπορεί να χρησιμοποιηθεί για να δρομολογήσει τα μηνύματα βασιζόμενη στο αν το εισερχόμενο μήνυμα ταιριάζει με την έκφραση που είναι εισαγμένη στο κουτί κειμένου. Μπορούν να προστεθούν κλάδοι Case στο κουτί

---

πατώντας το '+'. Η Switch είναι παρόμοια με την If αλλά η συνθήκη ελέγχει αν η εισερχόμενη τιμή του μηνύματος ταιριάζει σε κάθε μια από τις τιμές στα κουτιά κειμένου. Έτσι τα κουτιά κειμένου περιέχουν μόνο τιμές. Αντιθέτως η If μπορεί να έχει τελείως διαφορετικές συνθήκες ελέγχου για κάθε βέλος εξόδου.

#### 5.6.11 VARIABLE



Η δραστηριότητα Variable επιτρέπει τη θέση ή την λήψη μιας τιμής από μια μεταβλητή. Για να επιλεγεί ο τύπος της μεταβλητής χρησιμοποιείται το μενού στη κάτω δεξιά πλευρά του κουτιού. Αν δεν έχουν καθοριστεί καθόλου μεταβλητές ή για να καθοριστούν από την αρχή, χρησιμοποιείται η εντολή Define Variables από το μενού ή με την εντολή Variables από το κυρίως μενού Edit, όπου αυτό θα εμφανίσει το παράθυρο διαλόγου Define Variables και από κει μπορεί να γίνει εισαγωγή μιας μεταβλητής και καθορισμός του τύπου της.

Τα ονόματα μιας μεταβλητής πρέπει να ξεκινούν με γράμμα και να συμπεριλαμβάνουν μόνο γράμματα ή αριθμούς. Επίσης επιτρέπει τη χρήση του χαρακτήρα '\_'.

Οι δραστηριότητες των μεταβλητών είναι απλά δημιουργήματα που υποστηρίζουν την σύνδεση GetValue, για ανάγνωση της τιμής και την σύνδεση SetValue για την καταχώριση της τιμής. Όταν χρησιμοποιείται η σύνδεση τύπου SetValue στη εξωτερική σύνδεση της δραστηριότητας, μεταφέρεται μαζί και η μεταβλητή της.

---

### 5.6.12 ΤΥΠΟΙ ΔΕΔΟΜΕΝΩΝ

Η VPL υποστηρίζει τύπους δεδομένων στη .NET C#. Αυτοί είναι:

Τύπος στη VPL	Περιγραφή
bool	Τιμές άλγεβρας Boole: αληθής, ψευδής
byte	8 bit μη-προσημασμένος ακέραιος
sbyte	8 bit προσημασμένος ακέραιος
char	χαρακτήρας
decimal	αριθμός σταθερης υποδιαστολής
double	αριθμός σταθερης υποδιαστολής διπλής ακρίβειας
float	αριθμός σταθερης υποδιαστολής απλής ακρίβειας
int	32 bit προσημασμένος ακέραιος
uint	32 bit μη-προσημασμένος ακέραιος
long	64 bit προσημασμένος ακέραιος
ulong	64 bit μη-προσημασμένος ακέραιος
short	16 bit προσημασμένος ακέραιος
ushort	16 bit μη-προσημασμένος ακέραιος
string	στοιχειοσειρά χαρακτήρων

Η VPL υποστηρίζει λίστες και όχι πίνακες, όμως οι λίστες μπορούν να χρησιμοποιηθούν στη θέση των πινάκων.

Ο πίνακας είναι μια ομάδα μεταβλητών του ίδιου τύπου δεδομένων και προσπελάσιμων μέσω του ίδιου ονόματος και ενός δείκτη, ενώ είναι συνήθως καθορισμένου μεγέθους. Οι λίστες είναι περισσότερο γενικής χρήσης και επιτρέπουν την εισαγωγή και διαγραφή στοιχείων οπουδήποτε στη λίστα.

---

## ΚΕΦΑΛΑΙΟ 6

# ΕΦΑΡΜΟΓΗ ΤΟΥ ΛΟΓΙΣΜΙΚΟΥ MRDS ΓΙΑ ΕΛΕΓΧΟ ΚΑΙ ΠΡΟΣΟΜΟΙΩΣΗ ΤΟΥ LEGO MINDSTORMS NXT.

### 6.1 ΕΙΣΑΓΩΓΗ

Με την ονομασία LEGO MINDSTORMS NXT αναφερόμαστε στο ρομποτικό kit της εταιρίας LEGO, που χρησιμοποιείται για την διδασκαλία ρομποτικής και αυτοματισμού. Η καρδιά του kit είναι ένας 32 bit επεξεργαστής που έχει την δυνατότητα να επικοινωνήσει με έναν Η/Υ μέσω Bluetooth. Στο κεφάλαιο αυτό περιγράφεται η διαδικασία ελέγχου της βασικής μονάδας ελέγχου (NXT) και του MRDS

Οι υπηρεσίες του LEGO MINDSTORMS NXT έχουν σχεδιαστεί για ευκολία στη χρήση μαζί με τη βοήθεια της γλώσσας VPL. Αυτές οι υπηρεσίες είναι εύκολες στη ρύθμιση και παρέχουν μια επεκτάσιμη αρχιτεκτονική, έτσι ώστε αισθητήρια τρίτων εταιρειών να μπορούν να προστεθούν χωρίς πρόβλημα, συμπεριλαμβάνοντας κινητήρες και αισθητήρια για τις πρότυπες συσκευές NXT και μερικά δείγματα αισθητηρίων διαθέσιμων από τις εταιρείες HiTechnic και MindSensors.

### 6.2 ΧΡΗΣΙΜΟΠΟΙΩΝΤΑΣ ΤΟ MRDS ΚΑΙ ΤΑ LEGO MINDSTORMS

Για να είναι δυνατή η σύνδεση του NXT, με οδηγούς RDS θα πρέπει να είναι εγκατεστημένο το τελευταίο firmware. Προκειμένου να γίνει αυτό θα πρέπει να γίνουν τα ακόλουθα:

Από το μενού, επιλέξτε 'Tools', έπειτα 'Update NXT Firmware' και στη συνέχεια 'Online Update: Check'. Αυτό θα ανοίξει την ιστοσελίδα του MINDSTORMS NXT Software Updates. Αφού ολοκληρωθεί η λήψη του τελευταίου υλικολογισμικού από την συγκεκριμένη ιστοσελίδα, χρειάζεται να γίνει ενημέρωση του υλικολογισμικού ακολουθώντας τις οδηγίες που παρέχονται από τη LEGO.

#### Ρυθμίζοντας το Bluetooth

---

Οι οδηγοί που παρέχονται μαζί με την σουίτα RDS υποστηρίζουν μόνο επικοινωνία μέσω Bluetooth. Αυτό σημαίνει ότι ο υπολογιστής που θα χρησιμοποιηθεί πρέπει να έχει ένα προσαρμογέα Bluetooth για να εκτελεστούν οι οδηγοί. Για να γίνει ρύθμιση του προσαρμογέα Bluetooth και να επικοινωνήσει με το NXT χρειάζεται ένας διαχειριστής Bluetooth. Αυτό συνήθως παρέχεται μαζί με τον προσαρμογέα Bluetooth και δημιουργεί μια σύνδεση μεταξύ υπολογιστή και συσκευής.

### 6.3 ΠΑΡΟΥΣΙΑΣΗ ΤΩΝ LEGO ΥΠΗΡΕΣΙΩΝ

Οι υπηρεσίες LEGO MINDSTORMS NXT είναι αρκετά εύκολες στη ρύθμιση και παρέχουν μια επεκτάσιμη αρχιτεκτονική, έτσι ώστε τα αισθητήρια τρίτων εταιρειών να μπορούν να προστεθούν οποτεδήποτε. Οι παρεχόμενες υπηρεσίες συμπεριλαμβάνουν κινητήρες και αισθητήρια για τις πρότυπες συσκευές του NXT αλλά και μερικά αισθητήρια που είναι διαθέσιμα από τις εταιρείες HiTechnic και MindSensors.

Αυτές είναι [65]:

LEGO NXT Brick (v2)



Ελέγχει τη σύνδεση σε ένα κεντρικό κουτί LEGO MINDSTORMS NXT, όπου από εδώ και στο εξής θα αναφέρεται απλά ως 'κουτί'.

LEGO NXT Motor (v2)



Ελέγχει την ισχύ σε ένα κινητήρα και, προαιρετικά, επιστρέφει αναφορές του κωδικοποιητή.

LEGO NXT Drive (v2)



---

Ελέγχει την ισχύ σε δύο κινητήρες ρυθμισμένους ως οδηγούς. Επίσης είναι πλήρως ενσωματωμένες προηγμένες ρουτίνες, όπως οι DriveDistance και RotateDegrees.

#### LEGO NXT Light Sensor (v2)



Επιτρέπει την ενεργοποίηση του φακού. Το αισθητήριο αποστέλλει ειδοποιήσεις όταν αλλάζει η ένταση του φωτός.

#### LEGO NXT Color Sensor (v2)



Επιτρέπει την ενεργοποίηση διαφορετικών LED (κόκκινο, πράσινο, μπλε κλπ). Το αισθητήριο στέλνει ειδοποιήσεις όταν αλλάζει η ένταση του φωτός, εκτός αν είναι στη κατάσταση 'Color' όπου εκεί αποστέλλει έναν αριθμό κώδικα χρώματος.

#### LEGO NXT Sound Sensor (v2)



Αποστέλλει ειδοποιήσεις υποδεικνύοντας την ένταση του ήχου που λαμβάνεται από το αισθητήριο ήχου.

#### LEGO NXT Touch Sensor (v2)



Αποστέλλει ειδοποιήσεις όταν αισθητήριο επαφής πιεστεί ή απελευθερωθεί.

#### LEGO NXT Ultrasonic Sensor (v2)



Παρέχει συνεχείς ειδοποιήσεις υποδεικνύοντας την απόσταση του κοντινότερου αντικειμένου που είναι μέσα στην εμβέλεια του αισθητηρίου υπερήχων.

LEGO NXT Battery (v2)



Παρέχει συχνές ειδοποιήσεις για την τρέχουσα τάση της μπαταρίας και ρυθμιζόμενες ειδοποιήσεις για το 'CriticalBattery'.

LEGO NXT Buttons (v2)



Στέλνει ειδοποιήσεις όταν τα κουμπιά του κουτιού NXT πιεστούν και/ή απελευθερωθούν.

MindSensors Acceleration Sensor



Στέλνει συνεχείς ειδοποιήσεις υποδεικνύοντας την περιστροφή των x,y,z στο επιταχυνσιόμετρο.

MindSensors Compass Sensor



Αποστέλλει συνεχείς ειδοποιήσεις υποδεικνύοντας την τρέχουσα ένδειξη της πυξίδας.

---

### HiTechnic Acceleration Sensor



Στέλνει συνεχείς ειδοποιήσεις υποδεικνύοντας την περιστροφή των x,y,z στο επιταχυνσιόμετρο.

### HiTechnic Compass Sensor



Αποστέλλει συνεχείς ειδοποιήσεις υποδεικνύοντας την τρέχουσα ένδειξη της πυξίδας.

### LEGO NXT Contact Sensor Array (v2)



Συναθροίζει μέχρι τέσσερα αισθητήρια LEGO NXT δημιουργώντας ένα πίνακα αισθητηρίων επαφής.

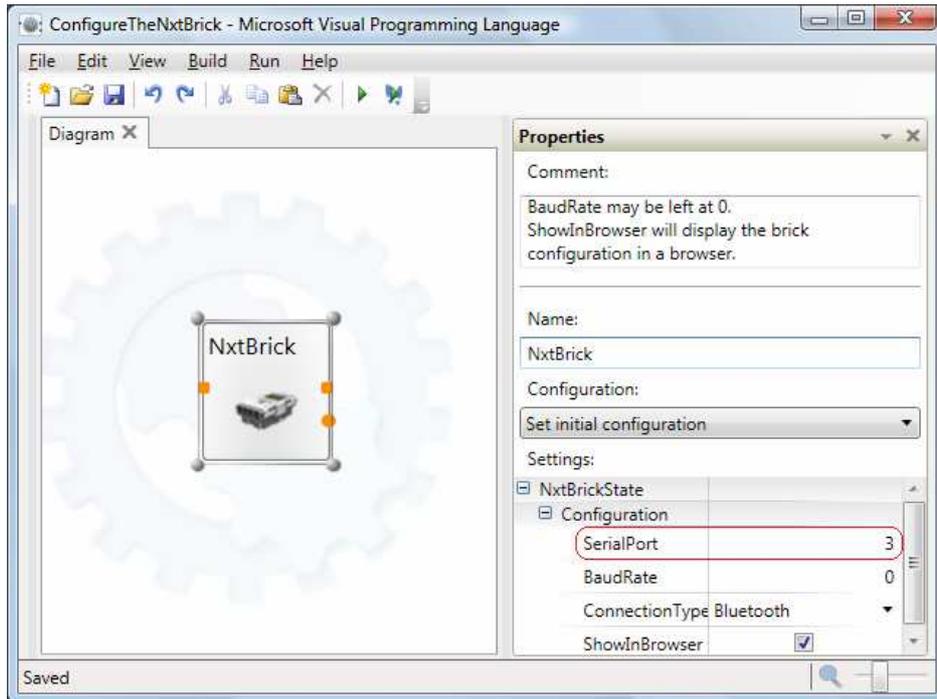
### LEGO NXT Brick I/O (v2)



Παρέχει πρόσβαση σε αρχεία και προγράμματα απευθείας στο κουτί NXT. Επιτρέπει επίσης την εκτέλεση των εντολών Get και Set στο όνομα του κουτιού. Η λειτουργία των παραπάνω υπηρεσιών θα αναλυθεί στη συνέχεια.

## 6.4 ΡΥΘΜΙΣΗ ΤΗΣ ΥΠΗΡΕΣΙΑΣ ΤΟΥ ΚΟΥΤΙΟΥ NXT

Αρχικά επιλέγουμε την υπηρεσία 'LEGO NXT Brick (v2)' και στη συνέχεια επιλέγουμε στην επιφάνεια του VPL το 'Set Initial Configuration' και αλλάζουμε την 'SerialPort' στην κατάλληλη θύρα για την σύνδεση του Bluetooth μας, προς το παρόν αυτό το πακέτο υποστηρίζει μόνο σύνδεση Bluetooth στο NXT (Εικόνα 6.1)



Εικόνα 6.1 Διάγραμμα εργασίας με την υπηρεσία LEGO NXT Brick (v2)

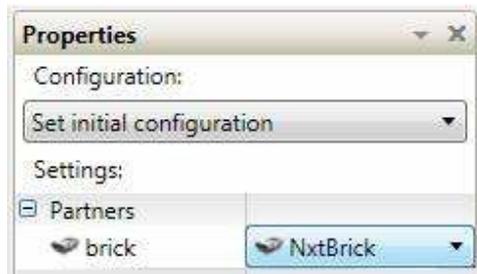
Βάζοντας προκαθορισμένο Baudrate ως 0, θα αναγνωρίσει αυτόματα το σωστό Baudrate. Επιλέγοντας το 'ShowInBrowser', τότε κάθε φορά που ξεκινά μια υπηρεσία θα ξεκινά από ένα φυλλομετρητή η διεπαφή χρηστών των υπηρεσιών του LEGO NXT. Αυτό είναι χρήσιμο για επίβλεψη των ρυθμίσεων και την κατάσταση εκτέλεσης των συσκευών που είναι συνδεδεμένες στο κουτί LEGO NXT.

## 6.5 ΡΥΘΜΙΣΕΙΣ ΔΙΑΜΟΡΦΩΣΗΣ ΣΥΣΚΕΥΩΝ

Για τη διαμόρφωση των κινητήρων και των αισθητηρίων είναι δυνατόν να ρυθμιστούν οι ακόλουθες ιδιότητες [66]

### Partners

Όταν χρησιμοποιούνται περισσότερα από ένα κουτιά NXT, αυτή η ρύθμιση αναγνωρίζει το κουτί στο οποίο είναι συνδεδεμένη η συγκεκριμένη συσκευή.



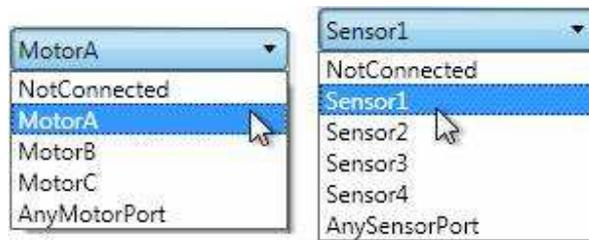
Εικόνα 6.2 Εμφάνιση της ρύθμισης Partners

### **Name**

Οι μη-μοναδικές συσκευές μπορούν να ονομαστούν και έτσι όταν συνδεθούν δυο όμοιες συσκευές στο NXT, έχουν μια ιδιότητα εν ονόματι 'Name' ώστε να μπορούν να διακριθούν μεταξύ τους. Το όνομα μπορεί να παραχθεί από την κατάσταση της υπηρεσίας.

### **Θύρες Motor και Sensor**

Το κουτί NXT έχει τρεις θύρες κινητήρων (MotorA, MotorB, MotorC) και τέσσερις θύρες αισθητηρίων (Sensor1-4). Όλες οι εξωτερικές συσκευές του NXT συνδέονται σε μια από αυτές τις θύρες και οι περισσότερες από αυτές απαιτούν να αναγνωριστεί σε ποια θύρα είναι συνδεδεμένες.



Εικόνα 6.3 Εμφάνιση των ρυθμίσεων Motor και Sensor

### **PollingFrequencyMs (Συχνότητα αναφορών)**

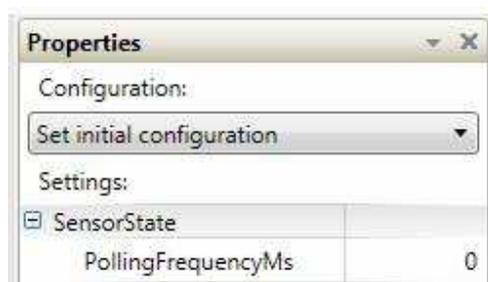
Τα περισσότερα αισθητήρια έχουν τη δυνατότητα αποστολής ειδοποιήσεων καθώς αλλάζει η τιμή τους. Αυτό γίνεται εφικτό με την συνεχόμενη επικοινωνία με το κουτί NXT. Η συχνότητα αναφορών επιτρέπει να γίνει ρύθμιση των αισθητηρίων για το κάθε πότε θα γίνεται η επικοινωνία με τους κωδικοποιητές των κινητήρων. Στις περισσότερες περιπτώσεις η τιμή της αφήνεται στο 0, που υποδηλώνει την προεπιλεγμένη συχνότητα του αισθητηρίου, όμως υπάρχουν περιπτώσεις που χρειάζονται πιο συχνά αναφορές ενός συγκεκριμένου αισθητηρίου από άλλα. Σε αυτές τις περιπτώσεις ρυθμίζετε την συχνότητα αναφορών σε τιμές ms (1 χιλιοστό του δευτερολέπτου). Ο χρόνος ανάγνωσης του αισθητηρίου για τους κωδικοποιητές των κινητήρων και των αισθητηρίων φως, ήχου και επαφής είναι περίπου 65ms. Το αισθητήριο υπερήχων και άλλα αισθητήρια τρίτων εταιρειών, έχουν χρόνο ανάγνωσης περίπου 200ms. Αν η τιμή της συχνότητας

---

αναφορών τεθεί πολύ χαμηλά για αρκετά αισθητήρια , το ρομπότ θα είναι πολύ αργό στο να απαντήσει σε όλες τις εντολές.

Όταν γίνεται βελτιστοποίηση ενός αισθητηρίου καλό θα ήταν η τιμή να είναι 50ms ή λιγότερη καθώς υποδεικνύει στο σύστημα ότι αυτή η συσκευή έχει προτεραιότητα, ακόμα και αν αργοπορήσουν κάποια άλλα αισθητήρια.

Επίσης θα ήταν καλό να απενεργοποιείται η συχνότητα αναφορών σε αισθητήρια που δε χρησιμοποιούνται (θέτοντας την τιμή -1) ή να ρυθμιστούν σε τιμή 500ms και άνω ώστε το κουτί να δέχεται μερικές ενδείξεις αλλά να επικεντρώνεται στο κύριο αισθητήριο.



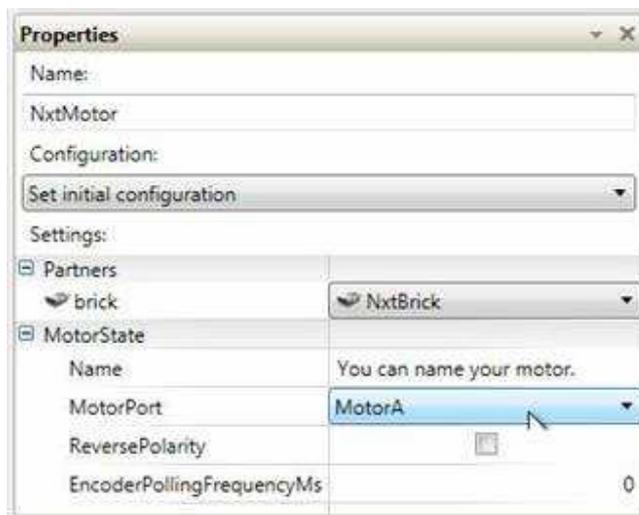
Εικόνα 6.4 Εμφάνιση της ρύθμισης *PollingFrequencyMS*

## 6.6 ΚΥΡΙΕΣ ΡΥΘΜΙΣΕΙΣ ΣΤΟ LEGO MINDSTORMS NXT

Παρακάτω παρουσιάζονται οι κύριες ρυθμίσεις των διαθέσιμων υπηρεσιών του MRDS για το LEGO MINDSTORMS NXT

### 6.6.1 ΡΥΘΜΙΣΕΙΣ ΥΠΗΡΕΣΙΑΣ ΚΙΝΗΤΗΡΩΝ

Οι κινητήρες μπορούν να ελεγχθούν μεμονωμένα ή ως ζεύγος οδηγών. Οι κωδικοποιητές είναι ενσωματωμένοι και στον κινητήρα και στις υπηρεσίες οδηγών και αναγνωρίζουν πόσες μοίρες έστριψε ο κινητήρας (360 μοίρες- μια πλήρης περιστροφή)



Εικόνα 6.5 Εμφάνιση των ιδιοτήτων της υπηρεσίας NxtMotor

Μερικές ρυθμίσεις της υπηρεσίας είναι [67]:

### **ReversePolarity**

Η ReversePolarity υποδεικνύει την πολικότητα του κινητήρα. Ενεργοποιώντας αυτή τη ρύθμιση, αντιστρέφεται η πολικότητα του κινητήρα.

### **MotorPort**

Ορίζει την θύρα (A,B,C) στην οποία είναι συνδεδεμένος ο κινητήρας. Αν ρυθμιστεί με την εντολή AnyMotorPort, η υπηρεσία θα την αντιστοιχήσει στην πρώτη διαθέσιμη θύρα κινητήρα. Αυτό μπορεί να είναι χρήσιμο όταν γίνεται εκκίνηση των κινητήρων από την υπηρεσία του πίνακα ελέγχου.

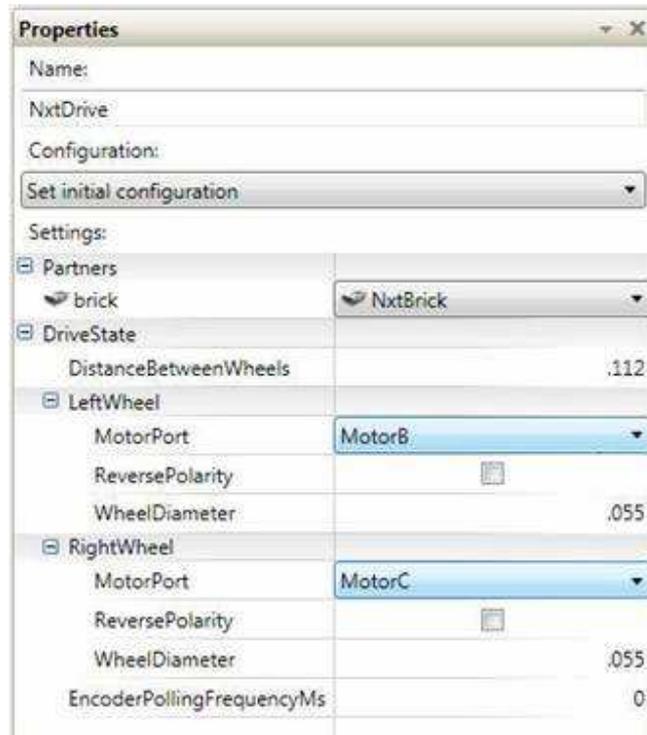
### **PollingFrequencyMs**

Αυτή η ιδιότητα καθορίζει την συχνότητα με την οποία δίνουν αναφορά οι κωδικοποιητές:

0	Προκαθορισμένος ρυθμός αναφορών
-1	Απενεργοποίηση των αναφορών
>0	Ρυθμός αναφορών σε ms

## **6.6.2 ΡΥΘΜΙΣΕΙΣ ΥΠΗΡΕΣΙΑΣ ΟΔΗΓΩΝ**

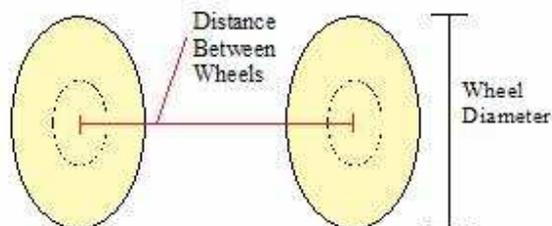
Για να λειτουργήσει κανονικά, η υπηρεσία οδηγού θα πρέπει να ρυθμιστεί ώστε να ταιριάζει με τα χαρακτηριστικά του ρομπότ στο οποίο χρησιμοποιείται [68]:



Εικόνα 6.6 Εμφάνιση των ιδιοτήτων της υπηρεσίας NxtDrive

### **DistanceBetweenWheels και WheelDiameter**

Θα πρέπει να μετρηθεί η διάμετρος κάθε τροχού και η απόσταση μεταξύ των δύο τροχών. Αυτό θα επιτρέψει στην υπηρεσία να γνωρίζει πως θα στρίψει ή πως θα οδηγήσει το ρομπότ σε μια καθορισμένη απόσταση [68]:



Εικόνα 6.7 Απόσταση μεταξύ τροχών και διάμετρος τους

**ΣΗΜΕΙΩΣΗ:** Στο MRDS οι μετρήσεις είναι πάντα σε μέτρα. Αυτό σημαίνει ότι για το LEGO MINDSTORMS NXT αν γίνει η μέτρηση σε χιλιοστά θα πρέπει να μετατραπούν μετακινώντας την υποδιαστολή κατά τρεις θέσεις αριστερά. Για παράδειγμα, αν οι

---

τροχοί μετρώνται περίπου στα 55 χιλιοστά, με την μετατροπή σε μέτρα η τιμή εισαγωγής είναι 0,055 μέτρα.

### **Κινητήρες LeftWheel και RightWheel**

Ρυθμίζει την υπηρεσία του οδηγού αναγνωρίζοντας πια θύρα κινητήρα είναι συνδεδεμένη στον κάθε τροχό

### **ReversePolarity**

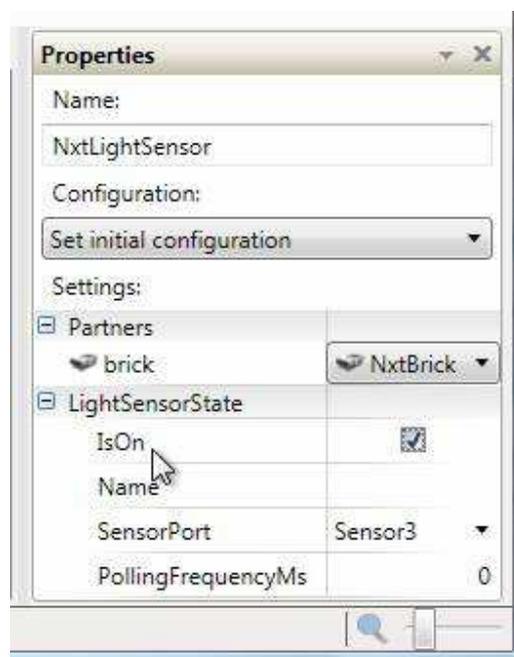
Υπάρχει περίπτωση ο κινητήρας να περιστραφεί ανάποδα, ανάλογα του πως έχει γίνει η σύνδεσή του. Στην περίπτωση αυτή, επιλέγεται το ReversePolarity.

### **PollingFrequencyMs**

Οι εντολές DriveDistance και RotateDegrees λειτουργούν τους κινητήρες για κάποια απόσταση και χρησιμοποιούν τους κωδικοποιητές που είναι ενσωματωμένοι στους κινητήρες για να γνωρίζουν πότε να σταματήσουν. Αν δεν χρησιμοποιηθούν οι κωδικοποιητές ή αυτές οι εντολές τότε στην PollingFrequencyMs συμπληρώστε -1.

## **6.6.3 ΡΥΘΜΙΣΕΙΣ ΥΠΗΡΕΣΙΑΣ ΑΙΣΘΗΤΗΡΙΟΥ ΦΩΤΟΣ**

Επιπρόσθετα από τις άλλες ιδιότητες το NxtLightSensor έχει και την ρύθμιση IsOn. Ενεργοποιώντας αυτή την παράμετρο θα έχει ως αποτέλεσμα να ανοίξει ο φακός και θα ξεκινήσει η υπηρεσία του αισθητηρίου φωτός. [69]



Εικόνα 6.8 Ιδιότητες υπηρεσίας αισθητηρίου φωτός

---

#### 6.6.4 ΡΥΘΜΙΣΕΙΣ ΥΠΗΡΕΣΙΑΣ ΑΙΣΘΗΤΗΡΙΟΥ ΧΡΩΜΑΤΟΣ

Επιπρόσθετα με της άλλες ρυθμίσεις, η υπηρεσία NxtColorSensor περιέχει την ρύθμιση SensorMode. Το αισθητήριο φωτός έχει τρία LEDs που μπορούν να ελεγχθούν από την ρύθμιση SensorMode. Οι επιλογές ελέγχου είναι [70]:

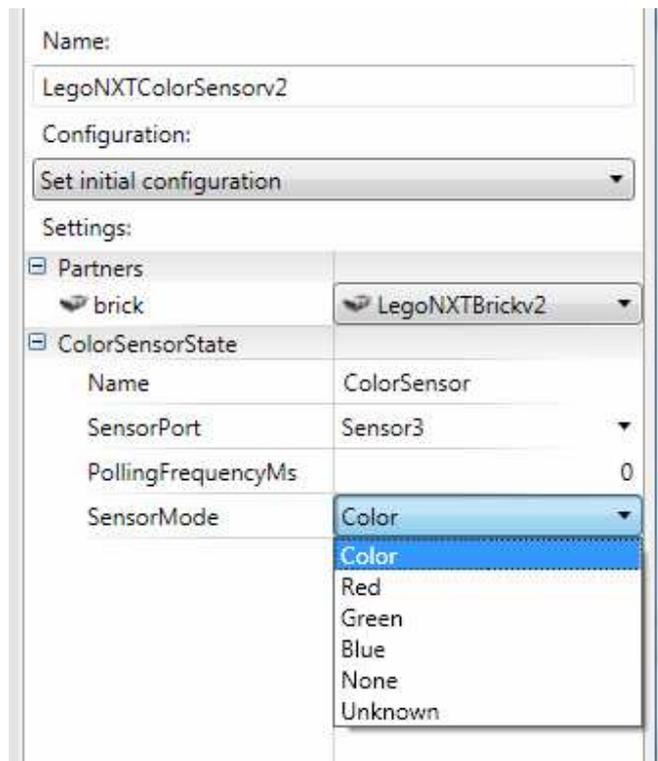
Color	Όλα τα LED ενεργοποιημένα και η ένδειξη είναι ένας αριθμός φωτός από 1-6 του επόμενου πίνακα
Red	Μόνο τα κόκκινα LED ενεργοποιημένα και η ένδειξη είναι μια τιμή έντασης από 0-1023
Green	Μόνο τα πράσινα LED ενεργοποιημένα και η ένδειξη είναι μια τιμή έντασης από 0-1023
Blue	Μόνο τα μπλε LED ενεργοποιημένα και η ένδειξη είναι μια τιμή έντασης από 0-1023
None	Κανένα LED ενεργοποιημένο και η ένδειξη είναι μια τιμή έντασης από 0-1023

Αυτές οι τιμές του SensorMode καθορίζονται στην ιδιότητα ‘enumerated type’ ‘ColorSensorMode’, έτσι ώστε να μπορούν να επιλεγούν ανά όνομα. Αν δεν καθοριστούν σε ένα αρχείο ρύθμισης, η προεπιλεγμένη τιμή είναι ‘Color’.

Στη κατάσταση ‘Color’ το αισθητήριο καθορίζει το χρώμα ενός αντικειμένου που είναι μπροστά του με βάση των ακόλουθων 6 τιμών που βασίζονται στα πιο συνηθισμένα χρώματα των LEGO MINDSTORMS NXT:

- 1 - Μαύρο (ή τίποτα εντός εμβέλειας)
- 2 - Μπλε
- 3 - Πράσινο
- 4 - Κίτρινο
- 5 - Κόκκινο
- 6 - Λευκό

Αυτοί οι αριθμοί χρωμάτων καθορίζονται από την ιδιότητα ‘enumerated type’ ‘ColorSensorMode’. Επίσης τα ονόματα εμφανίζονται στην ιστοσελίδα της υπηρεσίας ώστε να φαίνεται γρήγορα το χρώμα που ‘βλέπει’ το αισθητήριο. Να σημειωθεί ότι το αισθητήριο ανιχνεύει το πλησιέστερο χρώμα. Έτσι για παράδειγμα, αν βρεθεί μπροστά του ένα πορτοκαλί αντικείμενο θα εμφανίσει είτε κόκκινη είτε κίτρινη τιμή ανάλογα πόσο έντονο είναι. Επίσης έχει πολύ περιορισμένη εμβέλεια, που μετράται μόνο σε 1-2 εκατοστά.



Εικόνα 6.9 Διάγραμμα που δείχνει τις ρυθμίσεις του αισθητηρίου χρώματος στη VPL

## ΚΕΦΑΛΑΙΟ 7

### TUTORIALS

---

## 7.1 ΕΙΣΑΓΩΓΗ

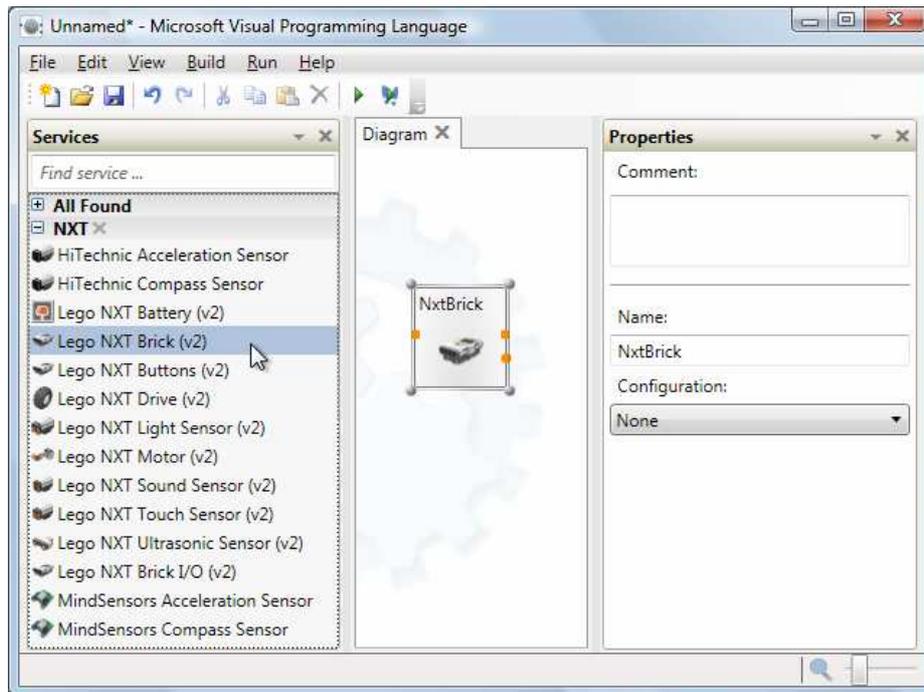
Σε αυτήν την ενότητα θα παρουσιαστούν δύο εκπαιδευτικά παραδείγματα (tutorials). Αυτοί οι οδηγοί του LEGO MINDSTORMS NXT, επιδεικνύουν πως γίνεται ο έλεγχος και η αλληλεπίδραση του LEGO MINDSTORMS NXT από τη Microsoft Visual Programming Language (VPL) [71]:

## 7.2 ΕΚΠΑΙΔΕΥΤΙΚΟ ΠΑΡΑΔΕΙΓΜΑ ΗΛΕΚΤΡΙΚΟΥ ΦΑΚΟΥ

Σε αυτό το εκπαιδευτικό παράδειγμα θα παρουσιαστεί ένα απλό πρόγραμμα που χρησιμοποιεί το αισθητήριο επαφής για να ενεργοποιήσει και να απενεργοποιήσει το αισθητήριο φωτός. Στόχος είναι η κατανόηση της δημιουργίας ενός προγράμματος LEGO NXT χρησιμοποιώντας την VPL γλώσσα και η ρύθμιση ενός κουτιού LEGO NXT με δύο αισθητήρια, χρησιμοποιώντας μια ειδοποίηση του αισθητηρίου επαφής για να ενεργοποιηθεί/απενεργοποιηθεί ο φακός του αισθητηρίου φωτός.

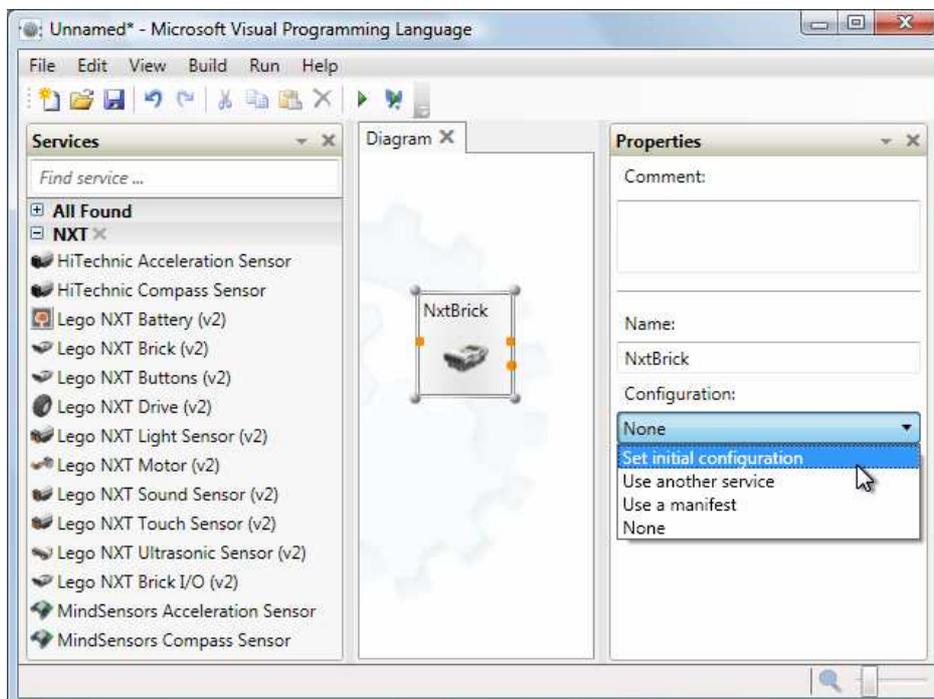
Κάθε πρόγραμμα LEGO ξεκινά γύρω από το κουτί NXT, αυτή είναι η βασική υπηρεσία που επικοινωνεί απευθείας με το υλισμικό του NXT.

Έτσι ανοίγουμε την Microsoft Visual Programming Language, επιλέγουμε το LEGO NXT Brick (v2) και το σέρνουμε στο διάγραμμα εργασίας (*Εικόνα 7.1*)



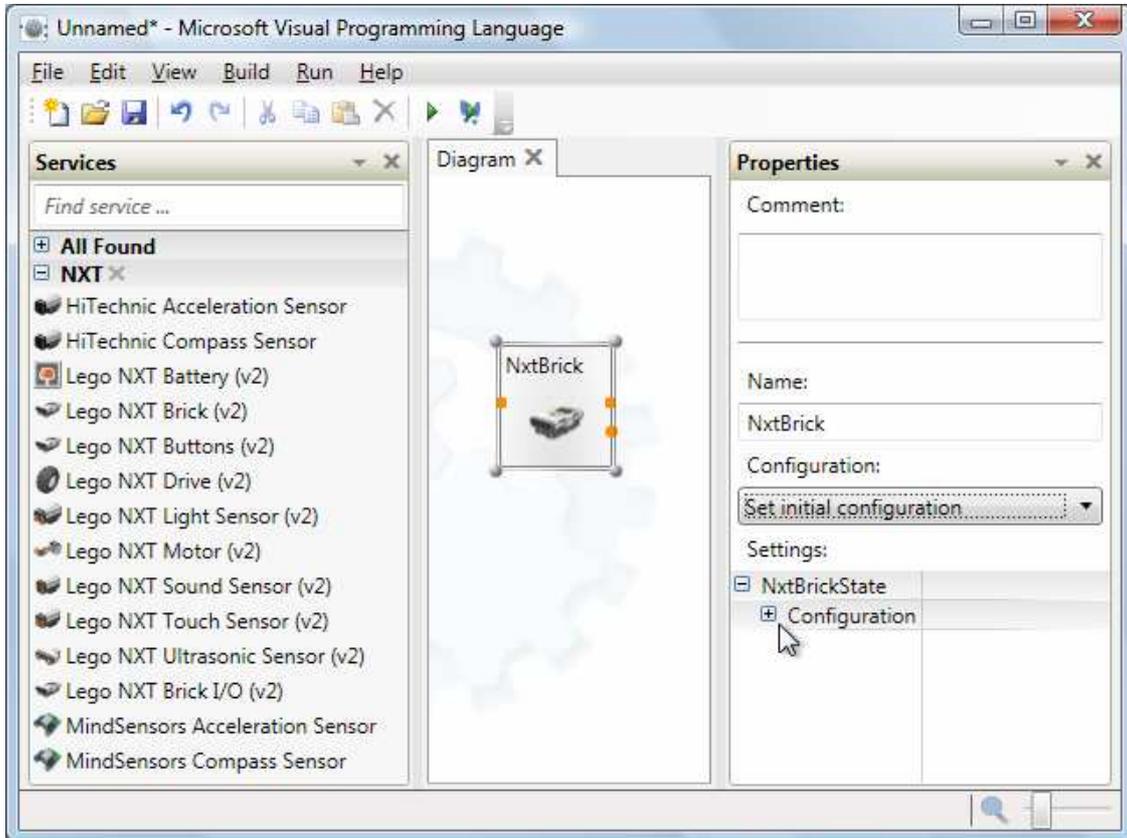
Εικόνα 7.1 Διάγραμμα εργασίας με την υπηρεσία LEGO NXT Brick (v2)

Στη συνέχεια, για την ρύθμιση του κουτιού NXT, από το παράθυρο 'Properties', στο πεδίο 'Configuration' επιλέγουμε 'Set initial Configuration' (Εικόνα 7.2)



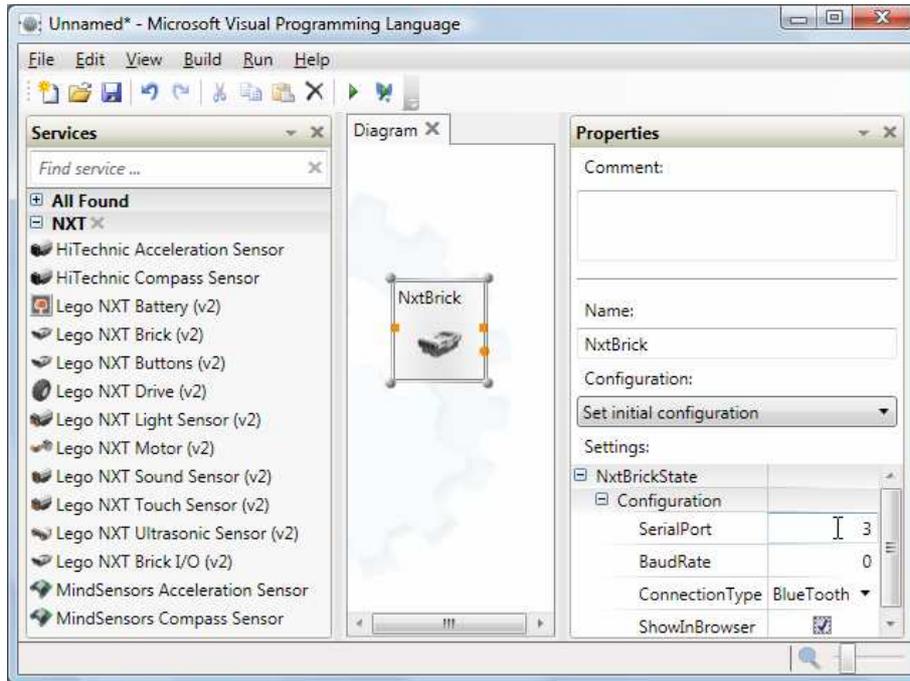
Εικόνα 7.2 Ιδιότητες του κουτιού NXT με τονισμένη την επιλογή 'Set initial configuration'

Έπειτα χρειάζεται ρύθμιση το 'NxtBrickState', η επόμενη κατάσταση δηλαδή του κουτιού NXT. Επιλέγουμε το '+' δίπλα από το πλαίσιο του 'Configuration' για να επεκταθεί το 'NxtBrickState' (Εικόνα 7.3)



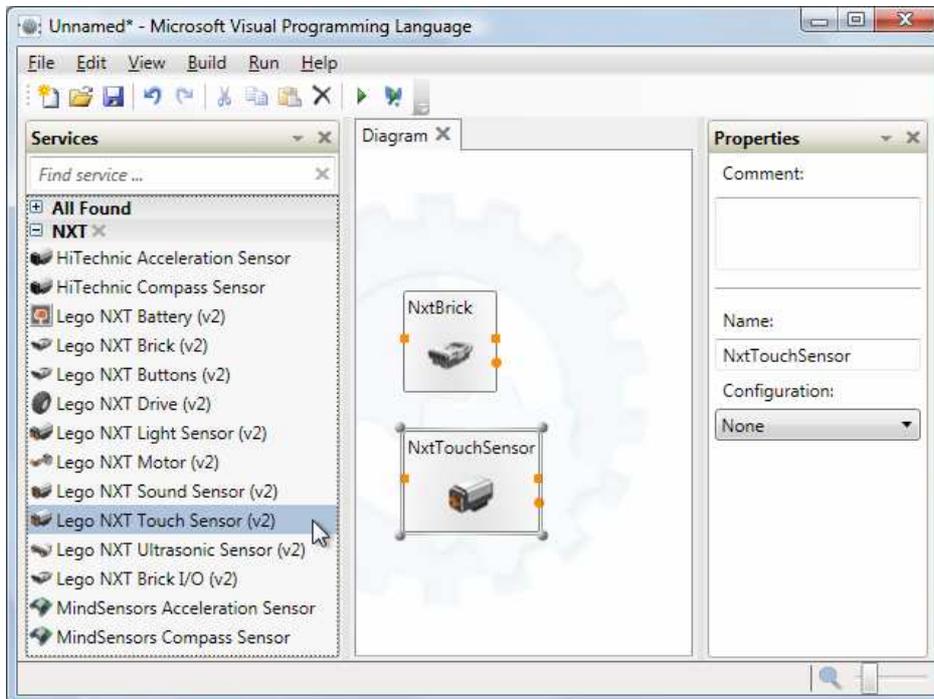
Εικόνα 7.3 Ιδιότητες του κουτιού NXT με τονισμένη την επιλογή 'NxtBrickState'

Αυτό θα εμφανίσει επιλογές για την σειριακή θύρα που επιτρέπει την σύνδεση με το LEGO NXT μέσω Bluetooth. Αν επιλεγεί το πεδίο 'ShowInBrowser', η διεπαφή χρήστη για την υπηρεσία LEGO NXT θα ανοίγει σε ένα φυλλομετρητή κάθε φορά που ξεκινά η υπηρεσία. Αυτό είναι χρήσιμο για επίβλεψη των ρυθμίσεων και την κατάσταση των εκτελούμενων συσκευών που είναι συνδεδεμένες στο κουτί NXT (Εικόνα 7.4)



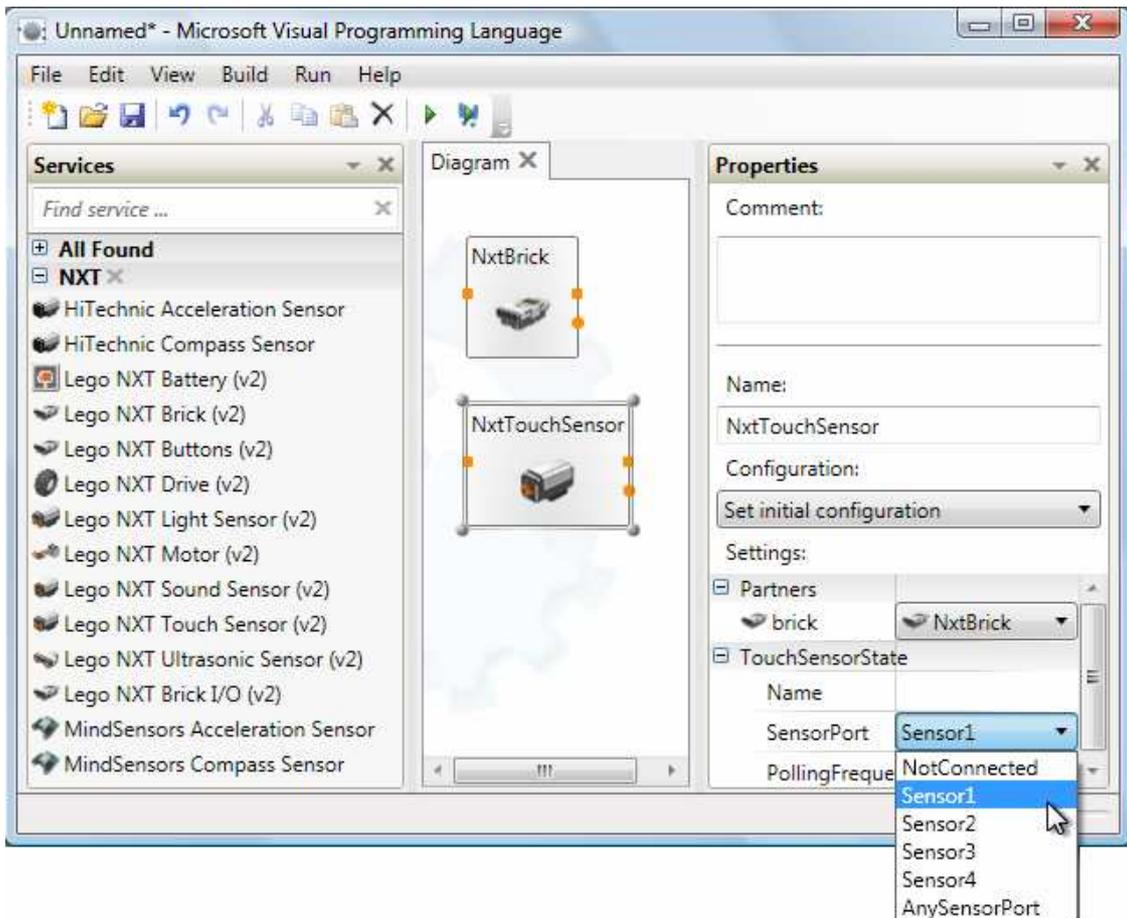
Εικόνα 7.4 Ιδιότητες του κουτιού NXT με επεκταμένη την επιλογή 'Configuration'

Στη συνέχεια προσθέτουμε την υπηρεσία 'NxtTouchSensor', που είναι η υπηρεσία του αισθητήριου επαφής, στο διάγραμμα (Εικόνα 7.5)



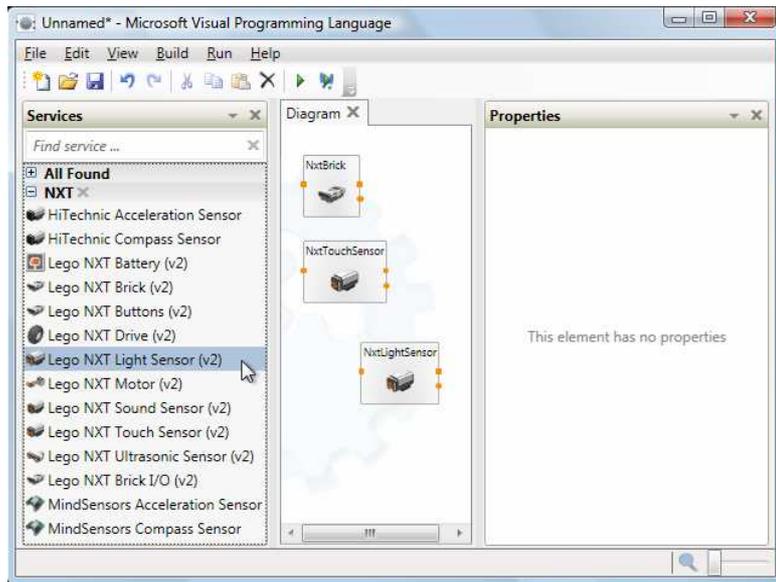
Εικόνα 7.5 Διάγραμμα με την υπηρεσία 'NxtTouchSensor'

Έπειτα ορίζουμε την αρχική διαμόρφωση του αισθητηρίου επαφής, συνδέοντας το αισθητήριο στην θύρα No1 του κουτιού και επιλέγοντας 'Sensor1' στο πεδίο 'Configuration'. Έπειτα χρειάζεται να ονομαστεί το αισθητήριο (προτεινόμενο όνομα θα ήταν 'Bumper' ή 'Switch') και αφήνουμε την προεπιλεγμένη τιμή της 'Polling Frequency' στο 0 (Εικόνα 7.6)



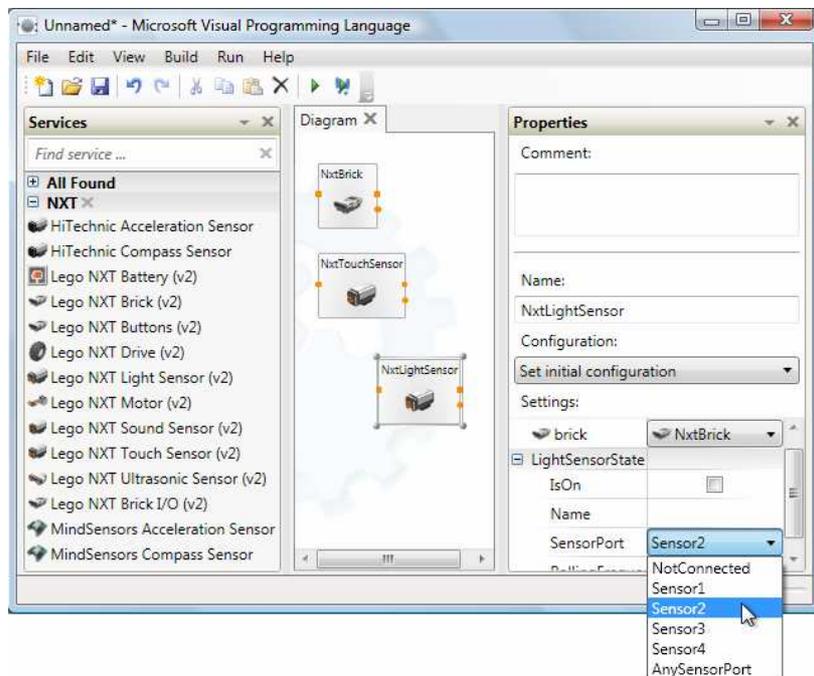
Εικόνα 7.6 Ιδιότητες του 'NxtTouchSensor' με επεκταμένη την επιλογή 'TouchSensorState'

Μετά προσθέτουμε την υπηρεσία 'NxtLightSensor', που είναι η υπηρεσία του αισθητηρίου φωτός, στο διάγραμμα (Εικόνα 7.7)



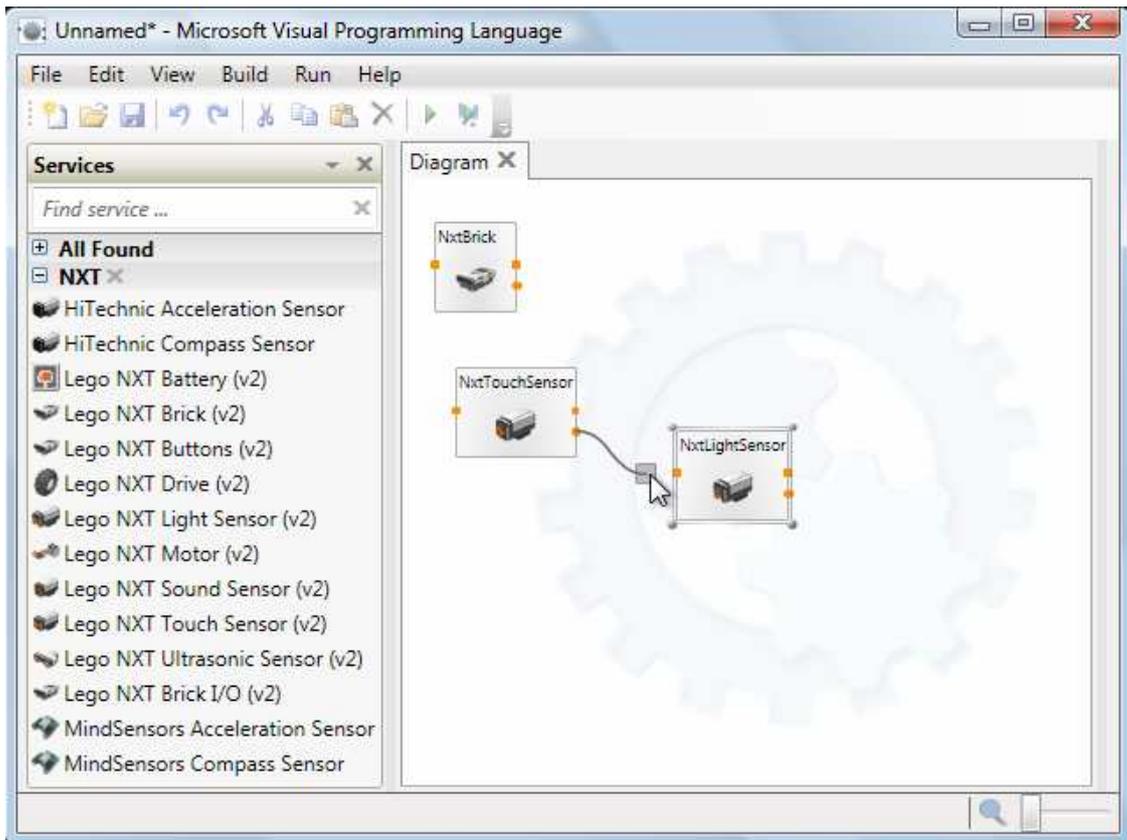
Εικόνα 7.7 Διάγραμμα με την υπηρεσία 'NxtLightSensor'

Πρέπει να συνδεθεί το αισθητήριο φωτός στη θύρα δύο των αισθητηρίων και να ρυθμιστεί ανάλογα. Δίνουμε ένα όνομα στο αισθητήριο πχ Αισθητήριο Φωτός και αφήνουμε τη συχνότητα αναφορών στο 0 που είναι η προεπιλεγμένη τιμή της. Η επιλογή IsOn καθορίζει αν ο φακός του αισθητηρίου θα είναι ανοικτός η κλειστός όταν ξεκινήσει η υπηρεσία, προς το παρών μένει κλειστός. (Εικόνα 7.8)



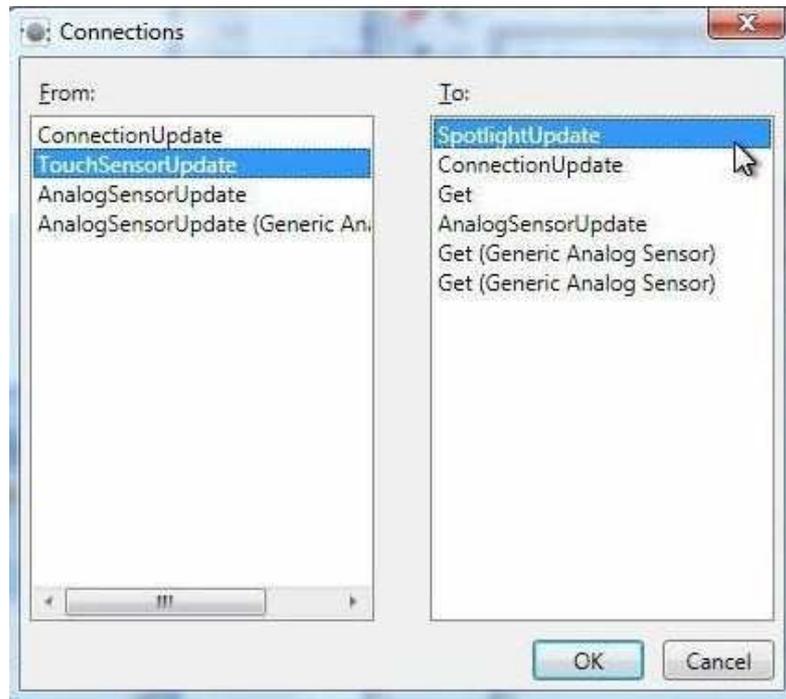
Εικόνα 7.8 Ιδιότητες του 'NxtLightSensor' με επεκταμένη την επιλογή 'LightSensorState'

Τώρα αποθηκεύονται ειδοποιήσεις από το NxtTouch κάθε φορά που το αισθητήριο επαφής πιέζεται ή ελευθερώνεται. Αυτή η τιμή θα χρησιμοποιηθεί για να ενεργοποιηθεί ή όχι ο φακός. Επιλέγουμε το στρογγυλό καρφί ειδοποίησης στο NxtTouchSensor και το ενώνουμε με το NxtLightSensor. Αυτό υποδεικνύει ότι θα αποθηκεύονται οι ειδοποιήσεις από το αισθητήριο επαφής και θα στέλνονται στο αισθητήριο φωτός. (Εικόνα 7.9)



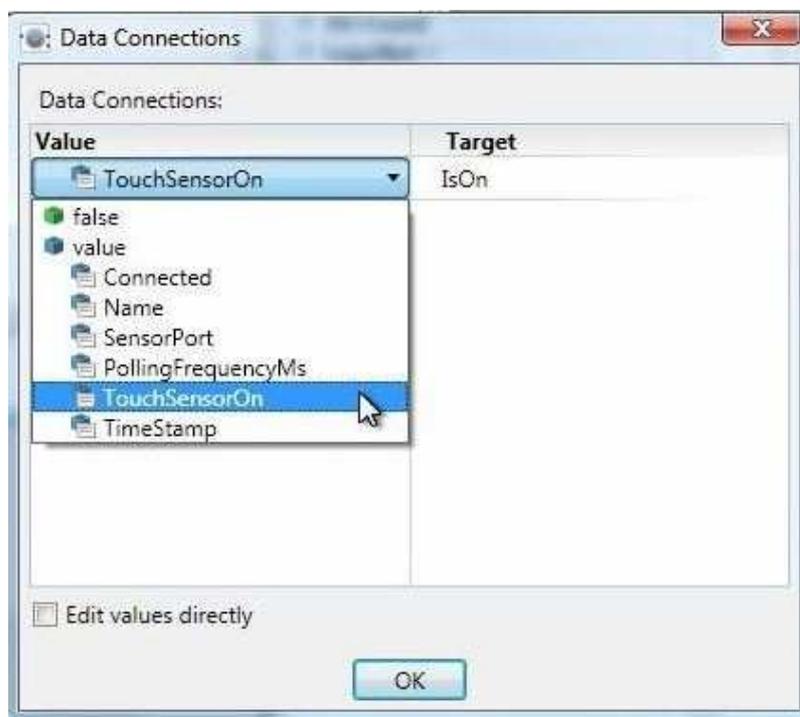
Εικόνα 7.9 Σύνδεση του 'NxtLightSensor' με το 'NxtTouchSensor'

Θα ανοίξει το παράθυρο διαλόγου 'Connections'. Επιλέγουμε την ειδοποίηση TouchSensorUpdate στείλτε την στην SpotlightUpdate και πατάμε OK. (Εικόνα 7.10)



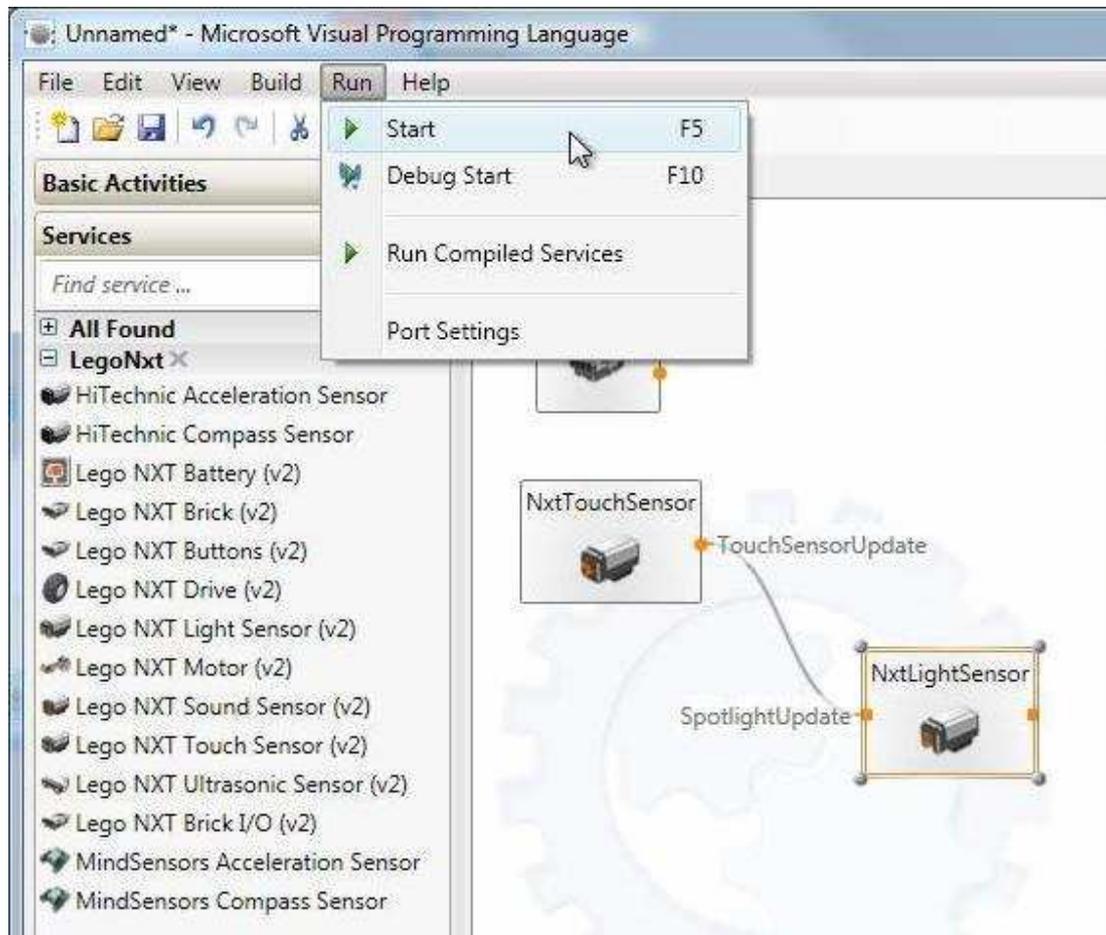
Εικόνα 7.10 Πίνακας διαλόγου μεταξύ 'NxtLightSensor' και 'NxtTouchSensor'

Έπειτα εμφανίζεται το παράθυρο διαλόγου “Data Connections”. Πρέπει να περιγραφούν τα δεδομένα τα οποία θα σταλούν στο SpotlightUpdate. Έτσι επιλέγουμε TouchSensorOn και πατάμε OK. (Εικόνα 7.11)



Εικόνα 7.11 Πίνακας συνδέσεων δεδομένων μεταξύ 'NxtLightSensor' και 'NxtTouchSensor'

Επιλέγουμε Start από το μενού Run. Το LEGO θα παράγει έναν ήχο για να υποδείξει μια επιτυχή σύνδεση. Τώρα μπορούμε να το δοκιμάσουμε ή να πατήσουμε και να κρατήσουμε το αισθητήριο επαφής. Όταν το κάνουμε αυτό, το φως αισθητηρίου θα ανάψει. Μόλις το αφήσουμε θα σβήσει (Εικόνα 7.12)

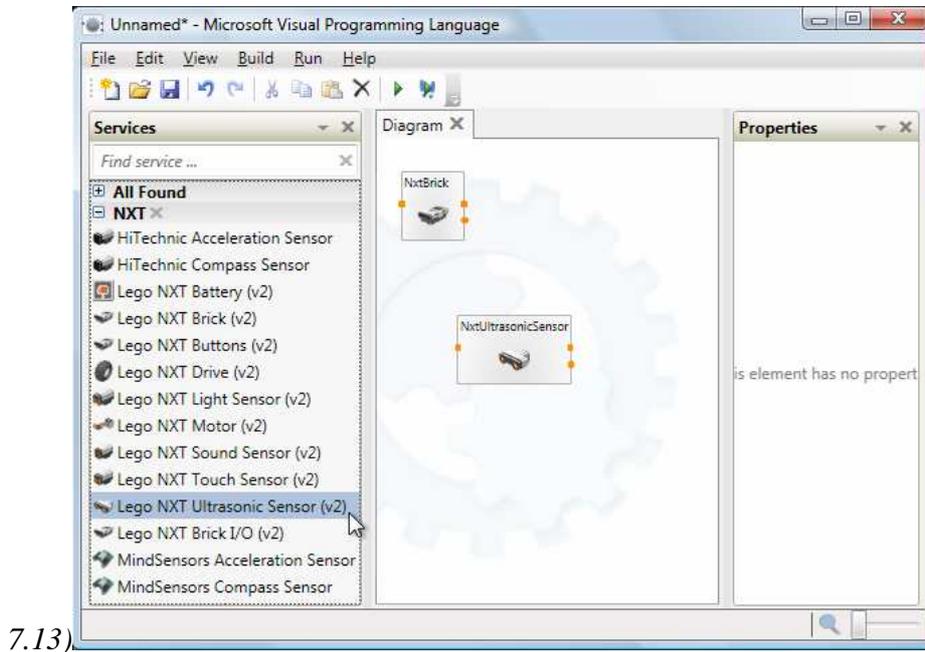


Εικόνα 7.12 Πατώντας την συντόμευση F5 εκτελείται το πρόγραμμα

### 7.3 ΕΚΠΑΙΔΕΥΤΙΚΟ ΠΑΡΑΔΕΙΓΜΑ ΑΙΣΘΗΤΗΡΙΟΥ ΥΠΕΡΗΧΩΝ

Αυτό το εκπαιδευτικό παράδειγμα θα περιγράψει βήμα προς βήμα την δημιουργία ενός προγράμματος οδήγησης στο LEGO MINDSTORMS NXT με μια ρύθμιση του Tribot χρησιμοποιώντας την προγραμματιζόμενη γλώσσα VPL, του Microsoft Robotics Studio. Αυτό το απλό πρόγραμμα θα χρησιμοποιεί ένα αισθητήριο υπερήχων για να εντοπίζει εμπόδια και να ελέγχει την οδήγηση του Tribot Ρύθμιση και οδήγηση του Tribot:

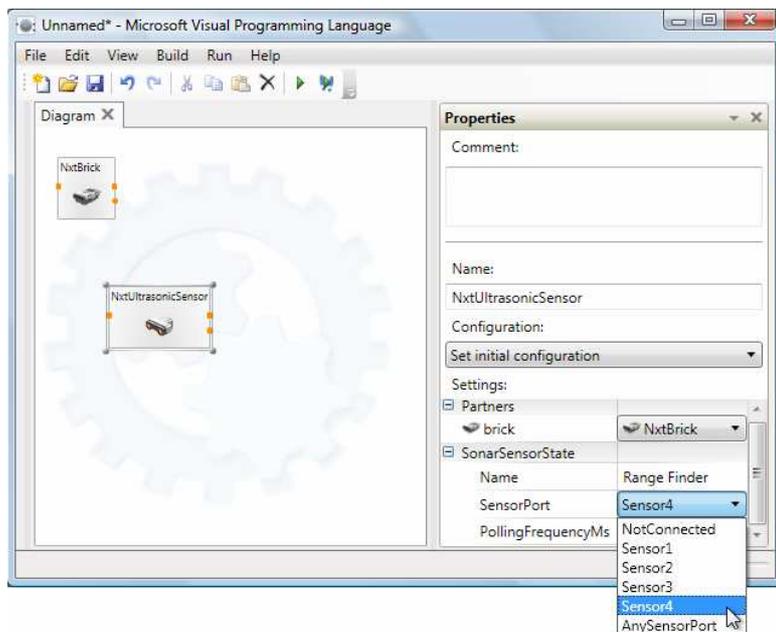
Κάθε πρόγραμμα LEGO ξεκινά προσθέτοντας και ρυθμίζοντας το κουτί NXT. Με το που ρυθμιστεί προσθέτουμε στο διάγραμμα και ένα αισθητήριο υπερήχων (Εικόνα



7.13)

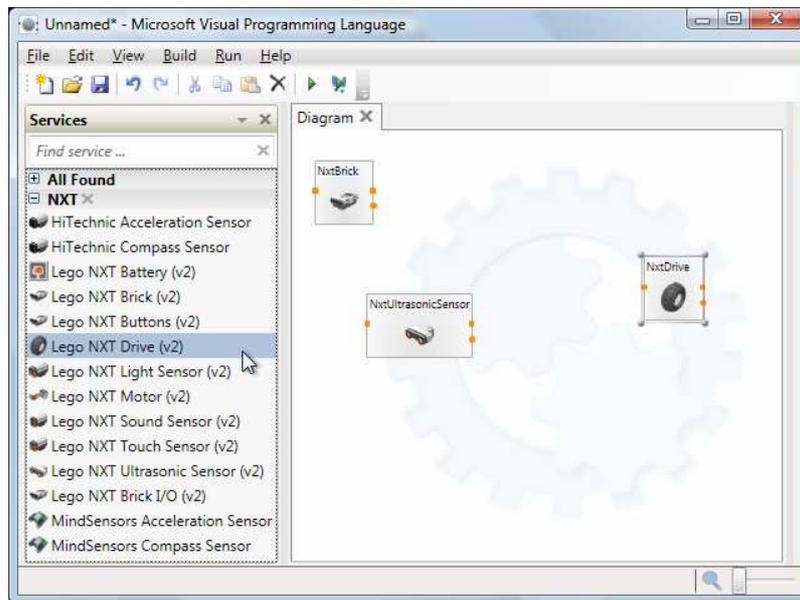
Εικόνα 7.13 Διάγραμμα με την υπηρεσία 'NxtUltrasonicSensor'

Επιλέγουμε τη ρύθμιση 'Set initial configuration'. Συνδέουμε το αισθητήριο υπερήχων στην τέταρτη θύρα του κουτιού NXT και επιλέγουμε sensor4 στην 'SensorPort'. Ονομάζουμε το αισθητήριο πχ 'RangeFinder' και αφήνουμε την συχνότητα αναφορών στο προεπιλεγμένο '0' (Εικόνα 7.14)



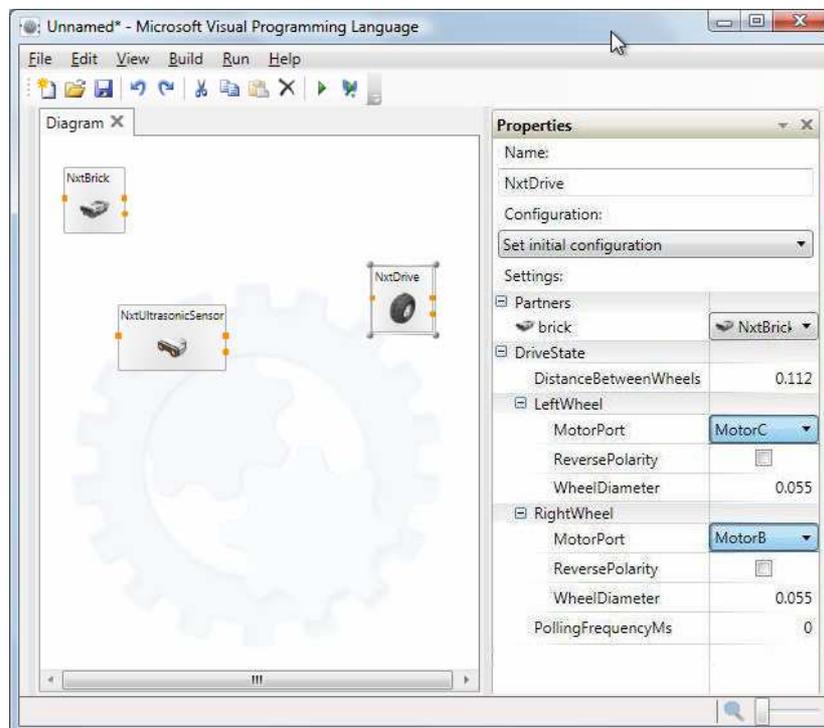
Εικόνα 7.14 Ιδιότητες του 'NxtUltrasonicSensor' με επεκταμένη την επιλογή 'SonarSensorState'

Προσθέτουμε ένα LEGO NxtDrive στο διάγραμμα (Εικόνα 7.15)



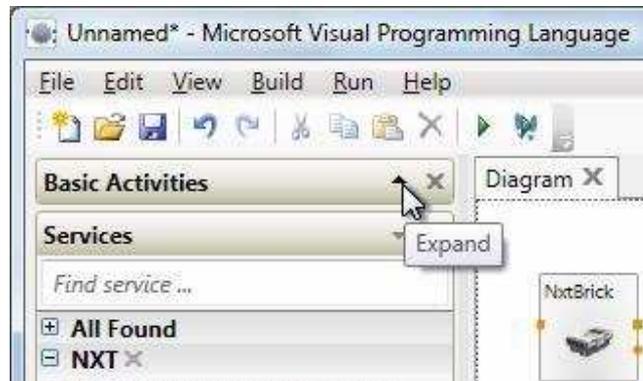
Εικόνα 7.15 Το διάγραμμα με την πρόσθεση του οδηγού 'NxtDrive'

Επιλέγουμε τη ρύθμιση 'Set initial configuration'. Συνδέουμε τον αριστερό κινητήρα στην C θύρα και τον δεξιό στην B θύρα, θέτουμε την DistanceBetweenWheels στο 0,112, τις δύο WheelDiameters στο 0,055 και αφήνουμε την PollingFrequencyMs στο προεπιλεγμένο '0' (Εικόνα 7.16)



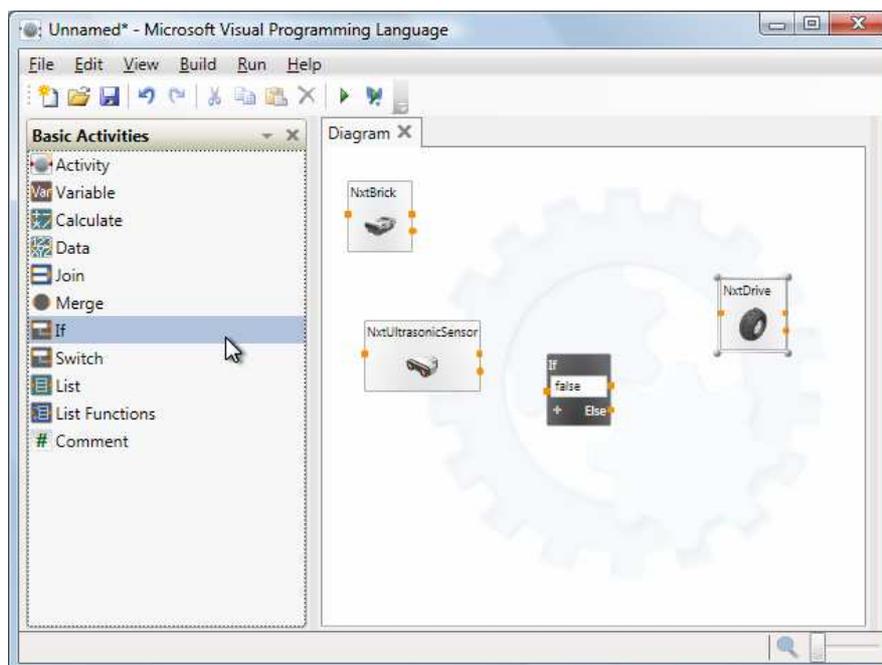
Εικόνα 7.16 Ιδιότητες του οδηγού 'NxtDrive'

Η πρόσθεση λογικής στο διάγραμμα είναι μια απλή διαδικασία, όμως χρειάζεται χώρος. Για να γίνει αυτό επεκτείνουμε το παράθυρο 'Basic Activities' (Εικόνα 7.17)



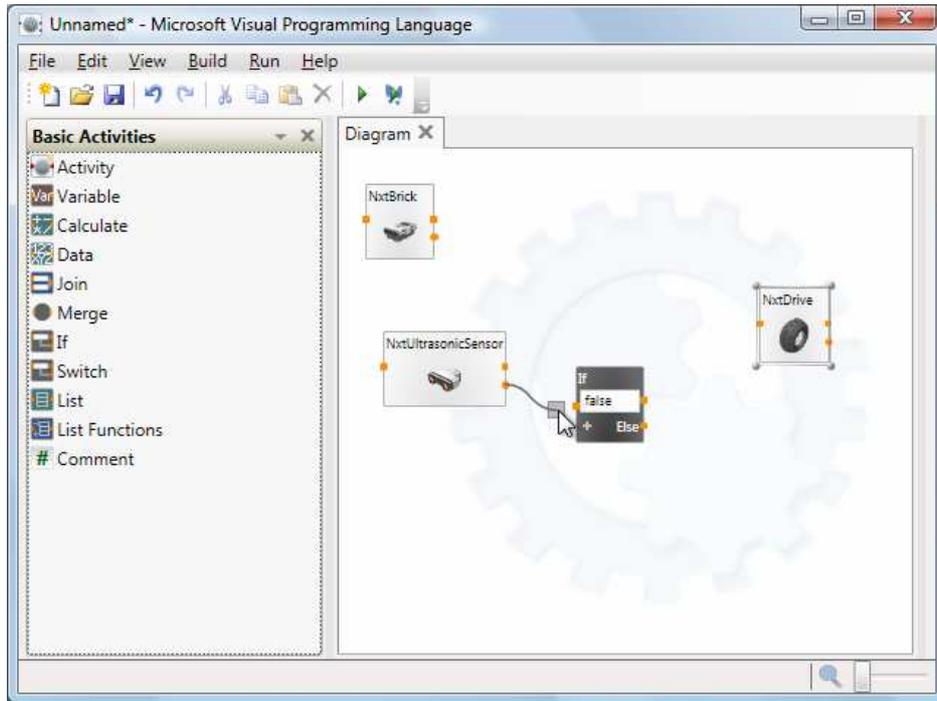
Εικόνα 7.17 Παράθυρο βασικών Δραστηριοτήτων

Από το παράθυρο φέρνουμε ένα κουτί 'If' στο διάγραμμα:



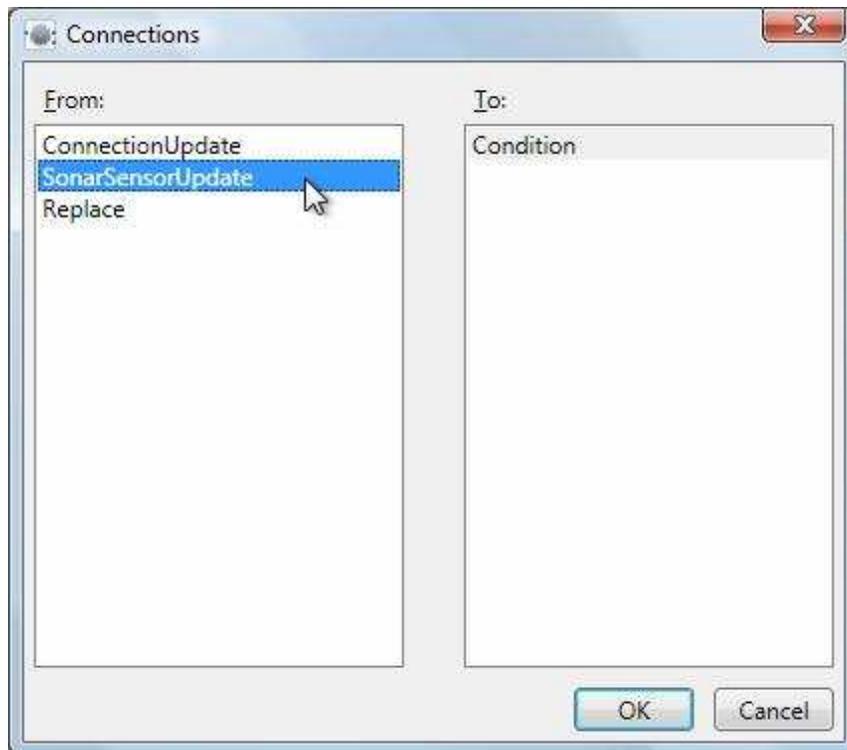
Εικόνα 7.18 Διάγραμμα με προστεθειμένη την δραστηριότητα If

Οι ειδοποιήσεις από το 'NxtUltrasonicSensor' θα αποθηκεύονται κάθε φορά που λαμβάνεται μια ένδειξη 'Distance'. Αυτή η τιμή θα χρησιμοποιηθεί για την οδήγηση των κινητήρων. Επιλέγουμε από το στρογγυλό καρφί ειδοποίησης του 'NxtUltrasonicSensor' και το συνδέουμε με το κουτί 'If'. Αυτό υποδεικνύει ότι θα αποθηκεύονται οι ειδοποιήσεις του αισθητηρίου πίεσης και θα αξιολογούνται (Εικόνα 7.19)



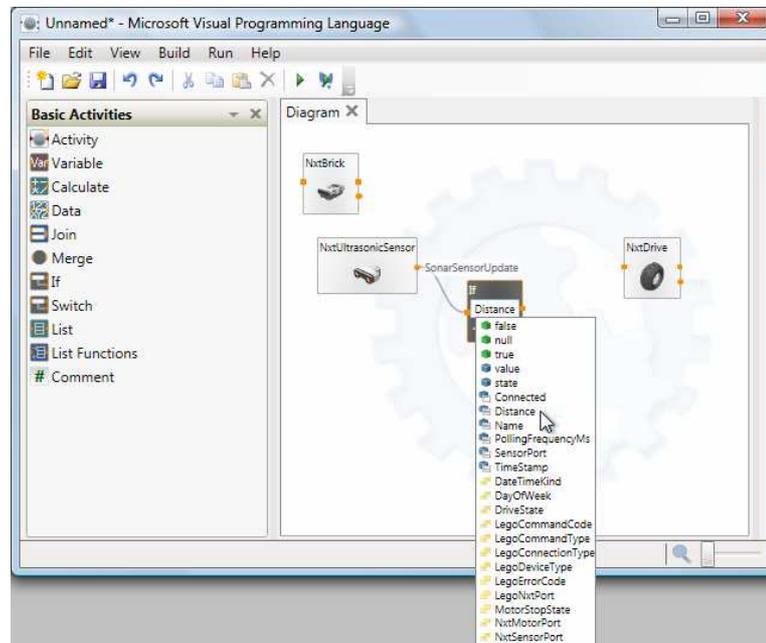
Εικόνα 7.19 Σύνδεση του αισθητηρίου με την δραστηριότητα If

Έπειτα θα ανοίξει ένα παράθυρο διαλόγου 'Connections'. Επιλέγουμε την ειδοποίηση από το 'SonarSensorUpdate' και πατάμε OK (Εικόνα 7.20)



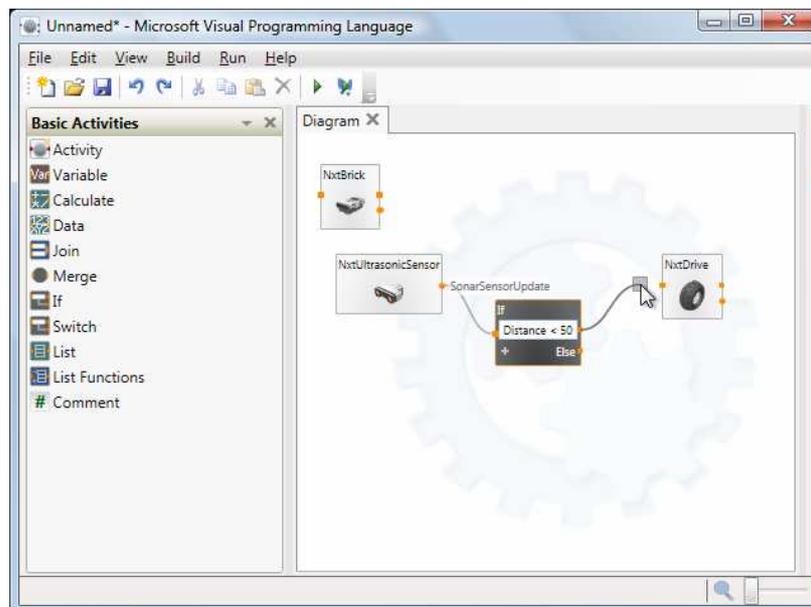
Εικόνα 7.20 Διάλογος Συνδέσεων

Στο κουτί 'If', πληκτρολογούμε ένα κενό. Παρατηρήστε ότι ανοίγει ένα μενού εμφανίζοντας τις διαθέσιμες τιμές του αισθητηρίου. Επιλέγουμε ή πληκτρολογούμε την Distance και προσθέτουμε το < 50. Αυτό σημαίνει 'όταν η απόσταση είναι μικρότερη από 50cm' (Εικόνα 7.21)



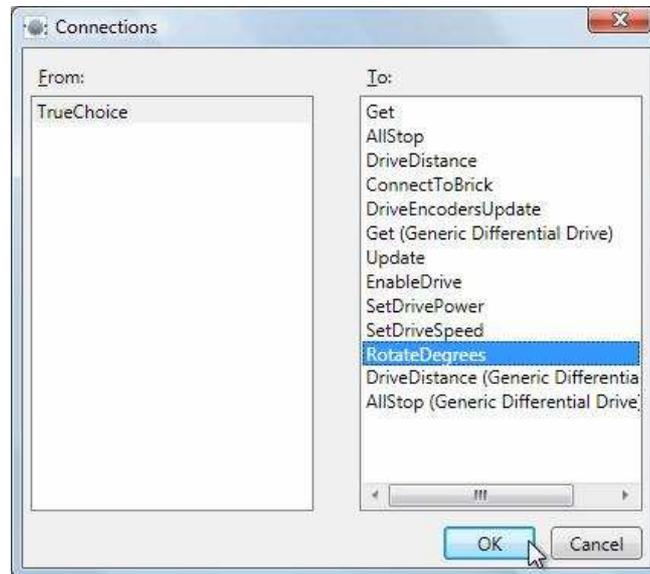
Εικόνα 7.21 Μενού με τους διάφορους τύπους μεταβλητών προς επιλογή

Συνδέουμε από τα αριστερά τη συνθήκη στην δραστηριότητα NxtDrive (Εικόνα 7.22)



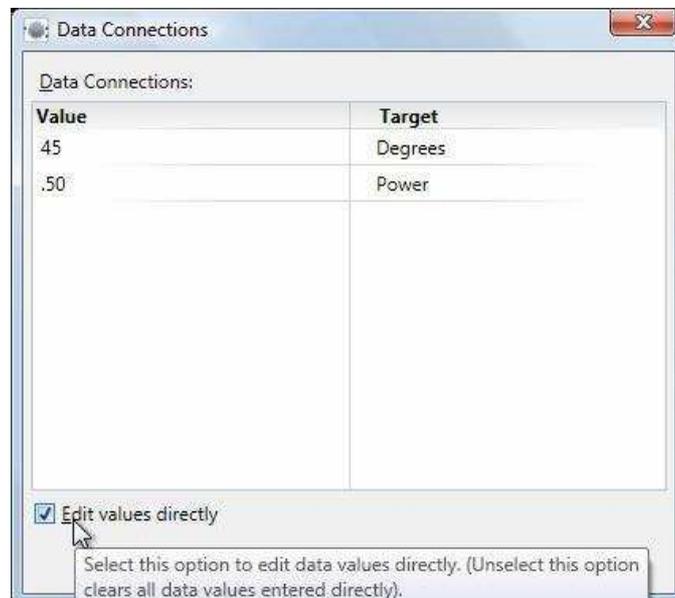
Εικόνα 7.22 Σύνδεση της δραστηριότητας If με την δραστηριότητα οδηγού 'NxtDrive'

Αυτό θα ανοίξει το παράθυρο διαλόγου 'Connections'. Επιλέγουμε από το 'TrueChoice' το RotateDegrees και πατάμε OK (Εικόνα 7.23)



Εικόνα 7.23 Διάλογος σύνδεσης δραστηριότητας οδηγού

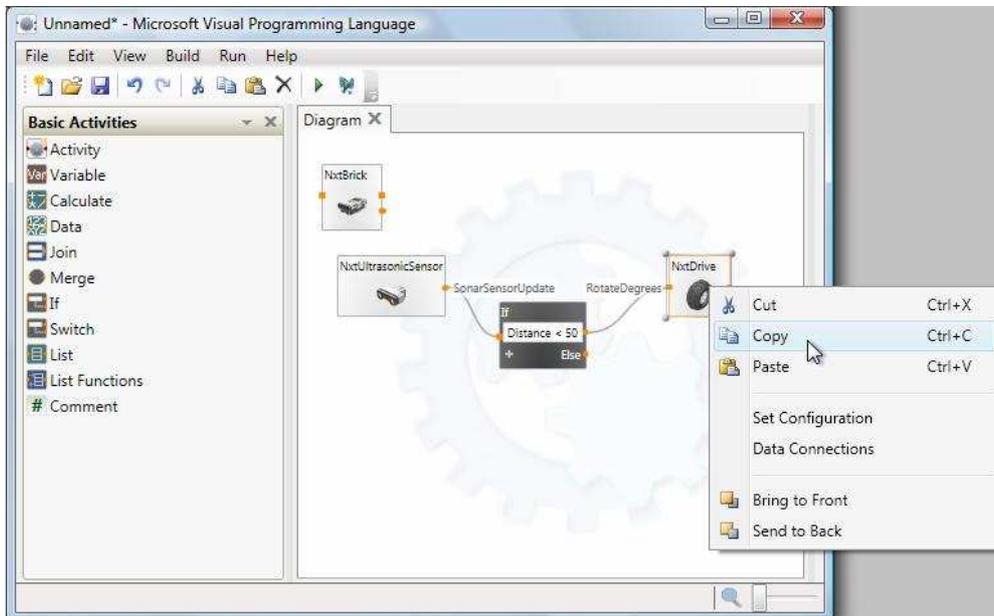
Έπειτα στο παράθυρο διαλόγου 'Data Connections' του 'RotateDegrees' που θα ανοίξει επιλέγουμε το 'Edit Values Directly' πληκτρολογούμε 45 Degrees και 0,50 Power και πατάμε OK. Υποθέτοντας ότι δεν έχουν δοθεί άλλες εντολές οδήγησης, αυτό θα γυρίσει το ρομπότ 45 μοίρες στα αριστερά (Εικόνα 7.24)



Εικόνα 7.24 Συνδέσεις δεδομένων για το RotateDegrees

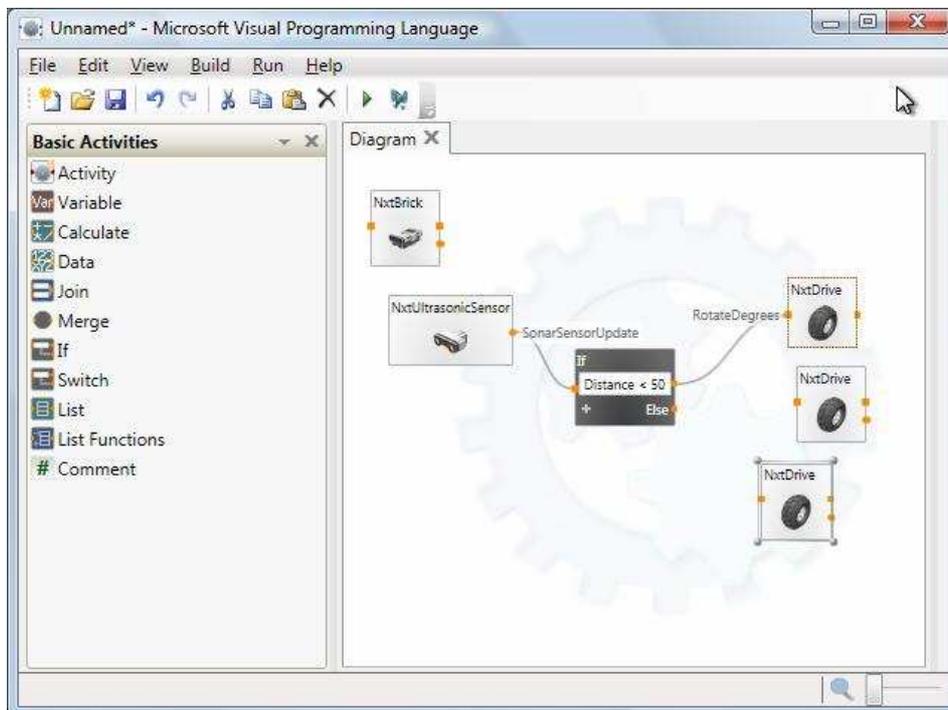
Η αποφυγή εμποδίων δεν είναι αρκετή για εξερεύνηση του περιβάλλοντος. Για να γίνει εξερεύνηση θα πρέπει να χρησιμοποιηθεί η δραστηριότητα NxtDrive άλλες δύο

φορές. Πατάμε δεξί κλικ με το ποντίκι στο κουτί NxtDrive και επιλέγουμε 'Copy' (Εικόνα 7.25)



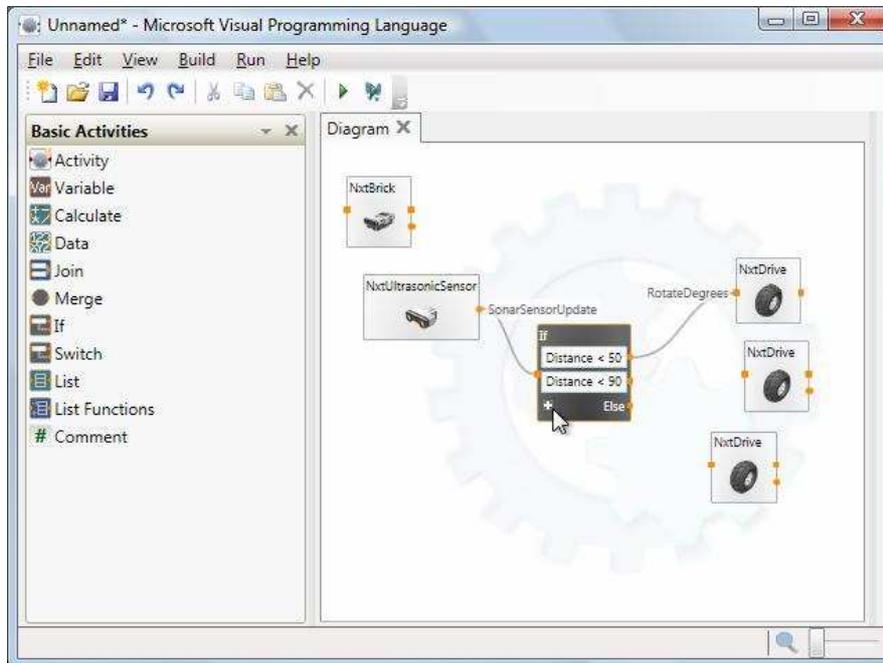
Εικόνα 7.25 Αντιγραφή της δραστηριότητας 'NxtDrive'

Πατάμε το 'Paste' από το μενού Edit δύο φορές για να αντιγράψουμε τον οδηγό στο διάγραμμα. Κάνοντας αυτήν την αντιγραφή, αντί να δημιουργούμε από την αρχή νέο αντικείμενο, μπορούμε να δίνουμε εντολές στον ίδιο οδηγό (Εικόνα 7.26)



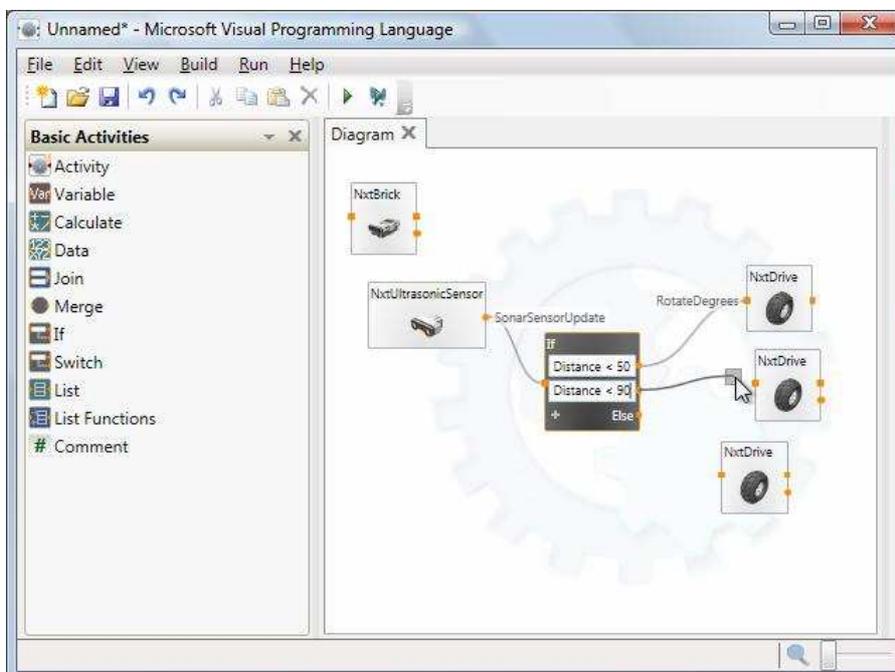
Εικόνα 7.26 Επικόλληση της δραστηριότητας 'NxtDrive'

Επεκτείνετε το κουτί 'If' πατώντας το '+' κάτω αριστερά και προσθέστε την νέα συνθήκη Distance <90 (Εικόνα 7.27)



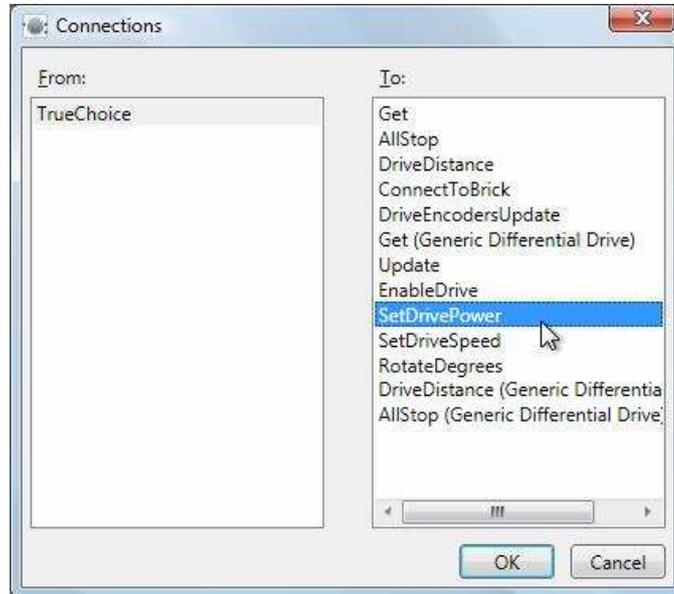
Εικόνα 7.27 Πρόσθεση συνθήκης στη δραστηριότητα If

Συνδέουμε τη δεύτερη συνθήκη, με τον δεύτερο οδηγό NxtDrive (Εικόνα 7.28)

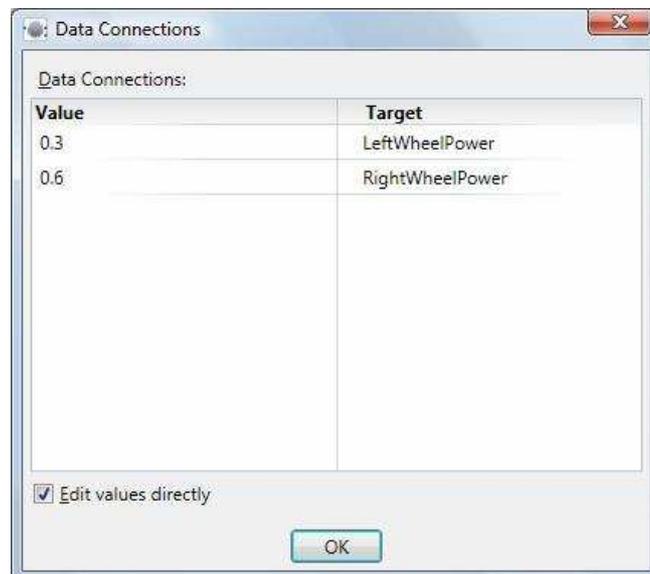


Εικόνα 7.28 Πρόσθεση συνθήκης στη δραστηριότητα If και σύνδεση με το δεύτερο οδηγό

Αυτή τη φορά επιλέγουμε SetDrivePower και εισάγουμε την τιμή 0,3 για την LeftWheelPower και 0,6 για την RightWheelPower. Έτσι όταν ένα εμπόδιο είναι ανάμεσα στα 50 και 90 εκατοστά το ρομπότ θα στρίψει αριστερά (Εικόνα 7.29)

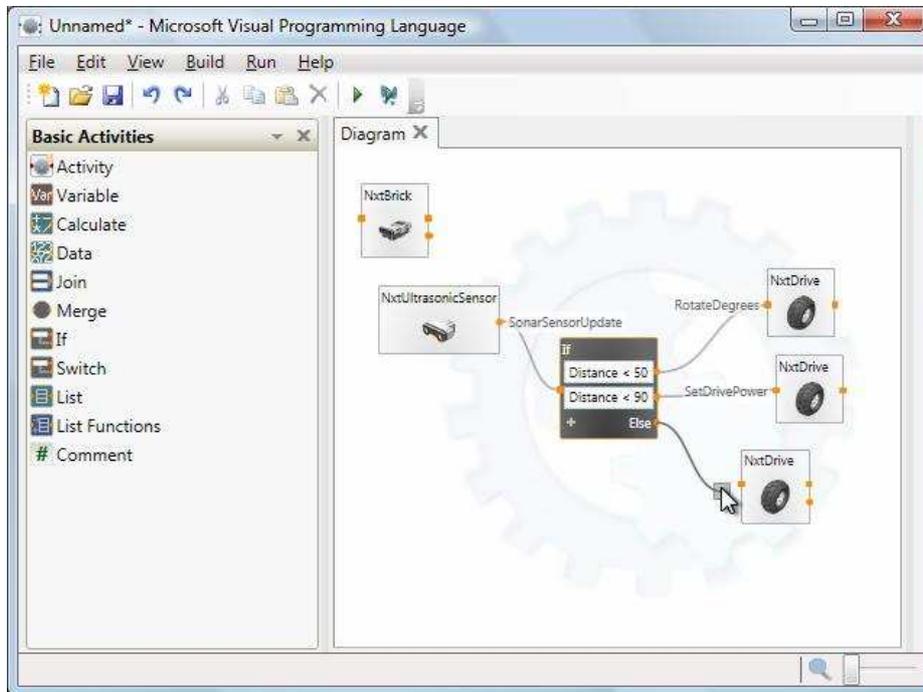


Εικόνα 7.29 Επιλογή SetDrivePower



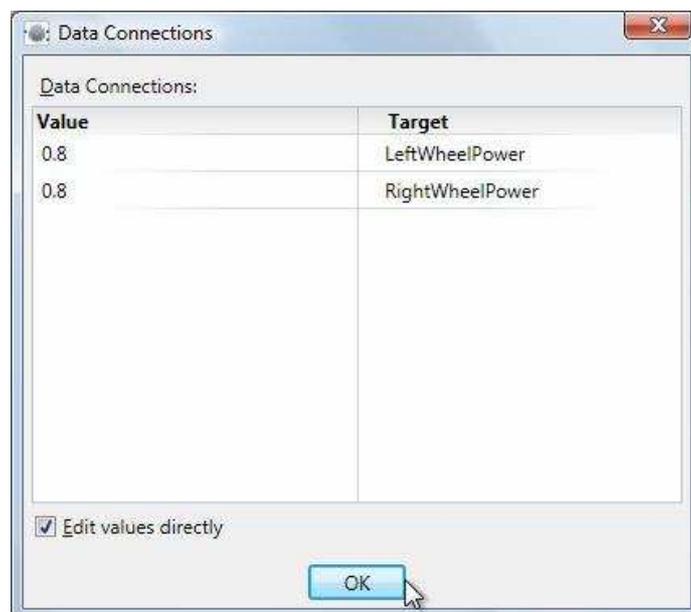
Εικόνα 7.30 Εισαγωγή διαφορετικών τιμών δεδομένων

Τέλος συνδέουμε την έξοδο 'Else' στο τρίτο οδηγό 'NxtDrive'. Στο διάλογο 'Connections' επιλέγουμε πάλι 'SetDrivePower' (Εικόνα 7.31)



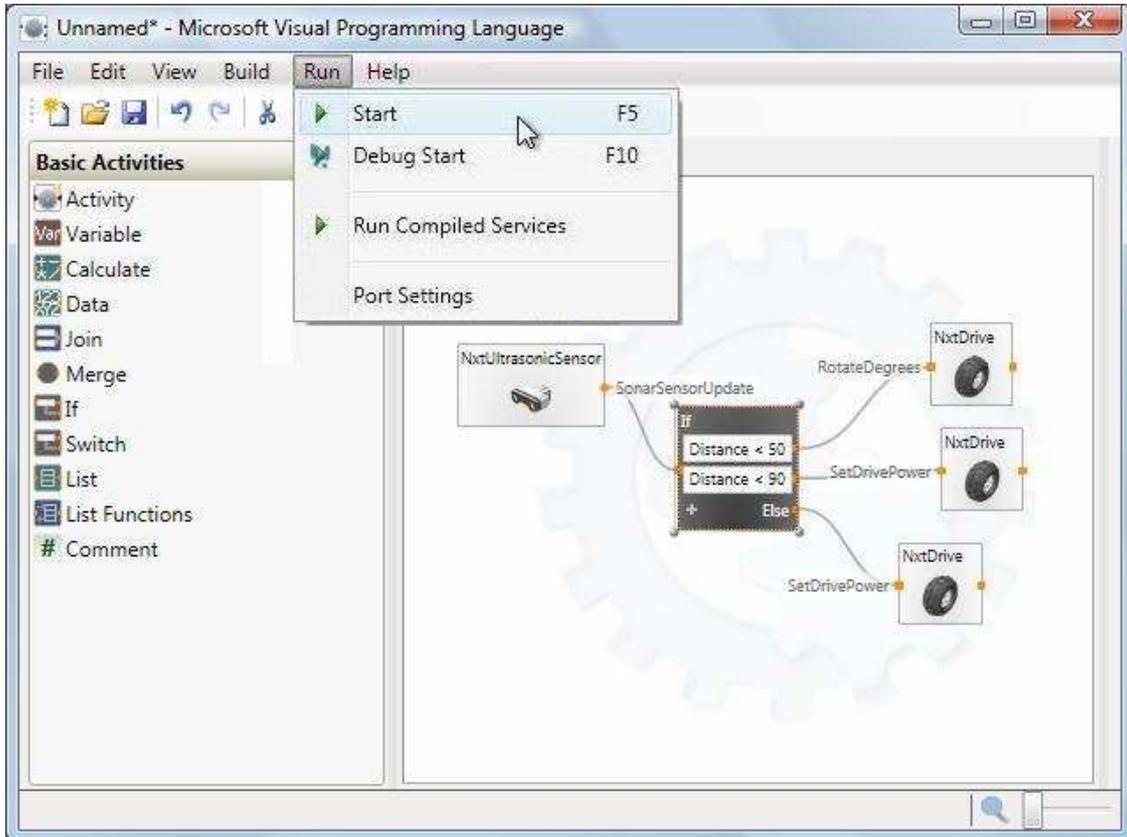
Εικόνα 7.31 Σύνδεση με τον τρίτο οδηγό

Όταν δεν υπάρχουν εμπόδια σε απόσταση μικρότερη από 90 εκατοστά από το ρομπότ θα προχωρήσει με ισχύ 80% ευθεία (Εικόνα 7.32)



Εικόνα 7.32 Εισαγωγή δεδομένων για 80% ισχύ

Τώρα το πρόγραμμα είναι έτοιμο προς εκτέλεση. Επιβεβαιώνουμε ότι είναι ενεργοποιημένο το ρομπότ και επιλέγουμε Run, Start από το μενού της VPL. Αν όλα πάνε καλά, το ρομπότ θα τριγυρίζει και κάθε φορά που πλησιάζει ένα εμπόδιο θα στρίβει προς μια διαφορετική κατεύθυνση (Εικόνα 7.33)



Εικόνα 7.33 Εκτέλεση του προγράμματος

---

## ΒΙΒΛΙΟΓΡΑΦΙΑ

- [1] M. Corporation. (2011, 5/2/2012). *Challenges Posed by Robotics Development*. Available: <http://msdn.microsoft.com/en-us/library/bb483076.aspx#Challenges>
- [2] Wikipedia. (2010). *Robot software*. Available: [http://en.wikipedia.org/wiki/Robot\\_software](http://en.wikipedia.org/wiki/Robot_software)
- [3] M. T. Jones. (2006 5/2/2012). Open source robotics toolkits. Available: <http://www.ibm.com/developerworks/linux/library/l-robotools/?ca=dgr-lnxw09OSRoboticToolkits>
- [4] M. Corporation. (2011, 5/2/2012). *Benefits of Simulation*. Available: <http://msdn.microsoft.com/en-us/library/bb483076.aspx#Benefits>
- [5] M. Corporation. (2011, 5/2/2012). *Simulation Drawbacks and Limitations*.
- [6] S. o. Robots. (2012, 5/2/2012). *Programming - ROBOT Simulation*. Available: [http://www.societyofrobots.com/programming\\_robot\\_simulation.shtml](http://www.societyofrobots.com/programming_robot_simulation.shtml)
- [7] A. A. Group, "Siemens PLM Software's Robotics Simulation: Validating & Commissioning the Virtual Work-cell," ed, 2008.
- [8] G. J. Xavier Cyril, Jean St-Pierre, "ADVANCED SPACE ROBOTICS SIMULATION FOR TRAINING AND OPERATIONS," *Institute of Aeronautics and Astronautics, Inc.*, 2000.
- [9] ROS. (2012). Available: <http://www.ros.org/wiki/electric>
- [10] ROS. (2012, 5/2/2012). Available: <http://www.ros.org/wiki/Repositories>
- [11] B. G. Morgan Quigley, Ken Conley, Josh Faust, Tully Foote, Jeremy Leibs, Eric Berger, Rob Wheeler, Andrew Ng., "ROS: an open-source Robot Operating System," 2010.
- [12] GOSTAI. (2012). *Why Open Source*. Available: <http://gostai.com/products/whyopensource/>
- [13] J.-C. D. Baillie, Akim; Nottale, Matthieu; Hocquet, Quentin; Tardieu, Samuel, "The Urbi Universal Platform for Robotics," 2008.
- [14] J.-C. D. Baillie, Akim; Nottale, Matthieu; Hocquet, Quentin; Tardieu, Samuel, "Tag: Job Control in urbiscript," 2010.
- [15] Wikipedia. (2012, 5/2/2012). *URBI*. Available: [http://en.wikipedia.org/wiki/URBI#Components\\_for\\_Robotics:\\_speech.2C\\_face.2Fobject\\_detection.2C\\_SLAM](http://en.wikipedia.org/wiki/URBI#Components_for_Robotics:_speech.2C_face.2Fobject_detection.2C_SLAM)
- [16] GOSTAI. (2012, 5/2/2012). *The Urbi Software Development Kit*. Available: <http://www.gostai.com/downloads/urbi/doc/>
- [17] Fedora. (5/2/2012). *FedoraProject*. Available: <http://fedoraproject.org/wiki/Robotics>

- 
- [18] Wikipedia. (2011, 5/2/2012). *Microsoft Robotics Developer Studio*. Available: [http://en.wikipedia.org/wiki/Microsoft\\_Robotics\\_Developer\\_Studio](http://en.wikipedia.org/wiki/Microsoft_Robotics_Developer_Studio)
- [19] (2007, 19/8/2010). *Technical Paper for the Darpa Challenge*. Available: [http://www.darpa.mil/grandchallenge/TechPapers/Princeton\\_University.pdf](http://www.darpa.mil/grandchallenge/TechPapers/Princeton_University.pdf)
- [20] D. Worthington. (2008, 5/2/2012). Microsoft's shift to parallel computing. Available: [http://www.sdtimes.com/MICROSOFT\\_S\\_SHIFT\\_TO\\_PARALLEL\\_COMPUTING/About\\_CLOUDCOMPUTING\\_and\\_MULTICORE\\_and\\_MICROSOFT/32651](http://www.sdtimes.com/MICROSOFT_S_SHIFT_TO_PARALLEL_COMPUTING/About_CLOUDCOMPUTING_and_MULTICORE_and_MICROSOFT/32651)
- [21] Wikipedia. (2010, 5/2/2012). *Webots*. Available: <http://en.wikipedia.org/wiki/Webots>
- [22] R. T. Lukas Hohl, Olivier Michel, Auke Jan Ijspeert. (2005, 5/2/2012). *Aibo and Webots: Simulation, wireless remote control and controller transfer*. Available: <http://birg2.epfl.ch/publications/fulltext/hohl06.pdf>
- [23] L. SarahDegallier, "Hand placement during quadruped locomotion in a humanoid robot:A dynamical system approach," presented at the IEEE/RSJ International Conference on Intelligent Robots and Systems, San Diego, CA, USA, 2007.
- [24] J. M. Pugh, Alcherio, "Distributed Adaptation in Multi-Robot Search using Particle Swarm Optimization," presented at the 10th International Conference on the Simulation of Adaptive Behavior Osaka, Japan, 2008.
- [25] J. N. Christopher M. Cianci, Amanda Prorok, and Alcherio Martinoli, "Assembly of Configurations in a Networked Robotic System: A Case Study on a Reconfigurable Interactive Table Lamp," 2008.
- [26] D. S. Louis-Emmanuel Martinet, Karim Benchenane, and Angelo Arleo, "Spatial Learning and Action Planning in a Prefrontal Cortical Network Model," 2011.
- [27] M. M. Mannella F., Baldassarre G., "A computational model of the amygdala nuclei's role in second order conditioning.," 2008.
- [28] M. A. A. Sproewitz, Y. Bourquin and A. J. Ijspeert, "An active connection mechanism for modular self-reconfigurable robotic systems based on physical latching," 2008.
- [29] Wikipedia. (2010). Available: [http://en.wikipedia.org/wiki/AnyKode\\_Marilou](http://en.wikipedia.org/wiki/AnyKode_Marilou)
- [30] H. magazine. (2012, 5/2/2012). *Image*. Available: <http://hplusmagazine.com/sites/default/files/images/articles/feb10/openrave-robots.gif>
- [31] R. Diankov, "Automated Construction of Robotic Manipulation Programs," The Robotics Institute, Carnegie Mellon University, 2010.
- [32] R. Diankov. (2011, 5/2/2012). *OpenRAVE Overview*. Available: <http://openrave.org/en/main/overview.html>
- [33] R. Diankov. (2010, 5/2/2012). *IKFast: The Robot Kinematics Compiler*. Available: <http://openrave.org/en/main/openravepy/ikfast.html#ikfast-the-robot-kinematics-compiler>

- 
- [34] B. Gerkey, Vaughan, R., and Howard, A., "The Player/Stage Project: Tools for Multi-Robot and Distributed Sensor Systems," presented at the Proceedings of the International Conference on Advanced Robotics, 2003.
- [35] T. H. J. Collet, MacDonald, B. A., and Gerkey, B., "Player 2.0: Toward a practical robot programming framework," presented at the Australasian Conference on Robotics and Automation (ACRA), 2005.
- [36] PlayerStage. (2011, 5/2/2012). Available: <http://playerstage.sourceforge.net/wiki/PlayerUsers>
- [37] PlayerStage. (2012). Available: [http://playerstage.sourceforge.net/doc/Player-cvs/player/supported\\_hardware.html](http://playerstage.sourceforge.net/doc/Player-cvs/player/supported_hardware.html)
- [38] N. L. R. Module. (2011, 5/2/2012). Available: <http://sine.ni.com/nips/cds/print/p/lang/en/nid/209856>
- [39] ROPSIM. (2012, 5/2/2012). Available: <http://www.camelot.dk/default.aspx>
- [40] Carmen. (2012). Available: [http://carmen.sourceforge.net/using\\_carmen.html](http://carmen.sourceforge.net/using_carmen.html)
- [41] Simbad. (2012, 5/2/2012). Available: <http://sourceforge.net/projects/simbad/>
- [42] N. B. Louis Hugues, "Simbad : an Autonomous Robot Simulation Package for Education and Research," presented at the Ginkgo-networks, Paris, France, 2007.
- [43] Dialogos. (2012, 5/2/2012). Available: <http://www.clt-st.de/produkte-losungen/dialogos/ubersicht/>
- [44] Wikipedia. (2012, 5/2/2012). *Mindstorms*. Available: [http://en.wikipedia.org/wiki/Lego\\_Mindstorms\\_NXT](http://en.wikipedia.org/wiki/Lego_Mindstorms_NXT)
- [45] MIT. (2006, 5/2/2012). Available: <http://ilk.media.mit.edu/projects.php?id=135>
- [46] LEGOSheets, 1994.
- [47] 000001. (5/2/2012). Available: <http://www.flickr.com/photos/brickjournal/3079324996/in/set-72157610670686416/>
- [48] Wikipedia. (2012, 5/2/2012). Available: <http://en.wikipedia.org/wiki/File:Nxt-brique.jpg>
- [49] MindStorms. (2011, 5/2/2012). Available: <http://mindstorms.lego.com/en-us/products/default.aspx>
- [50] P. Toolkit. (2012, 5/2/2012). Available: [http://upload.wikimedia.org/wikipedia/commons/0/0c/SENA\\_robot\\_navigation\\_demo\\_mlg2007.jpg](http://upload.wikimedia.org/wikipedia/commons/0/0c/SENA_robot_navigation_demo_mlg2007.jpg)
- [51] J. G. J.L. Blanco, J. and J.A. Fernández-Madriral, "Consistent observation grouping for generating metric-topological maps that improves robot localizatio," presented at the IEEE International Conference on Robotics and Automation (ICRA), 2006.
- [52] J. M. C. A. Harris, "Survey of popular robotics simulators, frameworks, and toolkits," presented at the IEEE Southeastcon, 2011.
- [53] J. H. Ryde, H., *Autonomous Robots*, 2010.
- [54] Z. a. R. Tuza, J. and Szederkényi, G, "Developing an integrated software environment for mobile robot navigation and control," presented at the International Conference on Indoor Positioning and Indoor Navigation, 2010.

- 
- [55] Papers. (2012, 5/2/2012). Available: <http://www.mrpt.org/Papers>
- [56] Wikipedia. (2012, 5/2/2012). *Metric Maps*. Available: [http://en.wikipedia.org/wiki/Mobile\\_Robot\\_Programming\\_Toolkit#Representation\\_of\\_metric\\_maps](http://en.wikipedia.org/wiki/Mobile_Robot_Programming_Toolkit#Representation_of_metric_maps)
- [57] Robotis. (2012, 5/2/2012). Available: <http://www.allonrobots.com/bioloid.html>
- [58] Azonano. (2012, 5/2/2012). Available: <http://www.azonano.com/news.aspx?NewsID=4498>
- [59] VexRobotics. (2010, 5/2/2012). Available: <http://www.vexrobotics.com/276-2231.html>
- [60] M. Corporation. (2011, 5/2/2012). *Products and Services*. Available: <http://msdn.microsoft.com/en-us/library/cc998476.aspx>
- [61] M. Corporation. (2011, 5/2/2012). *VPL Introduction*. Available: <http://msdn.microsoft.com/en-us/library/bb483088.aspx>
- [62] M. Corporation. (2011, 5/2/2012). *Simulation Overview*. Available: <http://msdn.microsoft.com/en-us/library/bb483076.aspx>
- [63] M. Corporation. (2011, 5/2/2012). *Robotics Common Overview*. Available: <http://msdn.microsoft.com/en-us/library/cc998481.aspx>
- [64] M. Corporation. (2011, 5/2/2012). *Basic Activities*. Available: <http://msdn.microsoft.com/en-us/library/bb466256.aspx>
- [65] M. Corporation. (2011, 5/2/2012). *LEGO Services Overview*. Available: <http://msdn.microsoft.com/en-us/library/bb870552.aspx>
- [66] M. Corporation. (2011, 5/2/2012). *LEGO Device Services*. Available: <http://msdn.microsoft.com/en-us/library/bb870566.aspx>
- [67] M. Corporation. (2011, 5/2/2012). *LEGO Motor Service*. Available: <http://msdn.microsoft.com/en-us/library/bb870564.aspx>
- [68] M. Corporation. (2011, 5/2/2012). *LEGO Drive Service*. Available: <http://msdn.microsoft.com/en-us/library/bb870567.aspx>
- [69] M. Corporation. (2011, 5/2/2012). *LEGO Light Sensor Service*. Available: <http://msdn.microsoft.com/en-us/library/bb870553.aspx>
- [70] M. Corporation. (2011, 5/2/2012). *LEGO Color Sensor Service*. Available: <http://msdn.microsoft.com/en-us/library/ff631052.aspx>
- [71] M. Corporation. (2011, 5/2/2012). *LEGO MINDSTORMS NXT Tutorials Overview*. Available: <http://msdn.microsoft.com/en-us/library/bb870550.aspx>

---

## ΠΑΡΑΡΤΗΜΑ

### Α. ΠΛΗΡΗΣ ΚΑΤΑΛΟΓΟΣ ΤΟΥ ΥΠΟΣΤΗΡΙΖΟΜΕΝΟΥ ΛΟΓΙΣΜΙΚΟΥ ΓΙΑ ΤΟ LEGO MINDSTORMS NXT 2.0

48

<u>Όνομα</u>	<u>Τύπος Γλώσσας</u>	<u>Σημειώσεις</u>
Actor-Lab	Τροποποιημένη γλώσσα τύπου flowchart	
Ada	Ada	Απαιτείται το nxtOSEK
Ada Interface to Mindstorms	Ada	
brickOS	C/C++	
Ch	C/C++	Έλεγχος των Mindstorms σε C/C++ μέσω αλληλεπίδρασης, χωρίς μεταγλώττιση
Enchanting	Drag and Drop όπως το NXT-G	Προγραμματισμός ρομπότ τραβώντας γραμμές ρουτινών
FLL NXT Navigation	NXT-G	
GCC	C/C++, Fortran, Java, Ada κλπ	
GNU Tool chain for h8300	C/C++	
jaraco.nxt	Python	Έλεγχος των NXT μέσω Bluetooth
LabVIEW	National Instruments LabVIEW visual programming language	Γλώσσα ανάπτυξης λογισμικού NXT
Lego.NET	Οτιδήποτε μπορεί να γίνει compile με CIL, δουλεύει άριστα με C#	Μετατρέπει CIL σε κώδικα μηχανής
Lego::NXT	Perl	Ομάδα modules για Perl
LegoNXTRemote	Objective-C	Πρόγραμμα ελέγχου από μακριά ενός NXT
leJOS	Java	Σύστημα βασισμένο σε Java
NXTGCC	Assembly, C, Eclipse, makefiles, Κλπ	Η πρώτη αλυσίδα εργαλείων βασισμένη σε GCC για το NXT
nxtOSEK	C/C++	

librex	C/C++	Βιβλιοθήκη GCC
Logitech SDK	Visual Basic, Visual C++	Μπορεί να συνδυαστεί με την βιβλιοθήκη ελέγχου του RCX
MicroWorlds EX Robotics Edition		Πρόγραμμα ελέγχου ενός NXT για μαθητές
NQC	NQC γλώσσα παρόμοια με C	
NXT++	C++	Επιτρέπει έλεγχο κατευθείαν από οποιοδήποτε πρόγραμμα C++
NXT_Python	Python	Πακέτο ελέγχου που χρησιμοποιεί την γλώσσα Python
NXT-Python	Python	Όπως το παραπάνω με περισσότερα χαρακτηριστικά
Lestat	C++	Έλεγχος οπουδήποτε προγράμματος C++ από Linux
OCaml Mindstorms	Objective Caml	Έλεγχος ρομπότ NXT μέσω Bluetooth/USB
Mindstorms SDK	Visual Basic, Visual C++, Mind Script, LASM	
OnScreen	εξατομικευμένη γλώσσα που προγραμματίζεται απευθείας στο RCX	
pbForth	Forth	
PBrickDev	Τροποποιημένη γλώσσα τύπου flowchart	
PRO-BOT	Ένας τύπος Visual Basic	
QuiteC	C	Βιβλιοθήκη GCC
RCX Code	Τροποποιημένη γλώσσα τύπου flowchart	Περιλαμβάνεται στις εμπορικές εκδόσεις του MINDSTORMS
ROBOLAB	Τροποποιημένη γλώσσα τύπου flowchart	Προγραμματιστικό περιβάλλον που προσφέρεται στις σχολικές εκδόσεις MINDSTORM

RoboRealm	Γλώσσα πολλών πλατφόρμων	
Robotc	C	
ROS	Βιβλιοθήκη βασισμένη στο Linux	
Robotics.NXT	Haskell	Υποστηρίζει άμεσες εντολές, μηνύματα κλπ
ruby-nxt	Ruby	Προσφέρει χαμηλού επιπέδου σύνδεση μέσω Bluetooth
RWTH – Mindstorms NXT Toolbox	MATLAB	Χειρισμός NXT μέσω MATLAB
SqLego	Squeak	
TclRCX	Tcl	
Terrapin Logo	LOGO	
TinySoar	Soar	Υλοποίηση της αρχιτεκτονικής τεχνητής νοημοσύνης SOAR για το RCX
TinyVM	Java	Πρόγονος της γλώσσας lejos
The Transterpreter	Occam	
TuxMinds	GUI για διάφορες γλώσσες	Σχεδόν κάθε είδους μικροελεγκτής μπορεί να υποστηριχθεί με κατάλληλη διαμόρφωση του XML αρχείου του
Gostai URBI for Lego Mindstorms NXT	URBI, C++, Java, MATLAB	Εύκολη στη χρήση παράλληλη scripting γλώσσα
Vision Command	RCX Code	Επίσημη γλώσσα προγραμματισμού της Lego Cam
XS	Lisp	
LegoLog	Prolog	Χρησιμοποίηση ενός NQC προγράμματος για ερμηνεία εντολών κώδικα PROLOG

Microsoft Visual Programming Language (VPL)	Γραφικό προγραμματιστικό περιβάλλον, βασισμένο στο .NET	Μέσω του RDS, χρησιμοποιεί το msrs πρόγραμμα για να στείλει και να δεχτεί μηνύματα ενός προγράμματος μέσω Bluetooth
DialogOS	Γραφικό προγραμματιστικό περιβάλλον	Συνδυάζει αναγνώριση φωνής και σύνθεση φωνής με τη ρομποτική
Processing	Java	Ανοικτή γλώσσα προγραμματισμού
Interactive C	Γλώσσα τύπου C	Γλώσσα ανεπτυγμένη από το διαγωνισμό MIT LEGO ROBOT DESIGN
pbLua	API για την γλώσσα Lua	Γραμμένη σε portable C