



ΤΕΧΝΟΛΟΓΙΚΟ ΕΚΠΑΙΔΕΥΤΙΚΟ ΊΔΡΥΜΑ ΚΡΗΤΗΣ

ΣΧΟΛΗ ΤΕΧΝΟΛΟΓΙΚΩΝ ΕΦΑΡΜΟΓΩΝ

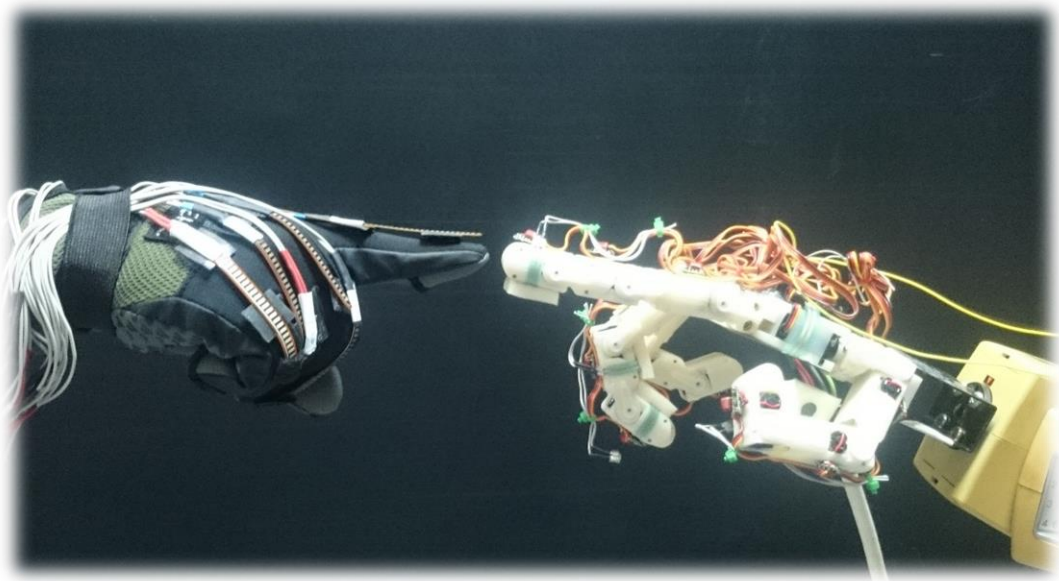
ΤΜΗΜΑ ΜΗΧΑΝΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ Τ.Ε.

Τηλεχειρισμός ανθρωπόμορφου ρομποτικού  
χεριού μέσω πρωτότυπης διάταξης αισθητήρων  
προσαρμοσμένων σε γάντι

---

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

Φυράι Κλάιντο 5856



ΕΠΙΒΛΕΠΩΝ:

Δρ. Φασουλάς Ιωάννης,

Επίκουρος καθηγητής Τ.Ε.Ι. Κρήτης

ΗΡΑΚΛΕΙΟ 2015



## **Περίληψη**

Η παρούσα πτυχιακή εργασία πραγματεύεται την σχεδίαση και κατασκευή ενός συστήματος συλλογής δεδομένων, προσαρμοσμένο σε γάντι, για τον τηλεχειρισμό ενός ρομποτικού χεριού 16 βαθμών ελευθερίας το οποίο πρόσφατα κατασκευάστηκε στο εργαστήριο Ρομποτικής και Αυτοματικής του Τ.Ε.Ι. Κρήτης. Αρχικά, γίνεται μία συνοπτική περιγραφή της κίνησης των δαχτύλων του ανθρώπινου χεριού, από την οποία προκύπτει ότι υπάρχει συσχέτιση στον τρόπο κίνησης των αρθρώσεων για κάθε δάχτυλο ξεχωριστά αλλά και μεταξύ γειτονικών δαχτύλων. Στην συνέχεια, παρουσιάζονται μαθηματικές σχέσεις που περιγράφουν τους περιορισμούς κίνησης στις αρθρώσεις των δαχτύλων, οι οποίες χρησιμοποιούνται στην υλοποίηση του συστήματος συλλογής δεδομένων που κατασκευάστηκε. Ακολούθως, παρουσιάζεται με λεπτομέρεια η σχεδίαση και η κατασκευή του συστήματος συλλογής δεδομένων η οποία βασίζεται στην χρήση εύκαμπτων αισθητήρων οι οποίοι τοποθετούνται με κατάλληλο τρόπο πάνω στο γάντι. Επίσης, αναλύεται ο τρόπος τηλεχειρισμού (ασύρματος ή ενσύρματος) του ρομποτικού χεριού από το παραπάνω σύστημα συλλογής δεδομένων και εξετάζεται η ορθότητα του συστήματος μέσα από πειράματα. Τέλος, αναπτύχθηκε ένα μοντέλο απεικόνισης των δεδομένων καταγραφής του γαντιού το οποίο χρησιμοποιήθηκε πριν την τελική σύνδεση του ρομποτικού χεριού με το γάντι.

## ***Abstract***

Purpose of this thesis project is the development of data acquisition system, placed on a glove, for the manipulation of the humanoid robotic hand with 16 DOF (Degrees of Freedom), which was recently constructed at the laboratory of Robotics & Automation of T.E.I. of Crete. Firstly, a brief description of the human hand fingers motion is presented, which shows that there is relation among the motion of the human finger joints for each finger, but also relation among fingers (e.g. fore, middle, ring and little finger). Then, mathematical relations are being presented describing the human hand finger constrains, which are used for the implementation of the data acquisition system that was constructed. Afterwards, a detailed description of the design and construction of the data acquisition system is presented which is based on flex sensors that are placed in a proper array to the glove. Moreover, the manipulation (wireless or wired) of the robotic hand from the data glove with the above mentioned data acquisition system is analyzed furthermore, the correctness of the system is examined with experiments. Last but not least, a virtual reality model was developed for the simulation of the data glove before it was integrated with the robotic hand.

# Περιεχόμενα

<i>Περίληψη</i> .....	i
<i>Abstract</i> .....	ii
<i>Περιεχόμενα</i> .....	iii
<i>Περιεχόμενα εικόνων</i> .....	vi
<b>1. Εισαγωγή</b> .....	<b>1</b>
1.1. Εισαγωγή.....	1
1.2. Κίνητρα και στόχοι της πτυχιακής .....	1
1.3. Επισκόπηση της πτυχιακής .....	2
<b>2. Καταγραφή των κινήσεων του ανθρώπινου χεριού με γάντια συλλογής δεδομένων .</b>	<b>4</b>
2.1. Εισαγωγή.....	4
2.2. Το ανθρώπινο χέρι.....	4
2.2.1. Περιγραφή των βασικών κινήσεων του ανθρώπινου χεριού .....	5
2.2.2. Οι αρθρώσεις και οι φάλαγγες του ανθρώπινου χεριού .....	6
2.3. Τα γάντια συλλογής δεδομένων και οι εφαρμογές τους .....	8
2.3.1. Το γάντι συλλογής δεδομένων <i>DG5 – Vhand Glove 3.0</i> .....	9
2.3.2. Το γάντι συλλογής δεδομένων <i>CyberGlove II</i> .....	9
2.3.3. Το γάντι συλλογής δεδομένων <i>5DT Data Glove 5 Ultra</i> .....	10
2.3.4. Το γάντι συλλογής δεδομένων <i>AcceleGlove</i> .....	11
2.3.5. Το γάντι συλλογής δεδομένων <i>Virtual Reality P5 Glove</i> .....	11
2.3.6. Το γάντι συλλογής δεδομένων <i>TALOS’ Glove</i> .....	12
<b>3. Συσχέτιση της κίνησης των αρθρώσεων του ανθρώπινου χεριού</b> .....	<b>13</b>
3.1. Εισαγωγή.....	13
3.2. Περιορισμοί που υπάρχουν στις κινήσεις των δακτύλων του ανθρώπινου χεριού .	
.....	13
3.2.1. Ένδο – δακτύλιοι περιορισμοί.....	14
3.2.2. Έσω – δακτύλιοι περιορισμοί.....	17

3.3.	Συμπεράσματα.....	20
<b>4.</b>	<b>Σχεδίαση συστήματος για τη καταγραφή της κίνησης των ανθρώπινων δαχτύλων..</b>	<b>21</b>
4.1.	Εισαγωγή.....	21
4.2.	Σύστημα καταγραφής κίνησης .....	21
4.3.	Ο μικροελεγκτής Arduino.....	23
4.4.	Αισθητήρας κάμψης .....	24
4.5.	Βαθμονόμηση αισθητήρων κάμψης.....	25
4.6.	Συμπεράσματα.....	35
<b>5.</b>	<b>Τηλεχειρισμός ανθρωπόμορφου ρομποτικού χεριού από το γάντι συλλογής δεδομένων .....</b>	<b>36</b>
5.1.	Εισαγωγή.....	36
5.2.	Το ανθρωπόμορφο ρομποτικό χέρι TALOS' HAND.....	36
5.3.	Κίνηση των αρθρώσεων του ρομποτικού χεριού TALOS'.....	38
5.4.	Ενσύρματη διασύνδεση .....	41
5.4.1.	Διασύνδεση με ελεγκτή SSC-32.....	42
5.5.	Ασύρματη διασύνδεση .....	44
5.5.1.	Ασύρματο δίκτυο.....	45
5.5.2.	Υλοποίηση ασύρματου δικτύου μέσω του XBee .....	46
5.5.3.	Διασύνδεση XBee με μικροελεγκτή .....	53
5.6.	Πειραματικά αποτελέσματα .....	58
5.7.	Συμπεράσματα.....	65
<b>6.</b>	<b>Εικονική αναπαράσταση δεδομένων που λαμβάνονται από το γάντι συλλογής δεδομένων .....</b>	<b>66</b>
6.1.	Εισαγωγή.....	66
6.2.	Απεικόνιση της κίνησης των δακτύλων με Η/Υ .....	66
6.2.1.	Εικονικά μοντέλα.....	66
6.2.2.	Σχεδίαση στο λογισμικό VRML.....	67
6.3.	Σειριακή επικοινωνία μέσω Matlab – Simulink.....	77

6.3.1.	Ενσύρματη διασύνδεση μικροελεγκτή με Η/Υ και Simulink.....	77
6.3.2.	Ασύρματη διασύνδεση με Η/Υ.....	82
6.4.	Πειραματικά αποτελέσματα.....	83
6.5.	Συμπεράσματα.....	86
<b>7.</b>	<b>Συμπεράσματα και μελλοντική έρευνα.....</b>	<b>87</b>
7.1.	Ανασκόπηση και συμπεράσματα της πτυχιακής εργασία.....	87
7.2.	Μελλοντική έρευνα .....	88
	<b>Βιβλιογραφία .....</b>	<b>90</b>
	<b>Παράρτημα Α Πλακέτα τυπωμένου κυκλώματος .....</b>	<b>94</b>
A.1.	Χαλκογραφία πλακέτας τυπωμένου κυκλώματος .....	94

## **Περιεχόμενα εικόνων**

Εικόνα 2.1. Ονομασίες δακτύλων ανθρώπινου χεριού. ....	5
Εικόνα 2.2. Οι κινήσεις των δακτύλων και τα επίπεδα του χεριού. ....	6
Εικόνα 2.3. Οι κινήσεις των δακτύλων στα δύο επίπεδα. ....	6
Εικόνα 2.4. Οι φάλαγγες του ανθρώπινου χεριού. ....	7
Εικόνα 2.5. Οι αρθρώσεις του ανθρώπινου χεριού. ....	8
Εικόνα 2.6. Το γάντι συλλογής δεδομένων <i>DG5 VHand 3.0</i> . ....	9
Εικόνα 2.7. Το γάντι συλλογής δεδομένων <i>CyberGlove II</i> . ....	10
Εικόνα 2.8. Γάντι συλλογής δεδομένων <i>5DT Data Glove 5 Ultra</i> . ....	10
Εικόνα 2.9. Γάντι συλλογής δεδομένων <i>AcceleGlove</i> . ....	11
Εικόνα 2.10. Γάντι συλλογής δεδομένων <i>Virtual Reality P5 Glove</i> . ....	12
Εικόνα 2.11. Γάντι συλλογής δεδομένων <i>TALOS' Glove</i> . ....	12
Εικόνα 3.1. Απεικόνιση της υπερέκτασης του δείκτη. ....	14
Εικόνα 3.2. Σχεδιάγραμμα για το δάχτυλο δείκτη του ανθρώπινου χεριού, όπου απεικονίζει τις αρθρώσεις του. ....	15
Εικόνα 3.3. Σχεδιάγραμμα για τον αντίχειρα του ανθρώπινου χεριού, όπου απεικονίζει τις αρθρώσεις του. ....	16
Εικόνα 3.4. Αναπαράσταση κίνησης για τους έσω – δακτύλιους περιορισμούς. ....	18
Εικόνα 3.5. Αναπαράσταση κίνησης για την ζεύξη των κινήσεων του μέσου και μικρού δαχτύλου. ....	18
Εικόνα 3.6. Απεικόνιση της κίνησης των δακτύλων. ....	19
Εικόνα 3.7. Απεικόνιση για την χαρακτηριστική κίνηση που συνδέει την περιστροφή του δείκτη με τον παράμεσο. ....	19
Εικόνα 4.1. Απεικόνιση της διάταξης των αισθητήρων πάνω στο γάντι. ....	22
Εικόνα 4.2. Αισθητήρας κάμψης ( <i>Flex Sensor</i> ). ....	23
Εικόνα 4.3. Κάτοψη μικροελεγκτή <i>Arduino Mega 2560</i> . ....	24



Εικόνα 4.4. Εκτυπωμένες επιφάνειές από τον τρισδιάστατο εκτυπωτή με ακτίνα καμπυλότητας 15mm.....	25
Εικόνα 4.5. Εκτυπωμένες επιφάνειές, 15 mm. ....	26
Εικόνα 4.6. Κύκλωμα καταγραφής της αντίστασης του αισθητήρα κάμψης. ....	26
Εικόνα 4.7. Τοποθέτηση του αισθητήρα σε διαφορετικά σημεία πάνω στην επιφάνεια. ....	27
Εικόνα 4.8. Αισθητήρας κάμψης διακρίνοντας τις χαρακτηριστικές γραμμές.....	27
Εικόνα 4.9. Κατάλληλο κύκλωμα συλλογής τάσης από τον αισθητήρα κάμψης. ....	29
Εικόνα 4.10. Τελεστικός ενισχυτής LM358P Dual Operational Amplifier. ....	30
Εικόνα 4.11. Σχηματικό κύκλωμα καταγραφής δεδομένων με τελεστικό ενισχυτή (μη αναστρέφουσα συνδεσμολογία) για ενίσχυση του σήματος.....	31
Εικόνα 4.12. Κύκλωμα συλλογής δεδομένων μέσω του Arduino.....	31
Εικόνα 4.13. Πλακέτα ενίσχυσης τάσης εξόδου αισθητήρων (στο τέλος του παραρτήματος Α απεικονίζονται τα σχέδια για την πλακέτα).....	32
Εικόνα 5.1. Οι σύνδεσμοι του ρομποτικού χεριού. ....	37
Εικόνα 5.2. Οι αρθρώσεις του ρομποτικού χεριού.....	37
Εικόνα 5.3. Δυνατότητες ανθρωπόμορφου ρομποτικού χεριού. ....	37
Εικόνα 5.4. Σερβοελεγκτής SSC-32.....	38
Εικόνα 5.5. Διάγραμμα ροής συστήματος ελέγχου. ....	41
Εικόνα 5.6. Σχηματική αναπαράσταση σύνδεσης ελεγκτή SSC - 32 με Arduino. ....	44
Εικόνα 5.7. Διάγραμμα ροής ασύρματης επικοινωνίας συλλογής δεδομένων και αξιοποίησή τους.....	45
Εικόνα 5.8. Σχηματική αναπαράσταση ασύρματου δικτύου.....	46
Εικόνα 5.9. Πλακέτα σύνδεσης XBee Explorer USB για σύνδεση του XBee με Η/Υ .....	46
Εικόνα 5.10. Παράδειγμα σύνδεσης του XBee Explorer Dongle με τον υπολογιστή. ....	47
Εικόνα 5.11. Πλακέτα σύνδεσης UartS Bee V4 για σύνδεση του XBee με Η/Υ . ....	47
Εικόνα 5.12. Διαχείριση Συσκευών. ....	48
Εικόνα 5.13. Λογισμικό XCTU.....	49
Εικόνα 5.14. Λογισμικό XCTU εντολή "Discover devices".....	49

Εικόνα 5.15. Αναγνώριση της σύνδεσης των XBee από το λογισμικό XCTU. ....	50
Εικόνα 5.16. Παράθυρο ρύθμισης παραμέτρων σειριακής επικοινωνίας του XBee με Η/Υ..	50
Εικόνα 5.17. Παράθυρο εμφανίσεις συσκευών σύνδεσης έπειτα από την ολοκλήρωση των ρυθμίσεων.....	51
Εικόνα 5.18. Καταχωρημένη συσκευή στο λογισμικό XCTU.....	51
Εικόνα 5.19. Παράμετροι πλακέτας XBee.....	52
Εικόνα 5.20. Πλακέτα σύνδεσης του XBee με τον Arduino. ....	53
Εικόνα 5.21. Κύκλωμα καταγραφής δεδομένων ασύρματα από τον Arduino I και XBee I (Στην εικόνα το XBee δεν απεικονίζεται να είναι συνδεδεμένο στην πλακέτα σύνδεσης, αλλά πρέπει να συνδεθεί κανονικά. Αυτό ισχύει και για της υπόλοιπες εικόνες).....	54
Εικόνα 5.22. Συνδεσμολογία αποδέκτη για ασύρματη καταγραφή δεδομένων. ....	54
Εικόνα 5.23. Χειρονομία ανοιχτής παλάμης με το ρομποτικό χέρι και το γάντι συλλογής δεδομένων.....	58
Εικόνα 5.24. Χειρονομία κλειστή παλάμη με το ρομποτικό χέρι και το γάντι συλλογής δεδομένων.....	59
Εικόνα 5.25. Χειρονομία point με το ρομποτικό χέρι και το γάντι συλλογής δεδομένων. ....	59
Εικόνα 5.26. Χειρονομία OK με το ρομποτικό χέρι και το γάντι συλλογής δεδομένων. ....	60
Εικόνα 5.27. Χειρονομία like με το ρομποτικό χέρι και το γάντι συλλογής δεδομένων. ....	60
Εικόνα 5.28. Χειρονομία άξονες X-Y-Z με το ρομποτικό χέρι και το γάντι συλλογής δεδομένων. ....	61
Εικόνα 5.29. Χειρονομία επαφή αντίχειρα - παράμεσου με το ρομποτικό χέρι και το γάντι συλλογής δεδομένων.....	61
Εικόνα 5.30. Ευσταθής λαβή χάρτινου κουτιού με το ρομποτικό χέρι και το γάντι συλλογής δεδομένων.....	63
Εικόνα 5.31. Ευσταθής λαβή κάρτας με το ρομποτικό χέρι και το γάντι συλλογής δεδομένων. ....	63
Εικόνα 5.32. Ευσταθής λαβή ποτηριού με το ρομποτικό χέρι και το γάντι συλλογής δεδομένων.....	64

Εικόνα 5.33. Ευσταθής λαβή εργαλείου με το ρομποτικό χέρι και το γάντι συλλογής δεδομένων.....	64
Εικόνα 6.1. Βιβλιοθήκη του VRML Builder.....	67
Εικόνα 6.2. Παράθυρο τροποποίησης του VRSIGNALExpander.....	68
Εικόνα 6.3. Παράθυρο διαχείρισης του VRSink.....	69
Εικόνα 6.4. Παράθυρο του VRML Builder.....	69
Εικόνα 6.5. Παράθυρο VRML Builder μετά την εισαγωγή του κυλίνδρου.....	70
Εικόνα 6.6. Παράθυρο του VRML με τα εισαγόμενα στοιχεία.....	71
Εικόνα 6.7. Παράθυρο του VRML Builder με σχεδιασμένο τον δείκτη του χεριού στον τρισδιάστατο χώρο.....	72
Εικόνα 6.8. Σχέση σύνδεσης των στοιχείων.....	73
Εικόνα 6.9. Λίστα με τα στοιχεία της άρθρωσης 2 στον εικονικού κόσμου.....	74
Εικόνα 6.10. Λίστα με τα στοιχεία της άρθρωσης 2 στον εικονικού κόσμου.....	74
Εικόνα 6.11. Ανθρώπινο χέρι σχεδιασμένο στο λογισμικό VRML.....	75
Εικόνα 6.12. Παράθυρο VRSink με φορτωμένο αρχείο .wrl.....	76
Εικόνα 6.13. Εντολή του VRSink με επεξεργάσιμες ιδιότητες.....	76
Εικόνα 6.14. Διάγραμμα ροής συστήματος ελέγχου.....	77
Εικόνα 6.15. Βιβλιοθήκη Instrument Control Toolbox.....	79
Εικόνα 6.16. Παράθυρο επεξεργασίας για την εντολή Query Instrument.....	79
Εικόνα 6.17. Παράθυρο της εντολής “Query Instrument” καρτέλα “Query” παράμετροι εισαγωγής δεδομένων.....	81
Εικόνα 6.18. Προγραμματισμός μέσω Simulink για την εισαγωγή των δεδομένων από το Arduino και διαχείρισης της εικονικής απεικόνισης του ανθρώπινου χεριού.....	81
Εικόνα 6.19. Σχηματική αναπαράσταση ασύρματου δικτύου με Η/Υ.....	82
Εικόνα 6.20. Διάγραμμα ροής ασύρματης επικοινωνίας με Η/Υ και διαχείρισης της εικονικής αναπαράστασης.....	82
Εικόνα 6.21. Χειρονομία ανοιχτής παλάμης με το ρομποτικό χέρι και το γάντι συλλογής δεδομένων όπως απεικονίζεται στην εικονική αναπαράσταση.....	84

Εικόνα 6.22. Χειρονομία κλειστή παλάμη με το ρομποτικό χέρι και το γάντι συλλογής δεδομένων όπως απεικονίζεται στην εικονική αναπαράσταση.....	84
Εικόνα 6.23. Χειρονομία point με το ρομποτικό χέρι και το γάντι συλλογής δεδομένων όπως απεικονίζεται στην εικονική αναπαράσταση. ....	85
Εικόνα 6.24. Χειρονομία OK με το ρομποτικό χέρι και το γάντι συλλογής δεδομένων όπως απεικονίζεται στην εικονική αναπαράσταση. ....	85
Εικόνα 6.25. Χειρονομία Like με το ρομποτικό χέρι και το γάντι συλλογής δεδομένων όπως απεικονίζεται στην εικονική αναπαράσταση. ....	86
Εικόνα 0.1. Κύκλωμα καταγραφής δεδομένων για δύο αισθητήρες. ....	95
Εικόνα 0.2. Πρότυπο σχέδιο PCB πλακέτας για χαλκογραφία. ....	95
Εικόνα 0.3. Χαρακτηριστικό σχέδιο για χαλκογραφία σε πλακέτα PCB. ....	96
Εικόνα 0.4. Εκτυπωμένο σχέδιο τοποθετημένο στην μηχανή έκθεσης φωτός UV. ....	96
Εικόνα 0.5. Πλακέτα εκτυπωμένου κυκλώματος τοποθετημένο σε μίγμα υδροξείδιο του νατρίου. ....	97
Εικόνα 0.6. Μηχάνημα που περιέχει χλωριούχο σίδηρο για την αφαίρεση του ανεπιθύμητου χαλκού από την πλακέτα.....	97
Εικόνα 0.7. Πλακέτα PCB στην τελική φάση. ....	98
Εικόνα 0.8. Σχηματική αναπαράσταση μη αναστρέφουσας συνδεσμολογίας για την ενίσχυση τάσης εξόδου για 11 αισθητήρες (FLi: αισθητήρες, i=1-11).....	98
Εικόνα 0.9. Σχηματική αναπαράσταση πάνω επιφάνειας πλακέτας.....	99
Εικόνα 0.10. Εικόνα 0.9. Σχηματική αναπαράσταση κάτω επιφάνειας πλακέτας. ....	99
Εικόνα 0.11. Σχηματική αναπαράσταση για την τοποθέτηση των στοιχείων. ....	100

## 1. Εισαγωγή

### 1.1. Εισαγωγή

*Σε αυτό το κεφάλαιο αναφέρονται τα κίνητρα που οδήγησαν στην υλοποίηση της παρούσας πτυχιακής εργασίας. Παρουσιάζονται επίσης οι στόχοι και η δομή της πτυχιακής.*

### 1.2. Κίνητρα και στόχοι της πτυχιακής

Περίπου πριν από τρεις δεκαετίες οι επιστήμονες μηχανικοί της ρομποτικής υπήρξαν αισιόδοξοι προβλέποντας μια ραγδαία εξέλιξη των ρομπότ, σε τέτοιο βαθμό ώστε να είναι ικανά να πραγματοποιούν αντίστοιχες εργασίες με αυτές που δύναται να υλοποιήσει ο άνθρωπος. Στην πραγματικότητα αν και έχει κατασκευαστεί μέχρι σήμερα ένας μεγάλος αριθμός από ρομποτικά συστήματα μεγάλης ακρίβειας, χρησιμοποιούνται κατά κύριο λόγο στην βιομηχανία σε απλές επαναληπτικές διαδικασίες. Έτσι, ακόμα και σήμερα τα ρομπότ χαρακτηρίζονται από αδεξιότητα όταν πρόκειται να χρησιμοποιηθούν σε εργασίες χειρισμού αντικειμένων που πραγματοποιεί ο άνθρωπος στην καθημερινή του ζωή.

Το ανθρώπινο χέρι χαρακτηρίζεται από την ικανότητα να χειρίζεται αντικείμενα διαφόρων μεγεθών και σχημάτων, χρησιμοποιώντας κυρίως την αίσθηση της αφής και της όρασης. Η συνεργασία πολλών ρομποτικών συστημάτων, ως ρομποτικά δάχτυλα, δίνει την δυνατότητα

να προσεγγιστεί η λειτουργικότητα των ανθρώπινων δαχτύλων για τον χειρισμό αντικειμένων. Τα τελευταία χρόνια κατασκευάστηκαν αρκετά γάντια συλλογής δεδομένων σε διάφορα ερευνητικά ιδρύματα και πανεπιστήμια, για εφαρμογές κατά κύριο λόγο σε συστήματα εικονικής πραγματικότητας (*virtual reality*). Έτσι, τα περισσότερα από αυτά δεν έχουν βρει εφαρμογή σε πραγματικά συστήματα αλλά μόνο σε εικονικά μοντέλα Η/Υ. Ο λόγος είναι ότι η αξιοπιστία και η ακρίβεια τους δεν έχουν αναπτυχθεί σε ικανοποιητικό επίπεδο ώστε, οι χειριστές να είναι σίγουροι για τα επιθυμητά αποτελέσματα. Έτσι, σε διεθνή βιβλιογραφία οι περισσότερες εργασίες αναφέρονται στην εκπαίδευση σε εικονικά μοντέλα σε τομείς όπως, η ιατρική και βιομηχανία.

Στόχος της παρούσας πτυχιακής εργασίας είναι ο σχεδιασμός και η κατασκευή ενός συστήματος καταγραφής της κίνησης των ανθρώπινων δαχτύλων. Το σύστημα θα αποτελείται από αισθητήρες κάμψης προσαρμοσμένους πάνω σε γάντι. Στην συνέχεια, το γάντι θα χρησιμοποιηθεί για τον τηλεχειρισμό ανθρωπόμορφου ρομποτικού χεριού αλλά και την απεικόνιση της κίνησης σε εικονικό περιβάλλον.

### **1.3. Επισκόπηση της πτυχιακής**

Η παρούσα πτυχιακή εργασία είναι χωρισμένη σε 7 κεφάλαια συμπεριλαμβανομένου και του παρόντος, που αποτελεί την εισαγωγή της πτυχιακής.

Στο **κεφάλαιο 2** γίνεται κινηματική περιγραφή για το ανθρώπινο χέρι και τα μέρη που το απαρτίζουν. Στην συνέχεια, παρουσιάζονται διάφορα γάντια συλλογής δεδομένων που υπάρχουν στο εμπόριο και που έχουν βρει εφαρμογή σε διάφορα παιδιά όπως η κινηματική ανάλυση του ανθρώπινου χεριού. Τέλος, παρουσιάζεται το γάντι συλλογής δεδομένων που αναλύετε στην παρούσα πτυχιακή εργασία.

Στο **κεφάλαιο 3** αναλύονται οι περιορισμοί κίνησης των ανθρώπινων δαχτύλων. Πιο συγκεκριμένα, τα ανθρώπινα δάχτυλα διαθέτουν περιορισμούς στην περιστροφή των αρθρώσεων τους. Σημαντική διαπίστωση στο παρών κεφάλαιο είναι ο καθορισμός περιστροφής της τελευταίας άρθρωσης (στην άκρη του δαχτύλου) με βάση την κίνηση των υπόλοιπων αρθρώσεων (π.χ. του δείκτη, του μέσου, του παράμεσου ή του μικρού), για την χρήση λιγότερων αισθητήρων.

Στο **κεφάλαιο 4** μελετάται το σύστημα καταγραφής της κίνησης των ανθρώπινων δαχτύλων. Στην αρχή, αναφέρεται ο μικροελεγκτής που χρησιμοποιείτε για την καταγραφή των δεδομένων. Περεταίρω, περιγράφεται και αιτιολογείτε η χρήση του γαντιού και του

αισθητήρα κάμψης (*flex sensor*). Εν κατακλείδι, μελετάτε ο αισθητήρας για την βέλτιστη καταγραφή των δεδομένων, αλλά και τον τρόπο τοποθέτησης του στο γάντι.

Στο **κεφάλαιο 5** παρουσιάζεται το ανθρωπόμορφο ρομποτικό χέρι που κατασκευάστηκε στο Τ.Ε.Ι. Κρήτης στο εργαστήριο Αυτοματικής και Ρομποτικής. Αρχικά, αναφέρεται ο τρόπος ενσύρματου χειρισμού του ρομποτικού χεριού. Στην συνέχεια, Παρουσιάζεται ο τρόπος ασύρματης επικοινωνίας μεταξύ των μικροελεγκτών και ο τηλεχειρισμός του ρομποτικού χεριού. Για την καλύτερη κατανόηση της πτυχιακής εργασίας, στο τέλος του κεφαλαίου παρουσιάζονται τα πειράματα από το σύστημα που έγιναν στο εργαστήριο Ρομποτικής και Αυτοματικής του Τ.Ε.Ι. Κρήτης.

Στο **κεφάλαιο 6** αναλύεται ο τρόπος διασύνδεσης του κυκλώματος συλλογής δεδομένων με Η/Υ. Αρχικά, γίνεται αναλυτική αναφορά για την σχεδίαση εικονικού ανθρώπινου δαχτύλου στο λογισμικό *VRML* του *MATLAB*. Στην συνέχεια, περιγράφεται αναλυτικά η διασύνδεση του μικροελεγκτή με το λογισμικό *MATLAB*. Ακολούθως, αναφέρεται η διαδικασία εικονικής αναπαράστασης από τα δεδομένα που συλλέγονται από το γάντι συλλογής δεδομένων και τέλος, παρουσιάζονται τα πειραματικά αποτελέσματα.

Στο τελευταίο κεφάλαιο (**κεφάλαιο 7**) αναφέρονται τα συμπεράσματα και οι δυνατότητες επέκτασης της παρούσας πτυχιακής εργασίας.

Στο **παράρτημα Α** αναφέρεται η διαδικασία χαλκογραφικής πλακέτας τυπωμένου κυκλώματος όπου, μέσω την χρήση μπορέσαμε να μειώσουμε τον όγκο της κατασκευής.

## **2. Καταγραφή των κινήσεων του ανθρώπινου χεριού με γάντια συλλογής δεδομένων**

### **2.1. Εισαγωγή**

*Στο κεφάλαιο αυτό αναλύεται το ανθρώπινο χέρι όσον αφορά τις αρθρώσεις και τις φάλαγγες που το αποτελούν. Στη συνέχεια, περιγράφονται τα γάντια συλλογής δεδομένων και αναφέρονται διάφορα από αυτά που έχουν κατασκευαστεί ή κυκλοφορούν στο εμπόριο, τα οποία έχουν βρει εφαρμογές σε διάφορες μελέτες. Τέλος, γίνεται σύντομη περιγραφή για το γάντι συλλογής δεδομένων που αναφέρεται σε αυτήν την πτυχιακή εργασία.*

### **2.2. Το ανθρώπινο χέρι**

Το χέρι είναι ένα από τα πιο σημαντικά όργανα του ανθρώπινου σώματος καθώς καθορίζει την αλληλεπίδραση μας με τον υλικό κόσμο. Με τέσσερα εύκαμπτα δάχτυλα και τον αντίχειρα, κάνει το 26% των δυνατοτήτων κίνησης του ανθρώπινου σώματος. Η λειτουργική μοναδικότητα του ανθρώπινου χεριού προκύπτει από την πολυπλοκότητα της γεωμετρικής διάταξης των συνδέσμων, αρθρώσεων μυών, που λειτουργούν ως μια συντονισμένη οντότητα για να παράγουν ένα ευρύ φάσμα ευέλικτων κινήσεων. Η περίπλοκη κινηματική



και δυναμική του μυοσκελετικού συστήματος του ανθρώπινου χεριού επιτρέπει την εκπληκτική λειτουργική ικανότητα του. Μάλιστα οι επιστήμονες αναφέρουν ότι «το ανθρώπινο χέρι αποτελεί την προέκταση του μυαλού», θέλοντας να δείξουν με αυτόν τον τρόπο το μέγεθος των ικανοτήτων του χεριού μας. Τα δάχτυλα και ο αντίχειρας μπορούν να λυγίσουν κάνοντας διάφορες λαβές έτσι μπορούμε να κρατήσουμε μια καρφίτσα ή να περιστρέψουμε τα δάχτυλα μας ξεβιδώνοντας το καπάκι από ένα μπουκάλι ή ακόμα και να κρατήσουμε δυνατά ένα σφυρί ώστε να καρφώσουμε ένα καρφί στον τοίχο. Στις άκρες των δαχτύλων υπάρχουν πυκνές περιοχές με νευρικές απολήξεις οι οποίες καθιστούν το χέρι την πλουσιότερη πηγή αισθητήριας ανάδρασης του ανθρώπινου σώματος. Τέλος, ο άνθρωπος χρησιμοποιεί επίσης τα χέρια του, σχεδόν καθημερινά, ως μέσο επικοινωνίας και έκφρασης. Αξίζει να αναφερθεί ότι οι χειρονομίες αποτελούν αναπόσπαστο μέρος της γλώσσας του ανθρώπινου σώματος και επιτρέπουν την ενίσχυση του προφορικού μας λόγου [1].

### 2.2.1. Περιγραφή των βασικών κινήσεων του ανθρώπινου χεριού

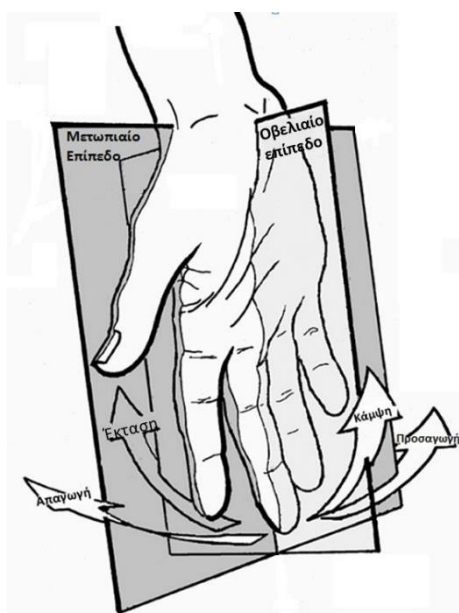
Το ανθρώπινο χέρι αποτελείται από πέντε δάχτυλα, τον αντίχειρα, τον δείκτη, τον μέσο, τον παράμεσο και τον μικρό (Εικόνα 2.1). Με δεδομένη όμως την ανατομική πολυπλοκότητα του ανθρώπινου χεριού, είναι απαραίτητο να οριστούν και να διευρυνθούν, σε αυτό το κεφάλαιο, μερικοί από τους συνήθεις όρους για την διευκόλυνση της περιγραφής του.



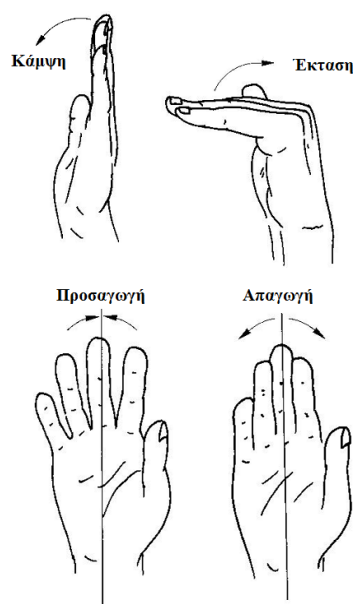
Εικόνα 2.1. Ονομασίες δακτύλων ανθρώπινου χεριού.

Για την περιγραφή της κίνησης των δαχτύλων χρησιμοποιούμε δύο επίπεδα αναφοράς, το μετωπιαίο και το οβελιαίο ή μεσαίο επίπεδο, ως προς τα οποία κινούνται τα δάχτυλα του (Εικόνα 2.2, Εικόνα 2.3). Το μετωπιαίο επίπεδο είναι αυτό που δημιουργείται όταν τα

δάχτυλα είναι σε πλήρη έκταση παράλληλα με την παλάμη του χεριού. Το οβελιαίο ή μεσαίο επίπεδο βρίσκεται κάθετα προς την παλάμη του χεριού, δηλαδή κάθετα στο μετωπιαίο επίπεδο. Η κάμψη και η έκταση για το χέρι χρησιμοποιεί το μετωπιαίο επίπεδο ως σημείο αναφοράς. Η κάμψη αναφέρεται στο λύγισμα των αρθρώσεων του δακτύλου, με σκοπό να κλείσει το χέρι, σε αυτή την περίπτωση απομακρύνεται από το μετωπιαίο επίπεδο. Η έκταση στο χέρι, είναι η κίνηση κατά την οποία τα δάχτυλα προσεγγίζουν το μετωπιαίο επίπεδο για τον σχηματισμό μιας επίπεδης - ανοιχτής παλάμης. Προσαγωγή είναι η κίνηση που φέρνει το δάχτυλο πιο κοντά στο μεσαίο/οβελιαίο επίπεδο. Η απαγωγή αναφέρεται στο τράβηγμα του δακτύλου μακριά από το μεσαίο/οβελιαίο επίπεδο [1].



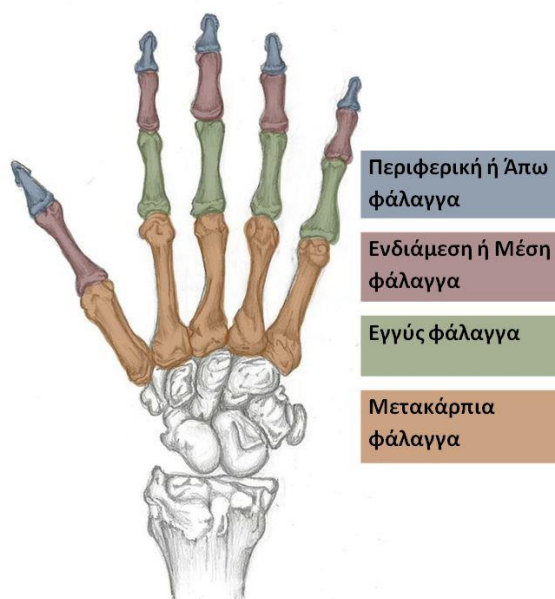
Εικόνα 2.2. Οι κινήσεις των δακτύλων και τα επίπεδα του χεριού.



Εικόνα 2.3. Οι κινήσεις των δακτύλων στα δύο επίπεδα.

### 2.2.2. Οι αρθρώσεις και οι φάλαγγες του ανθρώπινου χεριού

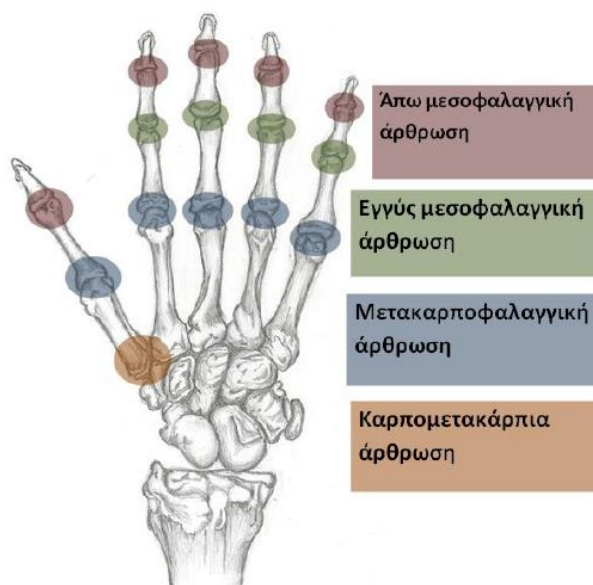
Το ανθρώπινο χέρι αποτελείται από φάλαγγες και αρθρώσεις. Οι φάλαγγες είναι παρένθετα τμήματα κοκάλων που αποτελούν (το άθροισμα τους) το κάθε δάχτυλο και χωρίζονται στην εγγύς φάλαγγα, την ενδιάμεση ή μέση φάλαγγα και την περιφερική ή άπω φάλαγγα. Ο αντίχειρας έχει τις δικές του ονομασίες για τις φάλαγγες που το απαρτίζουν οι οποίες είναι, η μετακάρπια φάλαγγα, η εγγύς φάλαγγα και η περιφερική ή άπω φάλαγγα. (Εικόνα 2.4)



Εικόνα 2.4. Οι φάλαγγες του ανθρώπινου χεριού.

Οι αρθρώσεις είναι αυτές που συνδέουν τις φάλαγγες μεταξύ τους και χωρίζονται στις μεσοφαλαγγικές αρθρώσεις, τις μετακαρποφαλαγγικές (ΜΚΦ) και την πηχεοκαρπική άρθρωση αυτή δηλαδή του καρπού. Οι μεσοφαλαγγικές αρθρώσεις έχουν ένα βαθμό ελευθερίας (κάμψη/επέκταση) καθώς επιτρέπουν στα δάχτυλα να κινηθούν προς μια κατεύθυνση. Οι μετακαρποφαλαγγικές είναι οι αρθρώσεις αμέσως μετά την παλάμη και έχουν δύο βαθμούς ελευθερίας (κάμψη/επέκταση και απαγωγή/συναγωγή). Η πηχεοκαρπική άρθρωση έχει τρεις βαθμούς ελευθερίας δίνοντας στο χέρι μεγάλη ευελιξία. Έτσι λοιπόν τα δάχτυλα του χεριού όπως ειπώθηκε και προηγουμένως, χάρις στις αρθρώσεις, είναι ικανά να κάνουν προσαγωγή και απαγωγή ως προς το οβελιαίο επίπεδο, έκταση και κάμψη ως προς το μετωπιαίο επίπεδο. (Εικόνα 2.3)

Το κάθε δάχτυλο έχει δική του ονομασία για την κάθε μία από τις τρεις αρθρώσεις. Ο αντίχειρας λόγω του ότι αποτελεί τον κύριο μοχλό για την δράση των υπόλοιπων δαχτύλων και όλου του χεριού συνολικά έχει ξεχωριστές ονομασίες για την κάθε άρθρωση του οι οποίες είναι (Εικόνα 2.5), η καρπομετακάρπια άρθρωση που συνδέεται με τα κόκαλα του καρπού, η μετακαρποφαλαγγική άρθρωση η οποία βρίσκεται πιο κοντά στην παλάμη και η μεσοφαλαγγική που είναι η τελευταία άρθρωση του αντίχειρα. Τα υπόλοιπα δάχτυλα έχουν ξεχωριστές ονομασίες για την κάθε άρθρωσή τους (Εικόνα 2.5), η μετακαρποφαλαγγική η οποία είναι αμέσως μετά τον καρπό του χεριού, η εγγύς μεσοφαλαγγική που είναι η μεσαία άρθρωση του δαχτύλου και η άπω μεσοφαλαγγική που είναι η τελευταία άρθρωση του δαχτύλου.



Εικόνα 2.5. Οι αρθρώσεις του ανθρώπινου χεριού.

### 2.3. Τα γάντια συλλογής δεδομένων και οι εφαρμογές τους

Τα γάντια συλλογής δεδομένων είναι μια διάταξη με πολλαπλούς αισθητήρες που μπορούν να καταγράψουν την κίνηση του ανθρώπινου χεριού, την θέση, τον προσανατολισμό και την κίνηση των δαχτύλων [2, 3, 4]. Λόγω των χαρακτηριστικών τους όπως η ευελιξία, το βάρος και η ακρίβεια δεδομένων βρίσκει εφαρμογές σε πολλά τεχνολογικά πεδία όπως, η εικονική αναπαράσταση (virtual reality) [5, 6], ο τηλεχειρισμός και έλεγχος ρομποτικών συστημάτων βιομηχανικών εφαρμογών [3, 7, 8, 9], ο τηλεχειρισμός και έλεγχος ρομποτικών συστημάτων στρατιωτικών εφαρμογών [10] και πολλά άλλα.

Είναι άξιο να αναφερθεί ότι τα τελευταία χρόνια έχει παρατηρηθεί ότι γίνονται εκτεταμένες μελέτες από επιστήμονες σχετικά με την καταγραφή της κίνησης των δαχτύλων του ανθρώπινου χεριού με κατάλληλο σύστημα καταγραφής εικόνας [11]. Ένα από αυτά τα συστήματα είναι το *Kinect* [12, 13]

Παρακάτω αναφέρονται κάποια από τα πιο γνωστά γάντια συλλογής δεδομένων που υπάρχουν στο εμπόριο, τα οποία είναι το *DG5 VHand 3.0*, *CyberGlove 2*, *5DT Data Glove 5 Ultra*, *AcceleGlove*, και *Virtual Reality P5 Glove*. Τέλος, γίνεται αναφορά για το γάντι συλλογής δεδομένων που παρουσιάζεται σε αυτήν την πτυχιακή εργασία και κατασκευάστηκε στο εργαστήριο Αυτοματικής και Ρομποτικής του Τ.Ε.Ι. Κρήτης.

### 2.3.1. Το γάντι συλλογής δεδομένων *DG5 – Vhand Glove 3.0*

Το γάντι συλλογής δεδομένων *DG5 – Vhand Glove 3.0* έχει εγκατεστημένους 8 αισθητήρες. Οι πέντε είναι εύκαμπτοι αισθητήρες (bend sensors), οι οποίοι δίνουν την δυνατότητα στο γάντι να έχει αρκετή ακρίβεια για την μέτρηση της κίνησης των δακτύλων. Ενώ οι τρεις αισθητήρες κίνησης (επιταχυνσιόμετρο (*accelerometer*), γυροσκόπιο (*gyroscope*), μαγνητόμετρο (*magnetometer*)), επιτρέπουν τον προσδιορισμό του προσανατολισμού (*roll*, *pitch* και *yaw*) και της κίνησης του ανθρώπινου χεριού στο χώρο. Επιπλέον, το γάντι διαθέτει ασύρματη επικοινωνία. Μπορεί και να τροφοδοτηθεί κατευθείαν από θύρα USB ή από μπαταρία για τελείως ασύρματη επικοινωνία (Εικόνα 2.6) [14].



Εικόνα 2.6. Το γάντι συλλογής δεδομένων *DG5 VHand 3.0*.

### 2.3.2. Το γάντι συλλογής δεδομένων *CyberGlove II*

Το γάντι συλλογής δεδομένων *CyberGlove II*, διαθέτει 18 εύκαμπτους αισθητήρες που έχουν κατανεμηθεί ανάλογα με το βαθμό ελευθερίας των αρθρώσεων. Υπάρχουν δύο σε κάθε δάχτυλο για την κάμψη/επέκταση και τέσσερις για την απαγωγή/συναγωγή των δακτύλων, επιπλέον αισθητήρες που ανιχνεύει την κίνηση του αντίχειρα για την καρπομετακάρπια άρθρωση, το τόξο της παλάμης, την έκταση/επέκταση και απαγωγή/συναγωγή του καρπού. Επιπλέον έχει την δυνατότητα να στέλνει ασύρματα τα δεδομένα. Λόγω των χαρακτηριστικών του, έχει εφαρμογές σε πολλούς τομείς όπως, η εικονική πραγματικότητα, η καταγραφή της κίνησης του ανθρώπινου χεριού και άλλα (Εικόνα 2.7) [15, 16, 17].



Εικόνα 2.7. Το γάντι συλλογής δεδομένων *CyberGlove II*.

### 2.3.3. Το γάντι συλλογής δεδομένων *5DT Data Glove 5 Ultra*

Το γάντι συλλογής δεδομένων *Data Glove 5 Ultra* διαθέτει δύο εκδόσεις· Η πρώτη έκδοση διαθέτει 5 εύκαμπτους αισθητήρες και στην δεύτερη έχουν εγκαταστήσει 14 εύκαμπτους αισθητήρες. Επιπρόσθετα διαθέτει ασύρματη επικοινωνία (Bluetooth) εμβέλειας 20 μέτρων. Λόγω των χαρακτηριστικών του όπως το ελάχιστο βάρος, η ευελιξία, η υποστήριξη ασύρματης επικοινωνίας αλλά και η παροχή λογισμικού βρίσκει ευρεία εφαρμογή σε πολλούς τομείς (Εικόνα 2.8) [18].



Εικόνα 2.8. Γάντι συλλογής δεδομένων *5DT Data Glove 5 Ultra*

### 2.3.4. Το γάντι συλλογής δεδομένων AcceleGlove

Το AcceleGlove διαθέτει 6 επιταχυνσιόμετρα, 5 από τα οποία είναι εγκατεστημένα στην άκρη κάθε δαχτύλου ενώ, το τελευταίο είναι εγκατεστημένο στην πάνω επιφάνεια του χεριού. Το γάντι έχει την δυνατότητα να μετατρέπει την κίνηση των δαχτύλων και του χεριού σε δεδομένα, για αυτόν το λόγο βρίσκει εφαρμογές σε διάφορους τομείς όπως Ρομποτική, Φυσικοθεραπεία, Τηλεϊατρική, Εκπαίδευση, εικονική πραγματικότητα και άλλα (Εικόνα 2.9) [19].



Εικόνα 2.9. Γάντι συλλογής δεδομένων AcceleGlove.

### 2.3.5. Το γάντι συλλογής δεδομένων Virtual Reality P5 Glove

Το γάντι συλλογής δεδομένων P5 Glove έχει εγκατεστημένους 5 εύκαμπτους αισθητήρες, επίσης παρακολουθεί την κίνηση και τον προσανατολισμό (6 βαθμοί ελευθερίας) μέσω συστήματος καταγραφής εικόνας. Παρέχει στους χρήστες ολοκληρωμένη διαισθητική αλληλεπίδραση σε τρισδιάστατο και εικονικό περιβάλλον όπως παιχνίδια, ιστοσελίδες και εκπαιδευτικά λογισμικά. Στην Εικόνα 2.10 απεικονίζεται το ρομποτικό γάντι P5 Glove. [20]





Εικόνα 2.10. Γάντι συλλογής δεδομένων Virtual Reality P5 Glove.

### 2.3.6. Το γάντι συλλογής δεδομένων TALOS' Glove

Το γάντι συλλογής δεδομένων (Εικόνα 2.11) που παρουσιάζεται σε αυτήν την πτυχιακή εργασία και κατασκευάστηκε στο εργαστήριο Ρομποτικής & Αυτοματικής του Τ.Ε.Ι. Κρήτης, διαθέτει 11 αισθητήρες κάμψης που ανιχνεύουν την κίνηση των ανθρώπινων δαχτύλων. Το γάντι μπορεί να καταγράψει την κίνηση για όλα τα δάχτυλα του ανθρώπινου χεριού. Περαιτέρω, μπορεί να στέλνει τα δεδομένα ασύρματα. Το γάντι συλλογής δεδομένων χρησιμοποιήθηκε για τον χειρισμό του ανθρωπόμορφου ρομποτικού χεριού *TALOS' HAND* [21, 22] αλλά και απεικόνιση της κίνησης των ανθρώπινων δαχτύλων σε εικονικό περιβάλλον.



Εικόνα 2.11. Γάντι συλλογής δεδομένων TALOS' Glove.



### **3. Συσχέτιση της κίνησης των αρθρώσεων του ανθρώπινου χεριού**

#### **3.1. Εισαγωγή**

*Σε αυτό το κεφάλαιο γίνεται αναφορά στους περιορισμούς κίνησης των ανθρώπινων δαχτύλων δηλαδή, τις σχέσεις που έχουν οι αρθρώσεις και τα δάχτυλα μεταξύ τους.*

#### **3.2. Περιορισμοί που υπάρχουν στις κινήσεις των δαχτύλων του ανθρώπινου χεριού**

Τα δάχτυλα του ανθρώπινου χεριού διαθέτουν δύο τύπους περιορισμών, είναι οι δυναμικοί και στατικοί περιορισμοί. Οι στατικοί περιορισμοί αφορούν το εύρος κίνησης που μπορούν να έχουν τα δάχτυλα και εξαρτώνται από την ανατομία του χεριού (διαστάσεις του ανθρώπινου χεριού). Οι δυναμικοί περιορισμοί αφορούν την σχετική κίνηση των αρθρώσεων είτε στα ίδια τα δάχτυλα είτε μεταξύ των δακτύλων (π.χ. ανάμεσα στον δείκτη και τον μέσο). Οι δυναμικοί περιορισμοί χωρίζονται σε ένδο – δακτύλιους και έσω – δακτύλιους.

Οι ένδο – δακτύλιοι περιορισμοί περιγράφουν τις σχέσεις ανάμεσα στις αρθρώσεις ενός δαχτύλου (π.χ. του δείκτη, του μέσου, του παράμεσου ή του μικρού). Οι έσω – δακτύλιοι

περιορισμοί περιγράφουν τις σχέσεις που έχουν οι αρθρώσεις ενός δαχτύλου (π.χ. του δείκτη, του μέσου, του παράμεσου ή του μικρού) με τις αρθρώσεις των υπόλοιπων δαχτύλων [23, 16, 17].

Υπάρχουν δύο τύποι κίνησης των δαχτύλων του ανθρώπινου χεριού, είναι η ενεργητική και παθητική. Ενεργητική είναι η κίνηση η οποία ενεργοποιείται από τους μύες και τους τένοντες. Παθητική είναι η κίνηση που δεν ενεργοποιείται από κάποιο νεύρο ή τένοντα, δηλαδή η κίνηση των δαχτύλων ενεργοποιείται από εξωτερικές δυνάμεις π.χ. η υπερέκταση ενός δαχτύλου (Εικόνα 3.1) ασκώντας στο άκρο αυτού μια δύναμη.



Εικόνα 3.1. Απεικόνιση της υπερέκτασης του δείκτη.

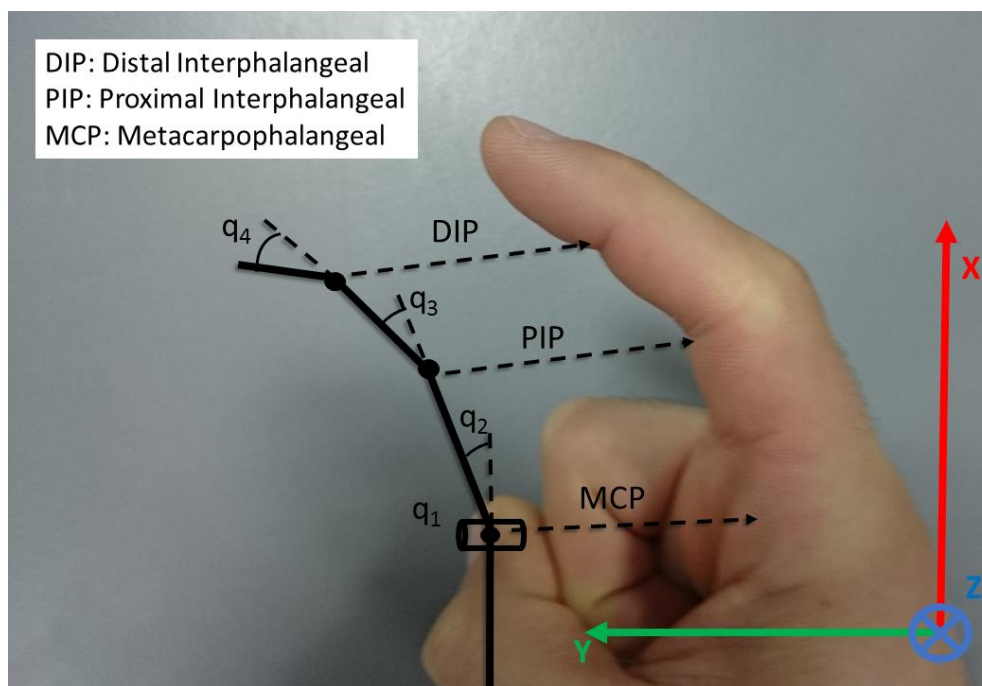
Η μελέτη για τους περιορισμούς στις αρθρώσεις των ανθρώπινων δαχτύλων έγινε για την εύρεση σχέσεων μεταξύ των αρθρώσεων που έχουν τα ανθρώπινα δάχτυλα ώστε, να μειωθούν οι αισθητήρες εν χρήσει άρα και ο όγκος των δεδομένων.

### **3.2.1. Ένδο – δακτύλιοι περιορισμοί**

Οι ενδο-δακτύλιοι περιορισμοί χωρίζονται σε δύο τμήματα, το πρώτο τμήμα είναι αυτό που περιέχει τα γενικά δάχτυλα (δείκτη, μέσο, παράμεσο και μικρό), ενώ το δεύτερο τμήμα είναι ο αντίχειρας. Οι περιορισμοί έχουν προταθεί σε διάφορες μελέτες έπειτα από κατάλληλα πειράματα του ανθρώπινου χεριού [23, 16, 17].

#### **3.2.1.1. Περιορισμοί για τα γενικά δάχτυλα δείκτη, μέσο, παράμεσο και μικρό**

Οι σχέσεις ισχύουν με την προϋπόθεση ότι τα δάχτυλα δεν ασκούν πίεση σε κάποια επιφάνεια, δηλαδή η κίνηση γίνεται ελεύθερα στον χώρο.



Εικόνα 3.2. Σχεδιάγραμμα για το δάχτυλο δείκτη του ανθρώπινου χεριού, όπου απεικονίζει τις αρθρώσεις του.

Η παρακάτω σχέση αναφέρει την συσχέτιση της άπω μεσοφαλαγγικής άρθρωσης (*distal interphalangeal (DIP)*) σε σχέση με την εγγύς μεσοφαλαγγική άρθρωση (*proximal interphalangeal (PIP)*).

$$q_4 \approx \frac{2}{3} q_3 \quad (3.1)$$

Από την σχέση (3.1) μπορούμε να συμπεράνουμε ότι, η περιστροφή της άρθρωσης  $q_4$  σχετίζεται με το 70% της περιστροφής της άρθρωσης  $q_3$ .

Ενώ στην παρακάτω σχέση περιγράφεται η συσχέτιση της εγγύς μεσοφαλαγγική άρθρωσης σε σχέση με την μετακαρποφαλαγγική άρθρωση (*metacarpophalangeal (MCP)*).

$$q_3 \approx \frac{3}{4} q_2 \quad (3.2)$$

Βάσει την παραπάνω σχέση (3.2) μπορούμε να συμπεράνουμε ότι, η περιστροφή της άρθρωσης  $q_2$  συνεπάγει το 75% της περιστροφής της άρθρωσης  $q_3$ . Επιπρόσθετα, από τις σχέσεις (3.1), (3.2) οι αρθρώσεις  $q_4$ ,  $q_3$  μπορούν να καθοριστούν γνωρίζοντας την περιστροφή της άρθρωσης  $q_2$ . Από τις σχέσεις (3.1) και (3.2) σχετίζεται η περιστροφή της άρθρωσης  $q_4$  με την  $q_2$ .

$$q_4 \approx \frac{1}{2} q_2 \quad (3.3)$$

Οι σχέσεις που αναφέρονται παραπάνω ισχύουν για την κάμψη/επέκταση των ανθρώπινων δαχτύλων, όμως υπάρχουν σχέσεις που σχετίζουν την απαγωγή/συναγωγή των δαχτύλων.

Στις περισσότερες περιπτώσεις είναι δύσκολο να γίνει απαγωγή/συναγωγή των δαχτύλων χωρίς να προκύψει κίνηση των υπόλοιπων δαχτύλων.

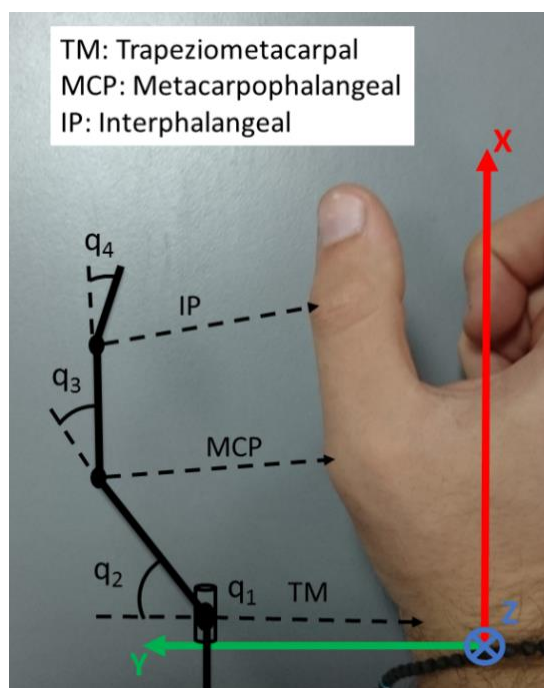
$$q_{1\_Μέσος} \approx \frac{1}{5} q_{1\_Δείκτης} \quad (3.4)$$

$$q_{1\_Παράμεσος} \approx \frac{1}{2} q_{1\_Μικρός} \quad (3.5)$$

Από τις παραπάνω σχέσεις (3.4),(3.5) συμπεραίνουμε ότι η απαγωγή του μέσου συνεπάγεται σε απαγωγή του δείκτη κατά 20%. Η απαγωγή του παράμεσου συνεπάγεται σε απαγωγή του μικρού κατά 50%.

### 3.2.1.2. Περιορισμοί του Αντίχειρα

Παρακάτω αναφέρονται οι περιορισμοί για τον αντίχειρα του ανθρώπινου χεριού. Θα πρέπει να αναφερθεί ότι οι παρακάτω σχέσεις ισχύουν για την ελεύθερη κίνηση και κάμψη/επέκταση του δαχτύλου.



Εικόνα 3.3. Σχεδιάγραμμα για τον αντίχειρα του ανθρώπινου χεριού, όπου απεικονίζει τις αρθρώσεις του. Στην παρακάτω σχέση αναφέρεται η συσχέτιση της άπω μεσοφαλαγγικής άρθρωσης (*interphalangeal (IP)*) και της μετακαρποφαλαγγικής άρθρωσης.

$$q_4 \approx \frac{1}{2} q_3 \quad (3.6)$$

Από την (3.6) μπορούμε να συμπεράνουμε ότι, η άρθρωση  $q_4$  περιστρέφεται περίπου το μισό από την άρθρωση  $q_3$ .

Αντίστοιχα στην παρακάτω σχέση περιγράφεται η συσχέτιση της μετακαρποφαλαγγικής άρθρωσης και της καρπομετακάρπιας άρθρωσης (*trapeziometacarpal (TM)*).

$$q_3 \approx \frac{5}{4} q_2 \quad (3.7)$$

Από την (3.7) μπορούμε να συμπεράνουμε ότι η περιστροφή της  $q_3$  συνεπάγει το 125% της περιστροφή της άρθρωσης  $q_2$ . Από τις σχέσεις (3.6), (3.7) σχετίζεται η περιστροφή της άρθρωσης  $q_4$  με την  $q_2$ .

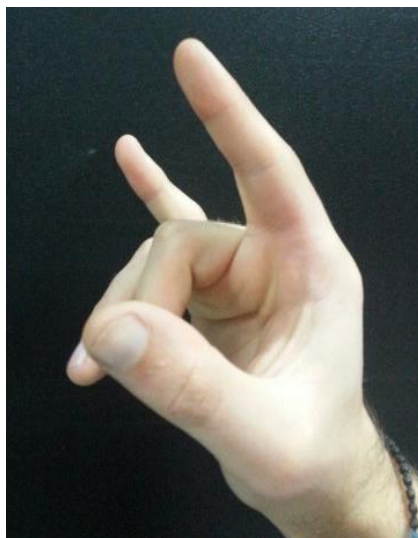
$$q_4 \approx \frac{5}{8} q_2 \quad (3.8)$$

### 3.2.2. Έσω – δακτύλιοι περιορισμοί

Οι έσω – δακτύλιοι περιορισμοί ισχύουν για τα όλα τα δάχτυλα εκτός του αντίχειρα και χωρίζονται σε δύο κατηγορίες κινήσεων, είναι οι περιπτώσεις συσχετιζόμενων κινήσεων ενός δακτύλου μεταξύ των υπόλοιπων δακτύλων και η συσχέτιση των αρθρώσεων μεταξύ των δακτύλων. Η συσχέτιση των αρθρώσεων περιγράφει την σχέση που έχουν τα δάχτυλα μεταξύ τους σε σχέση με την κίνηση των υπόλοιπων δακτύλων. Οι περιπτώσεις συσχετιζόμενων κινήσεων ενός δακτύλου μεταξύ των υπόλοιπων δακτύλων, περιγράφουν την σχέση των αρθρώσεων στα δάχτυλα, για ίδια περιστροφή σε μοίρες, όταν γίνονται συγκεκριμένες κινήσεις των υπόλοιπων δακτύλων.

Υπάρχει συσχέτιση ανάμεσα στις κινήσεις των δακτύλων, όταν γίνεται κάμψη της άρθρωσης  $q_2$  στον μέσο αντίστοιχη κάμψη εμφανίζεται στην άρθρωση  $q_2$  του παράμεσο. Όταν γίνεται αυτή η χαρακτηριστική κίνηση (Εικόνα 3.4) τότε ισχύει η παρακάτω σχέση.

$$q_{2\_Μέσος} \approx q_{2\_Παράμεσος} \quad (3.9)$$



Εικόνα 3.4. Αναπαράσταση κίνησης για τους έσω – δακτύλιους περιορισμούς.

Επίσης, κατά την πλήρη κάμψη του παράμεσου συνεπάγεται ταυτόχρονη κάμψη της άρθρωσης  $q_2$  για τον μέσο και τον μικρό (Εικόνα 3.5).

$$q_{2\_Μέσος} \approx q_{2\_Μικρός} \quad (3.10)$$



Εικόνα 3.5. Αναπαράσταση κίνησης για την ζεύξη των κινήσεων του μέσου και μικρού δαχτύλου.

Ο μικρός και ο παράμεσος ακολουθούν την παρακάτω συσχέτιση: ξεκινώντας από την αρχική θέση του χεριού, στην οποία όλα τα δάχτυλα σε πλήρη έκταση, μια κάμψη του μικρού προκαλεί μια αντίστοιχη αλλά μικρότερου μεγέθους κάμψη στον παράμεσο (Εικόνα 3.6). Η κίνηση αυτή υπόκειται στους παρακάτω περιορισμούς [23, 16, 17].

$$q_{2\_Παράμεσος} \approx \frac{7}{12} q_{2\_Μικρός} \quad q_{2\_Παράμεσος} - q_{2\_Μικρός} < 50^\circ \quad (3.11)$$



Εικόνα 3.6. Απεικόνιση της κίνησης των δαχτύλων.

Η κάμψη του παράμεσου είναι περισσότερο αισθητή όταν η σχετική γωνία  $q_2$  του μικρού ως προς την σχετική γωνία  $q_2$  του παράμεσο πλησιάζει τις  $50^\circ$ .

Επιπροσθέτως όταν ο μικρός βρίσκεται σε πλήρη κάμψη και ο δείκτης σε πλήρη έκταση τότε η κάμψη του παράμεσου προκαλεί αντίστοιχη κάμψη στον μέσο:

$$q_{2\_Μέσος} \approx \frac{3}{2} q_{2\_Παράμεσος} \quad q_{2\_Παράμεσος} - q_{2\_Μέσος} < 60^\circ \quad (3.12)$$

Ξεκινώντας σε πλήρη έκταση τα δάχτυλα, η κάμψη του δείκτη συνεπάγει σε κάμψη του μέσου (Εικόνα 3.7):

$$q_{2\_Μέσος} \approx \frac{1}{5} q_{2\_Δείκτη} \quad (3.13)$$



Εικόνα 3.7. Απεικόνιση για την χαρακτηριστική κίνηση που συνδέει την περιστροφή του δείκτη με τον παράμεσο.

Παρατηρώντας τα παραπάνω, μπορούμε να πούμε ότι γενικά ο δείκτης δεν επηρεάζεται από την κάμψη των υπόλοιπων δαχτύλων.

### **3.3. Συμπεράσματα**

Σε αυτό το κεφάλαιο παρουσιάστηκαν οι περιορισμοί για τα δάχτυλα του ανθρώπινου χεριού. Από την χρήση των περιορισμών μπορούμε να συμπεράνουμε ότι γνωρίζοντας την περιστροφή σε μια ή δύο αρθρώσεις μπορεί να γίνει καθορισμός των υπόλοιπων αρθρώσεων που συνεπάγεται στην χρήση λιγότερων αισθητήρων καταγραφής της κίνησης. Περαιτέρω, μπορούν να χρησιμοποιηθούν για την λαβή αντικειμένων με απλή γεωμετρία.



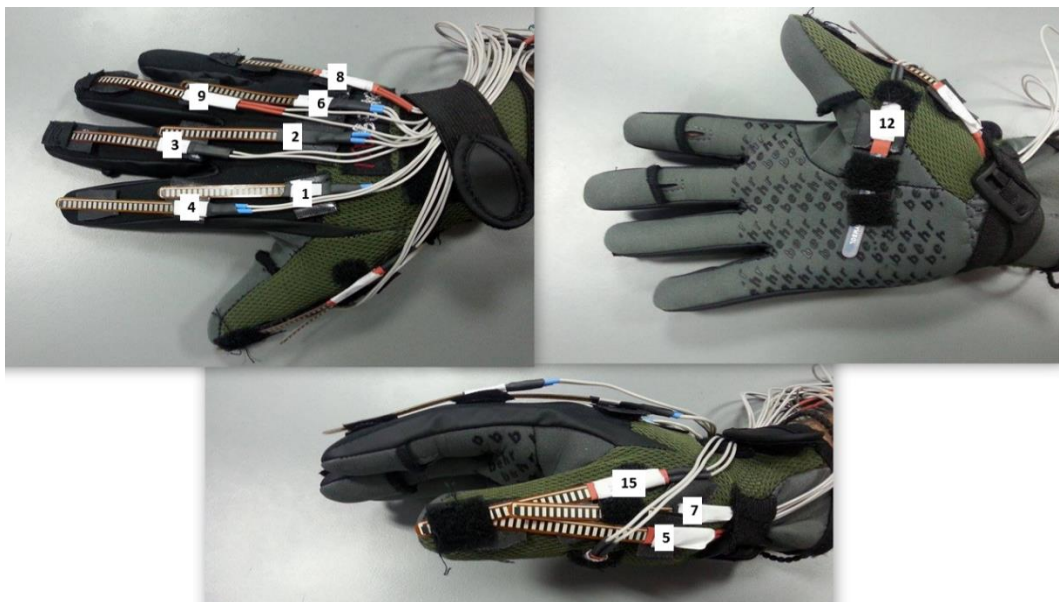
## 4. Σχεδίαση συστήματος για τη καταγραφή της κίνησης των ανθρώπινων δαχτύλων

### 4.1. Εισαγωγή

Αυτό το κεφάλαιο αφορά την σχεδίαση και κατασκευή συστήματος καταγραφής της κίνησης των ανθρώπινων δαχτύλων. Χρησιμοποιούνται αισθητήρες κάμψης (*flex sensor*), οι οποίοι προσαρμόζονται πάνω σε γάντι. Επικεντρώνουμε την μελέτη μας στους αισθητήρες για την πιο ακριβή και αξιόπιστη καταγραφή της κίνησης. Προτείνεται τρόπος τοποθέτησης των αισθητήρων πάνω στο γάντι για την βέλτιστη καταγραφή της κίνησης των ανθρώπινων δαχτύλων.

### 4.2. Σύστημα καταγραφής κίνησης

Η καταγραφή της κίνησης των ανθρώπινων δαχτύλων γίνεται μέσω των αισθητήρων κάμψης (*flex sensor*), οι οποίοι είναι τοποθετημένοι σε γάντι (επιλέχθηκε γάντι κατάδυσης λόγω της ποιότητας του) (Εικόνα 4.1).

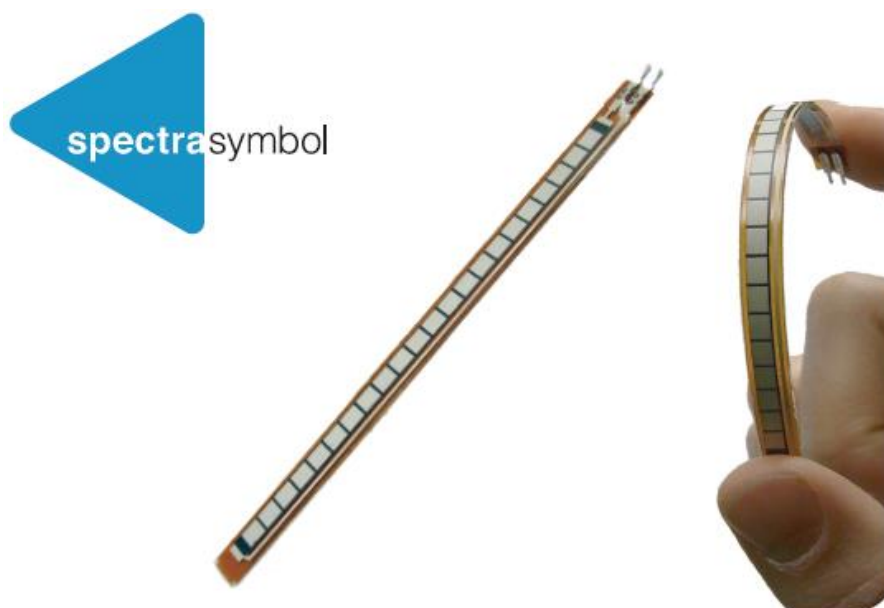


Εικόνα 4.1. Απεικόνιση της διάταξης των αισθητήρων πάνω στο γάντι.

Για την πλήρη καταγραφή της κίνησης των ανθρώπινων δακτύλων και βάσει όσων αναφέρθηκαν στο κεφάλαιο 3, χρησιμοποιούνται 11 αισθητήρες. Πιο αναλυτικά, τοποθετούνται 2 αισθητήρες στον δείκτη (αισθητήρες 1, 4), μέσο (αισθητήρες 2, 3), παράμεσο (αισθητήρες 6, 9) και 1 αισθητήρας στον μικρό (αισθητήρας 8)· οι υπόλοιποι 4 αισθητήρες τοποθετούνται στον αντίχειρα (αισθητήρες 12, 7, 5, 15). Επίσης, οι αισθητήρες τοποθετούνται στην 3<sup>η</sup> γραμμή στο σημείο κάμψης όπως παρουσιάζεται παρακάτω. Θα πρέπει να σημειωθεί ότι η αρίθμηση των αισθητήρων είναι τυχαία διότι, δοκιμάστηκαν διάφοροι αισθητήρες ώστε να γίνει επιλογή ανάμεσα σε αυτούς με την καλύτερη συμπεριφορά.

Οι αισθητήρες (Εικόνα 4.2) επιλέχθηκαν κυρίως λόγω των χαρακτηριστικών που διαθέτουν, τα πιο σημαντικά χαρακτηριστικά που τα απαρτίζουν είναι:

- ✓ Χαμηλό βάρος
- ✓ Ευελιξία κίνησης
- ✓ Ευκολία εγκατάστασης λόγω μικρού μεγέθους
- ✓ Εύκολη καταγραφή δεδομένων



Εικόνα 4.2. Αισθητήρας κάμψης (*Flex Sensor*).

Βάσει των παραπάνω, κατασκευάσαμε ένα σύστημα το οποίο έχει μικρό βάρος, ευελιξία κίνησης αλλά και φορητότητα για ασύρματη επικοινωνία.

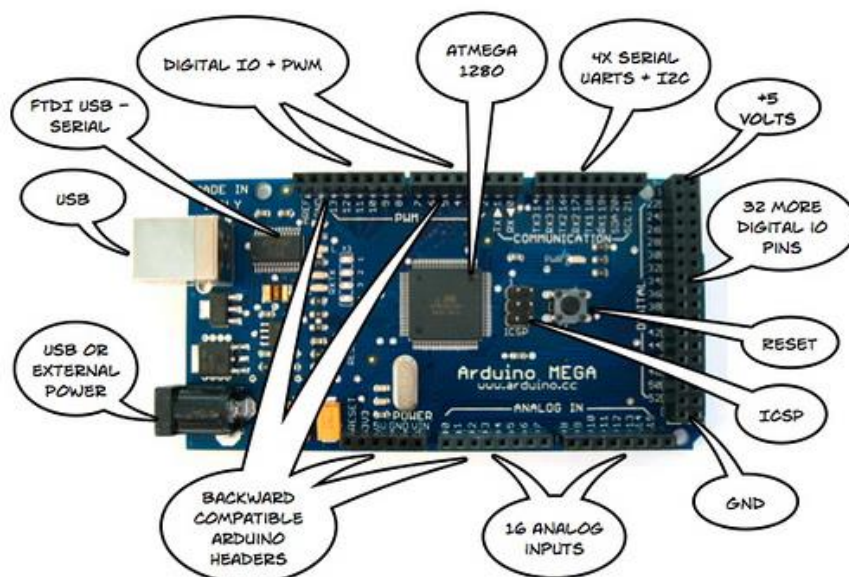
Τέλος, για την καταγραφή των δεδομένων από τους αισθητήρες χρησιμοποιήθηκε ο μικροελεγκτής Arduino.

### 4.3. Ο μικροελεγκτής Arduino

Ο μικροελεγκτής Arduino Mega 2560 είναι βασισμένος στον μικροεπεξεργαστή ATmega1280. Διαθέτει 54 ψηφιακά pin εισόδου/εξόδου (Όπου τα 14 από αυτά μπορούν να χρησιμοποιηθούν ως έξοδοι με *PWM (Pulse Width Modulation)* σήμα, 16 αναλογικές εισόδους, 4 UARTs (Σειριακές θύρες), 16 MHz κεραμικό συντονιστή, σύνδεση με USB, υποδοχή τροφοδοσίας, κεφαλή *ICSP (In Circuit Serial Programming)*, κουμπί επαναφοράς. Γενικά ο μικροελεγκτής διαθέτει όσα χρειάζονται ώστε να υποστηρίξει τον μικροεπεξεργαστή. Τέλος το Arduino μπορεί να προγραμματιστεί με την γλώσσα *Wiring* (ουσιαστικά πρόκειται για την γλώσσα *C++* με κάποιες μετατροπές), παρακάτω φαίνονται συνοπτικά τα χαρακτηριστικά του μικροελεγκτή επίσης, στην εικόνα 4-3 φαίνεται η κάτοψη του μικροελεγκτή [24].

Χαρακτηριστικά μικροελεγκτή:

- Μικροεπεξεργαστής ATmega1280
- Ονομαστική Τάση Λειτουργίας 5 V
- Τάση Εισόδου (Συνιστάται) 7 – 12 V
- Τάση Εισόδου (Όριο) 6 – 20 V
- Ψηφιακές Εισόδους/Εξόδους 54 (Όπου τα 15 διαθέτουν PWM έξοδο)
- Αναλογικές Εισόδους 16
- DC Ρεύμα για κάθε Είσοδο/Εξοδο 40 mA
- DC Ρεύμα για λειτουργία στα 3,3V 50 mA
- Flash Memory 128KB όπου τα 4KB χρησιμοποιούνται για εκκίνηση
- SRAM 8 KB
- EEPROM 4 KB
- Ταχύτητα Μικροεπεξεργαστή 16 MHz



Εικόνα 4.3. Κάτοψη μικροελεγκτή Arduino Mega 2560.

#### 4.4. Αισθητήρας κάμψης

Ο αισθητήρας κάμψης μπορεί να χρησιμοποιηθεί σε ρομπότ για ανίχνευση αντικειμένων σε μικρή απόσταση, ρομποτικά γάντια που ανιχνεύουν την κίνηση των δακτύλων ([NintedoPowerGlove](#)), πόρτες που ανιχνεύουν πόσο έχουν ανοίξει και διάφορες άλλες εφαρμογές. Παρακάτω φαίνονται τα χαρακτηριστικά του αισθητήρα [25, 26].

- Ωμική αντίσταση σε επίπεδη θέση  $25k\Omega$
- Ανοχή αντίστασης  $\pm 30\%$
- Εύρος ωμικής αντίστασης κατά της κάμψη (εξαρτάται από την κάμψη του αισθητήρα)  
 $45k\Omega - 125k\Omega$
- Ονομαστική ισχύ λειτουργίας  $0,5W - 1W$
- Θερμοκρασία λειτουργίας  $-35^{\circ}C - +80^{\circ}C$
- Κύκλος ζωής  $>1$  εκατομμύριο κάμψης

Η αντίσταση του αισθητήρα αλλάζει καθώς λυγίζει από την όψη που είναι τα μεταλλικά επιθέματα ειδικότερα, με την κάμψη του ανθρώπινου δαχτύλου η τιμή της αντίστασης του αισθητήρα μεταβάλλεται. Έτσι μπορούμε να καταγράψουμε την περιστροφή των αρθρώσεων των δαχτύλου ανάλογα με την μεταβολή της αντίστασης.

Επειδή η συμπεριφορά της τάσης εξόδου του αισθητήρα σε σχέση με την κάμψη του είναι μη γραμμική επίσης, έχει μεγάλη απόκλιση στις τιμές του ( $\pm 30\%$ ) γι' αυτόν τον λόγω ακολουθούνται συγκεκριμένα βήματα για την βαθμονόμηση του.

#### 4.5. Βαθμονόμηση αισθητήρων κάμψης

Αρχικά κατασκευάστηκαν 5 δοκίμια με γωνίες κάμψης  $20^{\circ}$ ,  $40^{\circ}$ ,  $60^{\circ}$ ,  $80^{\circ}$  και  $90^{\circ}$  (Εικόνα 4.4) στις οποίες τοποθετήθηκε ο αισθητήρας κάμψης με έξι διαφορετικούς τρόπους.



Εικόνα 4.4. Εκτυπωμένες επιφάνειες από τον τρισδιάστατο εκτυπωτή με ακτίνα καμπυλότητας 15mm.

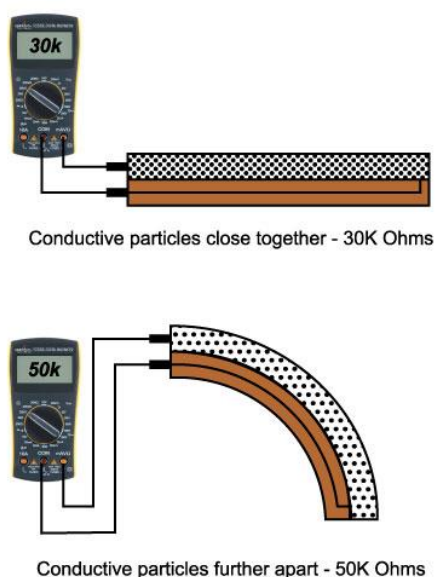
Η ακτίνα καμπυλότητας, για κάθε δοκίμιο (π.χ.  $20^{\circ}$ ,  $40^{\circ}$  κ.λπ.), είναι 15mm διότι, η ακτίνα καμπυλότητας για τις αρθρώσεις των ανθρώπινων δαχτύλων ενός μέσου ενήλικα έχοντας φορέσει το γάντι είναι 15mm.



Εικόνα 4.5. Εκτυπωμένες επιφάνειες, 15 mm.

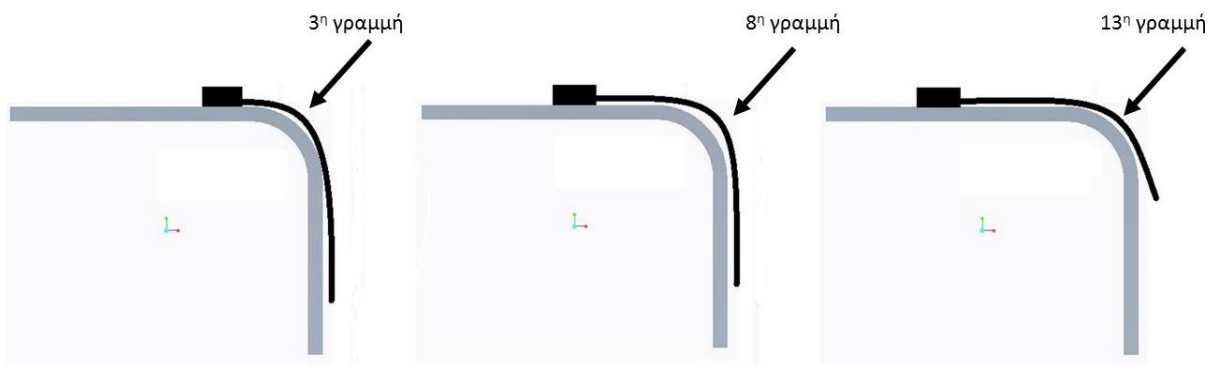
Η εκτύπωση έγινε στα μηχανήματα τρισδιάστατης εκτύπωσης (*3D printer*) στο εργαστήριο Αυτοματικής και Ρομποτικής του Τ.Ε.Ι. Κρήτης.

Εν συνέχεια, πάρθηκαν μετρήσεις της αντίστασης του αισθητήρα βάσει της παρακάτω συνδεσμολογίας (Εικόνα 4.6).



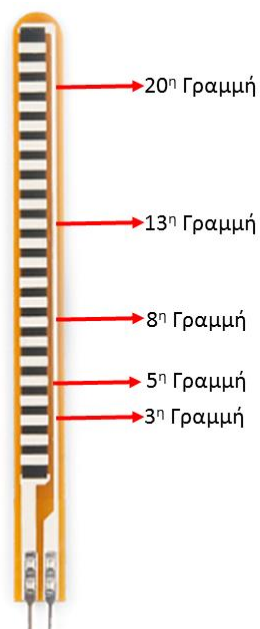
Εικόνα 4.6. Κύκλωμα καταγραφής της αντίστασης του αισθητήρα κάμψης.

Οι μετρήσεις της ωμικής αντίστασης των αισθητήρων γίνονται σε διαφορετικά σημεία πάνω στα δοκίμια που εκτυπώθηκαν (Εικόνα 4.7).



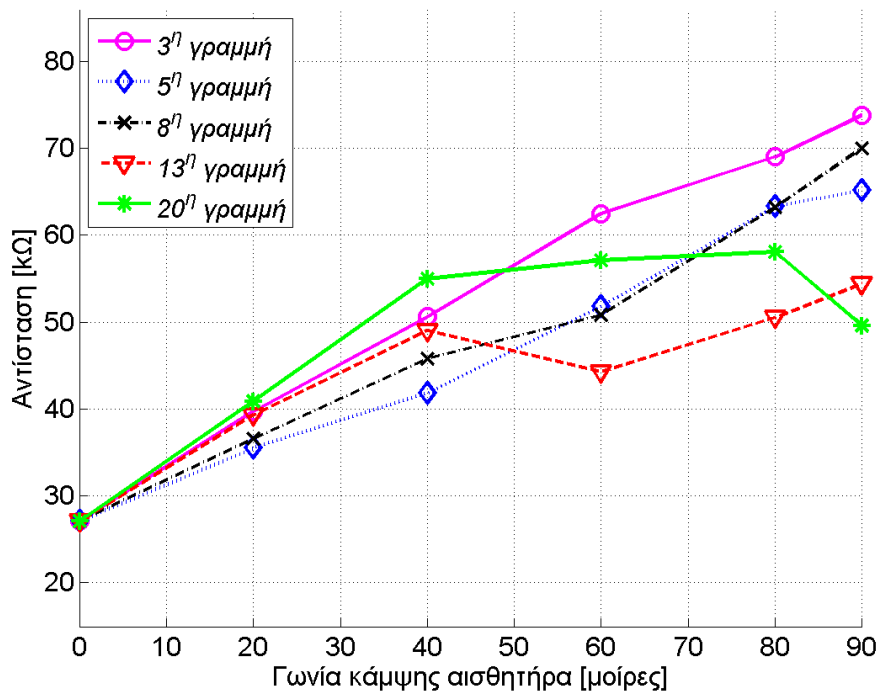
Εικόνα 4.7. Τοποθέτηση του αισθητήρα σε διαφορετικά σημεία πάνω στην επιφάνεια.

Ο αισθητήρας κάμψης (Εικόνα 4.8) διαθέτει κάποιες χαρακτηριστικές γραμμές, βάσει των γραμμών μπορούμε να ορίσουμε το σημείο τοποθέτησης των αισθητήρων πάνω στο σημείο κάμψης.



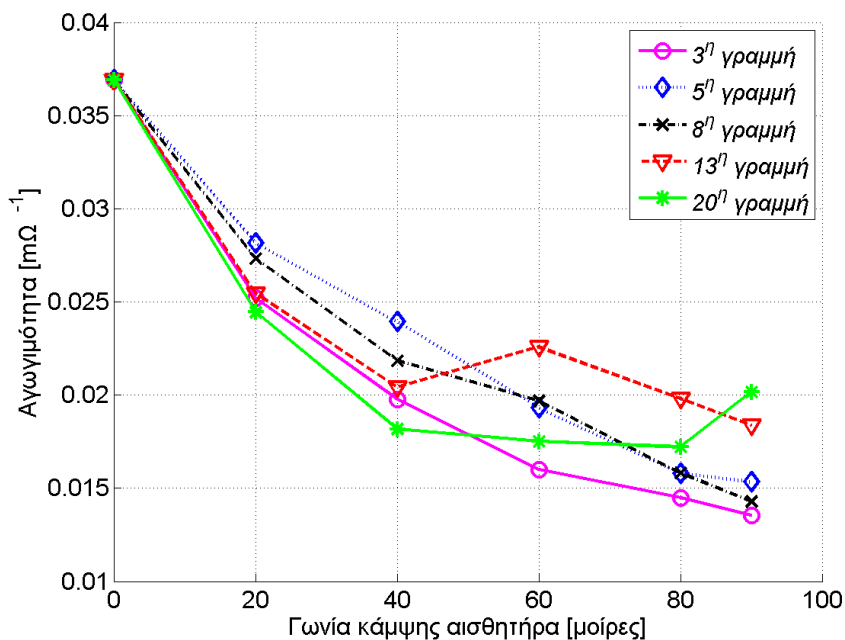
Εικόνα 4.8. Αισθητήρας κάμψης διακρίνοντας τις χαρακτηριστικές γραμμές.

Στο διάγραμμα 4-1 απεικονίζεται η ωμική αντίσταση σε σχέση με την γωνία κάμψης του αισθητήρα για διαφορετική θέση τοποθέτησης.



Διάγραμμα 4-1. Συμπεριφορά αντίστασης αισθητήρα κάμψης για διαφορετική τοποθέτηση.

Στο διάγραμμα 4-2 φαίνεται η συμπεριφορά της αγωγιμότητας σε σχέση με την γωνία κάμψης του αισθητήρα για διαφορετική θέση τοποθέτησης.



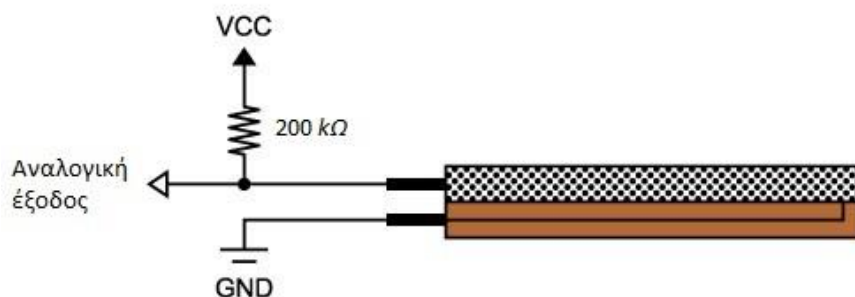
Διάγραμμα 4-2. Συμπεριφορά αγωγιμότητας του αισθητήρα κάμψης για διαφορετική τοποθέτηση.

Παρατηρούμε ότι ο αισθητήρας όσο πιο κοντά βρίσκεται στην άκρη που συνδέεται τόσο πιο γραμμική συμπεριφορά υπάρχει ανάμεσα στην ωμική αντίσταση και την γωνία



κάμψης. Έτσι οι μετρήσεις από τον αισθητήρα γίνονται τοποθετώντας την 3<sup>η</sup> γραμμή στο σημείο κάμψης (Εικόνα 4.7).

Για την συλλογή των δεδομένων από τον μικροελεγκτή υλοποιείται το παρακάτω κύκλωμα διαιρέτη τάσης (Εικόνα 4.9), αποτελούμενο από τον αισθητήρα σε σειρά με μια αντίσταση 200 kΩ.



Εικόνα 4.9. Κατάλληλο κύκλωμα συλλογής τάσης από τον αισθητήρα κάμψης.

Η τάση εξόδου δίνεται από την σχέση:

$$V_d = VCC \cdot \left( \frac{R_f}{R_f + R_c} \right) \quad (4.1)$$

Όπου:

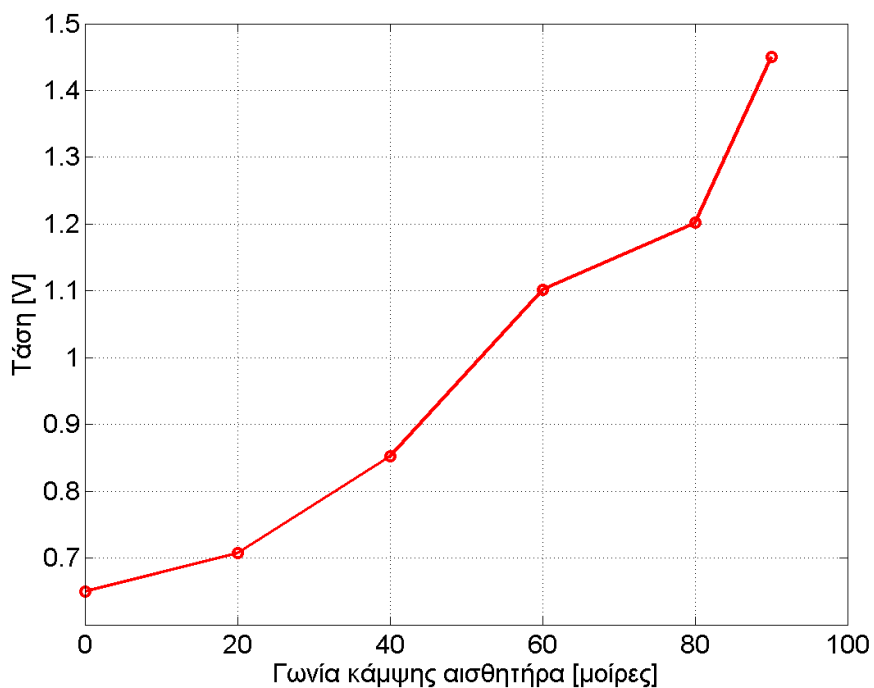
$R_f$ : Αντίσταση του αισθητήρα κάμψης.

$R_c$ : Σταθερή αντίσταση σε σειρά με τον αισθητήρα κάμψης.

$VCC$ : Τάση εισόδου στο κύκλωμα.

$V_d$ : Τάση εξόδου από το κύκλωμα.

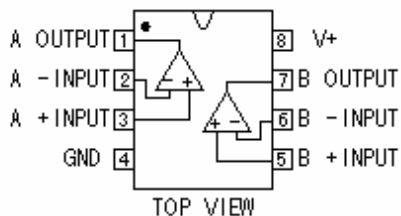
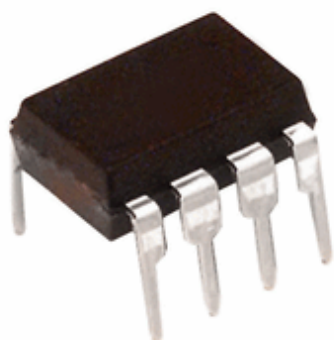
Στο διάγραμμα 4-3 απεικονίζεται η συμπεριφορά της τάσης εξόδου του αισθητήρα (μετρήσεις τοποθετώντας τον αισθητήρα στην 3<sup>η</sup> γραμμή) από το κύκλωμα του διαιρέτη τάσης (Εικόνα 4.9).



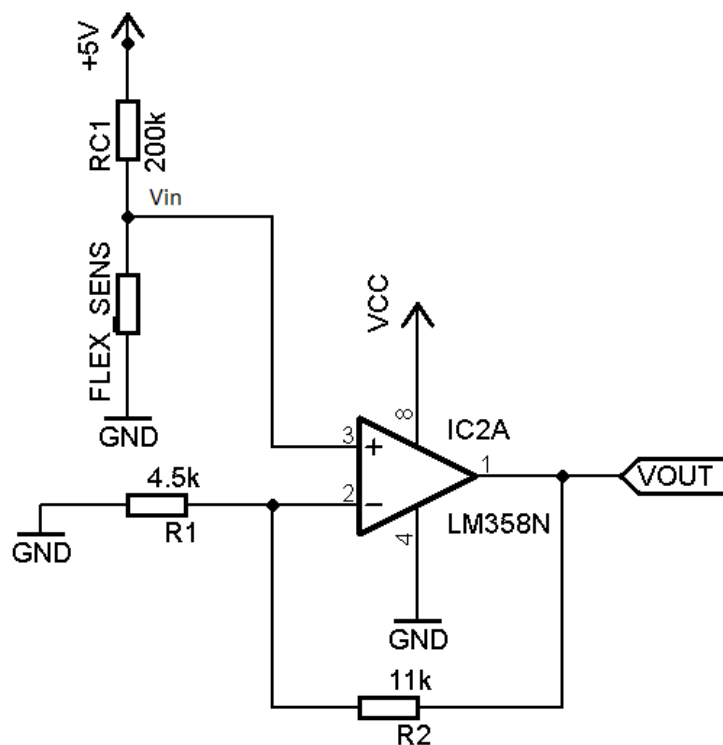
Διάγραμμα 4-3. Συμπεριφορά της τάσης εξόδου του αισθητήρα βάσει της γωνίας κάμψης (αισθητήρας 1).

Από το διάγραμμα 4-3 μπορούμε να συμπεράνουμε ότι το εύρος τιμών του αισθητήρα είναι μικρό (0,8 Volt), σαν αποτέλεσμα η καταγραφή της τάσης να μην πραγματοποιείται με καλή ακρίβεια.

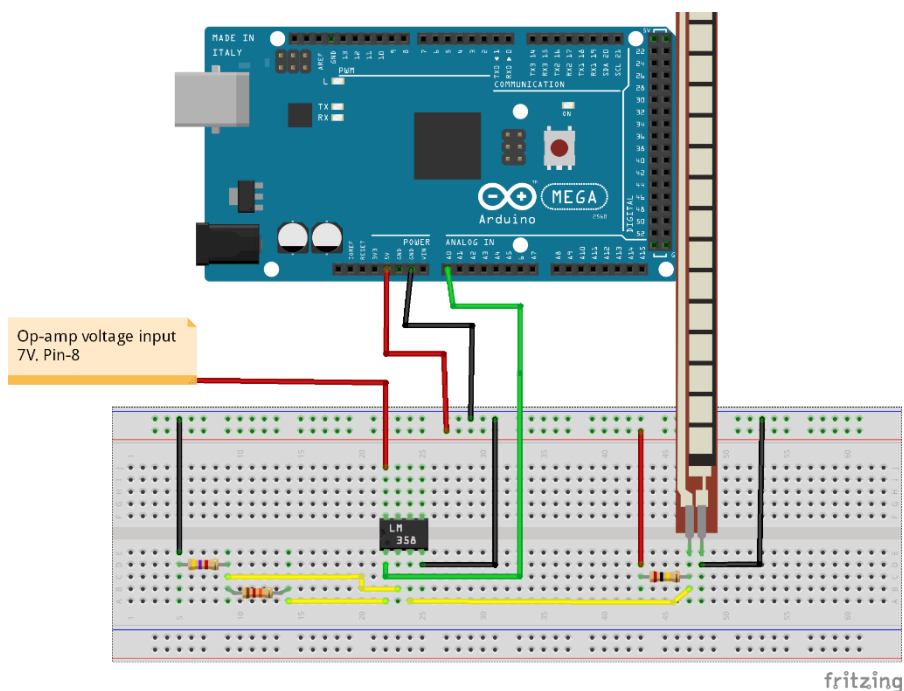
Προκειμένου να προσαρμόσουμε το εύρος της τάσης εξόδου (0,5 – 1,4 V), από τον διαιρέτη τάσης, στην είσοδο του A/D μετατροπέα του μικροελεγκτή σε (2,3 – 5 V), χρησιμοποιούμε τον τελεστικό ενισχυτή LM358P (Εικόνα 4.10) σε μη αναστρέφουσα συνδεσμολογία (Εικόνα 4.11, Εικόνα 4.12) όπως παρουσιάζεται παρακάτω:



Εικόνα 4.10. Τελεστικός ενισχυτής LM358P Dual Operational Amplifier.

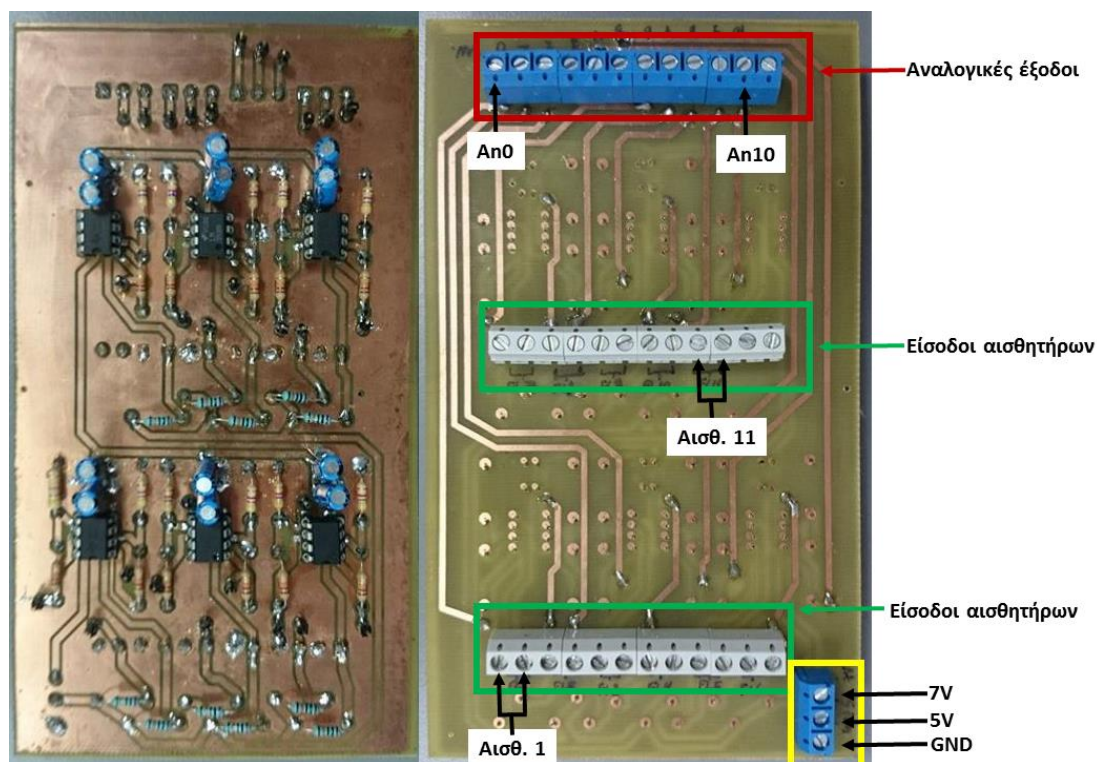


Εικόνα 4.11. Σχηματικό κύκλωμα καταγραφής δεδομένων με τελεστικό ενισχυτή (μη αναστρέφουσα συνδεσμολογία) για ενίσχυση του σήματος.



Εικόνα 4.12. Κύκλωμα συλλογής δεδομένων μέσω του Arduino.

Όπως παρατηρούμε στην εικόνα 4.12, ο τελεστικός ενισχυτής τροφοδοτείται από εξωτερική πηγή 7 Volt ενώ, ο αισθητήρας τροφοδοτείται από τον μικροελεγκτή με 5 Volt. Παρακάτω απεικονίζεται η κάτοψη τις πλακέτας τυπωμένου κυκλώματος που κατασκευάστηκε ώστε να ενισχύει την τάση εξόδου για 11 αισθητήρες (Εικόνα 4.13).

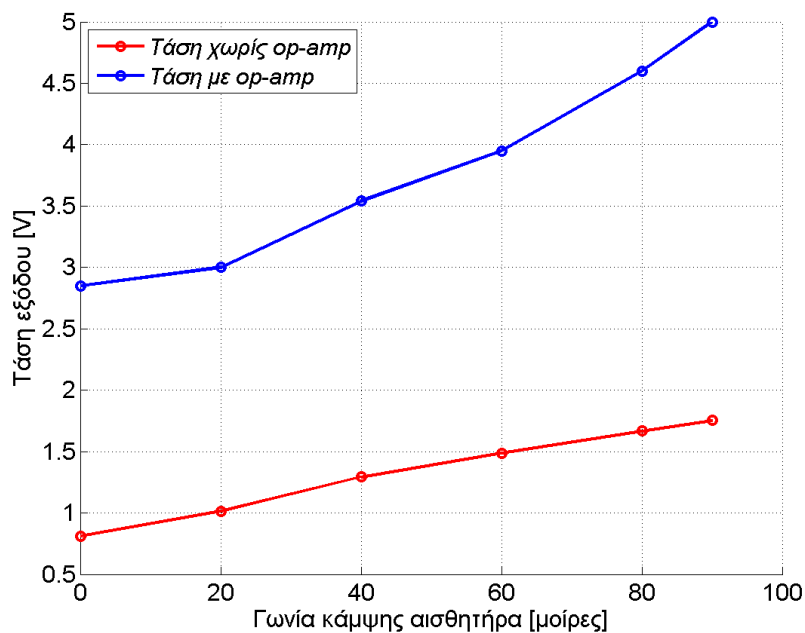


Εικόνα 4.13. Πλακέτα ενίσχυσης τάσης εξόδου αισθητήρων (στο τέλος του παραρτήματος Α απεικονίζονται τα σχέδια για την πλακέτα).

Η τάση εξόδου δίνεται από την παρακάτω σχέση:

$$V_{out} = V_{in} \cdot \left(1 + \frac{R_2}{R_1}\right) \quad (4.2)$$

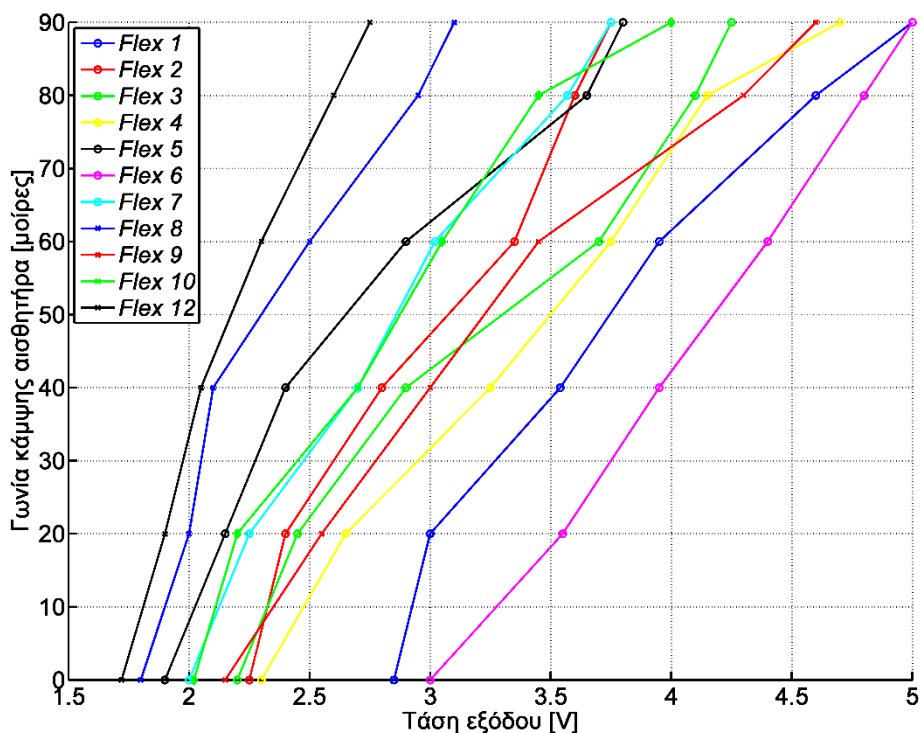
Από την σχέση (4.2) και από το κύκλωμα (Εικόνα 4.11) μπορούμε να παρατηρήσουμε ότι το σήμα εισόδου ( $V_{in}$ ) στον τελεστικό ενισχυτή ενισχύεται περίπου κατά 3,3 φορές. Στο διάγραμμα 4-4 μπορούμε να παρατηρήσουμε την τάση εξόδου και το εύρος τιμών του αισθητήρα.



Διάγραμμα 4-4. Σύγκριση της συμπεριφοράς του αισθητήρα συνδεδεμένου με τελεστικό ενισχυτή.

Από το διάγραμμα 4-4 μπορούμε να παρατηρήσουμε ότι, το εύρος τιμών της τάσης εξόδου παρουσιάζει μια μεταβολή της τάξης των 2,7 Volt' ενώ η συμπεριφορά του αισθητήρα είναι σχεδόν γραμμική.

Τέλος, πήραμε μετρήσεις από όλους τους αισθητήρες για την βαθμονόμησή τους (Διάγραμμα 4-5).



Διάγραμμα 4-5. Τάση εξόδου αισθητήρων κάμψης σε σχέση με την γωνία κάμψης.

Βάσει του διαγράμματος 4-5 συμπεραίνουμε ότι μπορούμε να περιγράψουμε την συμπεριφορά των αισθητήρων με μια γραμμική σχέση (4.3).

$$y_i = A_i \cdot x_i - B_i \quad (4.3)$$

Όπου  $y_i$  η τιμή της γωνίας κάμψης σε μοίρες ενώ  $x_i$  η τιμή της τάσης σε volt. Τα A, B είναι συντελεστές που καθορίζουν την συμπεριφορά του αισθητήρα και είναι ξεχωριστοί για κάθε αισθητήρα. Η παραπάνω σχέση (4.3) σχηματίστηκε από τις μετρήσεις των αισθητήρων τοποθετημένη στα δοκίμια που εκτυπώθηκαν. Η σχέση όμως δεν ισχύει όταν οι αισθητήρες είναι τοποθετημένη στο γάντι διότι, η επιφάνεια που είναι τοποθετημένη (πάνω στο γάντι) δεν είναι η ίδια επιφάνεια με αυτήν που έχουν τα δοκίμια. Για τον λόγω αυτό αναπτύχθηκε η παρακάτω σχέση.

$$y_i = b_{i\_min} + (x_i - a_{i\_min}) \cdot (b_{i\_max} - b_{i\_min}) / (a_{i\_max} - a_{i\_min}) \quad (4.4)$$

Όπου  $x_i$  η τιμή της τάσης εξόδου (volt) και  $y_i$  η τιμή της γωνίας κάμψης (σε μοίρες) για τον αισθητήρα ( $i$  είναι ο τρέχων αισθητήρας από 1 έως 11). Τα  $b_{i\_min}$ ,  $b_{i\_max}$  είναι η ελάχιστη και μέγιστη τιμή της γωνίας κάμψης (π.χ. 0°, 90°), αντίστοιχα. Ενώ τα  $a_{i\_min}$ ,  $a_{i\_max}$  είναι η ελάχιστη και μέγιστη τιμή της τάσης εξόδου σε volt κατά την κάμψη στο ελάχιστο και μέγιστο σημείο, αντίστοιχα. Κάθε αισθητήρας έχει διαφορετικούς συντελεστές  $a$  διότι, ο κάθε ένας έχει διαφορετική συμπεριφορά. Παρακάτω αναγράφονται οι τιμές για τον κάθε αισθητήρα.

Πίνακας 4.1. Τιμές βαθμονόμησης των αισθητήρων.

Αισθητήρας	$a_{i\_min}$ [volt]	$a_{i\_max}$ [volt]	$b_{i\_min}$ [μοίρες]	$b_{i\_max}$ [μοίρες]
1	3,20	5,00	0	90
4	3,05	5,00	0	90
2	2,40	3,55	0	90
3	2,45	4,00	0	90
6	3,20	4,70	0	90
9	3,80	5,00	0	90
8	2,10	3,75	0	90
12	2,90	4,50	270	135
7	4,00	4,60	0	90
5	2,80	4,50	0	90
15	3,60	5,00	0	90

#### **4.6. Συμπεράσματα**

Σε αυτό το κεφάλαιο εξετάσαμε το σύστημα καταγραφής της κίνησης των ανθρώπινων δαχτύλων. Επιπλέον, παρουσιάστηκε η διασύνδεση του κυκλώματος καταγραφής της κίνησης με τον μικροελεγκτή για την συλλογή των δεδομένων από τους αισθητήρες. Έπειτα από την τροποποίηση του κυκλώματος συλλογής δεδομένων η συμπεριφορά της τάσης εξόδου σε σχέση με την γωνία κάμψης των αισθητήρων από μη γραμμική έγινε σχεδόν γραμμική. Η γραμμική συμπεριφορά του αισθητήρα συντέλεσε στην εύκολη συλλογής και μετατροπή των δεδομένων.

## 5. Τηλεχειρισμός ανθρωπόμορφου ρομποτικού χεριού από το γάντι συλλογής δεδομένων

### 5.1. Εισαγωγή

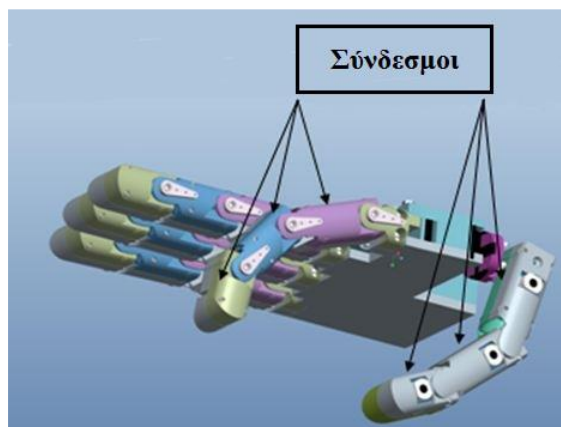
Σ' αυτό το κεφάλαιο αναφέρεται ο τρόπος τηλεχειρισμού του ρομποτικού χεριού. Παρουσιάζεται το ανθρωπόμορφο ρομποτικό χέρι και οι δυνατότητες του. Μελετάτε ο τρόπος διασύνδεσης του ρομποτικού χεριού, ενσύρματα και ασύρματα, με τον μικροελεγκτή. Τέλος, αναφέρονται τα πειραματικά αποτελέσματα από την μελέτη.

### 5.2. Το ανθρωπόμορφο ρομποτικό χέρι TALOS' HAND

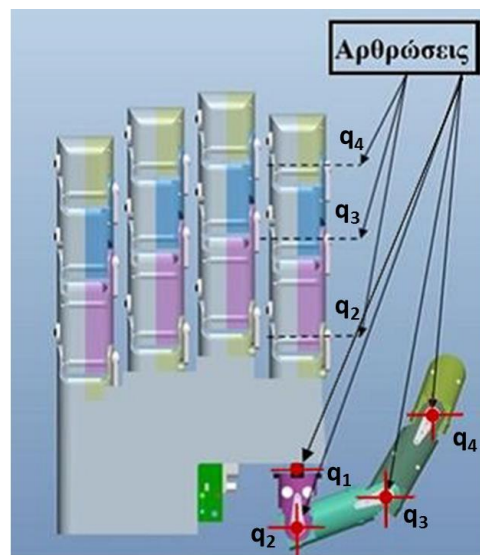
Το ανθρωπόμορφο ρομποτικό χέρι έχει πέντε δάχτυλα και παλάμη. Τα δάχτυλα αποτελούνται από τρεις συνδέσμους το καθένα (Εικόνα 5.1), έχουν το ίδιο μήκος μεταξύ τους με μικρές διαφορές για την περίπτωση του αντίχειρα. Γενικά, οι διαστάσεις του χεριού είναι λίγο μεγαλύτερες από αυτές ενός ανθρώπινου χεριού. Το κάθε δάχτυλο έχει τρεις αρθρώσεις ενώ ο αντίχειρας έχει τέσσερις (Εικόνα 5.2), έτσι το ρομποτικό χέρι έχει 16 συνολικά βαθμούς ελευθερίας. Σε κάθε μία από τις αρθρώσεις του ρομποτικού χεριού υπάρχει ένας σερβοκινητήρας ενώ τα χαρακτηριστικά κίνησης του σχεδιάστηκαν ώστε να ανταποκρίνονται



στην εκτέλεση διαφόρων χειρονομιών αλλά και προτυποποίησης κάποιων πρωτογενών λαβών. Η κατασκευή του ρομποτικού χεριού έγινε σε μηχανές ταχείας μοντελοποίησης (3D printers) που διαθέτει το εργαστήριο. Περισσότερες λεπτομέρειες για τον σχεδιασμό, την κατασκευή και την κινηματική ανάλυση του ρομποτικού χεριού μπορούν να βρεθούν στις πτυχιακές εργασίες [21, 22].

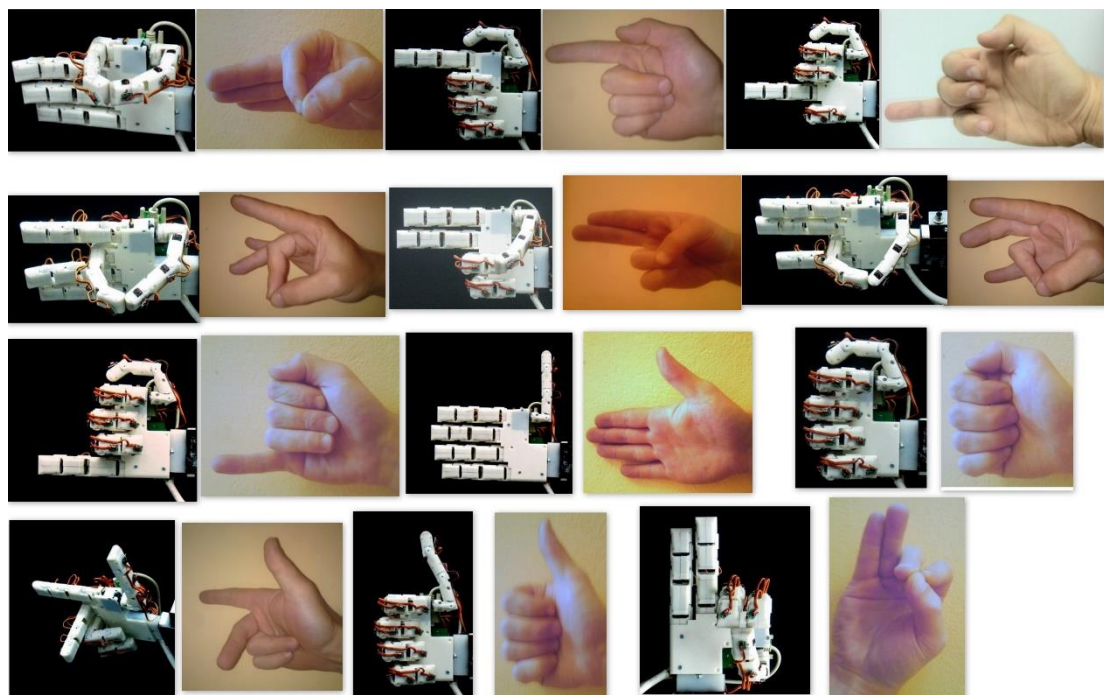


Εικόνα 5.1. Οι σύνδεσμοι του ρομποτικού χεριού.



Εικόνα 5.2. Οι αρθρώσεις του ρομποτικού χεριού.

Στην συνέχεια απεικονίζονται κάποιες δυνατότητες που ρομποτικού χεριού (Εικόνα 5.3)



Εικόνα 5.3. Δυνατότητες ανθρωπόμορφου ρομποτικού χεριού.

### 5.3. Κίνηση των αρθρώσεων του ρομποτικού χεριού TALOS'

Το ρομποτικό χέρι διαθέτει σερβοκινητήρες για την περιστροφή κάθε άρθρωσης. Ο έλεγχος των σερβοκινητήρων γίνεται μέσω του ελεγκτή SSC-32 (Εικόνα 5.4).



Εικόνα 5.4. Σερβοελεγκτής SSC-32.

Ο ελεγκτής SSC-32 έχει την δυνατότητα να ελέγχει ταυτόχρονα 32 σερβοκινητήρες, επίσης μπορεί να συνδεθεί ενσύρματα μέσω σειριακής επικοινωνίας με Η/Υ αλλά και με μικροελεγκτή (TX, RX). Ο ελεγκτής SSC-32 συνδέεται με τον μικροελεγκτή Arduino, ο μικροελεγκτής στέλνει μέσω σειριακής επικοινωνίας τις εντολές ελέγχου στον σερβοελεγκτή SSC-32.

Ο έλεγχος των σερβοκινητήρων γίνεται προσδιορίζοντας το κανάλι που είναι συνδεδεμένος ο κάθε σερβοκινητήρας, το εύρος παλμού που αντιστοιχεί στη επιθυμητή θέση του κινητήρα και ο χρόνος εκτέλεσης της τροχιάς του π.χ.

```
# (ChannelNum.) P (Pulse) T (time)
```

Για τον μικροελεγκτή Arduino υπάρχει έτοιμη βιβλιοθήκη προγραμματισμού, η "SSC32.h", όπου βρίσκεται στον παρακάτω σύνδεσμο.

<http://blog.martinperis.com/2011/05/libssc32-arduino-ssc32.html>

Ο κώδικας προϋποθέτει ότι ο ελεγκτής SSC-32 πρέπει να συνδεθεί με την σειριακή θύρα RX, TX (pin 0,1 αντίστοιχα) του Arduino. Ο κώδικας τροποποιήθηκε ώστε ο σερβοελεγκτής SSC32 να συνδέεται με τις θύρες TX1 και RX1 (pin 18, 19) του Arduino MEGA. Για να την χρήση της βιβλιοθήκης στον Arduino χρειάζεται να συμπεριλήφθη στην αρχή του κώδικα δηλαδή,

```
#include <SSC32.h>
```

Με την βιβλιοθήκη μπορεί να σταλεί από τον Arduino το κατάλληλο σήμα στον ελεγκτή SSC-32 ώστε να ελέγχει τους σερβοκινητήρες. Παρακάτω αναφέρονται κάποιες από τις πιο χρησιμοποιημένες χρήσιμες εντολές.

```
servoMoveTime(int channel, int position, int time);
```

Με αυτήν την εντολή έχουμε την δυνατότητα να ελέγξουμε τον σερβοκινητήρα ανάλογα με τον χρόνο που θέλουμε να πραγματοποιηθεί η τροχιά της κίνησης π.χ.

```
servoMoveTime(3, 1200, 1500);
```

Στο παραπάνω παράδειγμα ο σερβοκινητήρας στο κανάλι 3 θα περιστρέφει στην θέση που αντιστοιχεί σε εύρος παλμών 1200 *μsec* σε χρόνο 1500 *msec*.

Για να γίνει η κίνηση του σερβοκινητήρα πρέπει οι υπόλοιπες σερβοκινητήρες να μην κάνουν οποιαδήποτε άλλη ενέργεια. Για την ταυτόχρονη διαχείριση των σερβοκινητήρων χρησιμοποιείτε η εντολή,

```
beginGroupCommand(int type);
```

Με αυτήν την εντολή ορίζεται το σημείο στον κώδικα ώστε, μετά από αυτό το σημείο η κίνηση των σερβοκινητήρων να γίνεται ταυτόχρονα και όχι μεμονωμένα. Με την μεταβλητή “int type” ορίζουμε τον τύπο διαχείρισης των σερβοκινητήρων. Χρησιμοποιούμε τον τύπο “SSC32\_CMDGRP\_TYPE\_SERVO\_MOVEMENT” ή τον ακέραιο αριθμό 1 ώστε, ο έλεγχος των σερβοκινητήρων να γίνεται μέσω της εντολής “servoMoveTime()”.

```
abortGroupCommand();
```

Με αυτή την εντολή μπορούμε να διακόψουμε την εντολή “beginGroupCommand()”

```
endGroupCommand();
```

Ενώ με αυτήν την εντολή ορίζεται το τέλος της εντολής “beginGroupCommand()” π.χ.

```
beginGroupCommand(SSC32_CMDGRP_TYPE_SERVO_MOVEMENT);
```

```
servoMoveTime(3, 1200, 1200);
```

```
servoMoveTime(4, 2000, 1200);
```

```
endGroupCommand();
```

Στο παραπάνω παράδειγμα ο σερβοκινητήρας στο κανάλι 3 θα κινηθεί στην θέση 1200 *μsec* ενώ ο σερβοκινητήρας στο κανάλι 4 θα κινηθεί στην θέση 2000 *μsec*. Οι κινήσεις εκκινούν και ολοκληρώνονται στον ίδιο χρόνο δηλαδή, σε 1200 *msec*, σε όποια θέση και αν βρίσκονταν πριν οι σερβοκινητήρες.

Για τον χειρισμό των δαχτύλων του ρομποτικού χεριού ΤΑΛΟΣ χρησιμοποιείται ο παρακάτω μαθηματικός τύπος που μετατρέπει την επιθυμητή τιμή της γωνίας, των αρθρώσεων των

δαχτύλων του ρομποτικού χεριού, στο κατάλληλο εύρος παλμού που πρέπει να δοθεί στον σερβοκινητήρα της συγκεκριμένης άρθρωσης [22].

$$PWM_{q_i} = \text{round} \left\{ a - \frac{(a-b)}{90^\circ} \cdot q_i \right\} \quad (5.1)$$

Όπου  $a$ ,  $b$ , οι τιμές του παλμού ( $PWM$ ) προκειμένου να βρίσκεται η άρθρωση στις γωνίες  $0^\circ$  και  $90^\circ$ , αντίστοιχα. Με τον δείκτη  $i$  συμβολίζεται ο αριθμός της άρθρωσης του ρομποτικού δαχτύλου, ενώ η μεταβλητή  $q$  είναι η επιθυμητή γωνία στροφής της άρθρωσης. Σημαντικό είναι να ειπωθεί ότι η γωνίες των  $0^\circ$  και  $90^\circ$  δεν είναι οι γωνίες των σερβοκινητήρων αλλά οι γωνίες που επιθυμούμε για την άρθρωση.

Παρακάτω (Πίνακας 5.1) φαίνεται σε ποιες θύρες του ελεγκτή SSC32 έχει συνδεθεί ο κάθε κινητήρας (κανάλι).

Πίνακας 5.1. Οι αρθρώσεις κάθε δαχτύλου και το κανάλι που τους αντιστοιχούν στον SSC32.

Δάχτυλο	Δείκτης			Μέσος			Παράμεσος			Μικρός			Αντίχειρας			
	q <sub>2</sub>	q <sub>3</sub>	q <sub>4</sub>	q <sub>2</sub>	q <sub>3</sub>	q <sub>4</sub>	q <sub>2</sub>	q <sub>3</sub>	q <sub>4</sub>	q <sub>2</sub>	q <sub>3</sub>	q <sub>4</sub>	q <sub>1</sub>	q <sub>2</sub>	q <sub>3</sub>	q <sub>4</sub>
<b>Αρθρώσεις</b>	q <sub>2</sub>	q <sub>3</sub>	q <sub>4</sub>	q <sub>2</sub>	q <sub>3</sub>	q <sub>4</sub>	q <sub>2</sub>	q <sub>3</sub>	q <sub>4</sub>	q <sub>2</sub>	q <sub>3</sub>	q <sub>4</sub>	q <sub>1</sub>	q <sub>2</sub>	q <sub>3</sub>	q <sub>4</sub>
<b>Pin-SSC32</b>	15	14	13	12	11	10	9	8	7	16	17	18	31	30	29	28

Παρακάτω (Πίνακας 5.2, Πίνακας 5.3, Πίνακας 5.4, Πίνακας 5.5, Πίνακας 5.6) φαίνεται το εύρος παλμού του κάθε κινητήρα, για κάθε δάχτυλο, που έχει στις  $0^\circ$  και  $90^\circ$ . Άξιο αναφοράς είναι ότι ο κινητήρας που κινεί την άρθρωση  $q_1$  του αντίχειρα δεν περιστρέφεται από  $0^\circ$  έως  $90^\circ$ , αλλά από  $180^\circ$  έως  $270^\circ$ .

Πίνακας 5.2. Η βαθμονόμηση των τριών σερβοκινητήρων του Δείκτη.

Αρθρώσεις Δείκτη	q <sub>2</sub>		q <sub>3</sub>		q <sub>4</sub>	
<b>Μοίρες</b>	0°	90°	0°	90°	0°	90°
<b>Εύρος Παλμού για το σήμα εισόδου του κινητήρα</b>	2050	1050	2100	1050	2050	1000

Πίνακας 5.3. Η βαθμονόμηση των τριών σερβοκινητήρων του Μέσου.

Αρθρώσεις Μέσου	q <sub>2</sub>		q <sub>3</sub>		q <sub>4</sub>	
<b>Μοίρες</b>	0°	90°	0°	90°	0°	90°
<b>Εύρος Παλμού για το σήμα εισόδου του κινητήρα</b>	1800	800	2000	950	2190	1190

Πίνακας 5.4. Η βαθμονόμηση των τριών σερβοκινητήρων του Παράμεσου.

Αρθρώσεις Παράμεσου	$q_2$		$q_3$		$q_4$	
Μοίρες	0°	90°	0°	90°	0°	90°
Εύρος Παλμού για το σήμα εισόδου του κινητήρα	2000	1030	2080	1030	1990	940

Πίνακας 5.5. Η βαθμονόμηση των τριών σερβοκινητήρων του Μικρού.

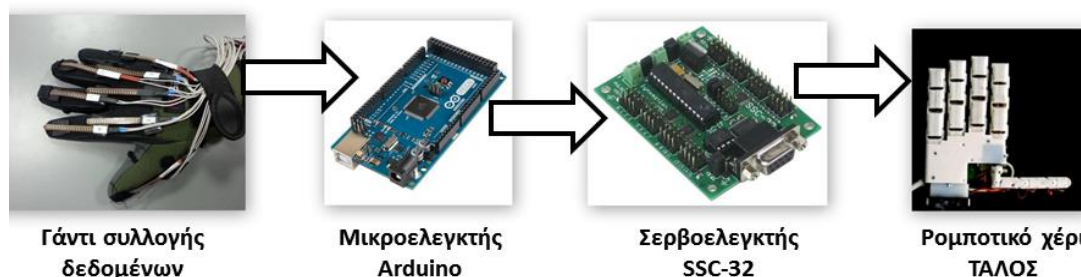
Αρθρώσεις Μικρού	$q_2$		$q_3$		$q_4$	
Μοίρες	0°	90°	0°	90°	0°	90°
Εύρος Παλμού για το σήμα εισόδου του κινητήρα	1950	990	2000	950	1970	950

Πίνακας 5.6. Η βαθμονόμηση των τεσσάρων σερβοκινητήρων του Αντίχειρα.

Αρθρώσεις Αντίχειρα	$q_1$		$q_2$		$q_3$		$q_4$	
Μοίρες	180°	270°	0°	90°	0°	90°	0°	90°
Εύρος Παλμού για το σήμα εισόδου του κινητήρα	1330	2270	650	500	2110	1100	2000	950

## 5.4. Ενσύρματη διασύνδεση

Το σύστημα ελέγχου για την ενσύρματη επικοινωνία αποτελείται, από το γάντι συλλογής δεδομένων, τον μικροελεγκτή Arduino, τον ελεγκτή SSC-32 και το ρομποτικό χέρι (Εικόνα 5.5).



Εικόνα 5.5. Διάγραμμα ροής συστήματος ελέγχου.

Ο μικροελεγκτής συλλέγει τα δεδομένα από το γάντι συλλογής δεδομένων δηλαδή, καταγράφει τις τάσεις των εύκαμπτων αισθητηρίων τις οποίες αντιστοιχεί σε γωνίες των αρθρώσεων των δαχτύλων με βάσει της σχέσης (4.4). Άξιο αναφοράς είναι ότι για τον χειρισμό των δαχτύλων, δείκτη, μέσω, παράμεσο και μικρό, η περιστροφή της 4<sup>ης</sup> ( $q_4$ ) άρθρωσης καθορίζεται μέσω της σχέσης (3.1). Ενώ, για τις υπόλοιπες δύο αρθρώσεις

χρησιμοποιούνται τα δεδομένα από τους αισθητήρες. Στην συνέχεια χρησιμοποιείται η σχέση (5.1), ώστε να υπολογιστεί το εύρος του σήματος PWM το οποίο θα χρησιμοποιήσει ο σερβοελεγκτής

SSC-32 για να κινήσει την αντίστοιχη άρθρωση του ρομποτικού χεριού. Ο ελεγκτής είναι συνδεδεμένος με τον Arduino μέσω σειριακής θύρας [27].

#### 5.4.1. Διασύνδεση με ελεγκτή SSC-32

Αρχικά προγραμματίζεται ο μικροελεγκτής Arduino. Στην αρχή του κώδικα δηλώνονται οι μεταβλητές που χρησιμοποιούνται στην συνέχεια, ρυθμίζεται η σειριακή επικοινωνία του μικροελεγκτή με τον ελεγκτή SSC-32 (Void setup). Το πρόγραμμα που εκτελείται σε συνεχή επανάληψη βρίσκεται στην συνάρτηση “void loop”. Ο μικροελεγκτής συλλέγει τα δεδομένα και αποθηκεύονται στην μεταβλητή “flxval” σε τιμές (0 έως 1023). Στην συνέχεια, τα δεδομένα μετατρέπονται από 0 έως 1023 σε 0V έως 5V μέσω της σχέση  $Q = x * 5,00/1023$ , όπου  $Q$  η τιμή της τάσης και  $x$  η τιμή της μεταβλητής (0 – 1023). Έπειτα, μέσω τις σχέσεις (4.4) οι τιμές μετατρέπονται σε μοίρες (0° έως 90°) ενώ, μέσω της σχέσης (3.1) καθορίζεται η περιστροφή της  $q_4$  (sscddeg13). Με την εντολή “conststrain” ορίζεται το διάστημα τιμών της “sscddeg” να είναι 0 έως 90. Στην μεταβλητή “p0” ορίζεται η τιμή του παλμού που πρέπει να δοθεί στον ελεγκτή ώστε να περιστραφεί στην επιθυμητή γωνία (σχέση (5.1)). Τέλος, ρυθμίζεται η ομαδική εκκίνηση των κινητήρων (στο παρών παράδειγμα υπάρχουν τρεις κινητήρες διότι, η διαδικασία προγραμματισμού των υπόλοιπων είναι ίδια) με την εντολή “myscc.beginGroupCommand (SSC32\_CMDGRP\_TYPE\_SERVO\_MOVEMENT)”, ενώ η εντολή “myscc.servoMoveTime(15,p15,t)” στέλνει την κατάλληλη εντολή στον ελεγκτή ώστε να εκκινήσει των σερβοκινητήρα που βρίσκεται στο κανάλι 15, να περιστρέφει στην θέση με εύρος παλμού “p15” και ολοκλήρωσης της τροχιάς σε 500 msec. Η εντολή “myscc.endGroupCommand()” σταματάει την ομαδική κίνηση ώστε, το πρόγραμμα να ξεκινήσει από την αρχή και να διαβάσει την καινούρια τιμή της τάση. Πρέπει να σημειωθεί ότι, το “Delay” πρέπει να έχει την ίδια τιμή με τον χρόνο ολοκλήρωσης της τροχιάς των σερβοκινητήρων δηλαδή, 500 ή περισσότερο. Ο παρών κώδικας ισχύει για 2 αισθητήρες δηλαδή, για την περιστροφή των σερβοκινητήρων του δείκτη. Για την να επέκταση του κώδικα, ώστε να χειρίζεται και τους 16 κινητήρες, γράφεται η εντολή “myscc.servoMoveTime(ch,p,t)” για κάθε κινητήρα ενδιάμεσα του “myscc.beginGroupCommand()” και “myscc.endGroupCommand()”.

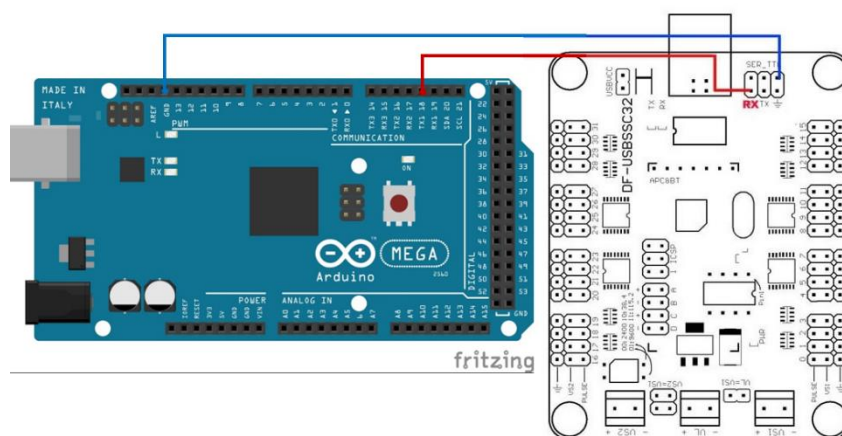
```
#include <SSC32.h>
// Defining the SSC32 board
```

```
SSC32 myssc = SSC32();
// Defining the flex sensor pin
int flx1pin = 0;
int flx2pin = 0;
// Initialization of the flex sensor values
int flx1val = 0;
int flx2val = 0;
// Initialization of the degree values
int sscdeg15 = 0;
int sscdeg14 = 0;
int sscdeg13 = 0;
// Initialization of the pulse values
int p15 = 0;
int p14 = 0;
int p13 = 0;
int t = 500;
void setup() {
    // Start communication with the SSC32 board Serial port 1
    //(Pins 19(RX),18(TX)) Arduino Mega 2560 instead of pins 0,1
    myssc.begin(9600);
}
void loop() {
    // Read the analog values of the flex sensor
    flx1val = analogRead(flx1pin)*5.00/1023;
    flx2val = analogRead(flx2pin)*5.00/1023;
    // Mapping of the flex sensor to degrees
    sscdeg15 = round((flx1val-3.20)*90/(5.00-3.20));
    sscdeg14 = round((flx2val-3.05)*90/(5.00-3.05));
    // Human hand constrain (eq. 3.1)
    sscdeg13 = round(sscdeg14*2/3);
    // Defining the minimum and maximum values, respectively
    sscdeg15 = constrain(sscdeg15,0,90);
    sscdeg14 = constrain(sscdeg14,0,90);
    sscdeg13 = constrain(sscdeg13,0,90);
    // Function for converting degrees to pulse
    p15 = 2050-(sscdeg15*((2050-1050)/90));
    p14 = 2100-(sscdeg14*((2100-1050)/90));
    p13 = 2050-(sscdeg13*((2050-1000)/90));
    // Function for group command of the servos. With this
    // command the servos move at the same time and reach the
```



```
// desired degrees.  
// 'SSC32_CMDGRP_TYPE_SERVO_MOVEMENT' is the type of group  
// command  
myssc.beginGroupCommand(SSC32_CMDGRP_TYPE_SERVO_MOVEMENT);  
myssc.servoMoveTime(15,p15,t);  
myssc.servoMoveTime(14,p14,t);  
myssc.servoMoveTime(13,p13,t);  
// Defining the end of group command  
myssc.endGroupCommand();  
delay(500);  
}
```

Ο σερβοελεγκτής συνδέεται στα pin 18 (TX), 19 (RX) και GND του Arduino (Εικόνα 5.6).



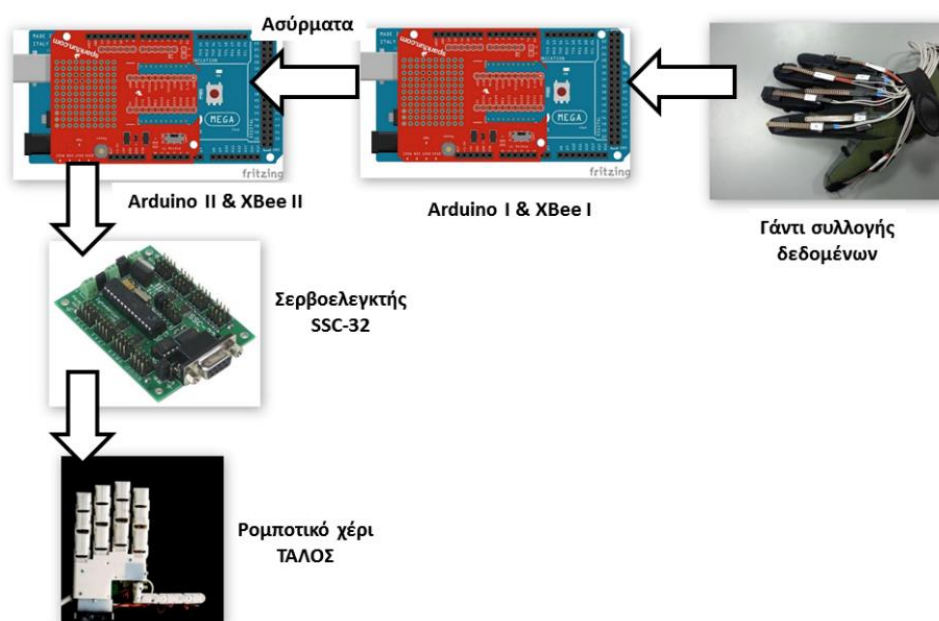
Εικόνα 5.6. Σχηματική αναπαράσταση σύνδεσης ελεγκτή SSC - 32 με Arduino.

Μέσω της ενσύρματης επικοινωνίας έχουμε γρήγορη μετάδοση των δεδομένων αλλά περιορισμό στην απόσταση χειρισμού του ανθρωπόμορφου ρομποτικού χεριού. Στην συνέχεια, γίνεται προσπάθεια τηλεχειρισμού του ρομποτικού χεριού, ώστε να έχουμε απομακρυσμένο έλεγχο.

## 5.5. Ασύρματη διασύνδεση

Το σύστημα που χρησιμοποιείται για την ασύρματη επικοινωνία αποτελείται από το σύστημα καταγραφή της κίνησης, 2 μικροελεγκτές Arduino, 2 πλακέτες XBee (στην παρούσα πτυχιακή χρησιμοποιείται το XBee 805.15.4 series 1), τον ελεγκτή SSC-32 και το ρομποτικό χέρι ΤΑΛΟΣ (Εικόνα 5.7).





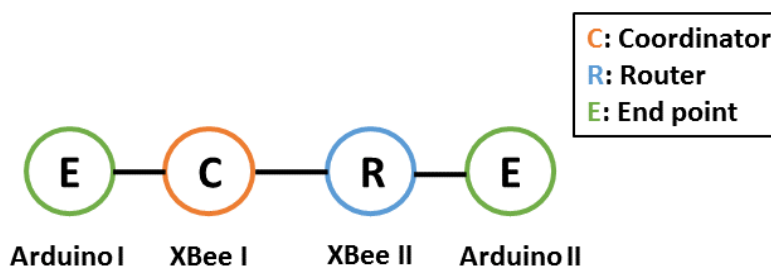
Εικόνα 5.7. Διάγραμμα ροής ασύρματης επικοινωνίας συλλογής δεδομένων και αξιοποίησή τους.

Ο μικροελεγκτής συλλέγει τα δεδομένα από το γάντι συλλογής δεδομένων και τα στέλνει ασύρματα μέσω του XBee I στο XBee II, που είναι συνδεδεμένο με τον Arduino II. Στην συνέχεια, ο Arduino II επεξεργάζεται τα δεδομένα ώστε να στείλει το κατάλληλο σήμα στον ελεγκτή SSC-32, όπου με την σειρά του χειρίζεται τους σερβοκινητήρες του ρομποτικού χεριού. Πρέπει να σημειωθεί ότι στην εικόνα 5.7 δεν απεικονίζεται τα XBee να είναι συνδεδεμένα με τον μικροελεγκτή, αλλά πρέπει να είναι συνδεδεμένα κανονικά ώστε να λειτουργεί το σύστημα.

### 5.5.1. Ασύρματο δίκτυο

Το XBee είναι ένας ασύρματος μικροελεγκτής κατασκευασμένος από το Zigbee [28]. Το πρωτόκολλο επικοινωνίας που χρησιμοποιεί το XBee, για την επικοινωνία του, είναι το IEEE 802.14.5 [29]. Διαθέτει 11 pin εισόδου/εξόδου, από τα οποία τα τέσσερα είναι αναλογικά pin. Η επικοινωνία μεταξύ δύο ή περισσότερων XBee γίνεται ορίζοντας ένα δίκτυο. Το δίκτυο αποτελείται από:

- Συντονιστή (*Coordinator*)
- Δρομολογητή (*Router*)
- Τελικό σημείο (*End point*)



Εικόνα 5.8. Σχηματική αναπαράσταση ασύρματου δικτύου.

Το τελικό σημείο δέχεται δεδομένα από τον δρομολογητή ή τον συντονιστή αλλά δεν μπορεί να στείλει δεδομένα (π.χ. αισθητήρας, Arduino): μπορούν να υπάρχουν πολλά τελικά στοιχεία στο δίκτυο. Ο δρομολογητής είναι το στοιχείο που επικοινωνεί με τα υπόλοιπα στο δίκτυο δηλαδή, επεκτείνει το εύρος του δικτύου (π.χ. XBee ή άλλος ασύρματος μικροελεγκτής) · μπορούν να υπάρχουν πολλοί δρομολογητές στο δίκτυο και πρέπει να λειτουργούν συνέχεια αλλιώς υπολειτουργεί το δίκτυο. Ο Συντονιστής είναι το στοιχείο που δημιουργεί το δίκτυο και ταυτόχρονα μπορεί να στείλει δεδομένα (π.χ. XBee ή άλλος ασύρματος μικροελεγκτής) · πρέπει να υπάρχει μόνο ένα στο δίκτυο και επίσης πρέπει να λειτουργεί συνέχεια αλλιώς δεν λειτουργεί το δίκτυο.

### 5.5.2. Υλοποίηση ασύρματου δικτύου μέσω του XBee

Για να συνδεθεί η πλακέτα του XBee στον Η/Υ θα πρέπει να χρησιμοποιηθούν οι πλακέτες σύνδεσης (*XBee explorer*) (Εικόνα 5.9, Εικόνα 5.10, Εικόνα 5.11). Υπάρχουν τρία είδη XBee explorer. [30]

- XBee Explorer USB



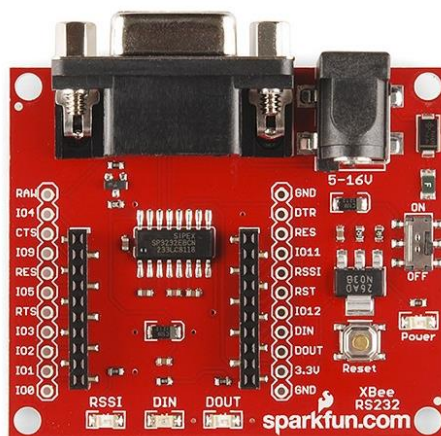
Εικόνα 5.9. Πλακέτα σύνδεσης XBee Explorer USB για σύνδεση του XBee με Η/Υ .

- XBee Explorer Dongle



Εικόνα 5.10. Παράδειγμα σύνδεσης του XBee Explorer Dongle με τον υπολογιστή.

- UartS Bee V4



Εικόνα 5.11. Πλακέτα σύνδεσης UartS Bee V4 για σύνδεση του XBee με Η/Υ .

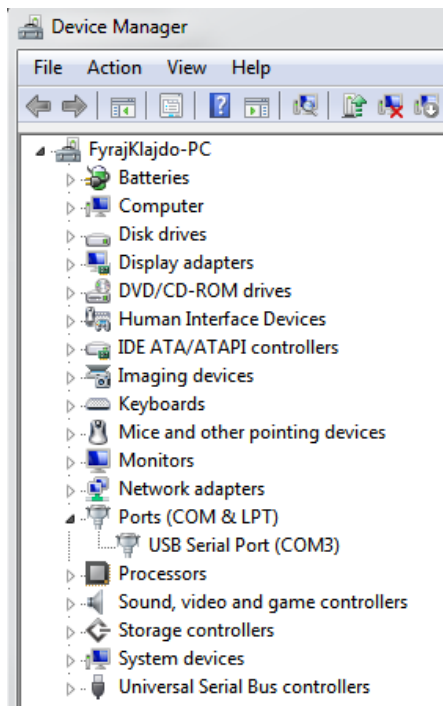
Όλες οι παραπάνω πλακέτες συνδέσεις συνδέουν το XBee με τον Η/Υ ώστε να επικοινωνεί με σειριακή επικοινωνία. Η διαφορά τους είναι ότι το XBee explorer USB συνδέεται κατευθείαν στον Η/Υ, η πλακέτα XBee explorer dongle συνδέει την πλακέτα σύνδεσης με τον Η/Υ μέσω καλωδίου micro-usb· ενώ, η πλακέτα UartS Bee V4 συνδέει την πλακέτα σύνδεσης με τον Η/Υ μέσω καλωδίου RS232.

#### 5.5.2.1. Πρόγραμμα Οδήγησης του XBee

Για να μπορέσει ο υπολογιστής να αναγνωρίσει την πλακέτα XBee, θα πρέπει να εγκατασταθούν τα προγράμματα οδήγησης του XBee· όπου βρίσκονται στον παρακάτω σύνδεσμο. Πρέπει να σημειωθεί ότι το πρόγραμμα οδήγησης είναι για την πλακέτα XBee / XBee-PRO 802.15.4 Modules.

<http://www.digi.com/support/productdetail?pid=3257&type=drivers>

Για να επιβεβαιωθεί η σύνδεση του XBee με τον Η/Υ χρησιμοποιείτε η εφαρμογή “Διαχείριση Συσκευών” (*Device Manager*) (Εικόνα 5.12), όπου φαίνεται η θύρα σύνδεσης του XBee (USB Serial Port (COM3)).

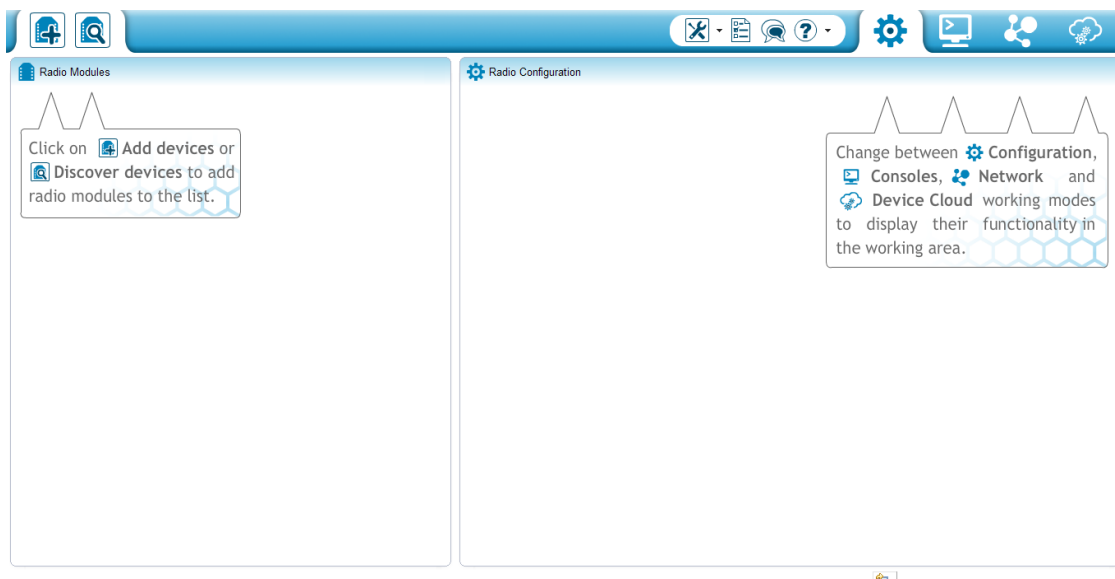


Εικόνα 5.12. Διαχείριση Συσκευών.


#### **5.5.2.2. Διαχείριση του XBee μέσω του Λογισμικού XCTU (Next Gen.)**

Το λογισμικό XCTU είναι ένα χρήσιμο εργαλείο για την διαχείριση της συσκευής XBee (Εικόνα 5.13). Μέσω του λογισμικού XCTU μπορεί να αναβαθμιστεί το λογισμικό (*Firmware*) της πλακέτας XBee, να γίνει έλεγχος σύνδεσης και ρύθμιση του δικτύου ανάμεσα στα XBee. Στον παρακάτω σύνδεσμο βρίσκεται το λογισμικό XCTU. Πρέπει να τονίσουμε ότι κάθε ένα XBee που χρησιμοποιείται στο δίκτυο θα πρέπει να ρυθμιστεί με βάση το παραπάνω πρόγραμμα.

<http://www.digi.com/support/productdetail?pid=3352>



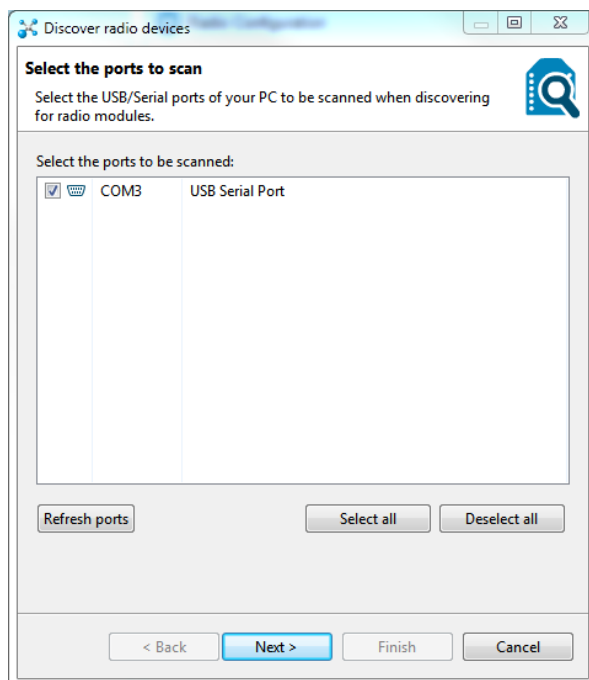
Εικόνα 5.13. Λογισμικό XCTU.

Αρχικά πρέπει να βρεθεί η συσκευή του XBee που είναι συνδεδεμένη με τον Η/Υ. Πατώντας την εντολή “Discover devices”  (Εικόνα 5.14).



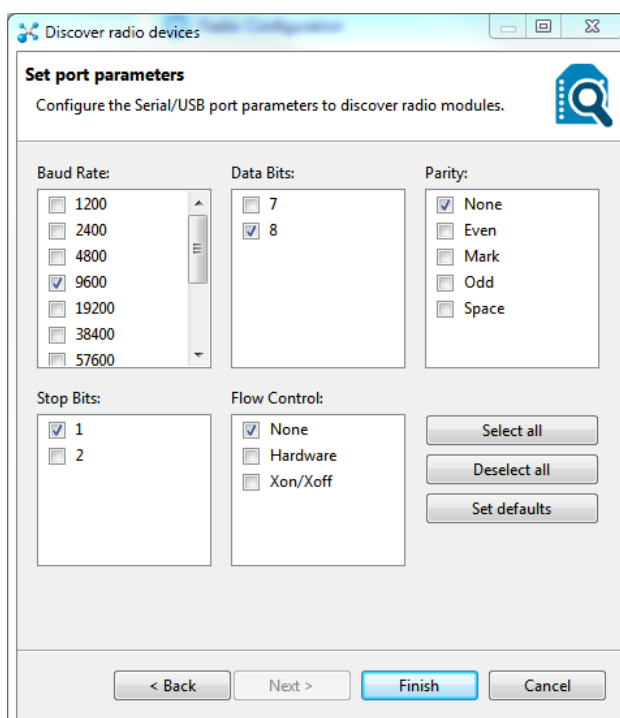
Εικόνα 5.14. Λογισμικό XCTU εντολή "Discover devices".

Στο επόμενο παράθυρο εμφανίζονται οι συσκευές που είναι συνδεδεμένες στον Η/Υ (Εικόνα 5.15).



Εικόνα 5.15. Αναγνώριση της σύνδεσης των XBee από το λογισμικό XCTU.

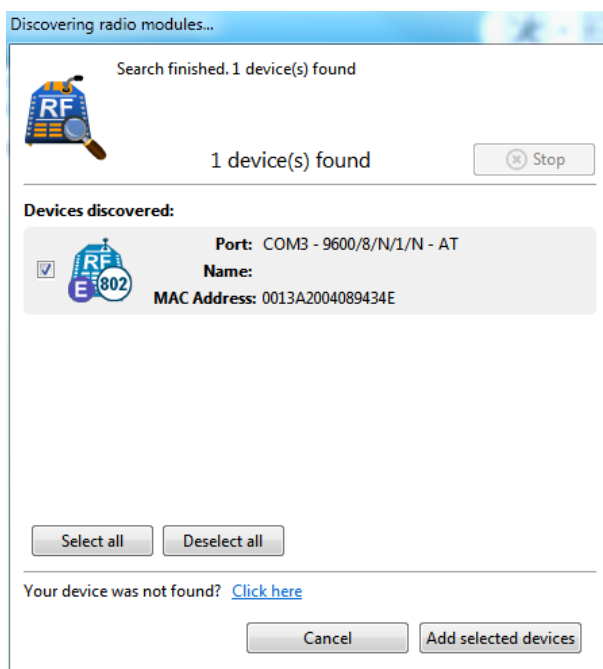
Πατώντας “Next” εμφανίζεται καινούριο παράθυρο (Εικόνα 5.16) όπου μπορούμε να ρυθμίσουμε τις παραμέτρους σύνδεσης της σειριακής επικοινωνίας του XBee με τον Η/Υ.



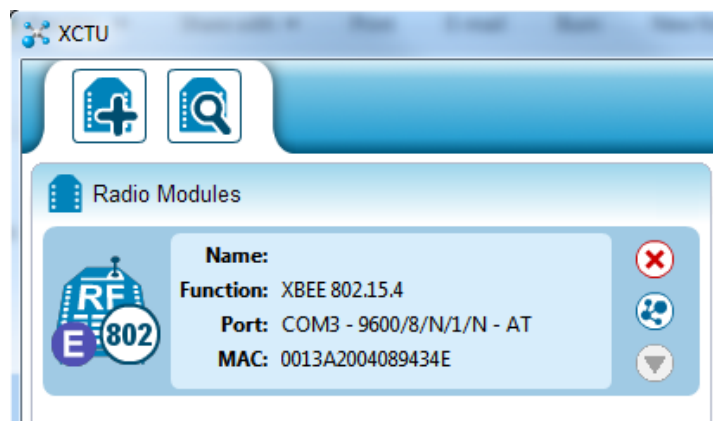
Εικόνα 5.16. Παράθυρο ρύθμισης παραμέτρων σειριακής επικοινωνίας του XBee με Η/Υ.

Πατάμε “Finish” ώστε να ολοκληρωθεί η διαδικασία. Στην συνέχεια, εμφανίζεται το παράθυρο (Εικόνα 5.17) όπου, δηλώνει την συσκευή που αναγνωρίστηκε και ρυθμίστηκαν οι

παράμετροι για την σύνδεση με τον Η/Υ. Πατάμε “Add selected devices” ώστε να εισάγουμε την συσκευή.



Εικόνα 5.17. Παράθυρο εμφανίσεις συσκευών σύνδεσης έπειτα από την ολοκλήρωση των ρυθμίσεων. Επιλέγουμε την συσκευή (Εικόνα 5.18) ώστε να εμφανιστούν οι παράμετροι του δικτύου (Εικόνα 5.19).

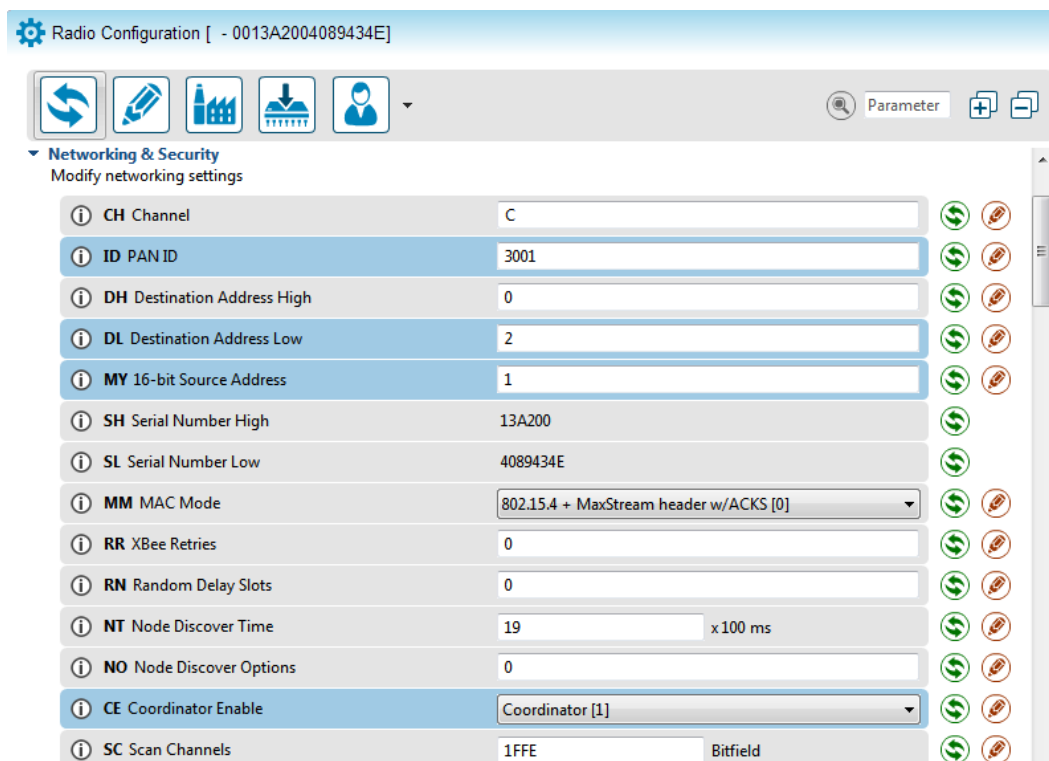


Εικόνα 5.18. Καταχωρημένη συσκευή στο λογισμικό XCTU.


Στην συνέχεια απαιτείται να ρυθμιστεί το δίκτυο που θα επικοινωνούν τα XBee μεταξύ τους (Εικόνα 5.19). Οι παράμετροι που πρέπει να ρυθμιστούν είναι:

- CH (*Channel*),
- ID (*Pan id*)
- DH (*Destination address high*)

- DL (*Destination address low*)
- MY (*16-bit source address*)
- CE (*Coordinator Enable*)



Εικόνα 5.19. Παράμετροι πλακέτας XBee.

Η παράμετρος CH ορίζει το κανάλι του δικτύου ενώ η παράμετρος ID ρυθμίζει την διεύθυνση του συντονιστή του δικτύου (*coordinator*). Με την παράμετρο DL ρυθμίζουμε την διεύθυνση που θα στέλνει δεδομένα ενώ, με την παράμετρο MY ρυθμίζουμε την διεύθυνση που θα λαμβάνει δεδομένα το συγκεκριμένο XBee. Τέλος, η παράμετρος DH πρέπει να οριστεί μηδέν ώστε τα XBee να επικοινωνούν με διεύθυνσης (DL) τύπου 16-bit και όχι 32-bit. Η παράμετρος CE ορίζει τον συντονιστή του δικτύου. Για την καταχώρηση των αλλαγών πατάμε το .

Με αντίστοιχη διαδικασία ρυθμίζουμε τις παραμέτρους επικοινωνίας του XBee II. Παρακάτω φαίνονται οι παράμετροι για τα δύο XBee.

XBee I:

- Channel (CH): C
- PAN ID (ID): 3001
- Address (MY): 1
- Destination address high (DH): 0



- Destination address low (DL): 2
- Coordinator Enable (CE): Coordinator [1]

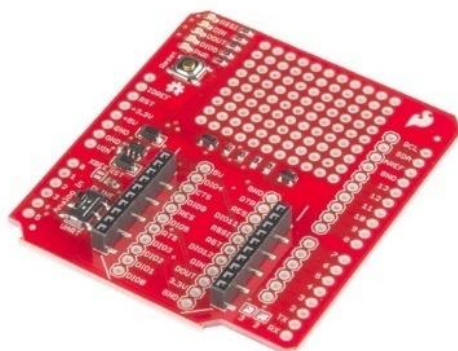
XBee II:

- Channel (CH): C
- PAN ID (ID): 3001
- Address (MY): 2
- Destination address high (DH): 0
- Destination address low (DL): 1

Οι παράμετροι DL και MY του XBee II έχουν οριστεί βάσει του XBee I δηλαδή, η DL παράμετρος του XBee I πρέπει να είναι ίδια με την παράμετρο MY του XBee II. Αυτό ισχύει και για το XBee II δηλαδή, η παράμετρος DL του XBee II πρέπει να είναι ίδια με την παράμετρο MY του XBee I.

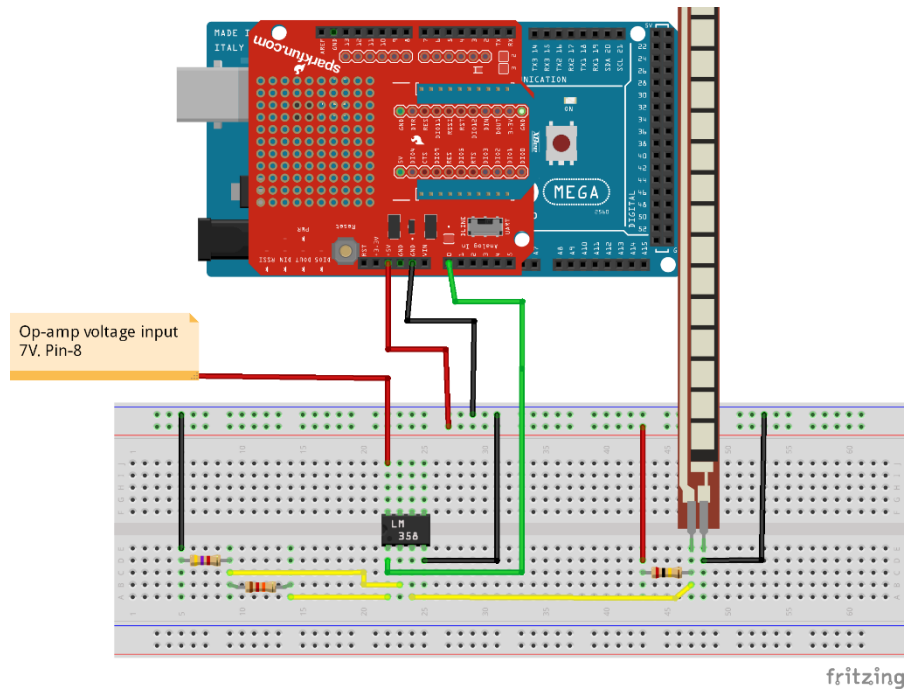
### 5.5.3. Διασύνδεση XBee με μικροελεγκτή

Αφού ρυθμιστούν τα XBee μπορούν να συνδεθούν με διάφορους μικροελεγκτές, όπως είναι ο Arduino. Η τοποθέτηση του XBee στον Arduino γίνεται μέσω της πλακέτας σύνδεσης (*XBee Shield*), μέσω της πλακέτας σύνδεσης το XBee συνδέεται στα κατάλληλα pin του Arduino, GND, 3,3V και RX0, TX0 (pin 0, 1). Οι υπόλοιπες είσοδοι μπορούν χρησιμοποιηθούν κανονικά (Εικόνα 5.20) [31].



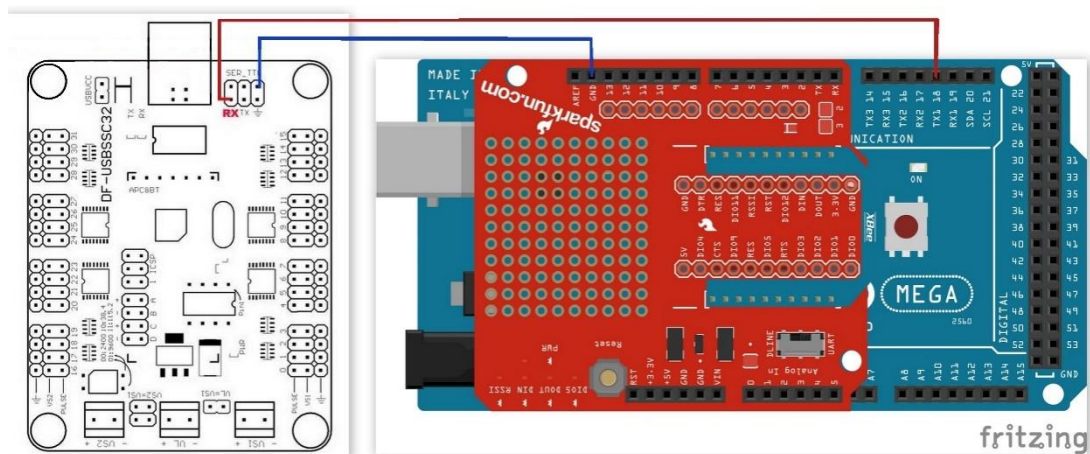
Εικόνα 5.20. Πλακέτα σύνδεσης του XBee με τον Arduino.

Στην Εικόνα 5.21 απεικονίζεται το κύκλωμα καταγραφής δεδομένων από ένα εύκαμπτο αισθητήρα για την ασύρματη αποστολή τους μέσω του XBee.



Εικόνα 5.21. Κύκλωμα καταγραφής δεδομένων ασύρματα από τον Arduino I και XBee I (Στην εικόνα το XBee δεν απεικονίζεται να είναι συνδεδεμένο στην πλακέτα σύνδεσης, αλλά πρέπει να συνδεθεί κανονικά. Αυτό ισχύει και για τις υπόλοιπες εικόνες).

Η συλλογή των δεδομένων γίνεται από τον Arduino II που είναι συνδεδεμένος με τον σερβοελεγκτή SSC-32 (Εικόνα 5.22), ο οποίος με την σειρά του ελέγχει τους σερβοκινητήρες.



Εικόνα 5.22. Συνδεσμολογία αποδέκτη για ασύρματη καταγραφή δεδομένων.

Τα XBee χρησιμοποιούν πάντα (εάν χρησιμοποιηθεί το XBee shield) την σειριακή θύρα (RX0, TX0 (pin 0, 1)) προκειμένου να επικοινωνήσουν με τον Arduino.

### 5.5.3.1. Προγραμματισμός μικροελεγκτή Arduino

Αρχικά γίνεται ο προγραμματισμός του Arduino I από τον Η/Υ και στην συνέχεια τοποθετείται η πλακέτα του XBee πάνω σε αυτόν. Για τον προγραμματισμό χρησιμοποιήθηκε ο παρακάτω κώδικας, όπου αρχικά ρυθμίζεται η σειριακή θύρα (RX0, TX0) για την επικοινωνία με το XBee με ρυθμό 9600 baud. Ο κώδικας που εκτελείται συνεχώς είναι ανάμεσα, της “void loop ()”. Στην αρχή ο Arduino διαβάζει τις τιμές του αισθητήρα (0 – 1023) και τις μετατρέπει σε τιμές της τάσης (0 - 5V), βάσει της σχέσης  $Q = x \cdot 5,00/1023$  (όπου  $x$  η τιμή του αισθητήρα). Στην συνέχεια, οι τιμές μετατρέπονται σε μοίρες, βάσει της σχέσης (4.4), έπειτα με την εντολή “constrain” ορίζουμε το ελάχιστο και μέγιστο όριο στην μεταβλητή (sscddeg) να είναι από 0 έως 90. Τέλος, τα δεδομένα στέλνονται σειριακή στο XBee I ώστε, να τα στείλει ασύρματα στο XBee II, κάθε 600 msec. Θα πρέπει να σημειώσουμε ότι ενδιάμεσα στα δεδομένα θα πρέπει να σταλεί ένας μη ακεραίος χαρακτήρας (π.χ. «,», «.», «\*»), ώστε ο Arduino II να μπορέσει να ξεχωρίσει από το μήνυμα τις τιμές για κάθε αισθητήρα. Το πρόγραμμα αφορά μόνο δύο αισθητήρες και μπορεί να επεκταθεί εύκολα προκειμένου να συμπεριλάβει περισσότερους αισθητήρες.

```
// Defining the sensor pin
int flx1pin = A0;
int flx2pin = A1;
// Initializing the sensor value
int flx1val = 0;
int flx2val = 0;
int sscdeg15 = 0;
int sscdeg14 = 0;
int sscdeg13 = 0;
// Time of delay
int t = 600;
void setup(){
// Starts the serial communication
Serial.begin(9600);
// Sets the mode of the flex sensor pin to input
pinMode(flx1pin, INPUT);
pinMode(flx2pin, INPUT);
}
void loop(){
// Reads the analog values of the flex sensor
flx1val = analogRead(flx1pin)*5.00/1023;
```

```
flx2val = analogRead(flx2pin)*5.00/1023;
// Converts analog values(0-1023) to degrees
sscdeg1 = round((flx1val-3.20)*90/(5.00-3.20));
sscdeg2 = round((flx2val-3.05)*90/(5.00-3.05));
// Human hand constrain (eq. 3.1)
sscdeg3 = round(sscdeg2*2/3);
// Constrains the values from 0-90 degrees
sscdeg15 = constrain(sscdeg15,0,90);
sscdeg14 = constrain(sscdeg14,0,90);
sscdeg13 = constrain(sscdeg13,0,90);
// Prints the analog values with serial communication
Serial.print(sscdeg15);
Serial.print(",");
Serial.print(sscdeg14);
Serial.print(",");
Serial.print(sscdeg13);
Serial.print('\n');
// Wait for the process of the receiver to complete
delay(t);
}
```

Στην συνέχεια προγραμματίζεται ο Arduino II (αποδέκτης) ώστε να διαβάζει τα δεδομένα από το XBee II, δηλαδή τις τιμές από τους αισθητήρες, ώστε να στείλει την κατάλληλη εντολή στον σερβοελεγκτή SSC-32. Αρχικά ο μικροελεγκτής ελέγχει άμα εισέρχονται δεδομένα μέσω της σειριακής θύρας ("Serial.available()") που είναι συνδεδεμένο το XBee II. Στην συνέχεια ο μικροελεγκτής συλλέγει τα δεδομένα από το XBee II σε μορφή ακέραιου (parseInt) (η parseInt συλλέγει έναν ακέραιο αριθμό μέχρις ότου συναντήσει έναν διαφορετικό χαρακτήρα μη ακέραιο) [32]. Τα δεδομένα αποθηκεύονται στην μεταβλητή "sscdeg15" και "sscdeg14" όπου μέσω της σχέσης (3.1) καθορίζεται η περιστροφή της άρθρωσης  $q_4$ . Έπειτα, ο μικροελεγκτής ελέγχει αν ο τελευταίος χαρακτήρας που έχει σταλεί είναι το enter ('\n') αλλιώς ξαναδιαβάζει τα δεδομένα. Τέλος τα δεδομένα μετατρέπονται σε εύρος παλμού (σχέση (5.1)) ώστε να σταλούν στους κινητήρες. Ο κώδικας μπορεί να αναπτυχθεί εύκολα για περισσότερους κινητήρες και αισθητήρες απλά προσθέτουμε περισσότερες μεταβλητές. Ο συγκεκριμένος κώδικας ελέγχει τους κινητήρες του δείκτη (κανάλι 15, 14, 13 στον SSC – 32).

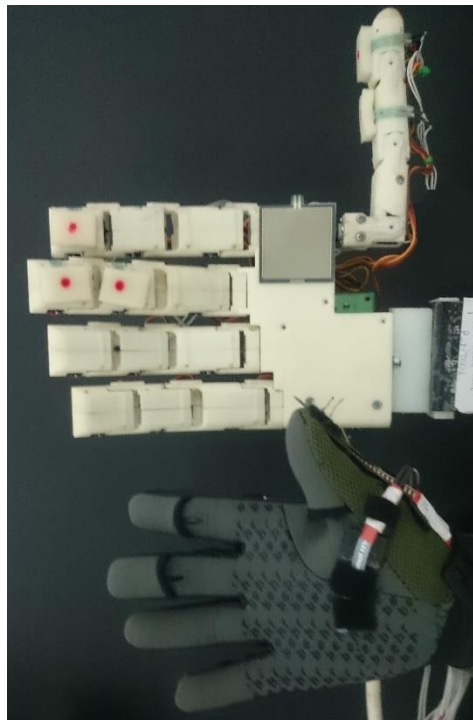
```
#include <SSC32.h>
// Defining the connection with SSC32 board
SSC32 myssc = SSC32();
// Initializing the flex sensor value
```

```
int sscdeg15 = 0;
int sscdeg14 = 0;
int sscdeg13 = 0;
// Initializing the pulse for SSC32
int p15 = 0;
int p14 = 0;
int p13 = 0;
// Defining the time that the SSC32 has to complete the trajectory
int t=500;
void setup(){
    // Start the serial communication with XBee
    Serial.begin(9600);
    // Start communication with the SSC32 board Serial port 1
    //(Pins 19(RX),18(TX)) Arduino Mega 2560
    myssc.begin(9600);
}
void loop(){
// Check if there is an incoming bit
while (Serial.available()){
    // Read the incoming bit integer format
    sscdeg15 = Serial.parseInt();
    sscdeg14 = Serial.parseInt();
    sscdeg13 = Serial.parseInt();
    if (Serial.read()=='\n'){
        // Converting the values of degrees to pulse
        p15 = 2050-(sscdeg15*((2050-1050)/90));
        p14 = 2100-(sscdeg14*((2100-1050)/90));
        p13 = 2050-(sscdeg13*((2050-1000)/90));
        // Sending the Command to SSC32 board
        myssc.beginGroupCommand(SSC32_CMDGRP_TYPE_SERVO_MOVEMENT);
        myssc.servoMoveTime(15,p15,t);
        myssc.servoMoveTime(14,p14,t);
        myssc.servoMoveTime(13,p13,t);
        myssc.endGroupCommand
        // Wait for the trajectory to be completed
        delay(t);
    }
}
}
```

## 5.6. Πειραματικά αποτελέσματα

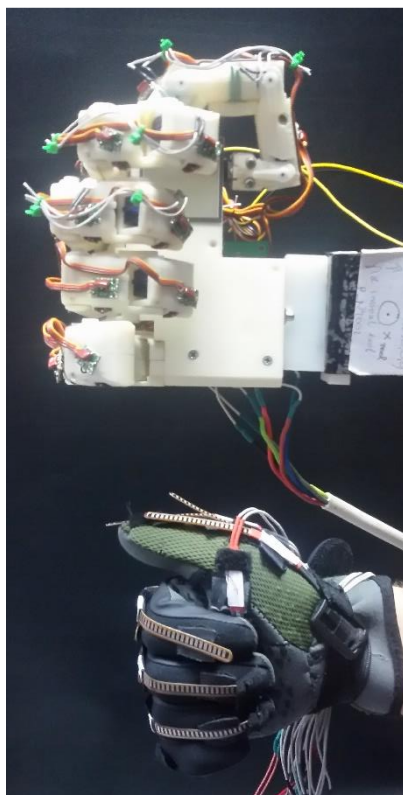
Παρακάτω παρουσιάζονται οι δυνατότητες ελέγχου του ρομποτικού χεριού από το γάντι συλλογής δεδομένων. Παράλληλα θα ειπωθούν διάφορα είδη λαβών που μπορεί να πραγματοποιήσει το ρομποτικό χέρι ελεγχόμενο από το γάντι συλλογής δεδομένων. Ο έλεγχος του ρομποτικού χεριού γίνεται με όσα αναφέρθηκαν παραπάνω (κεφ. 3, 4, 5).

**Ανοιχτή Παλάμη:** Είναι μια χειρονομία στην οποία όλα τα δάχτυλα βρίσκονται σε πλήρη έκταση ως προς το μετωπιαίο επίπεδο (Εικόνα 5.23).



Εικόνα 5.23. Χειρονομία ανοιχτής παλάμης με το ρομποτικό χέρι και το γάντι συλλογής δεδομένων.

**Κλειστή παλάμη – Γροθιά:** είναι μια χειρονομία στην οποία όλα τα δάχτυλα και ο αντίχειρας βρίσκονται σε πλήρη κάμψη (Εικόνα 5.24).



Εικόνα 5.24. Χειρονομία κλειστή παλάμη με το ρομποτικό χέρι και το γάντι συλλογής δεδομένων.

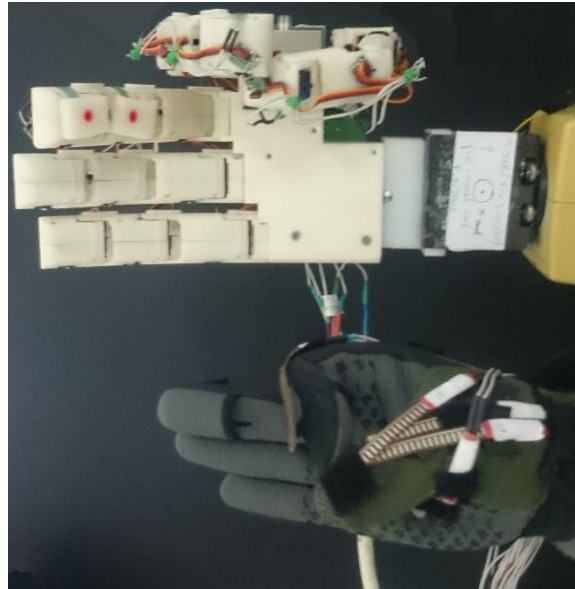
**Point:** είναι μια χειρονομία στην οποία ο δείκτης βρίσκεται σε πλήρη έκταση ως προς το μετωπιαίο επίπεδο ενώ όλα τα υπόλοιπα δάχτυλα βρίσκονται σε κάμψη ως προς αυτό.



Εικόνα 5.25. Χειρονομία point με το ρομποτικό χέρι και το γάντι συλλογής δεδομένων.

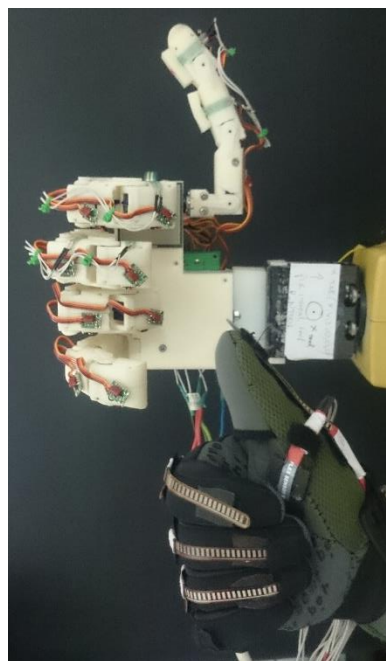
**OK:** είναι μια χειρονομία στην οποία ο δείκτης και ο αντίχειρας εφάπτονται μεταξύ τους στο οβελιαίο επίπεδο ενώ όλα τα υπόλοιπα δάχτυλα είναι σε πλήρη ή μερική έκταση (Εικόνα

5.26). Στην συγκεκριμένη χειρονομία ο αντίχειρας δεν πραγματοποιεί την χειρονομία με αρκετή ακρίβεια διότι, οι αισθητήρες που καταγράφουν την κίνηση του αντίχειρα δεν είναι τοποθετημένη με βέλτιστη διάταξη στο γάντι που συνεπάγεται σε απόκλιση στην άρθρωση  $q_3$ .



Εικόνα 5.26. Χειρονομία OK με το ρομποτικό χέρι και το γάντι συλλογής δεδομένων.

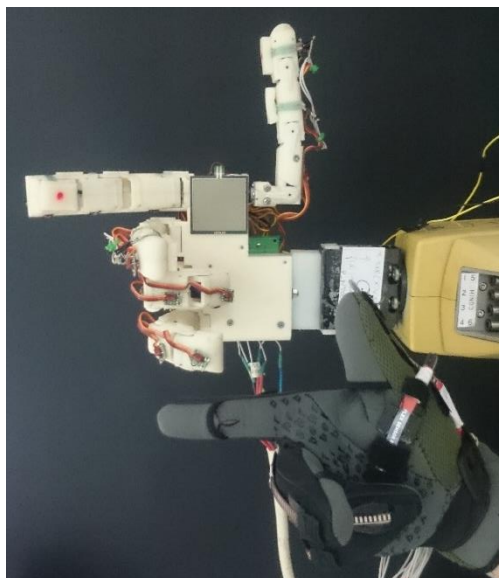
**Like:** είναι μια χειρονομία στην οποία ο αντίχειρας βρίσκεται σε μερική έκταση ως προς το οβελιαίο επίπεδο, ενώ τα υπόλοιπα δάχτυλα είναι σε πλήρη κάμψη στο μετωπιαίο επίπεδο (Εικόνα 5.27).



Εικόνα 5.27. Χειρονομία like με το ρομποτικό χέρι και το γάντι συλλογής δεδομένων.

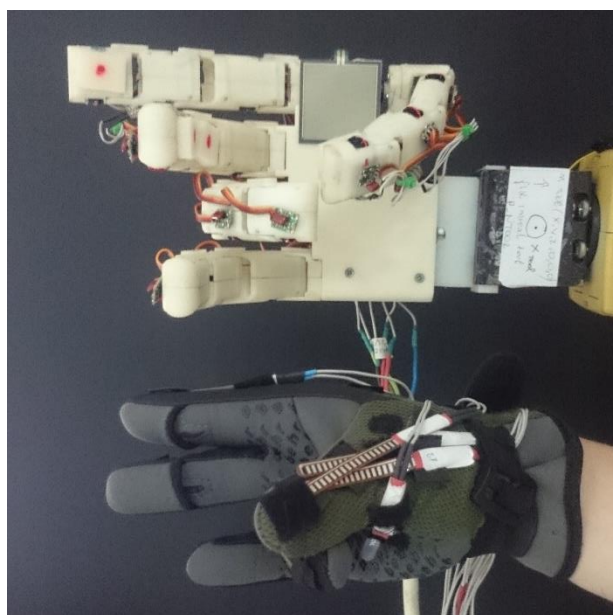


**Άξονες X-Y-Z:** είναι μια χειρονομία στην οποία ο αντίχειρας είναι σε πλήρη έκταση ως προς το οβελιαίο επίπεδο, ο μέσος είναι σε έκταση ως προς το μετωπιαίο επίπεδο, ο δείκτης είναι σε έκταση κάθετος όμως στο μετωπιαίο επίπεδο και ο παράμεσος με τον μικρό είναι σε πλήρη κάμψη (Εικόνα 5.28).



Εικόνα 5.28. Χειρονομία άξονες X-Y-Z με το ρομποτικό χέρι και το γάντι συλλογής δεδομένων.

**Επαφή Αντίχειρα – Παράμεσου:** είναι η χειρονομία στην οποία ο αντίχειρας και ο παράμεσος είναι σε κάμψη και εφάπτονται μεταξύ τους, ενώ όλα τα υπόλοιπα δάχτυλα είναι σε πλήρη ή μερική έκταση (Εικόνα 5.29).



Εικόνα 5.29. Χειρονομία επαφή αντίχειρα - παράμεσου με το ρομποτικό χέρι και το γάντι συλλογής δεδομένων

Ο Άνθρωπος καθημερινά χειρίζεται διάφορα αντικείμενα και εργαλεία για να πραγματοποιήσει διάφορες εργασίες. Τα αντικείμενα αυτά μπορούν να έχουν από πολύ απλό μέχρι πολύ σύνθετο γεωμετρικό σχήμα. Για να τα χειριστεί σωστά θα πρέπει να κάνει και την ανάλογη λαβή με την βοήθεια των δαχτύλων, του αντίχειρα του αλλά και της παλάμης του. Όταν λυγίζουν όλα τα δάχτυλα και ο αντίχειρας και με την βοήθεια της παλάμης λέμε ότι το χέρι κάνει μια **περιβάλλουσα λαβή**. Συγκεκριμένα με τον όρο αυτό εννοούμε όταν γίνεται σταθερή συγκρότηση αντικειμένων όταν επιβάλλονται εξωτερικές διαταραχές από το περιβάλλον στο αντικείμενο.

Όταν έχουμε την συντονισμένη κίνηση των δαχτύλων για την λαβή ενός αντικειμένου, τότε λέμε ότι το χέρι κάνει μια **ευσταθής λαβή**. Η λαβή αυτή μπορεί να επιτευχθεί είτε στο αρχικό σημείο επαφής των δύο δαχτύλων, είτε σε γειτονικές και προεπιλεγμένες θέσεις με τα δάχτυλα να κυλούν ή να ολισθαίνουν πάνω στην επιφάνεια επαφής του αντικειμένου.

Το ρομποτικό χέρι έχοντας τέσσερα δάχτυλα, αντίχειρα και παλάμη είναι ικανό να κάνει περιβάλλουσες αλλά και ευσταθής λαβές. Παρακάτω θα δούμε τις διάφορες αυτές λαβές που έγιναν μέσω τηλεχειρισμού από το γάντι συλλογής δεδομένων. Σημαντικό είναι να ειπωθεί ότι επειδή το ρομποτικό χέρι δεν έχει αισθητήρια και δεν μπορεί να εντοπίσει την θέση του αντικειμένου που θέλει να πιάσει, το τοποθετούμε εμείς σε συγκεκριμένο σημείο μπροστά από την παλάμη του.

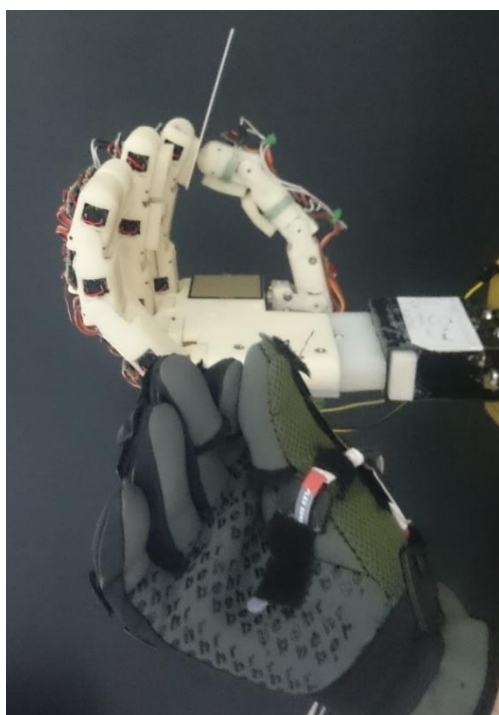
Οι λαβές που θα παρουσιαστούν είναι: **Ευσταθής λαβή χάρτινου κουτιού, ευσταθής λαβή κάρτας με δείκτη και αντίχειρα, περιβάλλουσα λαβή άδειου ποτηριού, ευσταθής λαβή εργαλείου (πένσα).**

**Ευσταθής λαβή χάρτινου κουτιού:** η λαβή αυτή επιτυγχάνετε με τον αντίχειρα να αγκαλιάζει το χάρτινο κουτί ενώ τα υπόλοιπα δάχτυλα να το πιέζουν προς αυτόν (Εικόνα 5.30).



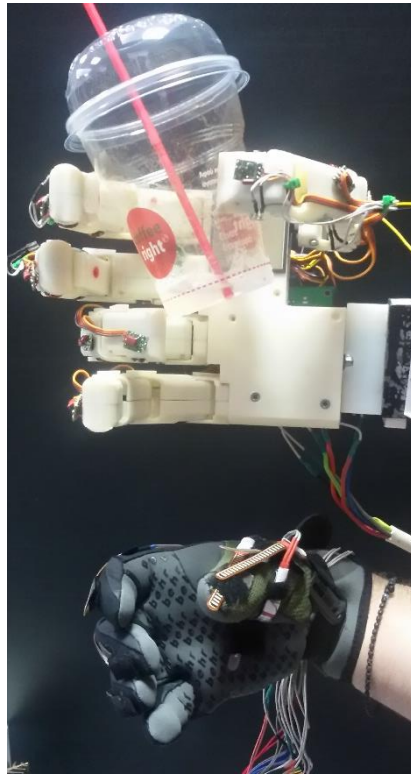
Εικόνα 5.30. Ευσταθής λαβή χάρτινου κουτιού με το ρομποτικό χέρι και το γάντι συλλογής δεδομένων.

**Ευσταθής λαβή κάρτας με δείκτη και αντίχειρα:** η λαβή αυτή επιτυγχάνετε με τον δείκτη και τον αντίχειρα να λυγίζουν την κάρτα (Εικόνα 5.31).



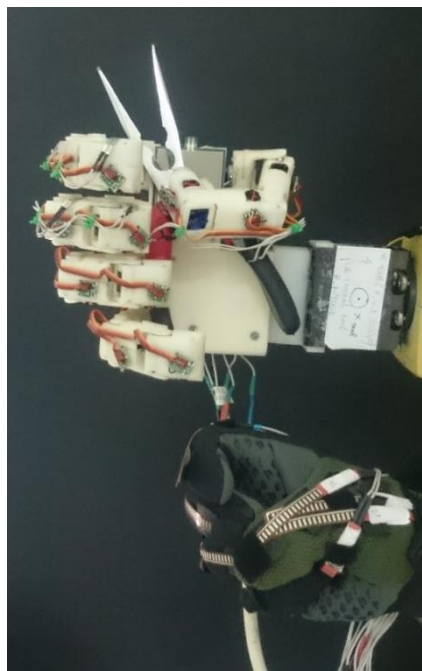
Εικόνα 5.31. Ευσταθής λαβή κάρτας με το ρομποτικό χέρι και το γάντι συλλογής δεδομένων.

**Περιβάλλουσα λαβή άδειου ποτηριού:** η λαβή επιτυγχάνεται με όλα τα δάχτυλα, τον αντίχειρα αλλά και την βοήθεια της παλάμης (Εικόνα 5.32).



Εικόνα 5.32. Ευσταθής λαβή ποτηριού με το ρομποτικό χέρι και το γάντι συλλογής δεδομένων.

**Ευσταθής λαβή εργαλείου (πένσα):** η λαβή αυτή επιτυγχάνεται με τον δείκτη, μέσο, παράμεσο και τον αντίχειρα ενώ ο αντίχειρας έρχεται βοηθητικά από πάνω τους. Η συγκεκριμένη λαβή δεν έγινε με απόλυτη ακρίβεια αλλά η λαβή του αντικειμένου έγινε με επιτυχία.



Εικόνα 5.33. Ευσταθής λαβή εργαλείου με το ρομποτικό χέρι και το γάντι συλλογής δεδομένων.

## **5.7. Συμπεράσματα**

Σε αυτό το κεφάλαιο παρουσιάστηκε ο τρόπος χειρισμού, ενσύρματα και ασύρματα, του ρομποτικού χεριού. Από τον τηλεχειρισμό του ρομποτικού χεριού πραγματοποιήθηκαν κάποιες χειρονομίες και λαβές διαφορών αντικειμένων. Από τις χειρονομίες και τις λαβές συμπεράναμε ότι το ρομποτικό χέρι μπορεί να ακολουθήσει την κίνηση του ανθρώπινου χεριού με σχετικά καλή ακρίβεια, εκτός από κάποιες χειρονομίες και λαβές που υπήρχε μια σχετική απόκλιση. Η απόκλιση οφείλεται κυρίως στους αισθητήρες (30% απόκλιση όπως αναφέραμε παραπάνω κεφ. (4.4)) αλλά κάλλιστα και στην τοποθέτηση τους πάνω στο γάντι.

## 6. Εικονική αναπαράσταση δεδομένων που λαμβάνονται από το γάντι συλλογής δεδομένων

### 6.1. Εισαγωγή

Αυτό το κεφάλαιο αφορά την εικονική αναπαράσταση των δεδομένων που συλλέγονται από το γάντι συλλογής δεδομένων. Αρχικά γίνεται αναλυτική αναφορά για το λογισμικό του **VRML** του **MATLAB**<sup>®</sup> που χρησιμοποιείται για την σχεδίαση της εικονικής αναπαράστασης. Στην συνέχεια, αναφέρεται η διαδικασία συλλογής δεδομένων του **Simulink**<sup>®</sup> από τον μικροελεγκτή *Arduino* αλλά και από το *XBee* για ασύρματη διασύνδεση.

### 6.2. Απεικόνιση της κίνησης των δακτύλων με Η/Υ

Σ' αυτήν την παράγραφο γίνεται αναλυτική αναφορά για την σχεδίαση ενός δακτύλου του ανθρώπινου χεριού στο λογισμικό VRML.

#### 6.2.1. Εικονικά μοντέλα

Για την ελαχιστοποίηση του χρόνου δοκιμών του πρωτότυπου γαντιού χρειάστηκε να σχεδιαστεί εικονική αναπαράσταση του ανθρώπινου χεριού. Στο εικονικό αυτό περιβάλλον

μπορούμε να αξιολογήσουμε την λειτουργικότητα για το γάντι συλλογής δεδομένων, αλλά και πιθανόν σφάλματα που μπορεί να έχει η κατασκευή. Επίσης, το περιβάλλον αυτό μπορεί να χρησιμοποιηθεί σε παρόμοια διάταξη για την αξιολόγηση των δεδομένων.

Για την σχεδίαση και προσομοίωση του εικονικού κόσμου χρησιμοποιείτε το λογισμικό VRML (*Virtual Reality Modeling Language*) και Matlab Simulink, αντίστοιχα. Στο λογισμικό VRML μπορούμε να σχεδιάσουμε οποιοδήποτε γεωμετρικό σχήμα αλλά δεν μπορούμε να κάνουμε οποιαδήποτε ανάλυση [33]. Για την ανάλυση και προσομοίωση του εικονικού κόσμου πρέπει να χρησιμοποιηθεί το Matlab Simulink.

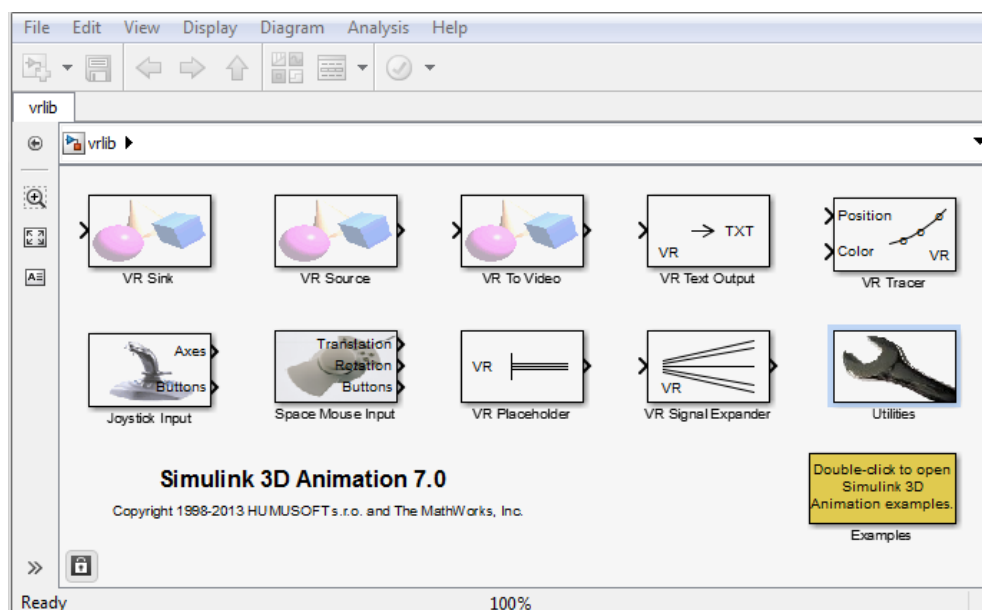
## 6.2.2. Σχεδίαση στο λογισμικό VRML

### 6.2.2.1. Εισαγωγή

Το λογισμικό VRML Builder είναι μια εφαρμογή του Matlab για την σχεδίαση εικονικών κόσμων. Υπάρχουν δύο τρόποι να σχεδιάσει κάποιος έναν εικονικό κόσμο. Ο πρώτος είναι μέσω της γλώσσας προγραμματισμού (text editor) του VRML Builder και ο δεύτερος είναι μέσω του VRML editor χωρίς ιδιαίτερη γνώση της γλώσσας προγραμματισμού VRML [34].

### 6.2.2.2. Βιβλιοθήκη VRML Builder

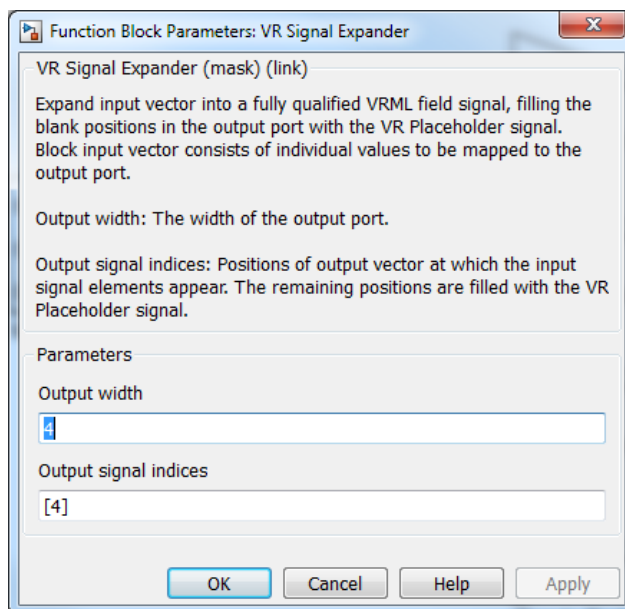
Εκτελώντας την εντολή “vrlib” στην γραμμή εντολών του Matlab, εμφανίζεται η βιβλιοθήκη του VRML Builder (Εικόνα 6.1). Στην βιβλιοθήκη υπάρχουν διάφορες εντολές για την σχεδίαση και επεξεργασία των εικονικών κόσμων όπως και έτοιμα παραδείγματα.



Εικόνα 6.1. Βιβλιοθήκη του VRML Builder.

Τα block που χρησιμοποιούνται στο παράδειγμα μας είναι το VRSink και το VR Signal Expander. Μέσω του VRSink σχεδιάζουμε στον εικονικό κόσμο τον δείκτη του ανθρώπινου χεριού, με απλοποιημένη γεωμετρία (π.χ. κυλίνδρους και σφαίρες). Ενώ με το VR Signal Expander μπορούμε να στείλουμε τα κατάλληλα δεδομένα από το Matlab Simulink στο VRSink για την διαχείριση των σχημάτων.

Το VR Signal Expander μετατρέπει τα δεδομένα σε πίνακα οποιασδήποτε μορφής. Πιο αναλυτικά, στην συγκεκριμένη περίπτωση θέλουμε να περιστρέψουμε ένα αντικείμενο στον χώρο. Για να οριστεί η περιστροφή του αντικειμένου στο VRSink πρέπει να δοθεί σαν εντολή ένας πίνακας  $4 \times 1$  όπου  $(x, y, z, \varphi)$ . Έτσι ορίζουμε τον άξονα περιστροφής (π.χ.  $x, y$  ή  $z$ ) και γωνία στροφής  $\varphi$  σε ακτίνια  $[rad]$ . Στην εικόνα 6.2 φαίνεται ένα παράδειγμα της χρήσης του VR Signal Expander όπου ορίζουμε να εξάγει έναν πίνακα  $4 \times 1$  (Output width) και η τιμή στο τέταρτο κελί θα είναι μεταβλητή (Output signal indices) δηλαδή η τιμή που θα του δίνουμε.



Εικόνα 6.2. Παράθυρο τροποποίησης του VRSignalExpander.

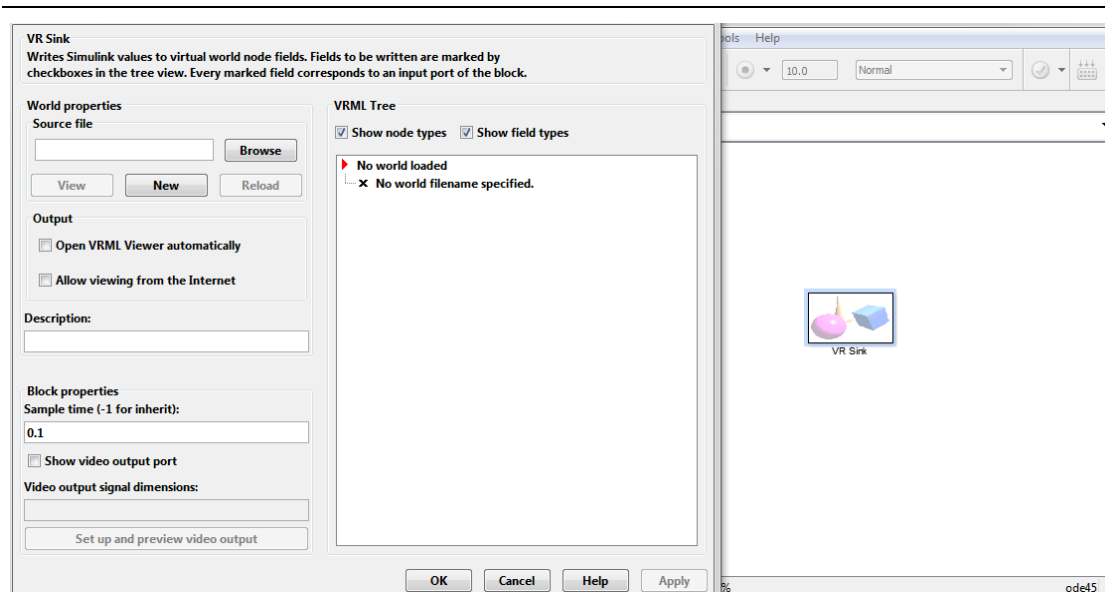
### 6.2.2.3. Σχεδίαση Τρισδιάστατου Κόσμου

Αρχικά, στο παράθυρο της βιβλιοθήκης του VRML ανοίγουμε ένα καινούριο μοντέλο προσομοίωσης, File → New → Model.

Εισάγουμε το block του VRSink στο μοντέλο του Simulink και πατάμε διπλό αριστερό κλικ ώστε να ανοίξουμε το παράθυρο τροποποίησης του VRSink (Εικόνα 6.3).

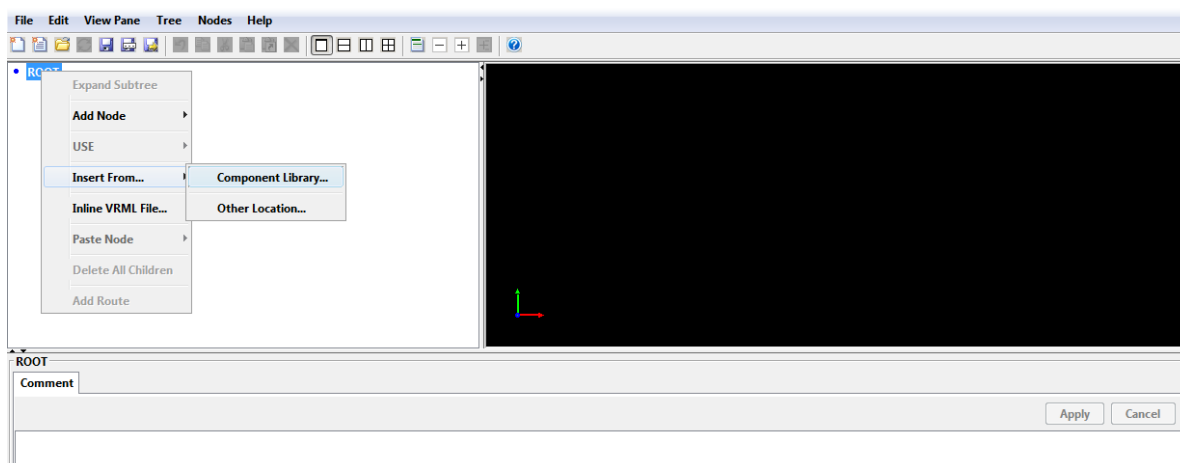


Κεφάλαιο 6<sup>ο</sup>: Εικονική αναπαράσταση δεδομένων που λαμβάνονται από το γάντι συλλογής δεδομένων

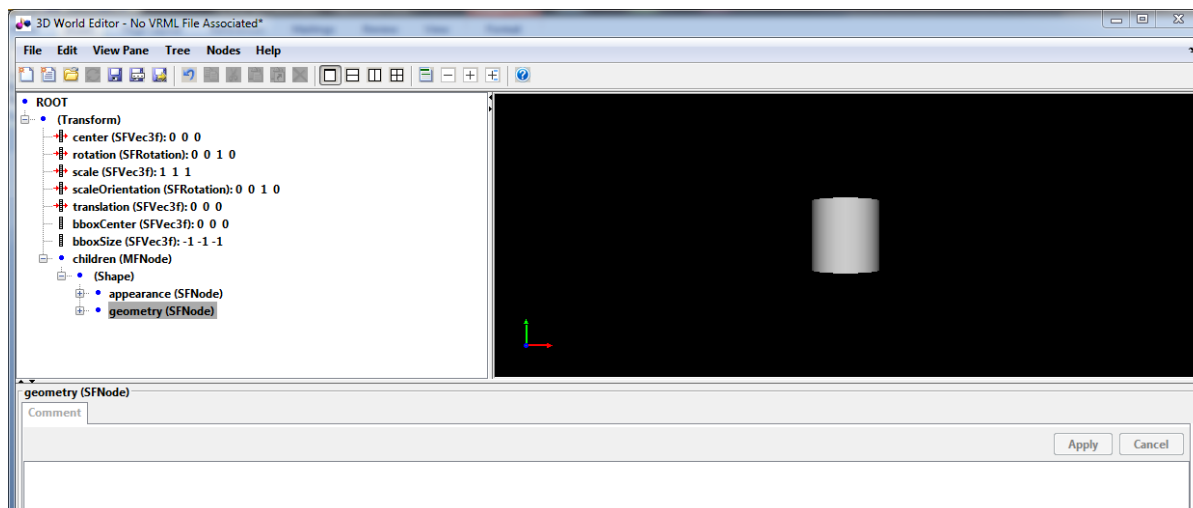


Εικόνα 6.3. Παράθυρο διαχείρισης του VRSink

Πατώντας την επιλογή “new” μας εμφανίζεται το παράθυρο του VRML Builder (Εικόνα 6.4) όπου και μπορούμε να σχεδιάσουμε τον εικονικό μας κόσμο. Για να εισάγουμε οποιοδήποτε γεωμετρικό σχήμα στον εικονικό κόσμο πρέπει πρώτα να ορίσουμε τις συντεταγμένες του στο χώρο. Πατάμε δεξί κλικ στο ROOT και επιλέγουμε Insert From→Component Library. Έπειτα θα μας εμφανιστούν κάποιες έτοιμες γεωμετρίες του VRML, όπου στον φάκελο Shapes υπάρχουν τα έτοιμα γεωμετρικά σχήματα και επιλέγουμε το Cylinder. Έτσι εισάγουμε έναν κύλινδρο μαζί με τις συντεταγμένες του στο χώρο (Εικόνα 6.5). Ο κύλινδρος που εισάγουμε αντιπροσωπεύει μια από τις φάλαγγες ενός δαχτύλου.



Εικόνα 6.4. Παράθυρο του VRML Builder



Εικόνα 6.5. Παράθυρο VRML Builder μετά την εισαγωγή του κυλίνδρου.

Κάνουμε κλικ στο + δίπλα από το **(Transform)** ώστε να το επεκτείνουμε τις ιδιότητες του σχήματος. Τότε μπορούμε να παρατηρήσουμε, όπως φαίνεται και στην Εικόνα 6.5, ότι στα αριστερά έχουμε μια λίστα με τις ιδιότητες του αντικειμένου. Από εκεί μπορούμε να διαχειριστούμε το αντικείμενο. Δηλαδή μπορούμε να αλλάξουμε τον προσανατολισμό του (**rotation**), την θέση του (**translation**) και το κέντρο (**center**). Παρακάτω υπάρχει η επιλογή σχήμα (**Shape**), όπου από εκεί μπορούμε να αλλάξουμε το γεωμετρικό σχήμα του αντικειμένου (**geometry**) και την εμφάνιση του (**appearance**).

Με την ιδιότητα **translation** έχουμε την δυνατότητα να τροποποιήσουμε την θέση του στοιχείου στο χώρο. Η ιδιότητα έχει 3 κελιά για τροποποίηση, δηλαδή τους άξονες  $x, y, z$ . Άρα άμα θέλουμε να μετακινήσουμε το στοιχείο 5  $m$  στον άξονα  $x$  εισάγουμε τις τιμές 5, 0, 0, αντίστοιχα στα κελιά.

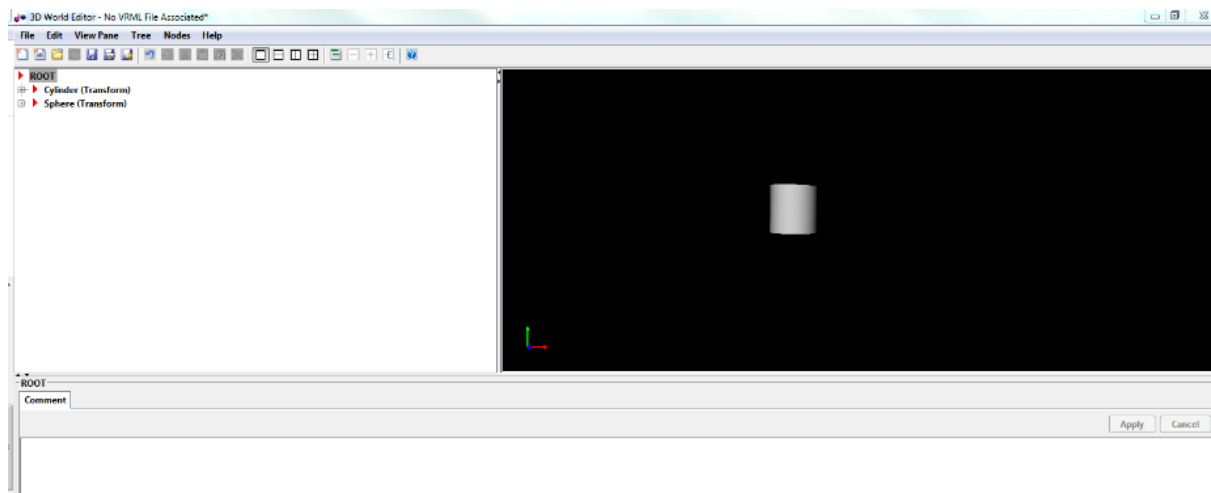
Με την ιδιότητα **rotation** έχουμε την δυνατότητα να τροποποιήσουμε τον προσανατολισμό του στοιχείου. Το **rotation** έχει 4 κελιά όπως και εξηγήσαμε παραπάνω που είναι της μορφής  $(x, y, z, \varphi)$ . Άρα άμα θέλουμε το στοιχείο να περιστραφεί στο  $x$  άξονα στις  $180^\circ$  τότε θα πρέπει να του εισάγουμε την τιμή  $[1, 0, 0, 3.14]$ .

Για την προσομοίωση και λόγους εμφάνισης διαφοροποιούμε την γεωμετρία του κυλίνδρου. Μόλις επεκτείνουμε το **geometry** φαίνονται τα γεωμετρικά χαρακτηριστικά του σχήματος. Με αυτόν τον τρόπο αλλάζουμε την γεωμετρία του κυλίνδρου. Όπου **height** (ύψος) ορίζουμε το ύψος του κυλίνδρου σε μέτρα και το ορίζουμε 5.5, ενώ στην επιλογή **radius** (ακτίνα) ορίζουμε την ακτίνα του κυλίνδρου σε μέτρα όπου και την ορίζουμε 0.9.

Το επόμενο βήμα είναι να εισάγουμε μία σφαίρα όπου θα αντιπροσωπεύει την άρθρωση του δαχτύλου. Ακολουθούμε τα ίδια βήματα όπως και με τον κύλινδρο δηλαδή, πατάμε δεξί κλικ

## Κεφάλαιο 6<sup>ο</sup>: Εικονική αναπαράσταση δεδομένων που λαμβάνονται από το γάντι συλλογής δεδομένων

στο ROOT επιλέγουμε το Insert From→Component Library και στον φάκελο Shapes ανοίγουμε το Cylinder. Αφού εισάγουμε στον τρισδιάστατο κόσμο τον κύλινδρο παρατηρούμε ότι δεν φαίνεται στον εικονικό κόσμο, αλλά έχει καταχωρηθεί στην λίστα με τα στοιχεία που υπάρχουν (Εικόνα 6.6).



Εικόνα 6.6. Παράθυρο του VRML με τα εισαγόμενα στοιχεία

Αυτό συμβαίνει διότι, όταν εισάγουμε ένα στοιχείο στον εικονικό κόσμο τότε αυτό χωροθετείται στις προεπιλεγμένες συντεταγμένες, δηλαδή (0, 0, 0). Για να μπορέσουμε να μετακινήσουμε το στοιχείο απλά τροποποιούμε την ιδιότητα translate.

Τέλος, εισάγουμε τέσσερις κυλίνδρους όπου θα αντιπροσωπεύουν τις φάλαγγες και τρεις σφαίρες όπου θα αντιπροσωπεύουν τις αρθρώσεις.

Στον παρακάτω πίνακα (Πίνακας 6.1) φαίνεται οι διαστάσεις για τις φάλαγγες δηλαδή, τους κυλίνδρους.

Πίνακας 6.1. Διαστάσεις για τις φάλαγγες του δαχτύλου.

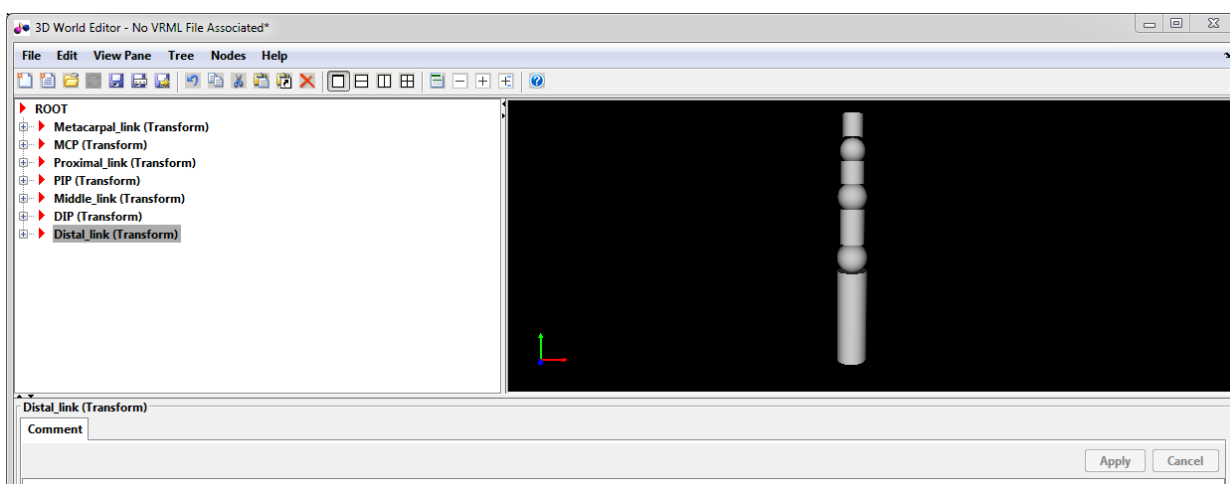
Φάλαγγες	Μετακάρπια	Εγγύς	Μέση	Άπω
Height [m]	5,5	2	1,25	1,25
Radius [m]	0,9	0,725	0,675	0,575
Translation [m]	[0 -0,75 0]	[0 4,6 0]	[0 7,65 0]	[0 10,2 0]

Ενώ στον παρακάτω πίνακα (Πίνακας 6.2) φαίνονται οι διαστάσεις για κάθε άρθρωση του δαχτύλου δηλαδή, τις σφαίρες.

Πίνακας 6.2. Διαστάσεις για τις αρθρώσεις του δαχτύλου.

Αρθρώσεις	1	2	3
Height [m]	0.9	0.85	0.725
Translation [m]	[0 2.8 0]	[0 6,3 0]	[0 8.85 0]

Στην εικόνα 6.7 φαίνεται το τελικό σχήμα για το δάχτυλο. Επίσης μπορούμε να παρατηρήσουμε ότι έχει ονομαστή το κάθε στοιχείο. Αυτό έγινε διότι, για να μπορέσουμε έπειτα να εισάγουμε δεδομένα στην κατάλληλη ιδιότητα (rotation) θα πρέπει να έχει ονομαστή, ώστε να οριστεί σαν μεταβλητή ιδιότητα.



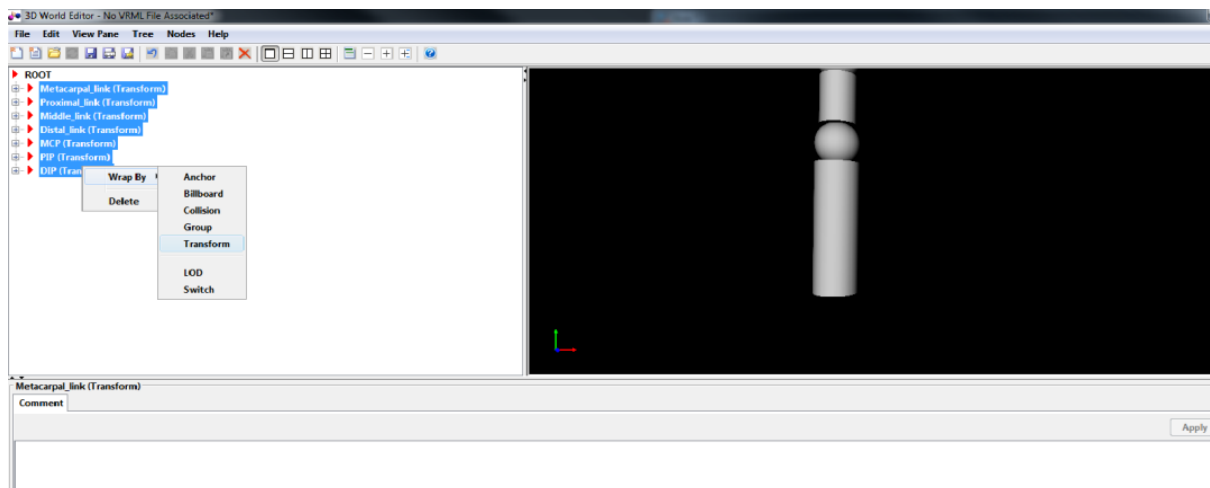
Εικόνα 6.7. Παράθυρο του VRML Builder με σχεδιασμένο τον δείκτη του χεριού στον τρισδιάστατο χώρο.

#### 6.2.2.4. Σύνδεση του κάθε μέλους

Έπειτα, χρειάζεται να σχετιστεί το σύστημα συντεταγμένων των στοιχείων ώστε να υπάρχει φυσιολογική κίνηση μεταξύ των αρθρώσεων, που θα σχετίζεται με την κίνηση των ανθρώπινων δαχτύλων.

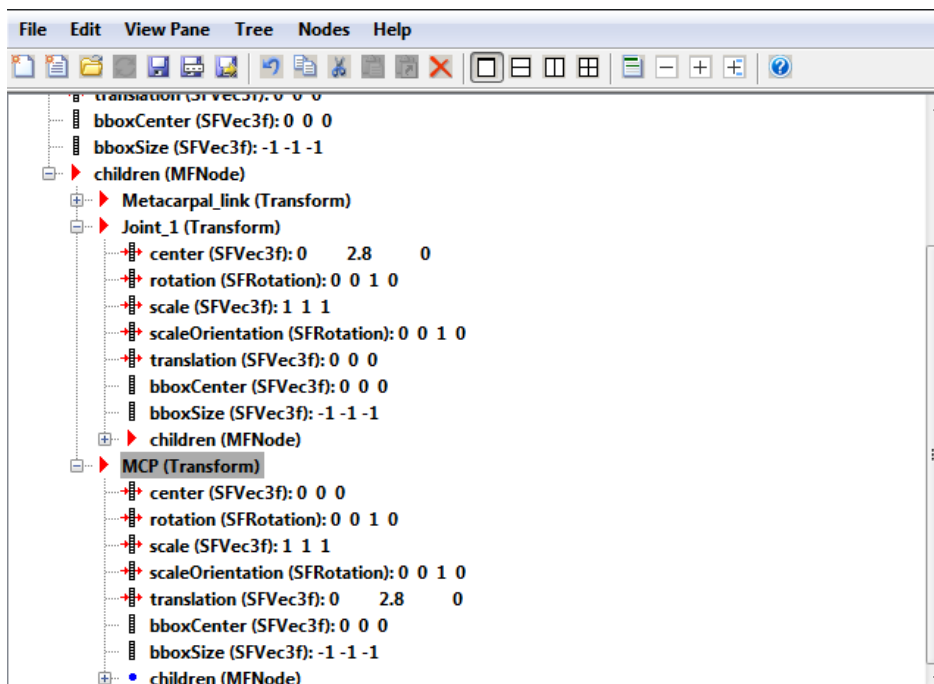
Ορίζουμε τα στοιχεία μεταξύ τους σαν ένα γεωμετρικό σχήμα. Πιο συγκεκριμένα, η περιστροφή στην άρθρωση 1 να περιστρέφει και τις υπόλοιπες αρθρώσεις. Επιλέγουμε όλα τα στοιχεία στην λίστα εκτός ROOT και πατάμε δεξί κλικ→Wrap By→Transform, έτσι εισάγουμε κοινό σύστημα συντεταγμένων για όλα τα στοιχεία. (Εικόνα 6.8).

## Κεφάλαιο 6<sup>ο</sup>: Εικονική αναπαράσταση δεδομένων που λαμβάνονται από το γάντι συλλογής δεδομένων



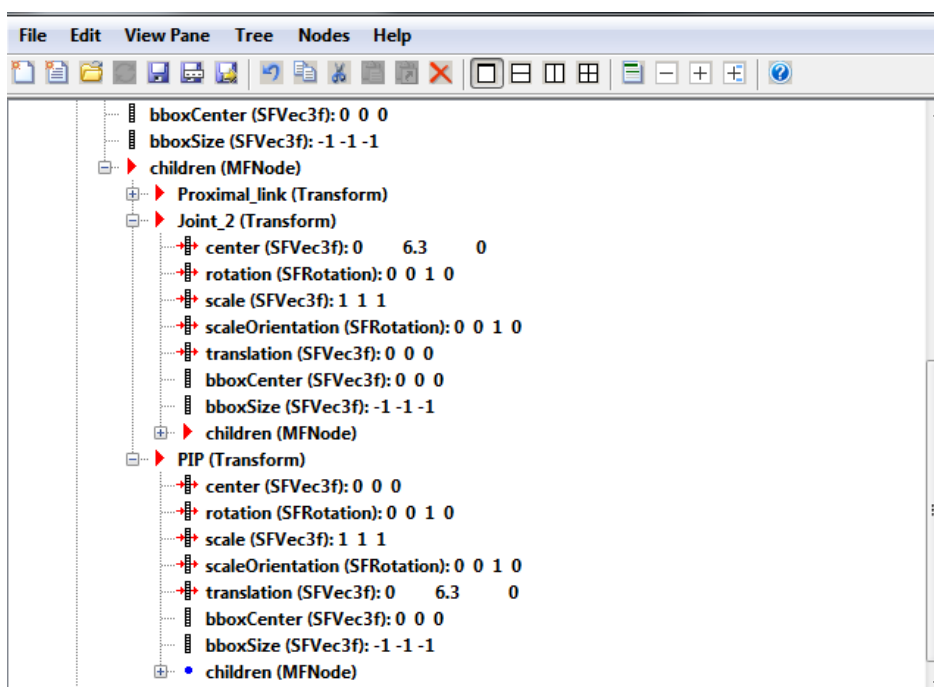
Εικόνα 6.8. Σχέση σύνδεσης των στοιχείων.

Στην συνέχεια, εισάγουμε κοινό σύστημα συντεταγμένων για κάθε άρθρωση. Δηλαδή, όταν η άρθρωση 1 περιστρέφεται, να ακολουθούν οι αρθρώσεις 2 και 3 και αντίστοιχα όταν περιστρέφεται η άρθρωση 2 να περιστρέφεται η άρθρωση 3. Οπότε επιλέγουμε όλα τα στοιχεία εκτός της Metacarpal\_link και του MCP, όπου και θα είναι σταθερά, στα υπόλοιπα στοιχεία πατάμε δεξί κλικ→Wrap By→Transform. Το στοιχείο που δημιουργείται (Transform) είναι στη ουσία η άρθρωση 1. Δηλαδή όταν περιστρέφουμε αυτό το στοιχείο θα περιστρέφονται τα υπόλοιπα στοιχεία, έτσι την ονομάζουμε Joint\_1. Για την σωστή περιστροφή των υπόλοιπων στοιχείων θα πρέπει να ορίσουμε το κέντρο (center) της άρθρωσης. Στην συγκεκριμένη περίπτωση είναι η απόσταση που έχει η σφαίρα MCP από τον άξονα y που είναι 2.8 (Εικόνα 6.9).



Εικόνα 6.9. Λίστα με τα στοιχεία της άρθρωσης 2 στον εικονικό κόσμο.

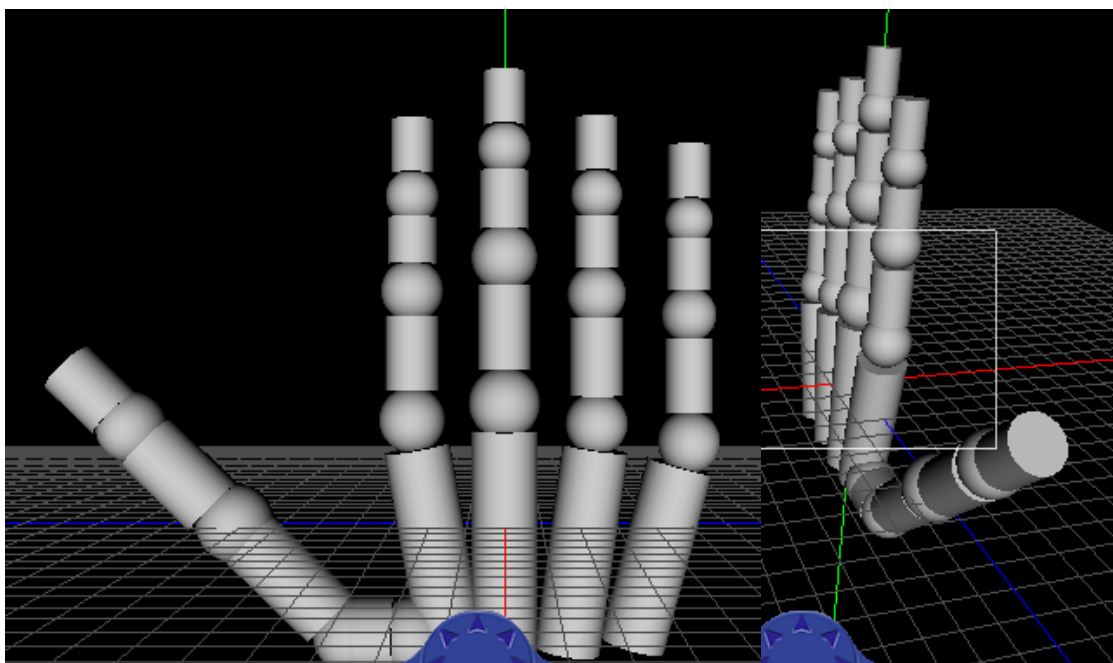
Άρα για να σχηματίσουμε τις υπόλοιπες άρθρωσης ακολουθούμε την ίδια διαδικασία. Αφήνουμε σταθερά τα στοιχεία Proximal\_link και PIP, ύστερα επιλέγουμε τα υπόλοιπα στοιχεία και πατάμε δεξί κλικ → Wrap By → Transform. Το καινούργιο στοιχείο (*Transform*) που δημιουργείται είναι η άρθρωση 2όπου και την ονομάζουμε Joint\_2. Αλλάζουμε το κέντρο της Joint\_2 και εισάγουμε την απόσταση από τον άξονα γπου έχει η σφαίρα PIP (Εικόνα 6.10).



Εικόνα 6.10. Λίστα με τα στοιχεία της άρθρωσης 2 στον εικονικό κόσμο.

Τέλος, για την άρθρωση 3 ακολουθούμε την ίδια διαδικασία. Επιλέγουμε το Distal\_link και πατάμε δεξί κλικ→Wrap By→Transform. Το καινούργιο στοιχείο (Transform) που δημιουργείται είναι η άρθρωση 3 όπου και την ονομάζουμε Joint\_3. Αλλάζουμε το κέντρο της Joint\_3 και εισάγουμε την απόσταση από τον άξονα y που έχει η σφαίρα DIP.

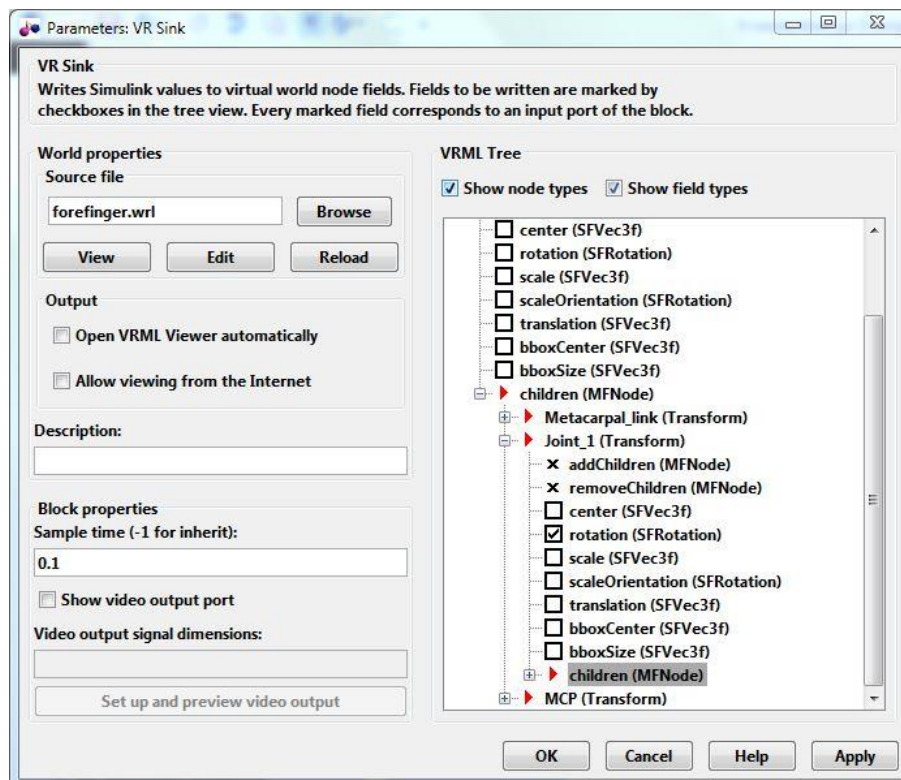
Ακολουθώντας την ίδια διαδικασία σχεδιάζεται το ανθρώπινο χέρι με όλες τις αρθρώσεις τους (Εικόνα 6.11).



Εικόνα 6.11. Ανθρώπινο χέρι σχεδιασμένο στο λογισμικό VRML.

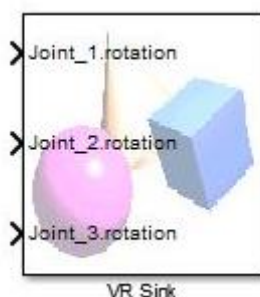
#### 6.2.2.5. Εξαγωγή Παραμέτρων

Αποθηκεύουμε την εργασία και την φορτώνουμε στο VRSink. Με την επιλογή “Browse” βρίσκεται η εργασία που αποθηκευτική ώστε, να φορτώσει στο VRSink. Μόλις ο εικονικός κόσμος φορτώσει στο VRSink (.wrl αρχείο) τότε στα δεξιά (VRML Tree) (Εικόνα 6.12) φαίνονται τα γεωμετρικά σχήματα (π.χ. κύλινδρος, σφαίρα) με τις ιδιότητες του, τις οποίες μπορούμε να τροποποιήσουμε μέσω κατάλληλων σημάτων από το Simulink.



Εικόνα 6.12. Παράθυρο VRSink με φορτωμένο αρχείο .wrl

Στο συγκεκριμένο παράδειγμα θέλουμε να περιστρέψουμε τις αρθρώσεις του δαχτύλου. Οπότε επιλέγουμε στην κάθε άρθρωση (Joint\_i, i=1 – 3) την επιλογή “rotation (SFRotation)”. Πλέον μπορούμε να περιστρέψουμε την κάθε άρθρωση δίνοντας το κατάλληλο σήμα, δηλαδή ένα πίνακα 4 × 1. Παρακάτω (Εικόνα 6.13) απεικονίζεται το VRSink με τις ιδιότητες που έχουμε εξάγει για επεξεργασία.



Εικόνα 6.13. Εντολή του VRSink με επεξεργάσιμες ιδιότητες.

Στην επόμενη παράγραφο εξηγούμε τον τρόπο τροποποίησης των ιδιοτήτων του VRSink δηλαδή, την αξιοποίηση των δεδομένων μέσω απεικόνισης της κίνησης των δακτύλων του χεριού.



### 6.3. Σειριακή επικοινωνία μέσω Matlab – Simulink

Το λογισμικό Matlab είναι ένα εύχρηστο εργαλείο για κάθε επιστήμονα, γενικά το Matlab έχει μεγάλο εύρος χρήσης διότι, διαθέτει βιβλιοθήκες για προγραμματισμό σε διάφορα επιστημονικά πεδία· όπως είναι η Μηχανολογία, Ιατρική κ.α.

Η βιβλιοθήκη που θα ασχοληθούμε στην παρούσα πτυχιακή είναι η “Instrument Control Toolbox”. Η βιβλιοθήκη δίνει την δυνατότητα στο Matlab να επικοινωνήσει άμεσα με κάποιο εξάρτημα όπως είναι ο παλμογράφος, οι μικροελεγκτές, τα τροφοδοτικά κ.α. Η βιβλιοθήκη συνδέεται με τα εξαρτήματα μέσω προγραμμάτων οδήγησης όπως IVI, VXIplug&play ή εντολές SCPI που βασίζονται σε κείμενο και επικοινωνούν μέσω κοινώς χρησιμοποιημένα πρωτόκολλα επικοινωνίας όπως GPIB, VISA, TCP/IP και UDP [35].

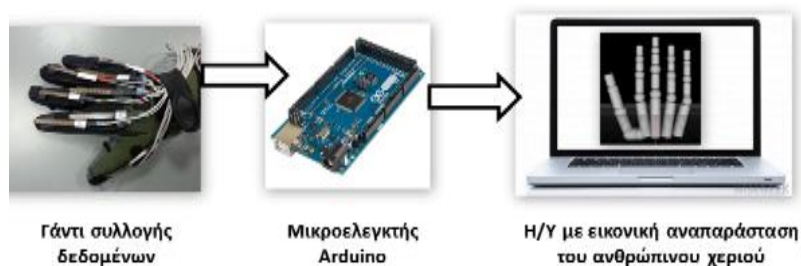
Η σειριακή επικοινωνία [36] είναι το πιο διαδεδομένο χαμηλού – επιπέδου πρωτόκολλο για επικοινωνία μεταξύ δύο ή περισσότερων συσκευών. Συνήθως, η μια συσκευή είναι ηλεκτρονικός υπολογιστής, ενώ η άλλη συσκευή μπορεί να είναι διαμορφωτής – αποδιαμορφωτής Η/Υ (modem), εκτυπωτής, άλλος Η/Υ ή μια επιστημονική συσκευή όπως είναι ο παλμογράφος.

Όπως υποδηλώνει το όνομα, η σειριακή θύρα στέλνει και λαμβάνει δεδομένα (bytes) πληροφοριών με σειριακό τρόπο – ένα bit κάθε φορά. Αυτά τα bytes μεταδίδονται χρησιμοποιώντας δυαδική μορφή ή μορφή κειμένου (ASCII).

Για πολλές εφαρμογές με σειριακή θύρα, μπορεί να γίνει η επικοινωνία χωρίς ιδιαίτερη γνώση το πώς λειτουργεί. Η επικοινωνία καθορίζεται μέσω της σειριακής θύρας του αντικειμένου, το οποίο δημιουργείτε σαν μεταβλητή στον χώρο εργασίας (workspace) του Matlab.

#### 6.3.1. Ενσύρματη διασύνδεση μικροελεγκτή με Η/Υ και Simulink

Ο μικροελεγκτής Arduino συνδέεται μέσω καλώδιο USB με τον Η/Υ και έπειτα διασυνδέεται με το Simulink μέσω σειριακής επικοινωνίας (Εικόνα 6.14).

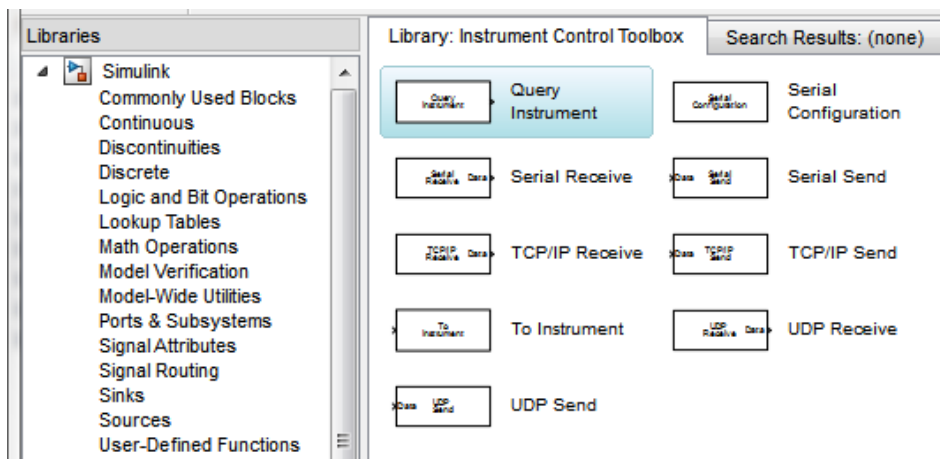


Εικόνα 6.14. Διάγραμμα ροής συστήματος ελέγχου.

Αρχικά προγραμματίζεται ο μικροελεγκτής ώστε να διαβάζει τις τάσεις εξόδου από το γάντι συλλογής δεδομένων, ο παρακάτω κώδικας είναι για 2 αισθητήρες δηλαδή, για την κίνηση του δάχτυλου δείκτη. Έπειτα, διαχειρίζεται τα δεδομένα ώστε να μετατρέψει την τάση εξόδου από κάθε αισθητήρα σε μοίρες (σχέση (4.4)) στην συνέχεια ορίζει το όριο στις τιμές που μπορούν να πάρουν οι μεταβλητές (constrain). Τέλος, στέλνει σειριακά τα δεδομένα στον Η/Υ υπολογιστή. Ο παρακάτω κώδικας μπορεί να επεκταθεί εύκολα για τους υπόλοιπους αισθητήρες.

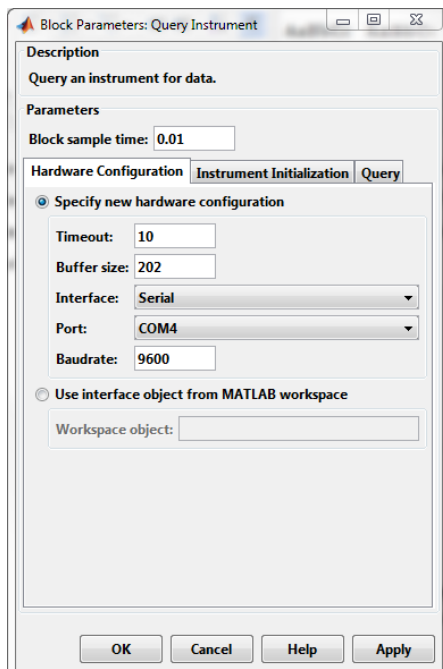
```
// Initialization of the variables
int flx1pin = 0;
int flx2pin = 1;
float flx1val = 0;
float flx2val = 0;
int degq2 = 0;
int degq3 = 0;
int degq4 = 0;
void setup(){
    // Starts the serial communication
    Serial.begin(9600);
    // Sets the mode of the flex sensor pin to input
    pinMode(flx1pin,INPUT);
    pinMode(flx2pin,INPUT);}
void loop(){
    // Reads the analog values of the flex sensor and converts
    // them to degrees (eq. (4.4))
    flx1val = analogRead(sensor1_pin)*5/1023;
    degq2 = round(0+(flx1val-3.20)*(90-0)/(5.00-3.20));
    flx2val = analogRead(sensor2_pin)*5/1023;
    degq3 = round(0+(flx2val-3.05)*(90-0)/(5.00-3.05));
    // Human hand constrain (eq. (3.1))
    degq4 = round(degq3*2/3);
    degq2 = constrain(degq2,0,90);
    degq3 = constrain(degq3,0,90);
    degq4 = constrain(degq4,0,90);
    // Prints the analog values with serial communication
    Serial.println(degq2);
    Serial.println(degq3);
    Serial.println(degq4);
    delay(200);
}
```

Στην συνέχεια γίνεται ρύθμιση του λογισμικού Simulink ώστε να συλλέγει δεδομένα από την σειριακή θύρα που είναι συνδεδεμένος ο μικροελεγκτής Arduino. Η βιβλιοθήκη που διαθέτει το Simulink για σειριακή επικοινωνία ονομάζεται **“Instrument Control Toolbox”** (Εικόνα 6.15), θα πρέπει να σημειωθεί ότι δεν είναι η μόνη βιβλιοθήκη για σειριακή επικοινωνία.



Εικόνα 6.15. Βιβλιοθήκη Instrument Control Toolbox.

Για την ανάγνωση των δεδομένων από το Simulink χρησιμοποιείτε το block με την εντολή “Query Instrument” (Εικόνα 6.16). Με την “Query Instrument” γίνεται εισαγωγή των δεδομένων με το δεκαδικό σύστημα δηλαδή, ακέραιους αριθμούς.



Εικόνα 6.16. Παράθυρο επεξεργασίας για την εντολή Query Instrument.

Αφού προστεθεί στο μοντέλο προσομοίωσης και γίνει διπλό κλικ στην εντολή τότε εμφανίζονται οι παράμετροι. Στην καρτέλα “Hardware Configuration” ρυθμίζουμε τα χαρακτηριστικά του μικροελεγκτή. Στην δικιά μας περίπτωση είναι τα παρακάτω:

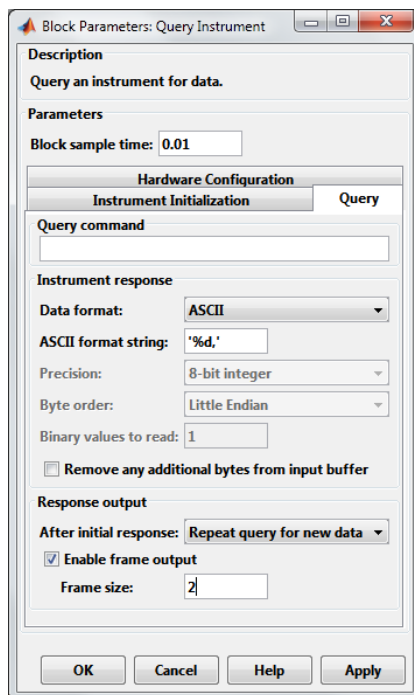
- Block sample time      0.01
- Timeout                    10
- Buffer size                202
- Interface                  Serial
- Port                         COM4
- Baudrate                  9600

Όπου:

- Block sample time:      Είναι η μόνη παράμετρος που δεν έχει σχέση με τις ρυθμίσεις. Ορίζοντας αυτή την παράμετρο με θετικό αριθμό ορίζουμε τον χρόνο δειγματοληψίας από τον μικροελεγκτή.
- Timeout:                 Ο χρόνος σε δευτερόλεπτα που επιτρέπεται στην εντολή να ολοκληρωθεί.
- Buffer size:              Ο συνολικός χώρος αποθήκευσης σε bytes που μπορεί να αποθηκευτεί στον μικροελεγκτή, κατά την διαδικασία που διαβάζει τα δεδομένα.
- Interface:                Ο τύπος της πλακέτας που συνδέεται.
- Port:                      Η θύρα σύνδεσης της πλακέτας.
- Baudrate:                Ο ρυθμός μεταφοράς σε bits.

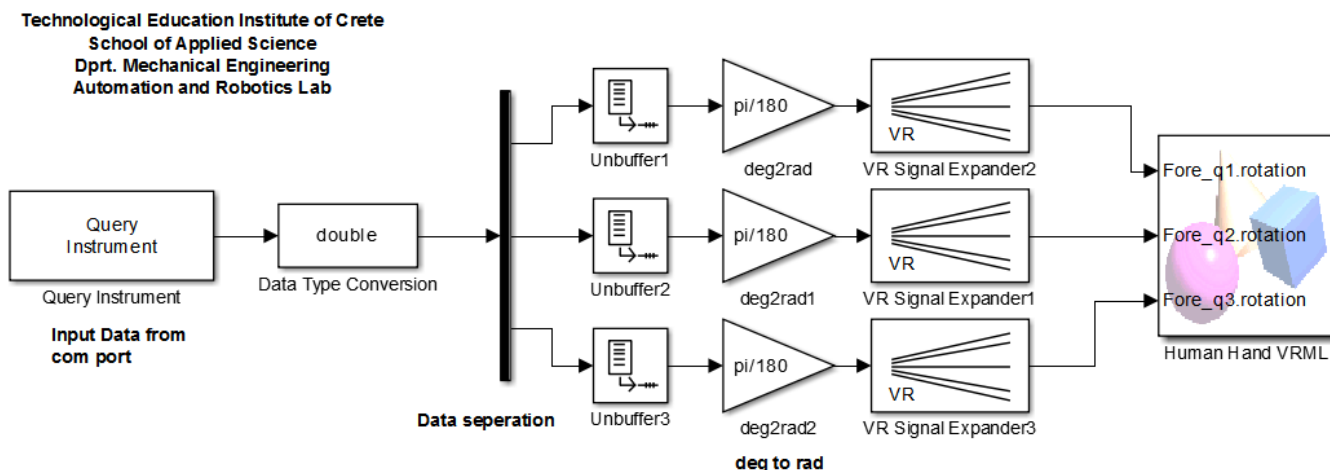
Στην καρτέλα “Instrument Initialization” μπορούμε να ορίσουμε κάποια εντολή να εκτελεστεί μόλις ανοίξει η επικοινωνία με την πλακέτα. Στην δικιά μας περίπτωση είναι κενό.

Στην καρτέλα “Query” μπορούμε να ορίσουμε προαιρετικές εντολές, τον τύπο δεδομένων που διαβάζει η εντολή και τον όγκο εισαγωγής δεδομένων (Εικόνα 6.17).



Εικόνα 6.17. Παράθυρο της εντολής “Query Instrument” καρτέλα “Query” παράμετροι εισαγωγής δεδομένων.

Ρυθμίζουμε να εισάγει δεδομένα σε μορφή “ASCII” και ακεραίους “%d”, επίσης να επαναλαμβάνει την διαδικασία κάθε φορά για καινούργια δεδομένα (Repeat query for new data). Με την επιλογή “Enable frame output” ρυθμίζουμε την εντολή “Query instrument” να δέχεται παραπάνω από ένα δεδομένο και να τα καταχωρεί σε πίνακα, όπου “Frame size” είναι το μέγεθος του πίνακα (π.χ.  $4 \times 1$ ). Παρακάτω (Εικόνα 6.18) φαίνεται ο προγραμματισμός μέσω Simulink.



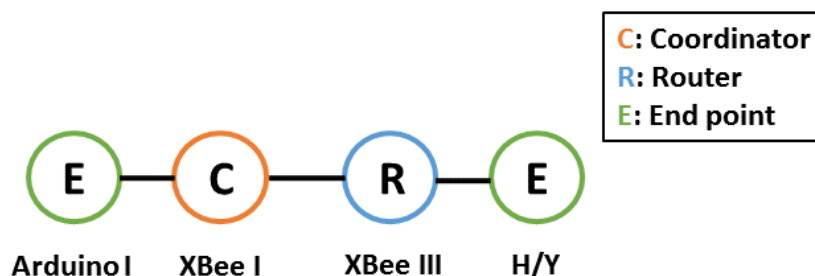
Εικόνα 6.18. Προγραμματισμός μέσω Simulink για την εισαγωγή των δεδομένων από το Arduino και διαχείρισης της εικονικής απεικόνισης του ανθρώπινου χεριού.

Στον παραπάνω κώδικα εισάγονται τα δεδομένα από τον μικροελεγκτή (*Query instrument*) και μετατρέπονται από “uint8 (unsigned integer 8-bit)” σε “double” ώστε, τα δεδομένα να

αναγνωρίζονται από τα υπόλοιπα στοιχεία. Στην συνέχεια, τα δεδομένα διαχωρίζονται από την εντολή “demux (separate data)” δηλαδή, “απομονώνεται” το στοιχείο σε κάθε κελί του πίνακα. Έπειτα, με την εντολή “Unbuffer” τα δεδομένα μετατρέπονται σε πίνακα  $1 \times 1$ . Τέλος, τα δεδομένα μετατρέπονται σε ακτίνια (deg2rad) ώστε να σταλούν στον εικονικό κόσμο μέσω του “VR SingalExpander”.

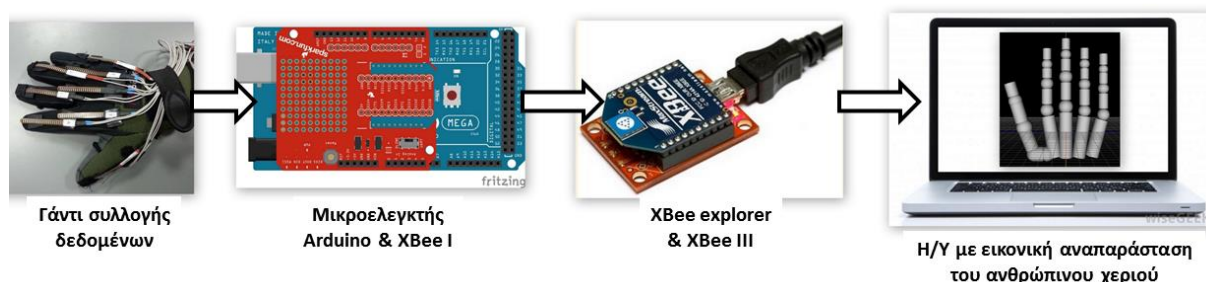
### 6.3.2. Ασύρματη διασύνδεση με Η/Υ

Για την ασύρματη διασύνδεση του Η/Υ με το γάντι συλλογής δεδομένων συνδέεται το XBee III με το Η/Υ και το XBee με τον Arduino I δηλαδή, με το σύστημα συλλογής δεδομένων (Εικόνα 6.19). Το XBee III ρυθμίζεται όπως αναφέραμε στο κεφάλαιο 5.5.2.



Εικόνα 6.19. Σχηματική αναπαράσταση ασύρματου δικτύου με Η/Υ.

Το XBee III (σύνδεση με Η/Υ) δέχεται δεδομένα από το XBee I (coordinator, σύνδεση με γάντι συλλογής δεδομένων), άρα η παράμετρο του DL πρέπει να είναι 1 και η MY 2. Στην συνέχεια ρυθμίζεται το Simulink να συλλέγει δεδομένα από την σειριακή θύρα (COM του XBee III) που είναι συνδεδεμένο το XBee III. Η διαδικασία ρύθμισης του Simulink ώστε να συλλέγει δεδομένα από το XBee είναι η ίδια όπως με τον μικροελεγκτή Arduino.



Εικόνα 6.20. Διάγραμμα ροής ασύρματης επικοινωνίας με Η/Υ και διαχείρισης της εικονικής αναπαράστασης.

Ο κώδικας που χρησιμοποιείτε για την συλλογή των δεδομένων, του Simulink από το XBee III, είναι ίδιος με τον κώδικα που χρησιμοποιείτε για τον μικροελεγκτή Arduino (κεφάλαιο 6.3.1.). Διότι, το Simulink συλλέγει δεδομένα από μια σειριακή θύρα (“COM”) που του

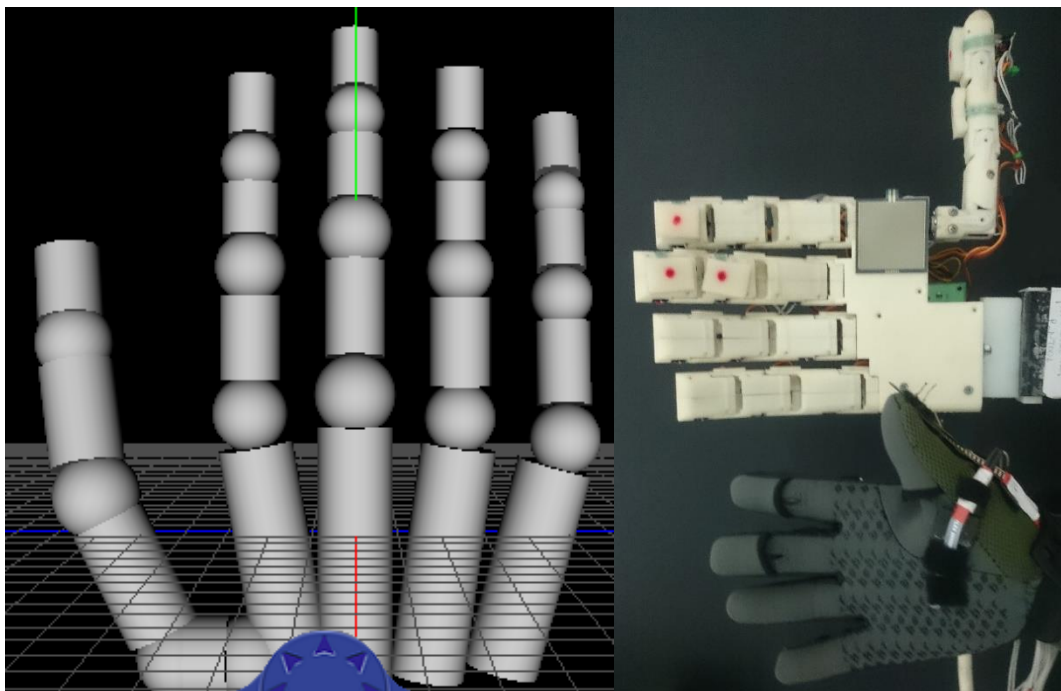
καθορίζουμε, χωρίς να ορίσουμε την συσκευή που είναι συνδεδεμένη στην συγκεκριμένη θύρα.

Τα δεδομένων που λαμβάνονται από το γάντι συλλογής δεδομένων μπορούν να χρησιμοποιηθούν ταυτόχρονα για τον χειρισμό του ρομποτικού χεριού και την απεικόνιση της κίνησης. Πιο συγκεκριμένα, στην περίπτωση της ενσύρματης επικοινωνίας ο μικροελεγκτής πρέπει να προγραμματιστή να στέλνει ταυτόχρονα δεδομένα, μέσω σειριακής επικοινωνίας, στον Η/Υ και στον σερβοελεγκτή SCC-32· έτσι τα δεδομένα μπορούν να αξιοποιηθούν ταυτόχρονα για τον χειρισμό του ρομποτικού χεριού αλλά και την απεικόνιση της κίνησης. Ενώ στην περίπτωση της ασύρματης επικοινωνίας είναι πιο εύκολο διότι, δεν χρειάζεται να γίνει κάποια μετατροπή απλά ρυθμίζεται το XBee III (σύνδεση με Η/Υ) να συλλέγει δεδομένα από το XBee I (coordinator) (σύνδεση με γάντι συλλογής δεδομένων).

#### **6.4. Πειραματικά αποτελέσματα**

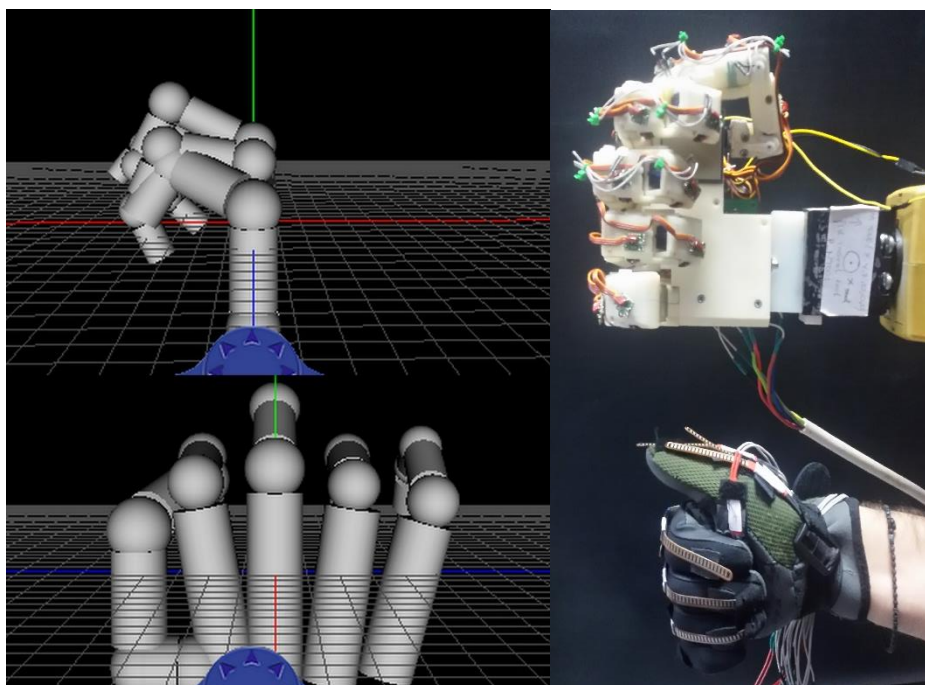
Παρακάτω παρουσιάζονται κάποιες χαρακτηριστικές χειρονομίες που μπορούν να επιτευχθούν με την εικονική αναπαράσταση, όταν συλλέγονται τα δεδομένα από το γάντι συλλογής δεδομένων. Οι χειρονομίες που παρουσιάζονται είναι κάποιες από τις χειρονομίες που πραγματοποίησε το ρομποτικό χέρι που αναφέραμε στο κεφάλαιο 5.6.

**Ανοιχτή παλάμη:** Είναι μια χειρονομία στην οποία όλα τα δάχτυλα βρίσκονται σε πλήρη έκταση ως προς το μετωπιαίο επίπεδο (Εικόνα 6.21).



Εικόνα 6.21. Χειρονομία ανοιχτής παλάμης με το ρομποτικό χέρι και το γάντι συλλογής δεδομένων όπως απεικονίζεται στην εικονική αναπαράσταση.

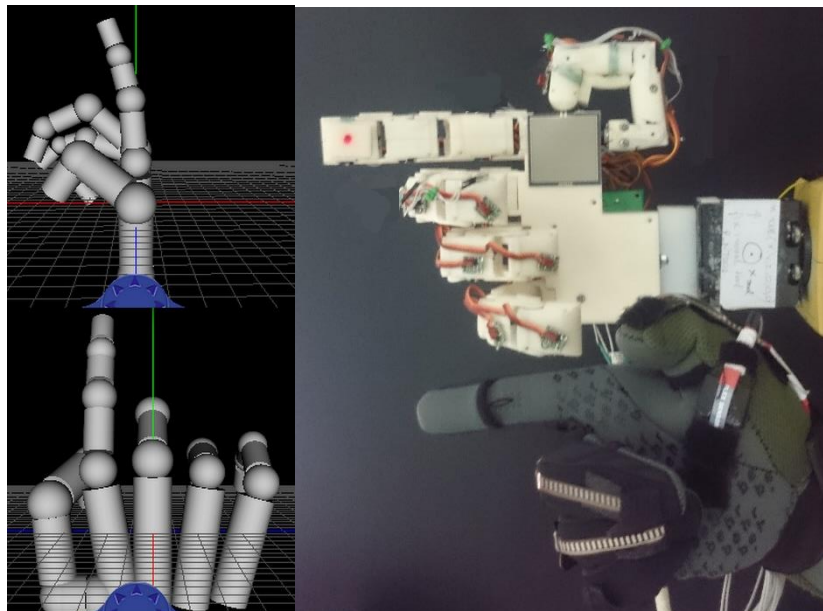
**Κλειστή παλάμη – Γροθιά:** είναι μια χειρονομία στην οποία όλα τα δάχτυλα και ο αντίχειρας βρίσκονται σε πλήρη κάμψη (Εικόνα 6.22).



Εικόνα 6.22. Χειρονομία κλειστή παλάμη με το ρομποτικό χέρι και το γάντι συλλογής δεδομένων όπως απεικονίζεται στην εικονική αναπαράσταση.

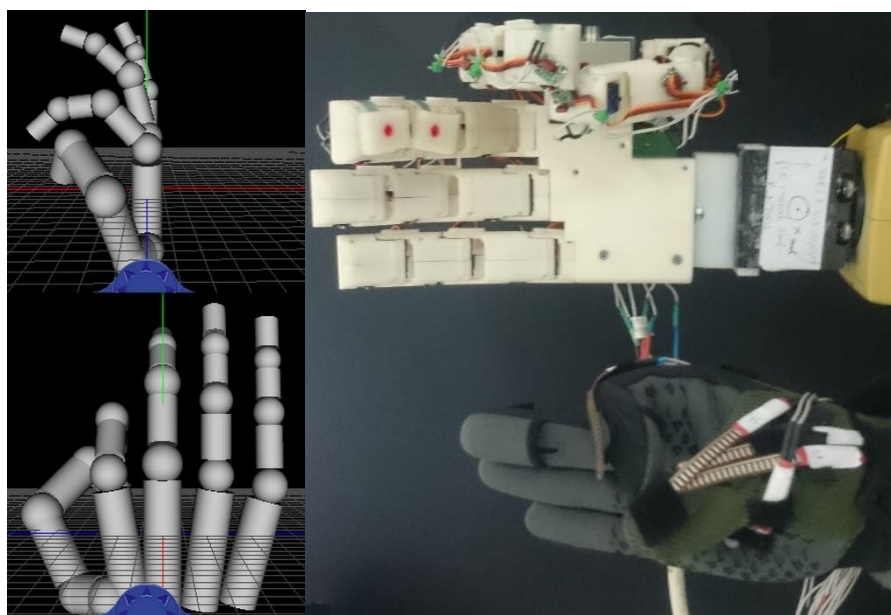


**Point:** είναι μια χειρονομία στην οποία ο δείκτης βρίσκεται σε πλήρη έκταση ως προς το μετωπιαίο επίπεδο ενώ όλα τα υπόλοιπα δάχτυλα βρίσκονται σε κάμψη ως προς αυτό (Εικόνα 6.23).



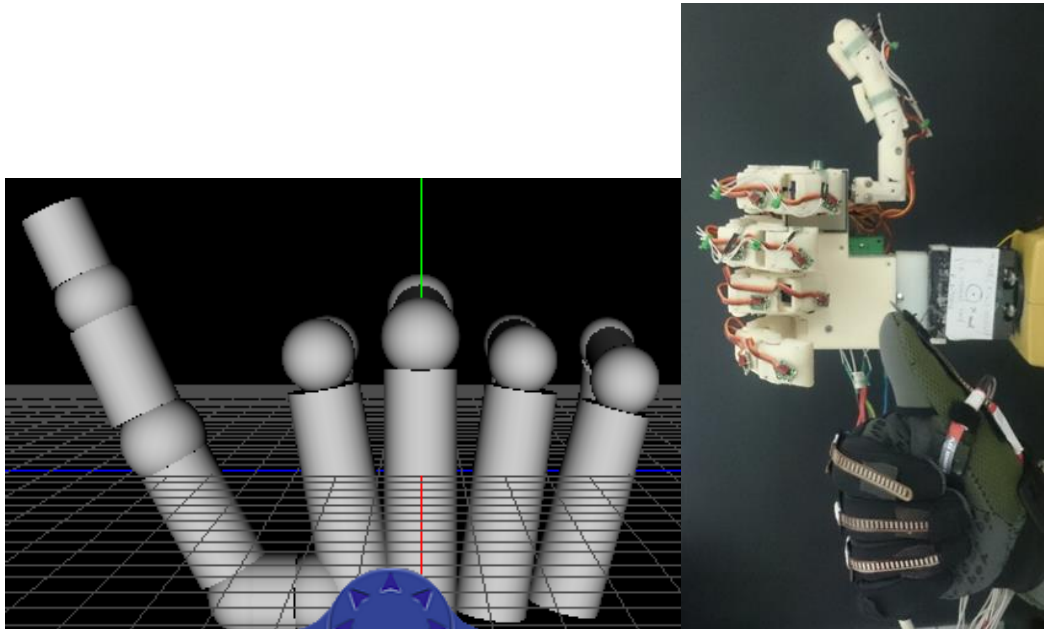
Εικόνα 6.23. Χειρονομία point με το ρομποτικό χέρι και το γάντι συλλογής δεδομένων όπως απεικονίζεται στην εικονική αναπαράσταση.

**OK:** είναι μια χειρονομία στην οποία ο δείκτης και ο αντίχειρας εφάπτονται μεταξύ τους στο οβελιαίο επίπεδο ενώ όλα τα υπόλοιπα δάχτυλα είναι σε πλήρη ή μερική έκταση (Εικόνα 6.24).



Εικόνα 6.24. Χειρονομία OK με το ρομποτικό χέρι και το γάντι συλλογής δεδομένων όπως απεικονίζεται στην εικονική αναπαράσταση.

**Like:** είναι μια χειρονομία στην οποία ο αντίχειρας βρίσκεται σε μερική έκταση ως προς το οβελιαίο επίπεδο, ενώ τα υπόλοιπα δάχτυλα είναι σε πλήρη κάμψη στο μετωπιαίο επίπεδο (Εικόνα 6.25).



Εικόνα 6.25. Χειρονομία Like με το ρομποτικό χέρι και το γάντι συλλογής δεδομένων όπως απεικονίζεται στην εικονική αναπαράσταση.

## 6.5. Συμπεράσματα

Σ' αυτό το κεφάλαιο έγινε αναλυτική αναφορά στην σχεδίαση ανθρώπινου χεριού σε εικονική περιβάλλον περαιτέρω, γίνεται συλλογή δεδομένων από το γάντι μέσω του Simulink. Έτσι, σχεδιάστηκε ένα ολοκληρωμένο σύστημα για την αξιολόγηση των δεδομένων από το γάντι συλλογής δεδομένων. Με την εικονική αναπαράσταση μπορούν να γίνουν δοκιμές στην διάταξη των αισθητήρων ώστε να αποφευχθούν τυχόν ατυχήματα (π.χ. σύγκρουση των δαχτύλων στο ρομποτικό χέρι).

## 7. Συμπεράσματα και μελλοντική έρευνα

### 7.1. Ανασκόπηση και συμπεράσματα της πτυχιακής εργασία

Στην παρούσα πτυχιακή εργασία, μελετήθηκε ο σχεδιασμός και η κατασκευή ενός συστήματος συλλογής δεδομένων (*data glove*) για τον τηλεχειρισμό του ανθρωπόμορφου ρομποτικού χεριού. Τα συμπεράσματα από την παρούσα εργασία συνοψίζονται ως εξής:

- Αρχικά μελετήθηκε το ανθρώπινο χέρι ως προς τις κινηματικές του ικανότητες. Η μελέτη στηρίχτηκε στην εύρεση περιορισμών για τις αρθρώσεις του ανθρώπινου χεριού ώστε, να ελαχιστοποιηθούν οι αισθητήρες εν χρήση.
- Η χρήση των περιορισμών αλλά και η μελέτη του αισθητήρα καταγραφής της κίνησης (*flex sensor*), συντέλεσε στην κατασκευή ενός συστήματος με αρκετή ακρίβεια και επαναληψιμότητα. Αναλυτικότερα:
  - Από την μελέτη του ανθρώπινου χεριού προέκυψε ότι μπορεί να γίνει πρόβλεψη της κίνησης των ανθρώπινων δαχτύλων, βάσει συγκεκριμένων κινήσεων των υπόλοιπων δαχτύλων. Λαμβάνοντας υπόψιν τα παραπάνω μπορούμε να πούμε ότι, στην περίπτωση που απενεργοποιήσουμε κάποιους αισθητήρες ή υπάρχει βλάβη τότε μπορεί να συνεχιστή η καταγραφή της κίνησης, αλλά όχι με την ίδια ακρίβεια.

- Η ανάλυση και βαθμονόμηση του αισθητήρα κάμψης συντέλεσε στην σωστή χωροθέτηση των αισθητήρων πάνω στο γάντι. Σαν αποτέλεσμα των παραπάνω συμπεραίνουμε ότι, το γάντι συλλογής δεδομένων μπορεί να καταγράψει την κάμψη/επέκταση ως προς το μετωπιαίο επίπεδο του δείκτη, μέσου, παράμεσου και μικρού. Άξιο αναφοράς είναι η καταγραφή της κίνησης, με αρκετή ακρίβεια, του αντίχειρα, ο οποίος διαθέτει 4 βαθμούς ελευθερίας.
- Η καταγραφή των δεδομένων από τον αισθητήρα κάμψης γίνεται με δύο τρόπους. Αυτό συμβαίνει διότι ο κάθε τρόπος έχει τα πλεονεκτήματα και μειονεκτήματα του. Αναλυτικότερα:
  - Όταν η καταγραφή των δεδομένων γίνεται με ενσύρματο τρόπο τότε ο χειρισμό του ρομποτικού χεριού γίνεται με απόκριση περίπου 100-200 msec.
  - Όταν η καταγραφή των δεδομένων γίνεται με ασύρματη επικοινωνία τότε έχουμε απομακρυσμένο χειρισμό του ρομποτικού χεριού. Αυτό συνεπάγεται σε περισσότερα πεδία εφαρμογής του συστήματος αλλά και ασφάλεια του χειριστή. Η ασύρματη επικοινωνία έχει το μειονέκτημα ότι η αποστολή και συλλογή των δεδομένων χρειάζεται περισσότερο χρόνο (περίπου 300-400 msec) για την απόκριση από το ρομποτικό χέρι.

## 7.2. Μελλοντική έρευνα

Από την παρούσα πτυχιακή εργασία αναδείχτηκε η δυνατότητα τηλεχειρισμού του ρομποτικού χεριού ΤΑΛΟΣ και απεικόνισης της κίνησης των δεδομένων από το γάντι συλλογής δεδομένων. Μελλοντικό πεδίο έρευνας μπορεί να αποτελέσει:

- Βελτίωση του συστήματος ενίσχυσης (μη αναστρέφουσα συνδεσμολογία με τελεστικό ενισχυτή) του σήματος από τον αισθητήρα ώστε, η συλλογή των δεδομένων από τους αισθητήρες να είναι πιο ακριβής.
- Βελτίωση του συστήματος ασύρματης αποστολής δεδομένων ώστε, ο χρόνο απόκρισης του ρομποτικού χεριού από τον τηλεχειρισμό να μειωθεί.
- Χρησιμοποίηση αισθητήρων στα ακροδάχτυλα και στην παλάμη του ρομποτικού χεριού. Μερικά τέτοια αισθητήρια είναι, τα αισθητήρια αφής (*Force Sensing Resistors (FSR)*) που χρησιμοποιούνται για την απόκτηση πληροφοριών σχετικά με την επαφή ανάμεσα στο ρομποτικό χέρι και αντικείμενο. Έτσι υπάρχει ανάδραση από το ρομποτικό χέρι στο γάντι συλλογής δεδομένων.

- Προσαρμογή αισθητήρων προσανατολισμού (*Gyroscope*) και επιταχυνσιόμετρα στο γάντι συλλογής δεδομένων ώστε, με κατάλληλες σχέσης που περιγράφουν το κινηματικό μοντέλο βραχίονα να χειρίζεται ρομποτικό βραχίονα. Έτσι ώστε να αποτελέσει ένα ολοκληρωμένο ανθρωπόμορφο βραχίονα και ρομποτικό χέρι, αλλά και για καλύτερη και πιο ευέλικτη εργασία του ρομποτικού βραχίονα.
- Χρήση του γαντιού συλλογής δεδομένων ώστε, όταν το ρομποτικό χέρι ακουμπήσει κάποιο αντικείμενο ακανόνιστου γεωμετρικού σχήματος να συλλέγει δεδομένα ώστε, να προσδιορίζει την γεωμετρία του αντικειμένου.

## Βιβλιογραφία

- [1] V. Bundhoo, «Design and evaluation of a shape memory alloy-based tendon-driven system for biomimetic artificial fingers,» University of Victoria, Victoria, Canada, 1999.
- [2] «McGraw-Hill Dictionary of Scientific & Technical Terms,» The McGraw-Hill Companies, Inc., 2003.
- [3] N. H. Adnan, K. Wan, A. Shahrman, S. K. Za'ba, H. Desa και M. A. A. Aziz, «The Development of a Low Cost Data Glove by Using Flexible Bend Sensor for Reistive Interfaces,» σε *IMIJS*, Malaysia-Ireland, 2012.
- [4] Chin-Shyurng Fahn και Herman Sun, «Develpoment of a Data Glove With Reducing Sensors Based on Magnetic Induction,» *IEEE Transactions on Industrial Electronics*, τόμ. 52, αρ. 2, pp. 585-594, 2005.
- [5] P. Kumar, J. Verma και S. Prasad, «Hand Data Glove: A Wearable Real-Time Device for Human-Computer Interaction,» *International Journal of Advanced Science and Technology*, τόμ. 43, pp. 15-25, June, 2012.
- [6] J. Weissmann και R. Salomon, «Gesture Recognition for Virtual Reality Applications using Data Gloves and Neural Networks,» σε *IEEE*, 1999.
- [7] C. P. Tung και A. C. Kak, «Automatic Learning of Assembly Tasks using a Dataglove System,» σε *IEEE*, 1995.
- [8] A. Sharma, K. Lewis, V. Ansari και V. Noronha, «Design and Implementation Of Anthropomorphic Robotic Arm,» *Journal of Engineering Research and Applications*, τόμ. 4, αρ. 1 (Version 4), pp. 73-79, January 2014.
- [9] M. Grimm, A. A. Arroyo και M. Nechyba, «A Robotic Hand with Realistic Thumb Pronation,» University of Florida, Florida.
- [10] N. X. Tran, H. Phan, V. V. Dinh, J. Ellen, B. Berg, J. Lum, E. Alcantara, M. Bruch, M. G. Ceruti, C. Kao, D. Garcia, S. Fugate και L. Duffy, «Wireless Data Glove for Gesture-Based Robotic Control,» σε *Human-Computer Interaction, Part II*, Berlin, 2009.

- [11] S. B. Kang και K. Ikeuchi, «A Robot System that Observes and Replicates Grasping Tasks,» σε *IEEE*, 1995.
- [12] I. Oikonomidis, N. Kyriazis και A. A. Argyros, «Efficient Model-based 3D Tracking of Hand Articulations using Kinect,» σε *BMVC*, 2011.
- [13] V. Frati και D. Prattichizzo, «Using Kinect for hand tracking and rendering in wearable haptics,» σε *IEEE World Haptics Conference*, Istanbul, Turkey, 2011.
- [14] «DG5 VHand 3.0,» Virtual Realities, [Ηλεκτρονικό]. Available: <http://www.vrealities.com/products/data-gloves/dg5-vhand-glove-3-0>. [Πρόσβαση 11 August 2014].
- [15] «CyberGlove II Wireless Glove,» CyberGlove Systems, [Ηλεκτρονικό]. Available: [http://www.cyberglovesystems.com/sites/default/files/CyberGloveII\\_Brochure\\_2009.pdf](http://www.cyberglovesystems.com/sites/default/files/CyberGloveII_Brochure_2009.pdf). [Πρόσβαση 11 August 2014].
- [16] S. Cobos, M. Ferre, M. A. S. Uran και J. Ortego, «Constraints for Realistic Hand Manipulation,» *Presence*, pp. 369-370, 2007.
- [17] J. Lin, Y. Wu και T. S. Huang, «Modeling the Constraints of Human Hand Motion,» σε *Human Motion*, Los Alamitos, CA, 2000.
- [18] «5DT Data Glove 5 Ultra,» Fifth Dimension Technologies, [Ηλεκτρονικό]. Available: <http://www.5dt.com/products/pdataglove5u.html>. [Πρόσβαση 11 August 2014].
- [19] «Accele Glove,» AnthroTronix, [Ηλεκτρονικό]. Available: <http://www.metamotion.com/images/AccelerGloveUserGuide.pdf>. [Πρόσβαση 11 August 2014].
- [20] «Virtual Reality P5 Glove,» CyberWorld, Inc., [Ηλεκτρονικό]. Available: [http://www.cwonline.com/store/view\\_product.asp?Product=1179](http://www.cwonline.com/store/view_product.asp?Product=1179). [Πρόσβαση 11 August 2014].
- [21] Γ. Κωνσταντουδάκης, «Σχεδίαση και κατασκευή ανθρωπόμορφου ρομποτικού χεριού,» Τ.Ε.Ι. Κρήτης, Ηράκλειο, 2012.
- [22] Ν. Κριτωτάκης, «Κινηματική ανάλυση και έλεγχος κίνησης-λαβής ανθρωπόμορφου ρομποτικού χεριού,» Τ.Ε.Ι. Κρήτης, Ηράκλειο, 2012.

- [23] S. Cobos, M. Ferre, R. Aracil, J. Ortego και M. A. S. Uran, «Simplified Human Hand Models for Manipulation Tasks,» *InTech*, 2010.
- [24] «Arduino Mega 2560,» Arduino, [Ηλεκτρονικό]. Available: <http://arduino.cc/en/Main/arduinoBoardMega>. [Πρόσβαση 11 Ιουλίου 2014].
- [25] Spectrasymbol, «Flex Sensor Datasheet,» [Ηλεκτρονικό]. Available: <https://www.sparkfun.com/datasheets/Sensors/Flex/flex22.pdf>. [Πρόσβαση 10 Ιουλίου 2014].
- [26] «SparkFun,» SparkFun Electronics (US), [Ηλεκτρονικό]. Available: <https://www.sparkfun.com/products/10264>.
- [27] «Arduino Serial,» Arduino Inc, [Ηλεκτρονικό]. Available: <http://arduino.cc/en/Reference/Serial>.
- [28] X. ZigBee, «XBee datasheet,» [Ηλεκτρονικό]. Available: [http://www.digi.com/pdf/ds\\_xbeemultipointmodules.pdf](http://www.digi.com/pdf/ds_xbeemultipointmodules.pdf).
- [29] «IEEE Standards Association,» [Ηλεκτρονικό]. Available: <http://standards.ieee.org/about/get/802/802.15.html>. [Πρόσβαση 04 Σεπτεμβρίου 2014].
- [30] A. Kurniawan, XBee IEEE 802.15.4 Programming, PE Preaa. Kindle Edition, 2014.
- [31] «XBee Shield Hookup Guide,» Sparkfun Inc., [Ηλεκτρονικό]. Available: <https://learn.sparkfun.com/tutorials/xbee-shield-hookup-guide>.
- [32] «Reference parseInt(),» Arduino, [Ηλεκτρονικό]. Available: <http://arduino.cc/en/Reference/ParseInt>.
- [33] «Simulink 3D Animation Product Description,» Mathworks, [Ηλεκτρονικό]. Available: <http://www.mathworks.com/help/sl3d/product-description.html>.
- [34] «VRML Editors,» The Mathworks, Inc., [Ηλεκτρονικό]. Available: <http://www.mathworks.com/help/sl3d/vrml-editors.html>.
- [35] «Instrument Control Toolbox,» The Mathworks Inc., [Ηλεκτρονικό]. Available: <http://www.mathworks.com/help/instrument/product-description.html>. [Πρόσβαση 11 Ιουλίου 2014].

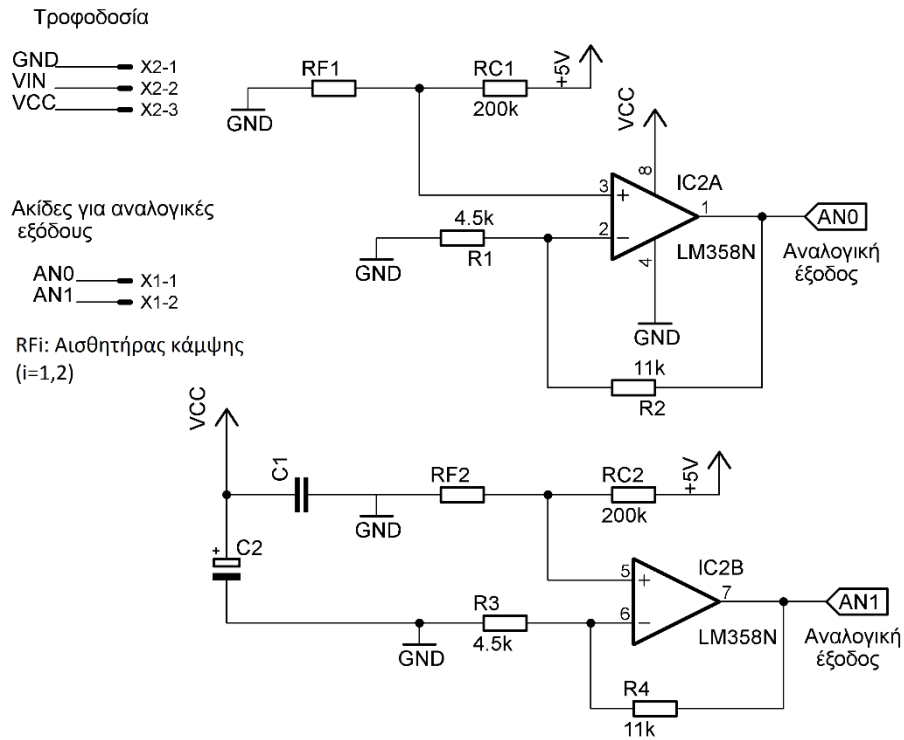


- [36] «Serial Port Overview,» Matworks Inc., [Ηλεκτρονικό]. Available: <http://www.mathworks.com/help/instrument/serial-port-overview.html>.
- [37] Γ. Μηνιάδης, «Εργαστήριο ηλεκτρικών & ηλεκτρονικών κατασκευών,» Τ.Ε.Ι. Κρήτης, Ηράκλειο, 2011.

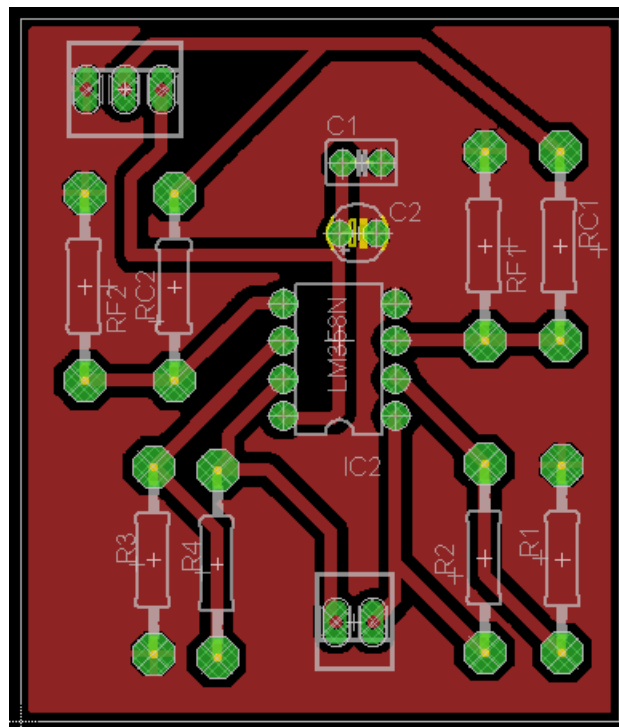
## Παράρτημα Α Πλακέτα τυπωμένου κυκλώματος

### Α.1. Χαλκογραφία πλακέτας τυπωμένου κυκλώματος

Για την χαλκογραφία της πλακέτας θα χρησιμοποιηθεί η μέθοδος του φωτοευαίσθητου υλικού. Αρχικά σχεδιάζετε το κύκλωμα σε CAD (*CadSoft Eagle*) λογισμικό (Εικόνα 0.1, Εικόνα 0.2) [37].



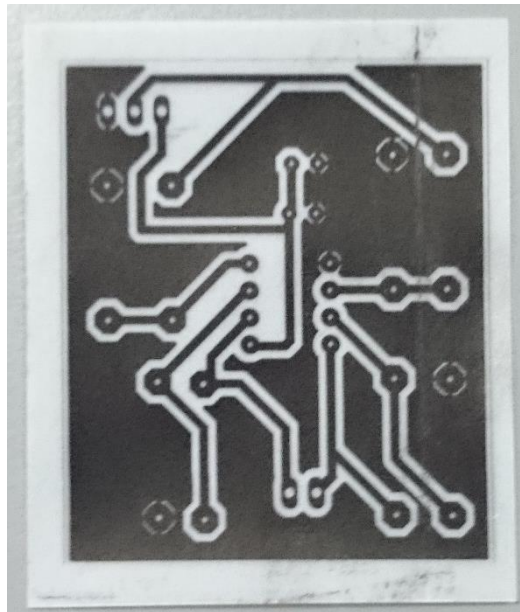
Εικόνα 0.1. Κύκλωμα καταγραφής δεδομένων για δύο αισθητήρες.



Εικόνα 0.2. Πρότυπο σχέδιο PCB πλακέτας για χαλκογραφία.

Η πλακέτα εκτυπώνεται σε ρυζόχαρτο ή κάποιο άλλο διαφανές φύλλο, πριν γίνει η εκτύπωση, ο εκτυπωτής θα πρέπει να ρυθμιστεί έτσι ώστε το εκτυπωμένο σχέδιο να είναι όσο πιο σκούρο γίνεται και να μην έχει ατέλειες. Οι ατέλειες στο σχέδιο χαλκογραφούνται στην

πλακέτα και τότε το κύκλωμα δεν λειτουργεί. Στην εικόνα 0.3 φαίνεται το πρότυπο εκτυπωμένο σχέδιο.



Εικόνα 0.3. Χαρακτηριστικό σχέδιο για χαλκογραφία σε πλακέτα PCB.

Στην συνέχεια, κόβεται η πλακέτα στις διαστάσεις του σχεδίου. Μετά, τοποθετούμε το εκτυπωμένο σχέδιο στην μηχανή έκθεσης φωτός UV. Το φύλλο ρυζόχαρτου πρέπει τοποθετηθεί με τέτοιο τρόπο ώστε η εκτυπωμένη μεριά να ακουμπάει στη επιφάνεια της πλακέτας που θα γίνει η χαλκογραφία (όψη με το χαρακτηριστικό μπλε αυτοκόλλητο). Στην εικόνα 0.4 φαίνεται το μηχάνημα και το φύλλο ρυζόχαρτού.



Εικόνα 0.4. Εκτυπωμένο σχέδιο τοποθετημένο στην μηχανή έκθεσης φωτός UV.

Αφού τοποθετηθεί και η πλακέτα πάνω στο σχέδιο τότε αφήνεται στην UV μηχανή για 2,5 με 3 λεπτά, ώστε το σχέδιο να εφαρμόσει πάνω στην πλακέτα. Στην ουσία με αυτόν τον τρόπο

όπου υπάρχει μαύρη μελάνι εφαρμόζει πάνω στην πλακέτα και προστατεύει τον χαλκό που υπάρχει. Μετά, η πλακέτα τοποθετείται σε μείγμα υδροξειδίου του νατρίου (καυστική σόδα) όπου αφαιρείτε το ανεπιθύμητο φιλμ (επιφάνεια χωρίς μελάνι), η διαδικασία φαίνεται στην παρακάτω εικόνα (Εικόνα 0.5).



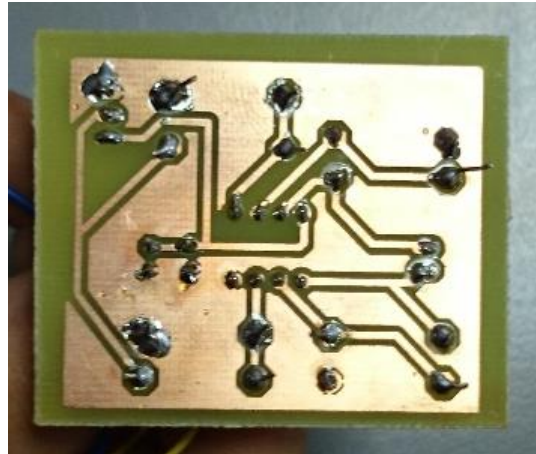
Εικόνα 0.5. Πλακέτα εκτυπωμένου κυκλώματος τοποθετημένο σε μίγμα υδροξειδίου του νατρίου.

Στην συνέχεια, η πλακέτα ξεπλένεται με νερό ώστε να τοποθετηθεί στο επόμενο μηχάνημα (Εικόνα 0.6) όπου περιέχει μίγμα χλωριούχου σιδήρου. Το μίγμα του χλωριούχου σιδήρου αφαιρεί τον ανεπιθύμητο χαλκό που υπάρχει ώστε να βγει η πλακέτα με τα επιθυμητά κύκλωμα. Η πλακέτα θα πρέπει να τοποθετηθεί στο μηχάνημα για αρκετό χρόνο για να αφαιρεθεί ο ανεπιθύμητος χαλκός, αλλά όχι για πάρα πολύ· γιατί τότε θα αφαιρεθεί και ο χαλκός που χρειαζόμαστε για το κύκλωμα.



Εικόνα 0.6. Μηχάνημα που περιέχει χλωριούχο σίδηρο για την αφαίρεση του ανεπιθύμητου χαλκού από την πλακέτα.

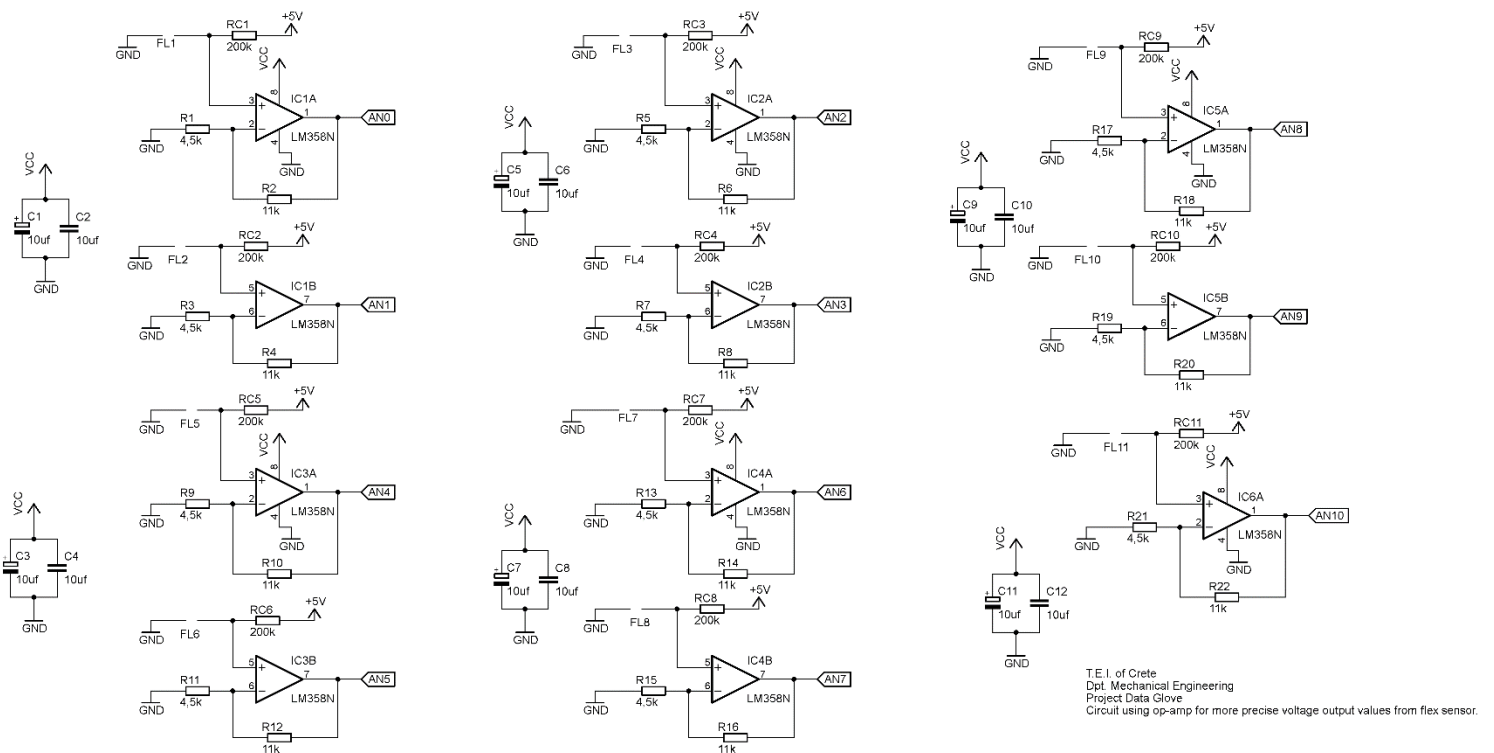
Στην εικόνα 0.7 φαίνεται χαρακτηριστικά πώς πρέπει να μοιάζει η πλακέτα μετά από την αφαίρεση της από το μίγμα.



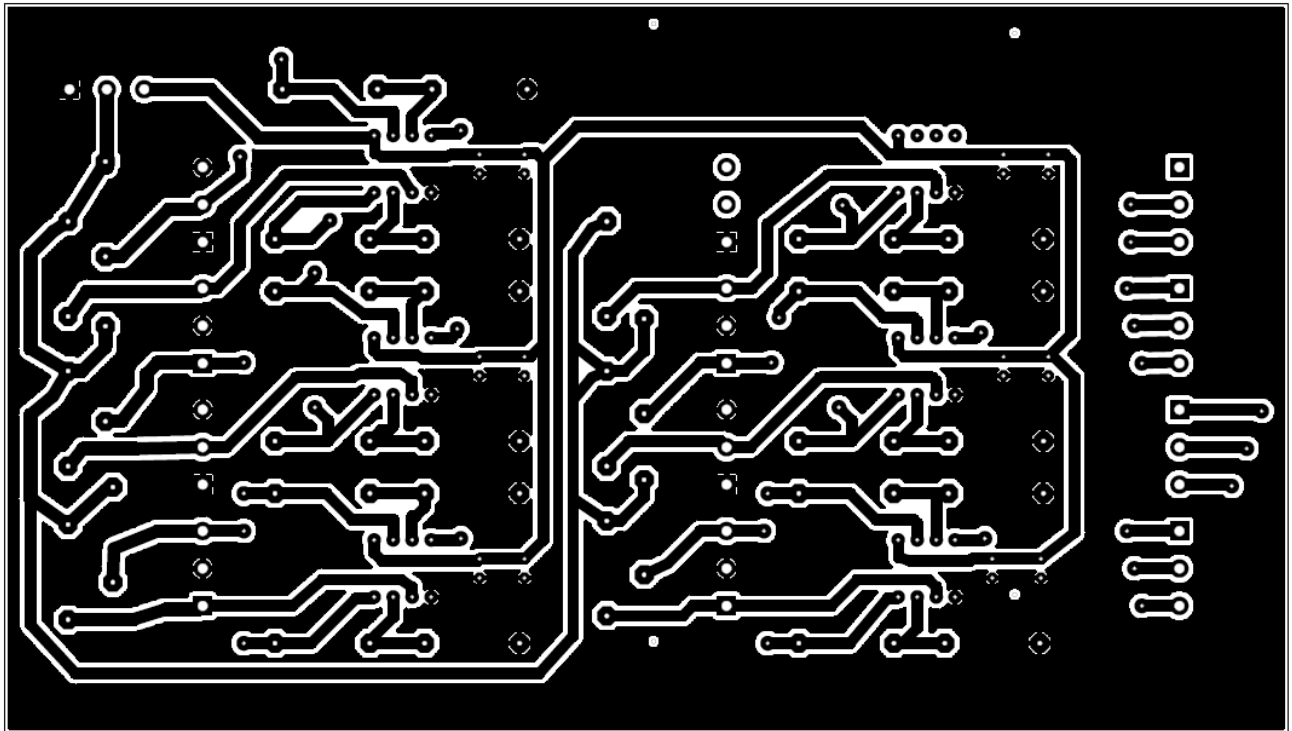
Εικόνα 0.7. Πλακέτα PCB στην τελική φάση.

Έπειτα, με κατάλληλο τρυπάνι γίνονται η τρύπες για την υποδοχή του κάθε στοιχείου. Τέλος, η πλακέτα καθαρίζεται με οινόπνευμα ή ασετόν ώστε να καθαριστεί από τα υπολείμματα και συγκολλούνται τα στοιχεία με καλώς για την σύνδεση τους με το κύκλωμα.

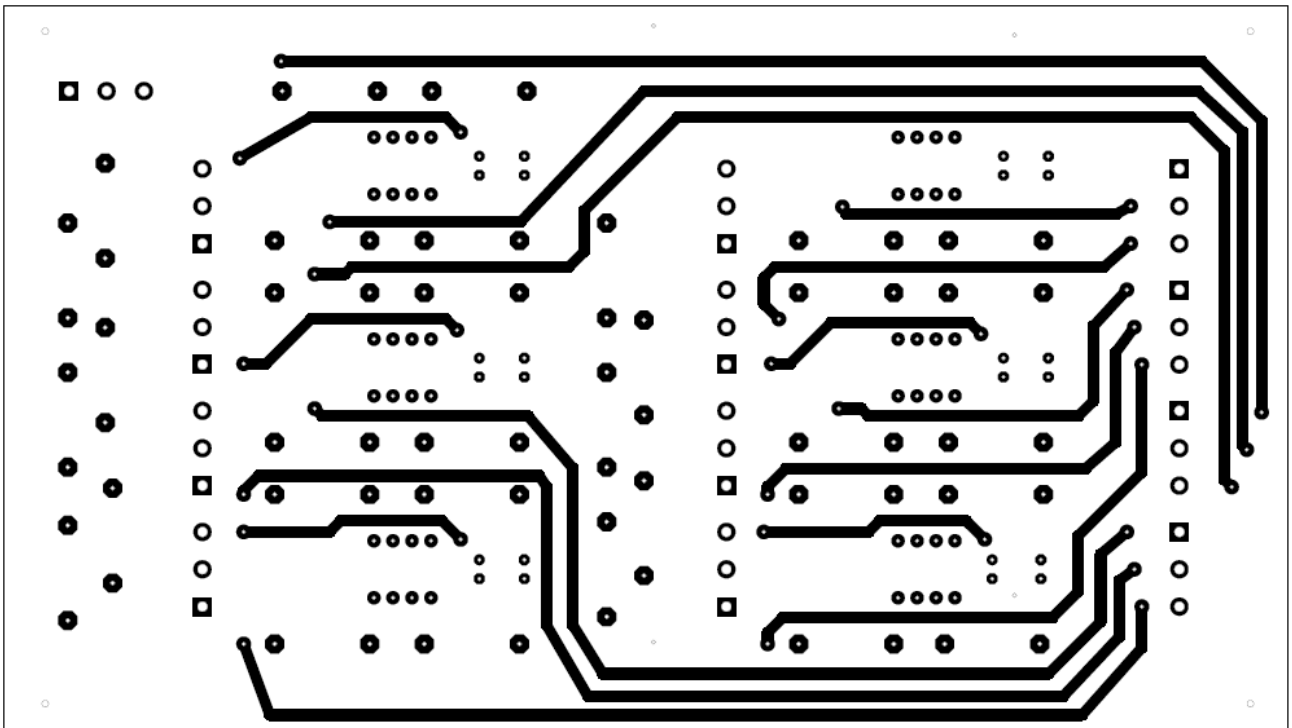
Παρακάτω απεικονίζονται τα τελικά σχέδια από την πλακέτα που κατασκευάστηκε για την ενίσχυση του σήματος σε 11 αισθητήρες. Το πρώτο σχέδιο (Εικόνα 0.8) είναι η σχηματική αναπαράσταση του κυκλώματος ενώ το δεύτερο (Εικόνα 0.9, Εικόνα 0.10) απεικονίζει το κύκλωμα του εκτυπώθηκε. Τέλος, το τρίτο σχέδιο (Εικόνα 0.11) απεικονίζει την τοποθέτηση των στοιχείων πάνω στην πλακέτα.



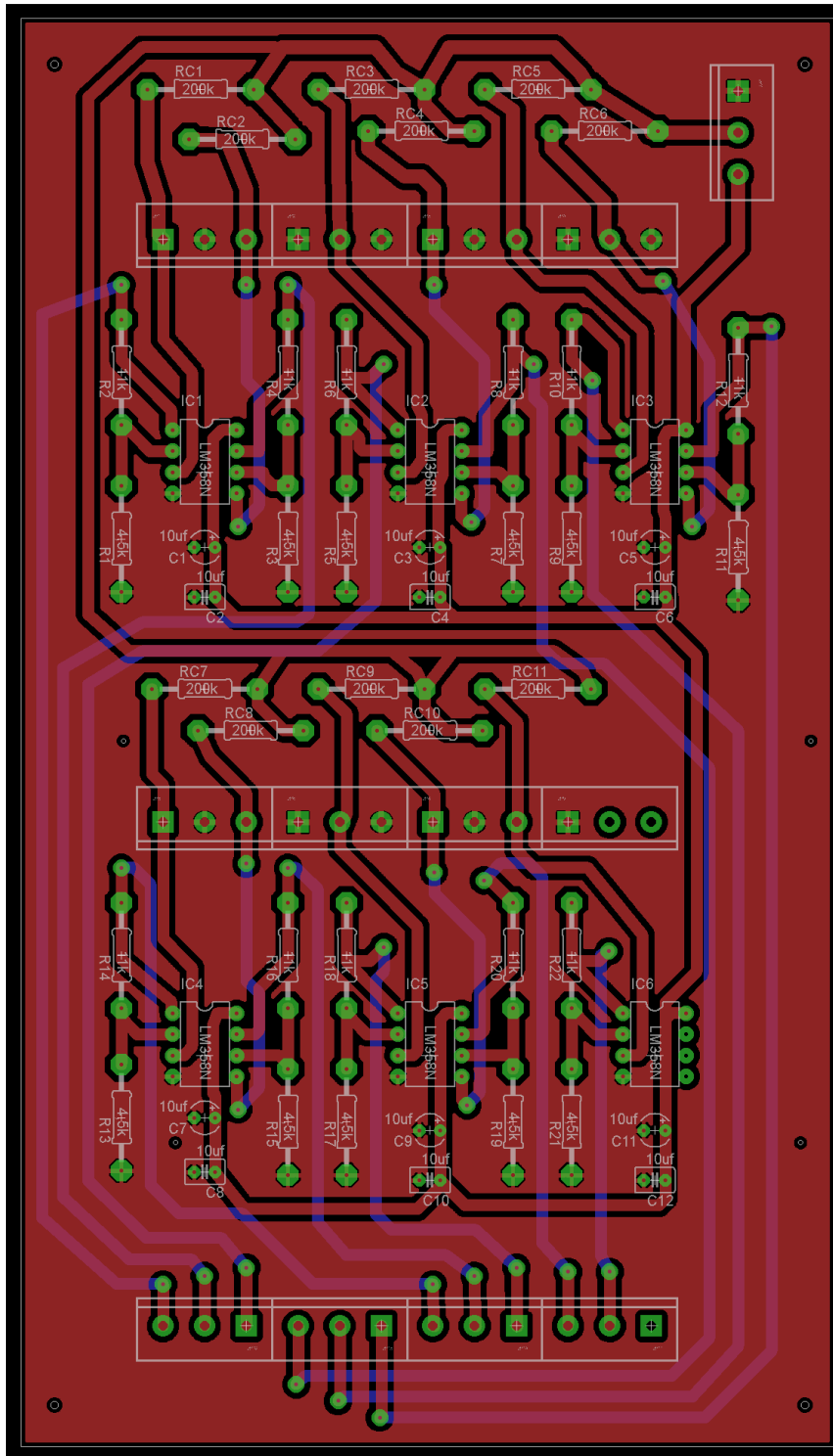
Εικόνα 0.8. Σχηματική αναπαράσταση μη αναστρέφουσας συνδεσμολογίας για την ενίσχυση τάσης εξόδου για 11 αισθητήρες (FLi: αισθητήρες, i=1-11).



Εικόνα 0.9. Σχηματική αναπαράσταση πάνω επιφάνειας πλακέτας.



Εικόνα 0.10. Εικόνα 0.9. Σχηματική αναπαράσταση κάτω επιφάνειας πλακέτας.



Εικόνα 0.11. Σχηματική αναπαράσταση για την τοποθέτηση των στοιχείων.