

Περιεχόμενα

Περίληψη	3
Abstract	3
Εισαγωγή.....	4
1. Ορισμός Ρομποτικού Συστήματος	5
2. Έρευνα Ρομπότ.....	6
2.1. Αλληλεπίδραση Ανθρώπου – Ρομπότ	6
3. Είδη Ρομπότ.....	8
4. Βιομηχανικοί Ρομποτικοί Βραχίονες: Βασικές Έννοιες και Είδη	12
4.1. Βαθμοί Ελευθερίας	14
4.2. Ωφέλιμο Φορτίο – Επαναληψιμότητα – Ακρίβεια	14
4.3. Ταξινόμηση Βραχιόνων Βάσει της Γεωμετρικής Διαμόρφωσής τους	15
4.4. Ρομποτικά Συστήματα Ελέγχου	18
4.5. Κινητήρες - Μοτέρ	18
4.5.1. Είδη Κινητήρων - Μοτερ	19
4.5.2. Οι Προδιαγραφές των Κινητήρων - Μοτέρ	19
A. Τάση Λειτουργίας.....	20
B. Ένταση Ρεύματος	20
C. Ταχύτητα.....	20
D. Ροπή Στρέψης	20
5. Οι Αλγόριθμοι	20
5.1. Η Σημασία των Αλγορίθμων	23
5.2. Χαρακτηριστικά των Αλγορίθμων	24
6. Δίκτυα Υπολογιστών	25
6.1. Τα Δίκτυα Υπολογιστών	26
6.2. Δομικά Στοιχεία Δικτύου	27
6.3. Συσκευές Διασύνδεσης.....	31
6.3.1. Γέφυρες (Bridges)	32
6.3.2. Δρομολογητές (Routers)	33
6.3.3. Πύλες (Gateways)	34
6.3.4. Άλλες Συσκευές.....	34
6.4. Το Πρωτόκολλο Επικοινωνίας TCP/IP	34
6.5. Η Client - Server Τεχνολογία	35
7. Οι Θύρες Επικοινωνίας του Υπολογιστή.....	36
7.1. Σειριακές Θύρες Επικοινωνίας (Serial Ports)	37
7.1.1. RS 232 (Recommended Standard 232).....	37
7.1.2. USB (Universal Serial Bus).....	38
7.1.3. FireWire (IEEE 1394 Interface).....	39

7.1.4. Ethernet.....	39
7.2. Η Παράλληλη Θύρα Επικοινωνίας (Parallel Port – IEEE 1284)	40
8. Περιγραφή της Μεταπτυχιακής Διατριβής	42
8.1. Το Υλικό (Hardware).....	42
8.1.1. Ο Ρομποτικός Βραχίονας.....	42
8.1.2. Πλακέτα 8 Ρελέ (KEMO B210 kit)	46
8.1.3. Η Διασύνδεση του Υλικού	48
8.2. Το Λογισμικό (Software)	50
8.3. Λειτουργία	51
8.3.1. Χρήση με Απευθείας Έλεγχο	53
8.3.2. Χρήση με Εξ' Αποστάσεως Έλεγχο	62
8.4. Σενάριο Μεταφοράς Αντικειμένου Μέσα στο Χώρο.....	65
8.5. Προγραμματισμός – Λειτουργία της Κλάσης.....	79
8.6. Προοπτικές Εξέλιξης της Εργασίας	83
9. Επίλογος.....	84
10. Παράρτημα	86
11. Βιβλιογραφία	98
11.1. Ιστορικές Πηγές	98
11.2. Πηγές Σχετικές με τις Θύρες Επικοινωνίας του Υπολογιστή	98
11.3. Πηγές Σχετικές με τα Δίκτυα Υπολογιστών.....	98
11.4. Πηγές Σχετικές με την Τηλεματική	99
11.5. Άρθρα και Γενικές Πληροφορίες για τη Ρομποτική.....	99
11.6. Πληροφορίες για την Πλακέτα KEMO B-210	99
11.7. Πηγές Σχετικές με την Επιστήμη των Υπολογιστών.....	99
11.8. Πηγές Προγραμματισμού στη Visual Basic 2008 .NET	99

Περίληψη

Η υλοποίηση ενός ρομποτικού συστήματος, βασίζεται στην ικανότητα συνεργασίας λογισμικού (software) και υλικού (hardware). Πίσω όμως από αυτές τις έννοιες (ρομποτικό σύστημα, λογισμικό, υλικό), κρύβεται μία πληθώρα γνωστικών αντικειμένων, τα οποία άρχισαν να αναπτύσσονται με την εμφάνιση της πληροφορικής επιστήμης και πρέπει να μελετηθούν προτού ένα ρομποτικό σύστημα αποκτήσει υπόσταση. Αυτά τα γνωστικά αντικείμενα όπου παρουσιάζονται σε αυτήν τη μεταπτυχιακή διατριβή είναι: βασικές γνώσεις για τους ρομποτικούς βραχίονες, οι κινητήρες ελέγχου τους, οι προδιαγραφές των κινητήρων, οι αλγόριθμοι, τα δίκτυα υπολογιστών, οι συσκευές δικτύωσης, τα πρωτόκολλα επικοινωνίας και οι θύρες επικοινωνίας.

Η μελέτη και η κατανόηση τους είναι το προσόν ώστε ένα ρομποτικό σύστημα να χαρακτηριστεί ως “επιτυχημένο”. Άλλωστε, πρέπει να σημειωθεί, ότι σαν “σύστημα”, οπουδήποτε μπορεί να εμφανιστεί στη φύση, χαρακτηρίζεται η “συνάθροιση οντοτήτων ή αντικειμένων, υλικών ή αφηρημένων, τα οποία αποτελούν σύνολο και το κάθε στοιχείο αλληλεπιδρά ή σχετίζεται με τουλάχιστον ένα στοιχείο του συνόλου...”.

Οπότε, κάθε γνωστικό αντικείμενο που προαναφέρθηκε, αποτελεί μέλος ενός συνόλου (δηλαδή ενός συστήματος) καθώς υπάρχει αλληλεπίδραση μεταξύ αυτών των μελών. Η αρμονική τους συνεργασία για τη δημιουργία ενός αποτελεσματικού συστήματος το οποίο λειτουργεί σύμφωνα με τους κανόνες για τους οποίους έχει σχεδιαστεί, αποτελεί την “πρόκληση” του κάθε μηχανικού που το πεδίο μελέτης του είναι η ρομποτική.

Η παρούσα μεταπτυχιακή διατριβή, είναι μία προσπάθεια προσέγγισης του πεδίου της ρομποτικής, δίνοντας έμφαση όχι τόσο στη “τεχνητή νοημοσύνη” αλλά στην “κίνηση” του ρομποτικού συστήματος, η οποία απαιτείται να είναι προσεκτικά μελετημένη και συναντάται στην πληθώρα των ρομποτικών συστημάτων που χρησιμοποιούνται στη βαριά βιομηχανία.

Abstract

The effectuation of a robotic system is based on the ability of the cooperation between software and hardware. However, behind these terms (robotic system, hardware, software) is hidden a plethora of cognitive disciplines, which started to evolve with the manifestation of the Computer Science and they must be considered before a robotic system obtain substance. The disciplines which are presented in this thesis are: main disciplines of the robotic arms, their motors, specifications of the motors, algorithms, computer networks, networking devices and communication protocols and data communication interface.

The disquisition and the comprehension of these disciplines is the asset in order to classify a robotic system as “successful”. Furthermore, it should be noted that as “system” wherever it may occur in nature, is characterized the “aggregation of entities or objects, which are physical or abstract and they constitute a set and each element interacts or is linked with at least one element of the set.

Whereupon, each subject mentioned above, constitutes a member of a set (system), as there is an interaction among these parts. The harmonious cooperation for the creation of an efficient system which operates according to the rules that it has been designed is a “challenge” for every engineer, whose field of study is Robotics.

This thesis is an attempt to approach the field of Robotics with emphasis not to the artificial intelligence but to the motion of a robotic system, which is required to be carefully studied and is found in the plentitude of the robotic systems which are used in the heavy Industry.

Εισαγωγή

Η παρούσα μεταπτυχιακή διατριβή, έχει σαν σκοπό την επίδειξη συνεργασίας ανάμεσα σε λογισμικό και υλικό (software και hardware αντίστοιχα). Με τον όρο “λογισμικό” περιγράφεται το υπολογιστικό λογισμικό και με τον όρο “υλικό” περιγράφονται τα υλικά στοιχεία που αποτελούν ένα υπολογιστικό ή μηχανικό σύστημα. Χαρακτηριστικό παράδειγμα σύγχρονου μηχανικού συστήματος με υπολογιστική νοημοσύνη αποτελεί το ρομποτικό σύστημα, όπου στη διατριβή που παρουσιάζεται κατέχει περίοπτη θέση.

Ετυμολογικά, η λέξη ρομπότ προέρχεται από τη σλάβικη λέξη “robota” η οποία σημαίνει εργασία. Σε μερικές γλώσσες που βασίζονται στη σλαβική, όπως η Πολωνική γλώσσα, η λέξη “robota” έχει την ελληνική απόδοση “καταναγκαστική εργασία”. Οπότε γίνεται εύκολα αντιληπτό ότι η λέξη “robot” σαν έννοια περιγράφει το μηχανικό εργάτη.

Κάνοντας μία ιστορική αναδρομή, τα ρομπότ εμφανίζονται στην ανθρώπινη ιστορία κατά τους αρχαίους χρόνους. Κανείς δεν μπορεί να διαχωρίσει με ακρίβεια σε τούτες τις ιστορικές αναφορές τα σαφή όρια ανάμεσα στη μυθοπλασία και την πραγματικότητα. Το μόνο που μπορεί να γραφτεί και να ειπωθεί με βεβαιότητα είναι το γεγονός ότι από τα αρχαία χρόνια συναντάται η φιλοδοξία του ανθρώπινου είδους να κατασκευάσει μηχανές οι οποίες θα σκέφτονται και θα ενεργούν. Σύμφωνα με τις περιγραφές και τις εικασίες που έχουν διασωθεί, ο τιτάνας Προμηθέας έπλασε την ανθρωπότητα από πυλό. Ο θεός Ήφαιστος, θεός της τεχνολογίας, της φωτιάς και της μεταλλουργίας στην αρχαία Ελλάδα, είναι ο πρώτος μηχανικός ο οποίος με την ευφυΐα και τα εργαλεία του, αναπτύσσει εφευρέσεις και ολοκληρωμένα μηχανικά συστήματα. Επιγραμματικά, στην Οδύσσεια του Ομήρου, αναφέρεται σχετικά ότι στις δύο πλευρές της εισόδου του παλατιού του Βασιλιά των Φαιάκων, Αλκίνου, υπάρχουν μαλαματένιοι (αργυρόχρυσοι) σκύλοι, έργα του ίδιου του θεού Ήφαιστου οι οποίοι «τον πύργο φυλάγουνε του Αλκίνου του μεγάλου / αθάνατοι κι αγέραστοι για πάντα και για πάντα».

Επίσης μία άλλη μορφή της αρχαίας Ελληνικής ιστορίας, εξίσου αινιγματική με το θεό Ήφαιστο είναι και ο Δαίδαλος. Θεωρείται από τους σπουδαιότερους αρχιτέκτονες, γλύπτες και μηχανικούς και του αποδίδονται εφευρέσεις σπουδαίες όπως πτητικές μηχανές, αρχιτεκτονήματα όπως ο λαβύρινθος και η κατασκευή του μυθικού ρομπότ της αρχαιότητας του Τάλω. Ο Τάλως μπορεί να θεωρηθεί σαν η σημαντικότερη ιστορική αναφορά ρομπότ.

Βέβαια η αρχαία ελληνική μυθολογία δεν μπορεί να κριθεί αντικειμενικά καθώς τα στοιχεία της περιορίζονται σε ιστορικές περιγραφές. Βέβαια, αν ληφθούν σαν αποκυήματα της φαντασίας των αρχαίων συγγραφέων όλες αυτές οι αναφορές, γίνεται εύκολα αντιληπτό ότι τους αρχαίους τους ένοιαζε και τους αφορούσε η τεχνολογική πρόοδος.

Στη σύγχρονη εποχή η εισαγωγή της έννοιας των ρομπότ έγινε από τον Τσέχο θεατρικό συγγραφέα Karel Čapek στο έργο του “R.U.R.” (Rossum's Universal Robots), όπου σατιρίζει την εξάρτηση της κοινωνίας από τους μηχανικούς εργάτες (τα ρομπότ) της τεχνολογικής εξέλιξης και που τελικά εξοντώνουν τους δημιουργούς τους. Σημαντική συνεισφορά στη φιλολογία της ρομποτικής έχει ο Ρώσος επιστήμονας και συγγραφέας Ισαάκ Ασίμοφ ο οποίος διατυπώνει – επινοεί - τους περίφημους “τρεις νόμους της ρομποτικής” με βάση τους οποίους γράφονται και τα διηγήματα του. Αυτοί οι νόμοι είναι:

1. Το ρομπότ δε θα κάνει κακό σε άνθρωπο, ούτε με την αδράνειά του θα επιτρέψει να βλαφτεί ανθρώπινο όν.
2. Το ρομπότ πρέπει να υπακούει τις διαταγές που του δίνουν οι άνθρωποι, εκτός αν αυτές οι διαταγές έρχονται σε αντίθεση με τον πρώτο νόμο.
3. Το ρομπότ οφείλει να προστατεύει την ύπαρξή του, εφόσον αυτό δεν συγκρούεται με τον πρώτο και το δεύτερο νόμο.

Προχωρώντας στις πιο σύγχρονες εποχές, μπορούμε να μιλήσουμε με περισσότερη ασφάλεια. Η κατασκευή ρομπότ είναι πλέον πραγματικότητα και γίνεται κατόπιν της εφεύρεσης των υπολογιστών τη δεκαετία του 1940. Όμως η έννοια του ρομπότ ακολουθείται από μία άλλη έννοια η οποία αναπτύσσεται σε θεωρητικό και πρακτικό επίπεδο σε όλους τους τομείς και ακολουθεί την ανθρώπινη ύπαρξη από τη δημιουργία της, αυτή του “συστήματος”. Η

ολοκληρωμένη έννοια, όπου και συναντάται σήμερα, είναι το “ρομποτικό σύστημα” όπου αποτελεί προέκταση της έννοιας “υπολογιστικό σύστημα”

Στα πανεπιστημιακά εργαστήρια, κυρίως των Η.Π.Α., συγκεντρώνονται λαμπρά επιστημονικά μυαλά και φοιτητές με διάθεση να προσφέρουν τις φρέσκιες ιδέες τους στο νέο αυτό επιστημονικό κλάδο που συνεχώς εξελίσσεται. Εκεί γίνονται οι πρώτες σημαντικές προσπάθειες ανάπτυξης ρομποτικών συστημάτων. Ένα από τα πρώτα ολοκληρωμένα ρομποτικά συστήματα που παρουσιάζονται είναι ο Σέικι του Stanford Research Institute (Η.Π.Α.), στα τέλη της δεκαετίας του 1960. Το εν λόγω ρομπότ ήταν σε θέση να τοποθετεί τουβλάκια σε κατακόρυφες στήλες, έχοντας μία βιντεοκάμερα ως οπτικό αισθητήρα και έναν Η/Υ ως κεντρική μονάδα για την επεξεργασία των πληροφοριών που λάμβανε. Την ίδια εποχή παρουσιάζονται τα πρώτα εμπορικά βιομηχανικά ρομποτικά συστήματα από τις εταιρείες Unimation και Fuji τα οποία περιορίζονταν στην παλετοποίηση (την τοποθέτηση σε παλέτες) της εκάστοτε παραγωγής. Τη δεκαετία του 1970 εμφανίζονται όλο και πιο σύνθετα βιομηχανικά ρομποτικά συστήματα που αποτελούν γνήσιους προγόνους των σημερινών.

Τα επόμενα όμως χρόνια που ακολούθησαν, η μελέτη της κίνησης στα ρομποτικά συστήματα πέρασε σε δεύτερη μοίρα, καθώς οι ερευνητές εστίασαν το ενδιαφέρον και τις προσπάθειες τους στην επίτευξη τεχνητής νοημοσύνης. Σε αυτά τα χρόνια που ακολούθησαν έγιναν πιο σαφή τα δύο ρεύματα ρομποτικής που παρουσιάστηκαν. Το ένα που είχε σαν στόχο και σκοπό τη μελέτη και την ανάπτυξη νοημόνων ρομποτικών συστημάτων και ένα δεύτερο που είχε σκοπό την ανάπτυξη ρομποτικών βιομηχανικών συστημάτων δίνοντας έμφαση στη ρομποτική κίνηση και τον καλύτερο τρόπο διασύνδεσης και επικοινωνίας με τον άνθρωπο – χειριστή. Τα δείγματα γραφής του δεύτερου ρεύματος είναι συνεχή και ολοένα εξελισσόμενα. Όσον αναφορά το ρεύμα με τα νοήμονα ρομποτικά – υπολογιστικά συστήματα τα πρώτα σημαντικά αποτελέσματα εμφανίζονται στους πιο σύγχρονους καιρούς και δη το 1997 όπου το νοήμον υπολογιστικό σύστημα της IBM ο Deep Blue κερδίζει σε σκακιστικό αγώνα τον τότε παγκόσμιο πρωταθλητή Γκάρι Κασπάροφ.

Πλέον η πρόοδος είναι αλματώδης και το εγγύς μέλλον υπόσχεται την αρμονική συνεργασία των δύο αυτών ρομποτικών ρευμάτων.

1. Ορισμός Ρομποτικού Συστήματος

Η “ρομποτική” είναι μία μορφή της μηχανικής επιστήμης και προκύπτει από το συνδυασμό επιστημών όπως της ηλεκτρονικής, της μηχανολογίας και της πληροφορικής. Η λέξη “σύστημα” καταγράφεται πλέον σε πολλές επιστήμες. Από την πολιτική επιστήμη για την περιγραφή κοινωνικών δομών, την ιατρική επιστήμη και σε κάθε πτυχή της για την περιγραφή και επεξήγηση του ανθρώπου, στην αστρονομία για την περιγραφή των κανόνων του σύμπαντος και γενικά στο καθετί. Οπότε φαντάζει αδιανόητο να απουσιάζει από τη μηχανική επιστήμη.

Κατά έναν επιστημονικό ορισμό, “σύστημα” αποτελεί “*συνάθροιση οντοτήτων ή αντικειμένων, υλικών ή αφηρημένων, τα οποία αποτελούν σύνολο και το κάθε στοιχείο αλληλεπιδρά ή σχετίζεται με τουλάχιστον ένα ακόμη στοιχείο του συνόλου. Κάθε αντικείμενο που δεν σχετίζεται ή αλληλεπιδρά με τουλάχιστον ένα ακόμη στοιχείο του συστήματος δεν αποτελεί μέρος του συστήματος*”. Ένα “υποσύστημα” είναι ένα σύνολο στοιχείων το οποίο αποτελεί σύστημα από μόνο του και αποτελεί υποσύνολο του όλου συστήματος.

Σύμφωνα με το Αμερικάνικο Ινστιτούτο Ρομποτικής (Robot Institute of America), ως ρομποτικό σύστημα μπορεί να περιγραφεί “*ένας μηχανισμός κατάλληλα σχεδιασμένος ώστε, μέσω προγραμματιζόμενων κινήσεων, να μεταφέρει υλικά, τεμάχια, εργαλεία ή ειδικευμένες συσκευές, με σκοπό την επίτευξη ποικιλίας εργασιών*”.

Ένας τέτοιος μηχανισμός περιλαμβάνει συνήθως τις ακόλουθες συνιστώσες:

- **Ένα μηχανολογικό υποσύστημα**, το οποίο ενσωματώνει τη δυνατότητα του ρομπότ για εκτέλεση έργου. Το συγκεκριμένο υποσύστημα αποτελείται από μηχανισμούς που επιτρέπουν στο ρομπότ να κινείται όπως αρθρώσεις, συστήματα μετάδοσης κίνησης, κινητήρες κτλ.

- **Ένα υποσύστημα αίσθησης**, μέσω του οποίου το ρομπότ συγκεντρώνει πληροφορίες για την κατάσταση στην οποία βρίσκονται τόσο το ίδιο όσο και το περιβάλλον του. Το εν λόγω υποσύστημα εκτός των άλλων είναι υπεύθυνο για την αποδοχή των εξωτερικών εντολών, την επεξεργασία τους, τη μετάφραση τους σε ηλεκτρική ισχύ που θα δοθεί στους κινητήρες του ρομπότ, καθώς επίσης και για την παραγωγή σημάτων εξόδου που θα πληροφορούν για την κατάσταση του συστήματος. Στο υποσύστημα αίσθησης περιλαμβάνονται όργανα μετρήσεως, αισθητήρες, ηλεκτρονικά στοιχεία κλπ.
- **Ένα σύστημα ελέγχου**, το οποίο συνδυάζει κατάλληλα την “αίσθηση με τη δράση”, έτσι ώστε το ρομπότ να λειτουργεί αποτελεσματικά με τον επιθυμητό τρόπο. Ο ελεγκτής του ρομπότ επιβλέπει και συντονίζει ολόκληρο το σύστημα. Για τη σχεδίαση και την υλοποίηση του απαιτείται ο συνδυασμός γνώσεων από πολλές γνωστικές περιοχές όπως είναι ο αυτόματος έλεγχος, η τεχνητή νοημοσύνη, η επιστήμη των υπολογιστών κλπ.

2. Έρευνα Ρομπότ

Η ρομποτική επιστήμη σε καμία περίπτωση δεν μπορεί να θεωρηθεί ότι έχει φτάσει στο ζενίθ της. Αντιθέτως, τώρα μπορούμε να θεωρήσουμε ότι έχει ξεκινήσει να αναπτύσσεται με όλους τους τομείς που την αφορούν να ξεκινούν από το ίδιο σημείο γνώσης, κάτι που δεν γινόταν παλιότερα, καθώς άλλοι τομείς όπως η ρομποτική κίνηση, παραχώρησαν τη θέση τους στην έρευνα και την ανάπτυξη άλλων τομέων, όπως της τεχνητής νοημοσύνης.

Η ανάπτυξη των ρομπότ μπορεί να επιτευχθεί με πολύ ταχύτερους ρυθμούς εάν υπάρχουν **“ρομποτικά συστήματα ανοιχτού κώδικα και επαναστατικές ιδέες που δεν γίνονται αντικείμενο μονόπλευρης έρευνας και ανάπτυξης από τις εταιρίες που αποσκοπούν καθαρά στο κέρδος”**, αναφέρει ο επιστήμονας και καθηγητής Hans Moravec του Carnegie Mellon University Robotics Institute, σε μία διάλεξη που έδωσε το 1997 για να περιγράψει την εξέλιξη του κλάδου. Ο καθηγητής Moravec ανέφερε και πρόβλεψε ότι, η **πρώτη γενιά των ρομπότ** τοποθετείται χρονολογικά γύρω 2010 (τωρινή εποχή) και θα βρίσκεται σε ένα αρχικό στάδιο μαθησιακών δυνατοτήτων. Η **δεύτερη γενιά των ρομπότ** θα είναι διαθέσιμη περί το 2020, θα είναι η εξέλιξη της πρώτης γενιάς και η νοημοσύνη των ρομπότ αυτής της γενιάς θα είναι συγκρίσιμη με τη νοημοσύνη που έχει ένα ποντίκι. Η **τρίτη γενιά** χαρακτηρίζεται ως “επαναστατική γενιά” και η νοημοσύνη των ρομπότ θα φτάνει αυτή του πιθήκου και η **τέταρτη γενιά** θα φτάνει σε νοημοσύνη τον άνθρωπο. Η τρίτη και η τέταρτη γενιά τοποθετούνται χρονικά γύρω στο 2040 με 2050.

Όποια και να είναι η εξέλιξη των ρομποτικών συστημάτων, ήδη χρησιμοποιούνται ευρέως. Είναι γνώριμη σε όλους μας η εικόνα της αυτοματοποιημένης γραμμής παραγωγής και είναι συνυφασμένη με τομείς όπως η αυτοκινητοβιομηχανία, η κλωστούφαντουργία, η βιομηχανία ηλεκτρικών και ηλεκτρονικών ειδών κτλ.

Σε αντιδιαστολή με τα προαναφερθέντα, βρίσκονται τα ευφυή και αυτόνομα ρομπότ - συστήματα, αυτά δηλαδή που δεν εκτελούν μία αλληλουχία προγραμματισμένων κινήσεων (όπως αυτά που λειτουργούν στις γραμμές παραγωγής), αλλά μηχανών που είναι σε θέση να παίρνουν αποφάσεις, να αυτενεργούν, να μαθαίνουν, και γενικά να επιτελούν εκείνες τις λειτουργίες που κατά κανόνα αποδίδουμε στην ανθρώπινη φύση.

2.1. Αλληλεπίδραση Ανθρώπου – Ρομπότ

Τα ρομπότ τα οποία προορίζονται για οικιακή χρήση, απευθύνονται σε ανθρώπους που δεν έχουν ειδικές γνώσεις επάνω στη ρομποτική. Γι' αυτό το λόγο πρέπει το σύστημα αλληλεπίδρασης ανάμεσα στον άνθρωπο και το ρομπότ να είναι όσο το δυνατόν απλούστερο. Δεν έχει αναπτυχθεί ένα συγκεκριμένο σύστημα αλληλεπίδρασης, αλλά ποικιλία τέτοιων συστημάτων που βασίζονται σε διαφορετικές τεχνικές, οι οποίες εκτός του ουσιαστικού αποτελέσματος επιτυγχάνουν και ένα διαισθητικό αποτέλεσμα. Τα ρομπότ αποκτούν όλο και περισσότερες ικανότητες “επικοινωνίας” με έναν μέσο χρήστη μέσω της ομιλίας που μπορούν να διαθέτουν, των χειρονομιών και εκφράσεων που μπορούν να σχηματίσουν στο πρόσωπό τους (Σχήμα 1).

Για τον άνθρωπο ο φυσιολογικός τρόπος επικοινωνίας με τους άλλους ανθρώπους είναι ο λόγος (είτε προφορικός είτε γραπτός). Κάτι τέτοιο όμως δεν συμβαίνει και με τα ρομπότ, όπου ο τρόπος επικοινωνίας τους περιορίζεται στη γραμμή εντολών που δέχονται. Βέβαια τα τελευταία χρόνια η επικοινωνία ανθρώπου – ρομπότ έχει εξελιχθεί και γίνεται με βάση την τεχνητή νοημοσύνη. Έτσι τα ρομπότ είναι σε θέση να καταλαβαίνουν τους ανθρώπους και να αντιδρούν κατάλληλα, στα πλαίσια που και τα ίδια έχουν διδαχθεί, μέσω της αναγνώρισης φωνής, χειρονομιών και εκφράσεων. Η συνέχεια σε όλα αυτά που έχουν αναφερθεί είναι η επίτευξη τεχνητών συναισθημάτων (Artificial emotions). Το στοίχημα των επιστημόνων για τα επόμενα χρόνια είναι τα ρομπότ μέσα από τις επιδράσεις που δέχονται από το περιβάλλον στο οποίο υπάρχουν, να μπορούν να αναπτύξουν τη δικιά τους ξεχωριστή προσωπικότητα. Με βάση την προηγούμενη διατύπωση, εύλογα γεννάται το ακόλουθο ερώτημα: “Μπορούν οι μηχανές στο μέλλον να φτάσουν σε επίπεδο ώστε να μπορούν να σκέφτονται;”. Αρχικά πρέπει να ξεκαθαρισθεί η διαφορά των όρων “σκέφτομαι” και “νοημοσύνη”. Αντικειμενική απάντηση, όπως είναι προφανές, δεν μπορεί να υπάρξει καθώς είναι ένας φιλοσοφικός διαχωρισμός που έχει απασχολήσει το σκεπτόμενο κόσμο εδώ και αιώνες και δεν είναι εύκολο να αναλυθεί.



Σχήμα 1: Εκφράσεις προσώπου που σχηματίζονται σε ένα ρομπότ

Παρ’ όλα αυτά είναι ξεκάθαρο ότι η νοημοσύνη έχει πολλές πλευρές, όπως η ικανότητα να σχεδιάζει, να μαθαίνει, να λύνει προβλήματα και να χρησιμοποιεί γλώσσα. Συμπεριφορά που επιδεικνύει οποιαδήποτε από αυτές τις ικανότητες θεωρείται και χαρακτηρίζεται από τους περισσότερους ανθρώπους “ευφυής” και σκοπός όσων εργάζονται επάνω στην τεχνητή νοημοσύνη είναι να κάνουν τους ηλεκτρονικούς υπολογιστές να επιδείξουν μία τέτοια συμπεριφορά.

Η συμπεριφορά είναι το βασικό κριτήριο νοημοσύνης που έθεσε ο Turing, για το αν μία μηχανή είναι ευφυής ή όχι. Για το λόγο αυτό, εδώ και περίπου 35 χρόνια υπάρχει το “test Turing” και η ουσία του τεστ αυτού είναι ο καθορισμός για το αν μία μηχανή είναι ευφυής ή όχι. Για το λόγο αυτό, διεξάγεται ένας αυθαίρετος διάλογος ανάμεσα σε έναν άνθρωπο και μία μηχανή και ένας παρατηρητής προσπαθεί να συμπεράνει ποιος είναι ο άνθρωπος και ποια η μηχανή. Για προφανείς λόγους και ο άνθρωπος και η μηχανή δεν είναι θεατοί στον παρατηρητή, ενώ το μέσο επικοινωνίας είναι ουδέτερο (π.χ. μηνύματα μέσω ενός πληκτρολογίου). Αν οι απαντήσεις της μηχανής σε τυχαίες δηλώσεις και ερωτήσεις είναι τέτοιες ώστε να μην μπορούν να διακριθούν με σιγουριά από τις ανθρώπινες, τότε το συμπέρασμα από τη δοκιμή Turing είναι ότι η μηχανή είναι πράγματι ευφυής. Βέβαια, αυτή είναι μόνο μία προσπάθεια για τον ορισμό της νοημοσύνης και έχει υποστεί πολλές κριτικές, καθώς η εξέταση αυτή αγνοεί το τι συμβαίνει μέσα στη μηχανή και αναφέρεται μόνο σε μία πλευρά της συμπεριφοράς της μηχανής.

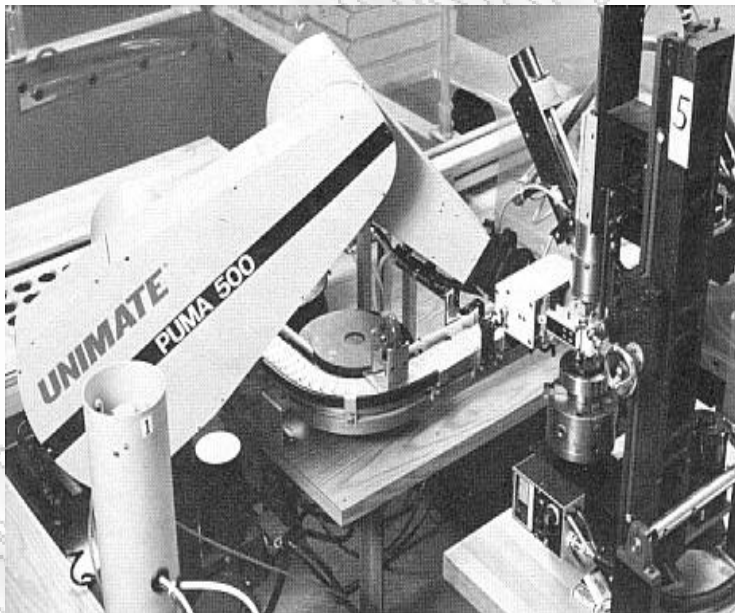
Παρακάτω αναφέρεται αποσπασματικά ένας πραγματικός διάλογος που έγινε σε ένα test Turing στα μέσα της δεκαετίας του ’60. Το πρόγραμμα ονομάζεται ELIZA και γράφτηκε από τον Joseph Weizenbaum. Το πρόγραμμα μιμείται τις αποκρίσεις ενός ψυχιάτρου κατά τη διάρκεια συζήτησης με έναν ασθενή (Οι παρατηρήσεις του ασθενή σημειώνονται με μικρά γράμματα, ενώ οι αποκρίσεις της ELIZA με κεφαλαία).

- *Οι άνδρες είναι όλοι ίδιοι;*
- *ΚΑΤΑ ΚΑΠΟΙΟ ΤΡΟΠΟ.*
- *Πάντα μας ενοχλούν για το ένα ή το άλλο θέμα.*
- *ΜΠΟΡΕΙΣ ΝΑ ΣΚΕΦΤΕΙΣ ΕΝΑ ΣΥΓΚΕΚΡΙΜΕΝΟ ΠΑΡΑΔΕΙΓΜΑ;*
- *Λοιπόν, ο φίλος μου με έκανε να έρθω εδώ.*
- *Ο ΦΙΛΟΣ ΣΟΥ ΣΕ ΕΚΑΝΕ ΝΑ ΕΡΘΕΙΣ ΕΔΩ;*
- *Λέει ότι είμαι θλιμμένη τον περισσότερο καιρό.*
- *ΛΥΠΑΜΑΙ ΠΟΥ ΑΚΟΥΩ ΟΤΙ ΕΙΣΑΙ ΘΛΙΜΜΕΝΗ*
- *Είναι αλήθεια, είμαι δυστυχισμένη.*
- *ΝΟΜΙΖΕΙΣ ΟΤΙ Ο ΕΡΧΟΜΟΣ ΣΟΥ ΕΔΩ ΘΑ ΣΕ ΒΟΗΘΗΣΕΙ ΩΣΤΕ ΝΑ ΜΗΝ ΕΙΣΑΙ ΔΥΣΤΥΧΙΣΜΕΝΗ;*
- *Χρειάζομαι κάποια βοήθεια, αυτό είναι το μόνο σίγουρο.*

Από αυτό το παράδειγμα φαίνεται ότι είναι δύσκολο να ξεχωρίσουμε τη συμπεριφορά της ELIZA από εκείνη των ψυχιάτρων μίας συγκεκριμένης σχολής και ότι το πρόγραμμα ικανοποιεί τη δοκιμή Turing μέσα σε ένα περιορισμένο πεδίο συζήτησης. Όμως, μία προσεχτικότερη εξέταση του προγράμματος δείχνει ότι οι αποκρίσεις της ELIZA δεν είναι τίποτα περισσότερο από μία μελετημένη “ηχώ” των δεδομένων του ασθενή.

3. Είδη Ρομπότ

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



Σχήμα 2: Βιομηχανικός Ρομποτικός Βραχίονας

Τα σπουδαιότερα είδη ρομπότ είναι τα παρακάτω:

- **Ρομπότ Σταθερής Βάσης:** τα ρομπότ αυτά αποτελούνται από διαδοχικά στερεά σώματα (σύνδεσμοι) που συνδέονται μέσω αρθρώσεων σχηματίζοντας μία κινηματική αλυσίδα. Η αλυσίδα αυτή έχει το ένα άκρο της (τη βάση) σταθερά συνδεδεμένο με κάποιο σημείο του περιβάλλοντος χώρου. Η μορφή αυτή ρομπότ είναι η παραδοσιακή

μορφή ενός βιομηχανικού ρομποτικού βραχίονα, και περιλαμβάνει το βραχίονα, τον καρπό και το εργαλείο (Σχήμα 2).

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

1. **AGV's:** τα AGV's (Automatic Guided Vehicles) έχουν περιορισμένη αυτονομία κίνησης, δεδομένου ότι η τροχιά τους είναι προκαθορισμένη μέσω καλωδίων στο έδαφος ή πομπών στον περιβάλλοντα χώρο (Σχήμα 3).

2. **Αυτόνομα Έντροχα Ρομπότ:** τα ρομπότ αυτά λειτουργούν με αρκετά υψηλό βαθμό αυτονομίας. Πιο συγκεκριμένα μπορούν και λειτουργούν χωρίς συνεχή εξωτερική επίβλεψη και είναι ικανά να εκτελούν εργασίες αυτόνομα δεχόμενα μόνο ορισμένες υψηλού επιπέδου εντολές (Σχήμα 4).



Σχήμα 3: AGV σε Βιομηχανικό Περιβάλλον



Σχήμα 4: Αυτόνομο Έντροχο Ρομπότ

3. **Βαδίζοντα Ρομπότ:** τα ρομπότ αυτά χρησιμοποιούν μηχανικά πόδια για την κίνησή τους και όχι συμβατικούς τροχούς όπως στις προηγούμενες δύο κατηγορίες. Τα κυριότερα πλεονεκτήματα της συγκεκριμένης υλοποίησης είναι η μεγάλη δυνατότητα αποφυγής εμποδίων και η ικανότητα αναρρίχησης σε ανώμαλα εδάφη και μη επίπεδες επιφάνειες. Από τα πιο συνηθισμένα ρομπότ αυτής της κατηγορίας είναι τα δίποδα ενώ δεν αποκλείονται και εφαρμογές με περισσότερα από δύο πόδια, π.χ. ρομπότ που μοιάζουν και κινούνται όπως οι αράχνες (Σχήμα 5).



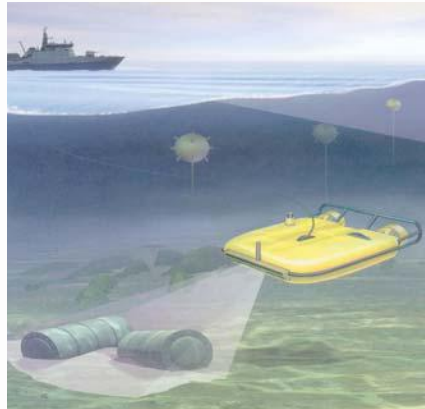
Σχήμα 5: Ο Dante II του Εργαστηρίου JPL της NASA κατά τη διάρκεια ανάβασης σε βουνό της Αλάσκα.

4. **ROV's:** τα ROVs (Remotely Operated Vehicles) ανήκουν στην κατηγορία των μη επανδρωμένων υποβρύχιων ρομπότ. Όπως υποδηλώνει το όνομά τους δεν έχουν μεγάλο βαθμό αυτονομίας, μιας και είναι συνδεδεμένα με το μητρικό πλοίο μέσω καλωδίου, το οποίο και καλύπτει τις ανάγκες του ρομπότ σε ενέργεια και επικοινωνίες. Τα ρομπότ αυτού του τύπου έχουν σχήμα κουτιού και κινούνται γενικά σε χαμηλές ταχύτητες (Σχήμα 6).



Σχήμα 6: Απόδοση ενός ROV

5. **AUVs:** τα AUVs (Autonomous Underwater Vehicles), αντίθετα με τα ROVs, είναι πλήρως αυτόνομα και κατά συνέπεια δεν έχουν την ανάγκη καλωδίου. Για τις ανάγκες τροφοδοσίας (ενέργεια) χρησιμοποιούνται ειδικές μπαταρίες, κάτι όμως που θέτει και περιορισμούς στη λειτουργία των ρομπότ αυτών. Τα AUVs έχουν σχήμα τορπιλών και μπορούν να κινούνται με αρκετά μεγάλες ταχύτητες (Σχήμα 7).



Σχήμα 7: Απόδοση ενός AUV

6. **Εναέρια ρομπότ:** πρόκειται για μη επανδρωμένα ιπτάμενα ρομπότ, όπως ελικόπτερα και αεροπλάνα. Τα ρομπότ αυτά έχουν διαρκώς αυξανόμενες εφαρμογές και η ανάπτυξη τους ξεκίνησε αρχικά για στρατιωτικούς σκοπούς (Σχήματα 8 και 9).



Σχήμα 8: Ρομποτικό ελικόπτερο



Σχήμα 9: Το ρομπότ Helios της NASA

7. **Έρποντα ρομπότ:** Όπως μαρτυράει και η ονομασία τους πρόκειται για ρομπότ τα οποία έρπονται στο έδαφος (Σχήμα 10). Μιμούνται την κίνηση των ερπετών και δη των φιδιών. Μπορούν να πορευθούν με άνεση σε κλειστούς και περιορισμένους χώρους. Η εφαρμογή τους γίνεται από τα σωστικά συνεργεία και συναντάται στην ανίχνευση παγιδευμένων ανθρώπων σε συντρίμια και ερείπια.

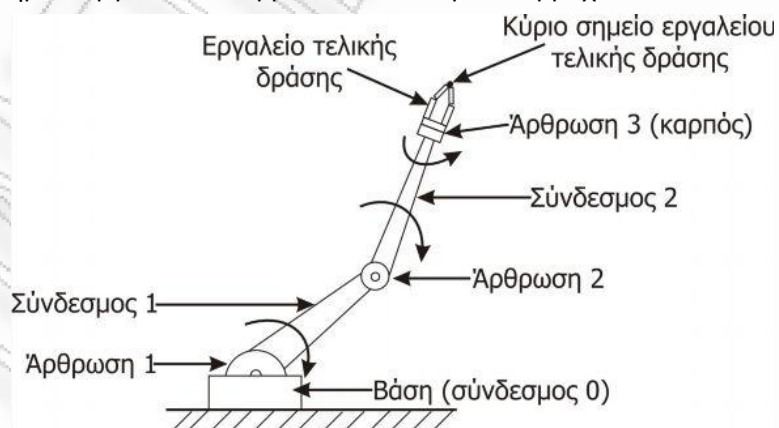


Σχήμα 10: Ρομπότ – ερπετό, που πορεύεται εξίσου τόσο στο έδαφος όσο και στο νερό

Όλες οι παραπάνω κατηγορίες ρομπότ αποτελούνται από τα βασικά υποσυστήματα που έχουν ήδη αναφερθεί.

4. Βιομηχανικοί Ρομπωτικοί Βραχίονες: Βασικές Έννοιες και Είδη

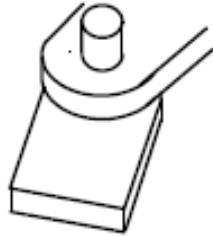
Ένας ρομπωτικός βραχίονας (ρομπότ σταθερής βάσης) αποτελείται από το κυρίως σώμα (δηλαδή το κορμό) του ρομπότ και από μια διαδοχή συνδέσμων ή μελών (links) που ενώνονται μεταξύ τους με αρθρώσεις ή ενώσεις (joints). Οι αρθρώσεις (ή ενώσεις) ελέγχουν τις κινήσεις των συνδέσμων (Σχήμα 11). Η ομάδα των αρθρώσεων που ελέγχουν την κίνηση του εργαλείου καθεαυτού που κρατάει ο βραχίονας υπάγονται στο καρπό. Κάθε μια από τις αρθρώσεις του βραχίονα και του καρπού παρέχει ένα βαθμό ελευθερίας στην κίνηση του τελικού στοιχείου δράσης. Ως τελικό στοιχείο δράσης ορίζεται το κατάλληλα σχεδιασμένο και προσαρμοσμένο μηχανικό εξάρτημα – εργαλείο, που βρίσκεται στο άκρο του βραχίονα.



Σχήμα 11: Αναπαράσταση ρομπωτικού βραχίονα

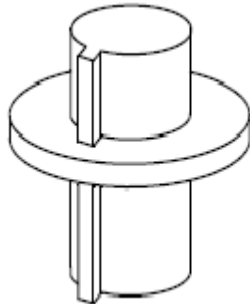
Οι αρθρώσεις μπορεί να είναι:

- **Πρισματικές:** σχετική μεταφορική κίνηση μεταξύ δύο διαδοχικών συνδέσμων (Σχήμα 12),



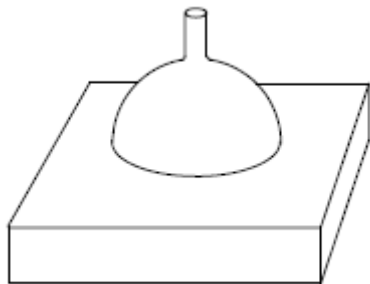
Σχήμα 12: Πρισματική άρθρωση

- **Περιστροφικές:** υλοποιούν σχετική περιστροφική κίνηση μεταξύ δύο διαδοχικών συνδέσμων (Σχήμα 13),



Σχήμα 13: Περιστροφική άρθρωση

- **Σφαιρικές:** υλοποιούν σφαιρική περιστροφική κίνηση μεταξύ δύο διαδοχικών συνδέσμων και παρέχουν στην κατασκευή από έναν βαθμό κινητικότητας (Σχήμα 14).
Με τη σειρά της, μία κινηματική αλυσίδα χαρακτηρίζεται ως ανοικτή όταν υπάρχει μία μόνο διαδοχή συνδέσμων που να συνδέει τα δύο άκρα του βραχίονα και κλειστή όταν οι σύνδεσμοι που τη συνιστούν σχηματίζουν βρόχο.



Σχήμα 14: Σφαιρική άρθρωση

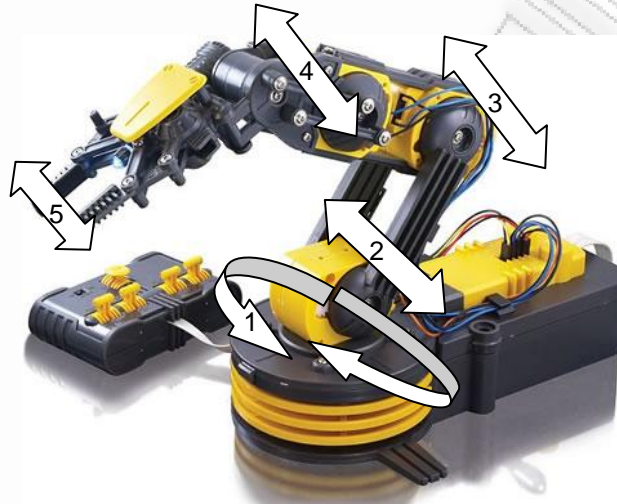
Πολύ σημαντικό ρόλο στη δομή του ρομπωτικού βραχίονα έχουν οι κινητήρες (ή μοτέρ). Κάθε άρθρωση χρειάζεται και από έναν κινητήρα ο οποίος αναλαμβάνει την κίνηση της. Γι' αυτό πολλές φορές οι κινητήρες αναφέρονται στη βιβλιογραφία και σαν "συστήματα μετάδοσης κίνησης". Οι κινητήρες μπορεί να είναι ηλεκτρικοί, υδραυλικοί ή πνευματικοί. Η επιλογή τους καθορίζεται από το κόστος και την ανάγκη χρήσης.

4.1. Βαθμοί Ελευθερίας

Βασικό γνώρισμα κάθε ρομποτικού βραχίονα αποτελεί ο Βαθμός Ελευθερίας του (Degree Of Freedom – DOF). Η έννοια αυτή δηλώνει το πόσο ευκίνητο είναι ένα ρομπότ στο χώρο. Συνήθως, κάθε ανεξάρτητα κινούμενη άρθρωση προσθέτει ένα βαθμό ελευθερίας στο ρομπότ. Ο ορισμός λέει: “Ο αριθμός των ανεξάρτητων παραμέτρων, που προσδιορίζουν τη θέση ενός σώματος στο χώρο, ονομάζεται Βαθμός Ελευθερίας”.

Έτσι ένα ρομπότ με η βαθμούς ελευθερίας περιέχει η αρθρώσεις ή γενικά η άξονες κίνησης.

Στο παρακάτω σχήμα (Σχήμα 15) αναπαριστώνται οι βαθμοί ελευθερίας του ρομποτικού βραχίονα της εργασίας. Οι βαθμοί ελευθερίας είναι 5.



Σχήμα 15: Βαθμοί ελευθερίας ρομποτικού βραχίονα της παρούσας εργασίας

4.2. Ωφέλιμο Φορτίο – Επαναληψιμότητα – Ακρίβεια

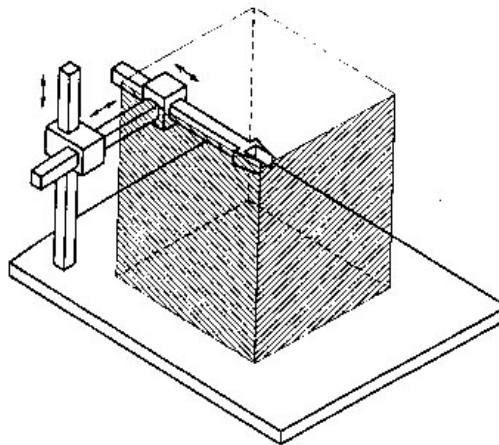
Από τα πιο σημαντικά μεγέθη ενός βιομηχανικού βραχίονα είναι το ωφέλιμο φορτίο, η επαναληψιμότητα και η ακρίβεια. Πιο συγκεκριμένα τα παραπάνω μεγέθη αναφέρονται στα εξής:

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

4.3. Ταξινόμηση Βραχιόνων Βάσει της Γεωμετρικής Διαμόρφωσής τους

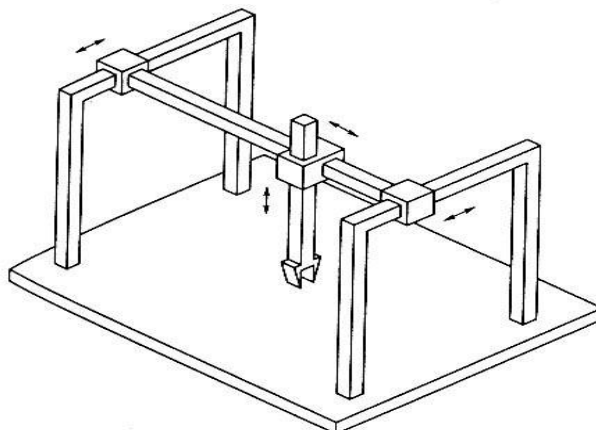
Ο τύπος και η διαδοχή των αρθρώσεων ενός βραχίονα επιτρέπει την ταξινόμησή των ρομπότ σε διάφορες κατηγορίες, οι οποίες αναφέρονται παρακάτω. Οι αρθρώσεις που μας απασχολούν στο σημείο αυτό είναι οι τρεις πρώτες του βραχίονα και κατά συνέπεια εξαιρούνται οι αρθρώσεις του καρπού. Θα έχουμε λοιπόν τα εξής:

- **Καρτεσιανοί Βραχίονες:** η καρτεσιανή γεωμετρία υλοποιείται με τρεις διαδοχικές πρισματικές αρθρώσεις. Οι άξονες των αρθρώσεων αυτών είναι ανά δύο κάθετοι μεταξύ τους (Σχήμα 16). Η καρτεσιανή δομή παρέχει μεγάλη δυσκαμψία και σταθερή ακρίβεια σε ολόκληρο το χώρο εργασίας που είναι ένα παραλληλεπίπεδο. Βασικό μειονέκτημα της κατασκευής είναι η μειωμένη επιδεξιότητα κίνησης, λόγω της πρισματικής φύσης των αρθρώσεων.



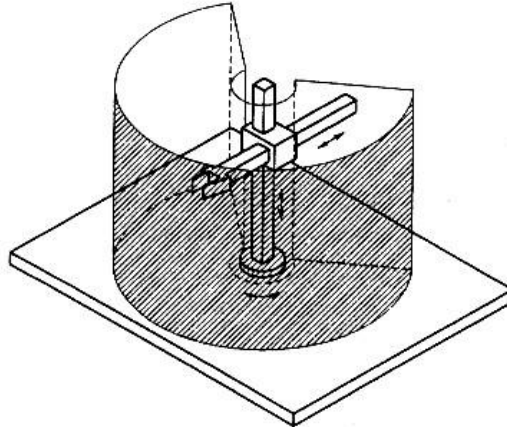
Σχήμα 16: Καρτεσιανός βραχίονας

- **Βραχίονες Gantry:** οι βραχίονες Gantry είναι στην ουσία καρτεσιανοί, διαφέρουν όμως από τους τελευταίους στον τρόπο προσέγγισης τους αντικείμενου ενδιαφέροντος (Σχήμα 17). Ειδικότερα ο βραχίονας Gantry προσεγγίζει το αντικείμενο από πάνω, τη στιγμή που ένας κλασικός καρτεσιανός βραχίονας προσεγγίζει το αντικείμενο από το πλάι. Άμεσες συνέπειες της διαφοροποίησης αυτής είναι η αύξηση του χώρου εργασίας και της δυσκαμψίας, καθώς επίσης και η δυνατότητα χειρισμού μεγάλων και βαριών αντικειμένων.



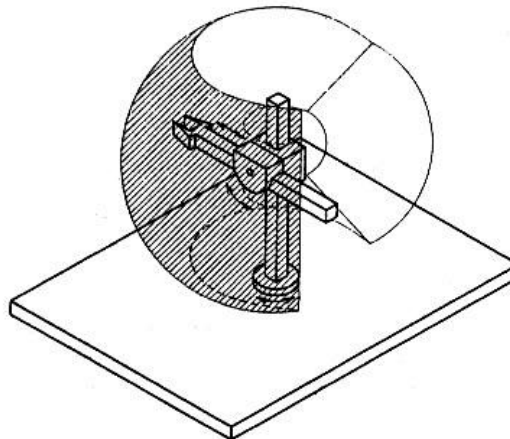
Σχήμα 17: Βραχίονας Gantry

- **Κυλινδρικοί Βραχίονες:** στους κυλινδρικούς βραχίονες η πρώτη πρισματική άρθρωση της καρτεσιανής δομής έχει αντικατασταθεί από μία περιστροφική άρθρωση (Σχήμα 18). Οι συγκεκριμένοι βραχίονες χαρακτηρίζονται από καλή δυσκαμψία, όμως η ακρίβεια της θέσης του καρπού μειώνεται καθώς η οριζόντια μετατόπιση αυξάνεται. Ο χώρος εργασίας στην περίπτωση αυτή είναι τμήμα κυλίνδρου. Σημαντικό μειονέκτημα της συγκεκριμένης γεωμετρίας είναι το ότι ο βραχίονας εισέρχεται στο χώρο εργασίας και τον περιορίζει.



Σχήμα 18: Κυλινδρικός βραχίονας

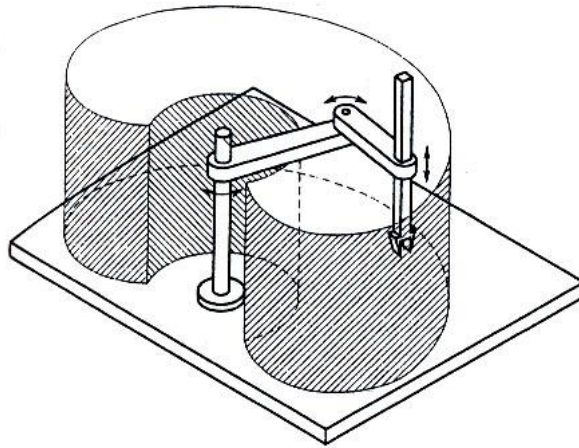
- **Σφαιρικοί Βραχίονες:** στους βραχίονες αυτούς αντικαθίσταται πλέον και η δεύτερη πρισματική άρθρωση της καρτεσιανής δομής με περιστροφική (Σχήμα 19). Η μηχανολογική πολυπλοκότητα αυξάνει, ενώ η δυσκαμψία μειώνεται. Επιπλέον η ακρίβεια του καρπού μειώνεται με την αύξηση της ακτινικής απόστασης. Ο χώρος εργασίας είναι τμήμα σφαίρας και περιέχει ένα μέρος της βάσης με άμεση συνέπεια τη δυνατότητα χειρισμού αντικειμένων που βρίσκονται στο έδαφος.



Σχήμα 19: Σφαιρικός βραχίονας

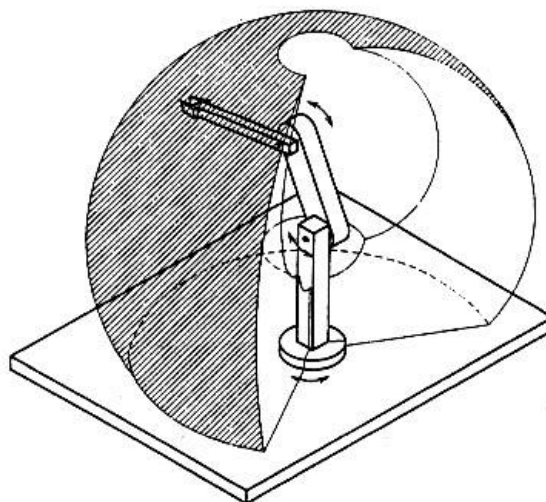
- **Βραχίονες SCARA:** η γεωμετρία SCARA είναι ειδική και περιλαμβάνει δύο περιστροφικές και μία πρισματική άρθρωση τοποθετημένες κατά τέτοιο τρόπο ώστε οι άξονες κίνησης να είναι παράλληλοι μεταξύ τους (Σχήμα 20). Το όνομα SCARA

προέρχεται από τα αρχικά των λέξεων Selective Compliance Assembly Robot Arm. Η συγκεκριμένη γεωμετρία παρέχει μεγάλη δυσκαμψία σε κατακόρυφη φόρτιση και ελαστικότητα σε οριζόντια. Η ακρίβεια τοποθέτησης του καρπού μειώνεται με την αύξηση της απόστασης του από τον άξονα της πρώτης άρθρωσης.



Σχήμα 20: Βραχίονας SCARA

- **Ανθρωπομορφικοί Βραχίονες:** η ανθρωπομορφική γεωμετρία υλοποιείται με τρεις διαδοχικές περιστροφικές αρθρώσεις. Ειδικότερα, ο άξονας περιστροφής της πρώτης άρθρωσης είναι κατακόρυφος και κάθετος στους άξονες περιστροφής των επόμενων δύο αρθρώσεων, οι οποίοι είναι παράλληλοι μεταξύ τους (Σχήμα 21). Η συγκεκριμένη δομή παρέχει τη μεγαλύτερη επιδεξιότητα από όλες τις προηγούμενες, καθώς όλες οι αρθρώσεις είναι περιστροφικές. Ωστόσο, η ακρίβεια του καρπού δεν είναι σταθερή εντός του χώρου εργασίας που έχει τη μορφή σφαίρας.



Σχήμα 21: Ανθρωπομορφικός βραχίονας

4.4. Ρομποτικά Συστήματα Ελέγχου

Μελετώντας τους ρομποτικούς βραχίονες με βάση τη θεωρία των συστημάτων αυτομάτου ελέγχου, οι ρομποτικοί βραχίονες και ευρύτερα τα ρομποτικά συστήματα, διακρίνονται σε δύο κατηγορίες:

- A. **Ανοιχτού Βρόχου.**
- B. **Κλειστού Βρόχου.**

Στην κατηγορία των συστημάτων Ανοιχτού Βρόχου, η έξοδος δεν έχει καμία επίδραση στην είσοδο, δεν υπάρχει δηλαδή ανατροφοδότηση. Σαν ένα παράδειγμα ας θεωρηθεί ένας ηλεκτρικός κινητήρας στον οποίο εφαρμόζεται μια σταθερή τάση, οπότε ο κινητήρας περιστρέφεται. Έξοδος του συστήματος εδώ είναι η ταχύτητα περιστροφής του άξονα και είσοδος η τάση που εφαρμόζεται στον κινητήρα. Εάν εφαρμοσθεί κάποιο φορτίο στον κινητήρα η ταχύτητά του θα αυξηθεί. Οι αυξομειώσεις αυτές της ταχύτητας με τη μεταβολή του φορτίου στην πράξη δεν είναι επιθυμητές. Ένα καλύτερο σύστημα προκύπτει εάν η έξοδος μετράται και ανατροφοδοτείται με κατάλληλο τρόπο στην είσοδο του συστήματος. Στο πιο πάνω παράδειγμα η ταχύτητα του κινητήρα μπορεί να μετρηθεί και να μετατραπεί σε τάση με κατάλληλο αισθητήριο όργανο (ταχογεννήτρια) και στη συνέχεια να συγκριθεί με την τάση εισόδου. Με βάση τη σύγκριση αυτή, γίνεται αυτόματα η απαραίτητη διόρθωση για να επιστρέψει η ταχύτητα του κινητήρα στην επιθυμητή τιμή.

Συστήματα στα οποία η έξοδος ανατροφοδοτείται στην είσοδο ονομάζονται Κλειστά Συστήματα Ελέγχου ή Κλειστού Βρόχου. Στα ρομποτικά συστήματα Κλειστού Συστήματος ελέγχου η θέση των αξόνων μετράται με κατάλληλα αισθητήρια όργανα ανατροφοδότησης, όπως είναι λ.χ. ο ψηφιακός (διαφορικός) κωδικοποιητής. Ο τύπος του στοιχείου οδήγησης κάθε άξονα ενός ρομπότ καθορίζεται ως επί των πλείστων από την απαιτούμενη ακρίβεια και ισχύ του ρομπότ.

Στα ρομπότ Ανοιχτού Βρόχου η κίνηση των αξόνων (αρθρώσεων) γίνεται με Βηματικούς Κινητήρες (Stepping motors). Ο βηματικός κινητήρας είναι ένας κινητήρας ο άξονας του οποίου περιστρέφεται κατά μια σταθερή γωνία για κάθε παλμό που δέχεται στην είσοδό του. Οι βηματικοί κινητήρες παρέχουν τον απλούστερο τρόπο μετατροπής μιας ακολουθίας ηλεκτρικών παλμών σε ανάλογη γωνιακή μετατόπιση και γι' αυτό αποτελούν μια σχετικά απλή και φθηνή λύση στο πρόβλημα ελέγχου των ρομπότ. Επειδή όμως δεν υπάρχει ανατροφοδότηση από τη θέση του άξονα, η ακρίβεια θέσης εξαρτάται αποκλειστικά από την ικανότητα του κινητήρα να προχωράει κατά τον ακριβή αριθμό βημάτων σύμφωνα με το πλήθος παλμών που δέχεται στην είσοδό του. Ένα από τα κύρια χαρακτηριστικά των βηματικών κινητήρων είναι ότι η μέγιστη ταχύτητά τους εξαρτάται από τη μηχανική ροπή του φορτίου. Όσο πιο μεγάλη είναι η ροπή του φορτίου, τόσο μικρότερη είναι η μέγιστη επιτρεπτή ταχύτητα του κινητήρα. Συνέπεια του γεγονότος αυτού είναι ότι οι βηματικοί κινητήρες δεν μπορούν να χρησιμοποιηθούν σε συστήματα με απρόβλεπτες μεταβολές των ροπών φορτίου, γιατί μια μεγάλη αύξηση της ροπής φορτίου μπορεί να κάνει το βηματικό κινητήρα να χάσει ορισμένα βήματα και έτσι να κινηθεί σε λάθος θέση. Στα βιομηχανικά ρομπότ οι κινητήρες πρέπει να μπορούν να αναπτύσσουν μηχανικές ροπές όχι μόνο ανάλογα με το φορτίο που σηκώνουν αλλά και με τη θέση του βραχίονά τους. Συνεπώς οι βηματικοί κινητήρες σήμερα δεν προτιμώνται στα βιομηχανικά ρομπότ.

4.5. Κινητήρες - Μοτέρ

Τα μοτέρ είναι η "κινητήρια δύναμη" του ρομπότ, αφού είναι εκείνα που δημιουργούν την κίνηση στις αρθρώσεις και τους συνδέσμους του. Υπάρχουν πολλά είδη μοτέρ και η επιλογή του κατάλληλου γίνεται ανάλογα με τη χρήση όπου προορίζεται. Έτσι αν αναφερόμαστε για παράδειγμα για βιομηχανική χρήση, κατάλληλα μοτέρ θεωρούνται τα σερβομοτέρ (servomotors) και αν αναφερόμαστε σε ρομπότ ερασιτεχνικής ή οικιακής χρήσης τότε τα κατάλληλα μοτέρ από οικονομική και χρηστική άποψη θεωρούνται τα μοτέρ συνεχούς ρεύματος (ή DC μοτέρ – Direct Current motors).

4.5.1. Είδη Κινητήρων - Μοτέρ

Όπως διατυπώθηκε και στο τέλος της προηγούμενης παραγράφου, για την κατασκευή ενός ρομπότ δεν χρησιμοποιείται οποιοδήποτε μοτέρ. Η επιλογή του πρέπει να διαθέτει κάποια χαρακτηριστικά, τα οποία θα το καθιστούν ικανό για να χρησιμοποιηθεί.

Τα μοτέρ που συναντώνται κατά κύριο λόγο στο εμπόριο είναι τα εξής:

- Μοτέρ Συνεχούς Ρεύματος (DC Motors) ή Εναλλασσομένου Ρεύματος (AC Motors).
- Μοτέρ Συνεχούς Κίνησης (Continuous motors) ή Βηματικά Μοτέρ (Stepping Motors).
- Μοτέρ με Σερβομηχανισμό (Servomotors).

Ακολουθεί μια σύντομη και γενική περιγραφή των παραπάνω κατηγοριών μοτέρ.

Το **Συνεχές Ρεύμα (Direct Current – DC)** είναι η κύρια πηγή τροφοδοσίας που κυριαρχεί στον τομέα της ρομποτικής, είτε γίνεται αναφορά σε στάσιμα είτε σε κινητά ρομπότ. Πολύ σπάνια χρησιμοποιούνται μοτέρ **Εναλλασσόμενου Ρεύματος (AC)**, ακόμα και αν πρόκειται για βιομηχανική χρήση. Αυτό που συναντάται στην εφαρμογή, είναι η μετατροπή του εναλλασσόμενου σε συνεχές ρεύμα (DC to AC) και μετά γίνεται η χρησιμοποίηση του συνεχούς ρεύματος στα εν λόγω υποσυστήματα. Ένα σημαντικό χαρακτηριστικό που πρέπει να λαμβάνεται υπ' όψιν κατά το σχεδιασμό ενός ρομπότ και την επιλογή ενός μοτέρ, είναι αν το μοτέρ πρέπει να είναι αναστρεφόμενο (στη βιβλιογραφία συναντώνται και σαν “δύο κατευθύνσεων”) ή μονόδρομο. Με τον όρο “αναστρεφόμενο”, εννοείται ότι το μοτέρ πρέπει να είναι ικανό να δημιουργεί (και να εκτελεί) κίνηση προς τις δύο κατευθύνσεις (δεξιόστροφα και αριστερόστροφα). Λίγες ρομποτικές εφαρμογές χρησιμοποιούν μονόδρομα μοτέρ (μοτέρ μιας κατεύθυνσης). Τα DC μοτέρ έχουν “έμφυτη” την ικανότητα να είναι δυο κατευθύνσεων, αλλά αρκετές φορές διάφοροι περιορισμοί στη σχεδίαση αναγκάζουν τον κατασκευαστή να χρησιμοποιήσει μοτέρ μιας μόνο κατεύθυνσης. Σημαντικό επίσης κριτήριο επιλογής, αποτελούν οι ψήκτρες του συλλέκτη (Commutator Brushes). Εάν οι ψήκτρες είναι επικλινείς (κεκλιμένες), το μοτέρ πιθανότατα να μην μπορεί να εκτελέσει κίνηση δύο κατευθύνσεων. Επίσης, η εσωτερική καλωδίωση που έχει το DC μοτέρ έρχεται να παίξει σημαντικό ρόλο στην ικανότητά του να κινηθεί σε δυο κατευθύνσεις. Ο καλύτερος τρόπος για να ελεγχθεί ο τρόπος λειτουργίας ενός μοτέρ, είναι με τη βοήθεια μιας συνεχούς (DC) πηγής ρεύματος (μπαταρία). Ανάλογα της πόλωσης της μπαταρίας με τα καλώδια του μοτέρ, παρατηρείται η κίνηση, η οποία θα είναι δεξιόστροφη ή αριστερόστροφη.

Τα DC μοτέρ μπορούν να αναπτυχθούν σε δύο περαιτέρω κατηγορίες, στα **Συνεχούς Κίνησης Μοτέρ (Continuous Motors)** και στα **Βηματικά Μοτέρ (Stepping Motors)**. Η διαφορά τους έγκειται στο παρακάτω χαρακτηριστικό: Στα **Μοτέρ Συνεχούς Κίνησης**, η παροχή ενέργειας κάνει τον άξονα του μοτέρ να περιστρέφεται συνεχώς. Ο άξονας σταματάει την περιστροφή του, μόνο όταν η παροχή ενέργειας στο μοτέρ σταματήσει ή εάν το μοτέρ φτάσει σε ένα σημείο όπου δεν μπορεί να περιστραφεί περισσότερο, ανεξαρτήτως της ενέργειας που θα του διοχετευτεί. Στα **Βηματικά Μοτέρ**, η παροχή ενέργειας κάνει τον άξονα του μοτέρ να περιστραφεί μερικές μοίρες και μετά να σταματήσει.

Υποκατηγορία των μοτέρ συνεχούς κίνησης (continuous motors) αποτελούν τα **Σερβομοτέρ (Servomotors)**, τα οποία στις περισσότερες περιπτώσεις συνδυάζουν ένα μοτέρ συνεχούς ρεύματος και συνεχούς κίνησης (Continuous DC Motor) με ένα μηχανισμό ανάδρασης προκειμένου να εξασφαλιστεί μεγάλη ακρίβεια στην περιστροφή του μοτέρ.

4.5.2. Οι Προδιαγραφές των Κινητήρων - Μοτέρ

Στο παρόν κεφάλαιο θα αναφερθούν συνοπτικά οι κυριότερες προδιαγραφές ενός μοτέρ που είναι:

- A. Η τάση λειτουργίας
- B. Η ποσότητα ρεύματος
- C. Η ταχύτητά
- D. Η ροπή στρέψης

A. Τάση Λειτουργίας

Όλα τα μοτέρ ταξινομούνται ανάλογα με την τάση λειτουργίας τους (operating voltage). Σε μικρά DC μοτέρ, η τιμή της τάσης κυμαίνεται συνήθως από 1.5 έως 6 Volt. Μερικά μοτέρ υψηλής ποιότητας χρησιμοποιούν τάση μεγέθους 12 ή 24 Volt. Η τιμή της τάσης για την κατασκευή ενός πειραματικού ρομπότ κυμαίνεται από 1.5 έως και 12 Volt. Τα πιο πολλά μοτέρ μπορούν να δουλέψουν και σε τάσεις χαμηλότερες από αυτές που οι προδιαγραφές τους ορίζουν. Δηλαδή ένα μοτέρ των 12 Volt μπορεί να δουλέψει στα 9 Volt χωρίς να δημιουργηθεί ιδιαίτερο πρόβλημα. Ακόμα εάν το μοτέρ είναι της τάξης των 12 Volt και του δοθεί τάση 6 Volt, τότε αυτό θα συνεχίζει να περιστρέφεται, αλλά με μειωμένη ταχύτητα και ροπή στρέψης. Ακόμα, παρατηρείται ότι ένα μοτέρ δεν δουλεύει καθόλου (ή δεν δουλεύει ικανοποιητικά) σε τιμές τάσης κάτω από το 50% της προδιαγραφής της. Παρομοίως, ένα μοτέρ των 12 Volt μπορεί να λειτουργεί στα 16 Volt, αλλά όπως είναι φυσιολογικό οι στροφές του θα αυξηθούν, όπως και η ροπή στρέψης του. Δεν συνιστάται πάντως να λειτουργεί ένα μοτέρ παραπάνω από το 30% της τιμής της προδιαγραφής του.

B. Ένταση Ρεύματος

Η ένταση ρεύματος εκφράζει την ποσότητα του ρεύματος που το μοτέρ απαιτεί από την πηγή ενέργειας για να δουλέψει σωστά. Η προδιαγραφή αυτή είναι πολύ σημαντική όταν περιγράφει μοτέρ που κινεί ή περιστρέφει κάτι. Η ένταση που απαιτείται για ένα μοτέρ που δεν χρειάζεται να περιστρέψει κάτι που απαιτεί σημαντική δύναμη, μπορεί να είναι σημαντικά χαμηλή. Αλλά για ένα μοτέρ που πρέπει να περιστρέψει κάποια άρθρωση, ή ένα σετ τροχών, η ένταση του ρεύματος είναι υπέρ-πολλαπλάσια. Η ένταση του ρεύματος που χρησιμοποιεί το μοτέρ για να κινηθεί, με τη δύναμη που επιδρά πάνω του (και την οποία πρέπει να κινήσει), είναι ανάλογες. Όσο περισσότερο βάρος προσπαθεί να περιστρέψει ένα μοτέρ, τόσο μεγαλύτερη είναι και η ποσότητα του ρεύματος που χρησιμοποιεί.

C. Ταχύτητα

Η περιστροφική ταχύτητα μετριέται σε στροφές ανά λεπτό (revolutions per minute – rpm). Τα πιο πολλά DC συνεχούς κίνησης μοτέρ έχουν ένα αριθμό στροφών της τάξης των 4000 με 7000 στροφών το λεπτό. Κάποια μοτέρ που χρησιμοποιούνται για ειδικούς σκοπούς, όπως σε σκληρούς δίσκους υπολογιστών, λειτουργούν σε αριθμό στροφών της τάξης των 2000 με 3000 στροφών το λεπτό. Όμως, για τις ρομποτικές εφαρμογές αυτές οι ταχύτητες είναι πολύ ψηλές. Πρέπει η ταχύτητα να μειωθεί δραστικά και να είναι της τάξης των 150 στροφών το λεπτό (και μάλιστα να είναι μικρότερη για μοτέρ που οδηγούν βραχίονα ή αρπάγες). Η μείωση του αριθμού στροφών μπορεί να επιτευχθεί είτε μέσω ηλεκτρονικού ελέγχου, είτε μέσω συστήματος μείωσης μετάδοσης της κίνησης (gear reduction system).

D. Ροπή Στρέψης

Με τον όρο “ροπή στρέψης” εννοείται η περιστροφική δύναμη που παράγεται στον άξονα του μοτέρ. Όσο μεγαλύτερη η ροπή, τόσο μεγαλύτερο μπορεί να είναι και το φορτίο που θα ασκηθεί στο μοτέρ, αλλά και τόσο περισσότερο το μοτέρ θα μπορεί να περιστρέψει τον άξονά του κάτω από αυτό το φορτίο. Εάν μειωθεί η ροπή τότε και το μοτέρ θα ελαττώσει τη δύναμη και την ταχύτητα του κάτω από το φορτίο που θα του ασκηθεί.

5. Οι Αλγόριθμοι

Οι εργασίες που διεκπεραιώνονται από ένα υπολογιστικό σύστημα, είναι εκείνες όπου προσδιορίζονται σύμφωνα με τις απλές εργασίες όπου μπορεί ο ίδιος ο υπολογιστής και κατά προέκταση το ίδιο το σύστημα, να επιτελέσει. Για να συμβεί αυτό, πρέπει κάποιος να του περιγράψει το πώς η εργασία θα διεκπεραιωθεί. Μία τέτοια περιγραφή λέγεται **αλγόριθμος (algorithm)**. Συνοπτικά, ένας αλγόριθμος περιγράφει τη μέθοδο με την οποία μία εργασία μπορεί να διεκπεραιωθεί. Ο αλγόριθμος αποτελείται από μία ακολουθία βημάτων η οποία αν εκτελεστεί σωστά, θα έχει σαν αποτέλεσμα η **διεργασία (process)** να διεκπεραιωθεί.

Η έννοια του αλγορίθμου δεν είναι ιδιαίτερο χαρακτηριστικό της επιστήμης των υπολογιστών. Κατ' επέκταση, για να εκτελεστεί ένας αλγόριθμος σωστά πρέπει να είναι

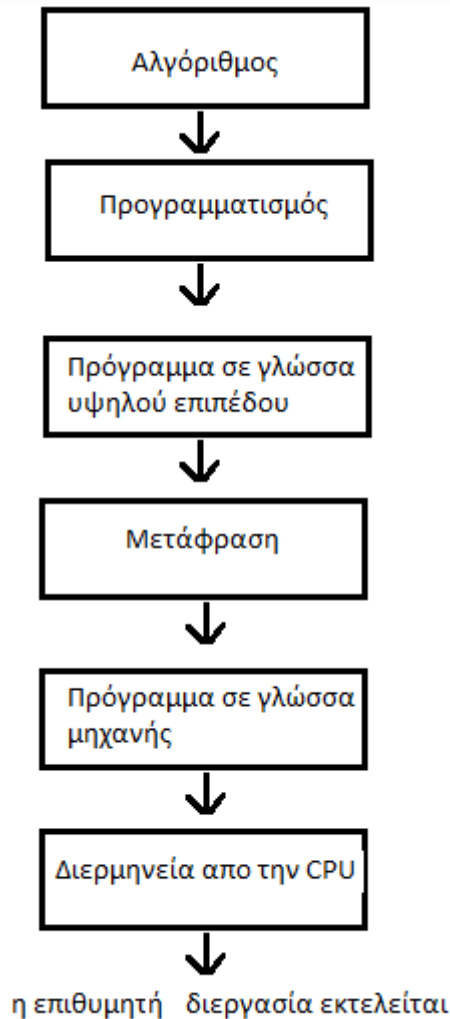
εκφρασμένος σε μορφή που να μπορεί να καταλάβει και να εκτελέσει ο αντίστοιχος επεξεργαστής. Όταν ο αντίστοιχος επεξεργαστής είναι ένας υπολογιστής, ο αλγόριθμος πρέπει να είναι εκφρασμένος σε μία μορφή που λέγεται **πρόγραμμα (program)**. Ένα πρόγραμμα είναι γραμμένο σε μία **γλώσσα προγραμματισμού (programming language)** και η δραστηριότητα έκφρασης του αλγόριθμου στην κάθε γλώσσα προγραμματισμού, λέγεται **προγραμματισμός (programming)**. Κάθε βήμα του αλγόριθμου εκφράζεται από μία **εντολή (instruction)** ή **πρόταση / δήλωση (statement)** στο πρόγραμμα. Έτσι ένα πρόγραμμα αποτελείται από μία ακολουθία προτάσεων, καθεμία από τις οποίες προσδιορίζει συγκεκριμένες λειτουργίες που πρόκειται να εκτελέσει ο υπολογιστής.

Η φύση των προτάσεων σε ένα πρόγραμμα εξαρτάται από τη γλώσσα προγραμματισμού που χρησιμοποιείται. Υπάρχει μεγάλος αριθμός γλωσσών προγραμματισμού και κάθε γλώσσα έχει το δικό της γλωσσάρι με προτάσεις. Οι απλούστερες γλώσσες, που λέγονται **γλώσσες μηχανής (machine languages)**, είναι σχεδιασμένες έτσι ώστε κάθε πρόταση να μπορεί να διερμηνευθεί απευθείας από τον υπολογιστή. Αυτό πρακτικά σημαίνει ότι η CPU είναι ικανή να καταλάβει κάθε πρόταση και να επιτελέσει τις αντίστοιχες λειτουργίες. Όμως, από τη στιγμή που η διατύπωση των προτάσεων είναι τόσο απλή, καθεμία μπορεί μόνο να εκφράσει ένα μικρό μέρος του αλγόριθμου. Αυτό σημαίνει ότι απαιτείται μεγάλος αριθμός προτάσεων για να εκφραστεί ένας πλήρης αλγόριθμος και ο προγραμματισμός σε γλώσσα μηχανής γίνεται κουραστικός.

Για να γίνει ο προγραμματισμός ευκολότερος, έχουν αναπτυχθεί άλλες μορφές γλωσσών προγραμματισμού. Αυτές είναι οι γλώσσες υψηλού επιπέδου (high level languages) και είναι περισσότερο βολικές από ότι οι γλώσσες μηχανής, γιατί κάθε πρόταση μπορεί να εκφράσει ένα μεγαλύτερο βήμα του αλγόριθμου. Φυσικά κάθε πρόταση πρέπει πάλι να διερμηνευθεί από τον υπολογιστή και θα μπορούσε να φανεί ότι οι ικανότητες της CPU θα έπρεπε να επεκταθούν ανάλογα, με σκοπό να καταστήσουν κάτι τέτοιο δυνατόν. Η επέκταση της CPU είναι βέβαια ένας τρόπος για να κάνουμε τη διερμηνεία των προγραμμάτων υψηλού επιπέδου εφικτή, αλλά στην πραγματικότητα υπάρχει μία φθηνότερη και πιο ευέλικτη λύση εναλλακτική λύση: **να μεταφραστεί (to translate)** τα προγράμματα υψηλού επιπέδου σε γλώσσα μηχανής πριν εκτελεστούν. Η μετάφραση (translation) συνίσταται στη μετατροπή κάθε πρότασης προγράμματος υψηλού επιπέδου, σε μία ισοδύναμη ακολουθία από προτάσεις διατυπωμένες σε γλώσσα μηχανής. Αυτές οι προτάσεις της γλώσσας μηχανής μπορούν τότε να διερμηνευθούν από τη CPU. Η όλη διαδικασία φαίνεται στο παρακάτω σχήμα (Σχήμα 22).

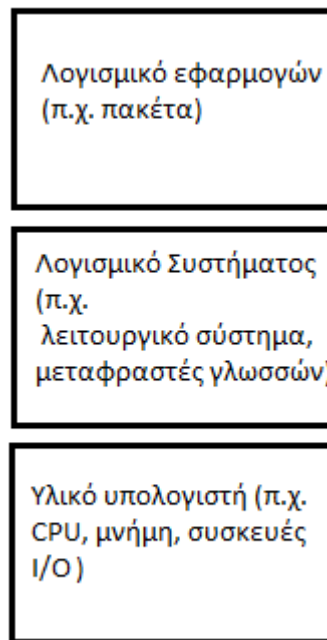
Πρέπει να αναφερθεί ότι, τα γενικά προγράμματα αναφέρονται με το γενικό όρο **λογισμικό (software)** για να τα διακρίνουμε από το **υλικό (hardware)**, από το οποίο είναι κατασκευασμένος ο υπολογιστής. Μέρος του λογισμικού γράφεται από τους χρήστες του υπολογιστικού συστήματος για να επιτελέσει διάφορες εργασίες που επιθυμούν να διεκπεραιώσουν, αλλά στην πραγματικότητα οι περισσότεροι χρήστες υπολογιστών δεν προγραμματίζουν καθόλου. Χρησιμοποιούν τους υπολογιστές σαν εργαλεία για να επιτελέσουν εργασίες, για τις οποίες οι αλγόριθμοι είναι ήδη γνωστοί και τα προγράμματα είναι ήδη γραμμένα. Μία συλλογή προγραμμάτων για χρήση σε συγκεκριμένη εφαρμογή λέγεται **πακέτο εφαρμογών (applications package)**.

Η συμμετοχή του χρήστη στον προγραμματισμό περιορίζεται στις περιπτώσεις εκείνες όπου το πακέτο μπορεί να χρειάζεται τροποποίηση, πιθανώς για να ανταπεξέλθει σε κάποιες καινούριες συνθήκες ή για να ικανοποιήσει κάποια πρόσθετη απαίτηση. Υπάρχουν όμως πολλές περιπτώσεις, όπου ο χρήστης του υπολογιστή θέλει να κάνει κάτι, για το οποίο κανένα πακέτο εφαρμογών δεν είναι διαθέσιμο. Σε αυτές τις περιπτώσεις ο χρήστης αναγκάζεται να γράψει δικά του προγράμματα ή να μισθώσει κάποιον να τα γράψει γι' αυτόν. Αυτά τα προγράμματα μπορεί αργότερα να πουληθούν σαν πακέτα για άλλους χρήστες με παρόμοιες απαιτήσεις.



Σχήμα 22: Στάδια για την εκτέλεση ενός αλγόριθμου στον υπολογιστή

Τα πακέτα εφαρμογών και τα προγράμματα που γράφονται από τους χρήστες ανήκουν στη γενική κατηγορία του **λογισμικού εφαρμογών (application software)**. Για λόγους ευχρηστίας, το λογισμικό εφαρμογών συνήθως είναι γραμμένο σε γλώσσα προγραμματισμού υψηλού επιπέδου και συνεπώς βασίζεται στην ύπαρξη των κατάλληλων μεταφραστών γλωσσών. Οι μεταφραστές γλωσσών δεν είναι τα μοναδικά προγράμματα πάνω στα οποία βασίζεται το λογισμικό εφαρμογών. Υπάρχει ένας αριθμός από άλλα προγράμματα, τα οποία παρέχουν ζωτικές υπηρεσίες και μαζί με τους μεταφραστές γλωσσών, αναφέρονται σαν **λογισμικό συστήματος (system software)**. Ένα αξιοσημείωτο στοιχείο του λογισμικού συστήματος είναι το **λειτουργικό σύστημα (operating system)**, οι υπηρεσίες του οποίου περιλαμβάνουν συνήθως τη διαχείριση των μονάδων εισόδου – εξόδου του υπολογιστή, την αποθήκευση των πληροφοριών για μεγάλες χρονικές περιόδους και τη δυνατότητα της πρόσβασης στον υπολογιστή από διάφορους χρήστες ταυτόχρονα.



Σχήμα 23: Η ιεραρχία υλικού - λογισμικού

Από τα παραπάνω είναι φανερό ότι ένα υπολογιστικό σύστημα μπορεί να θεωρηθεί ως ιεραρχία των τμημάτων του, λογισμικού και του υλικού (Σχήμα 23). Στο υψηλότερο επίπεδο είναι το λογισμικό εφαρμογών με το λογισμικό συστήματος από κάτω του. Το λογισμικό συστήματος με τη σειρά του, βασίζεται πάνω στη CPU, τη μνήμη και τις μονάδες εισόδου – εξόδου του υπολογιστή. Η ιεραρχία δεν σταματά στην πραγματικότητα εδώ, αφού αυτά τα τμήματα του υλικού είναι κατασκευασμένα από απλούστερα τμήματα. Βέβαια, στο σχήμα δεν γίνεται σαφής η διάκριση ανάμεσα στο ρόλο του υλικού και του λογισμικού.

Πρακτικά, η επιλογή μεταξύ υλικού και λογισμικού εξαρτάται από τις εκτιμήσεις ταχύτητας και κόστους που κάνουμε. Πολύπλοκες λειτουργίες, όπως η μετάφραση και ο σχεδιασμός, επιτελούνται γενικά από το λογισμικό. Απλές λειτουργίες ή λειτουργίες που απαιτούνται πολύ συχνά, επιτελούνται από το υλικό, ώστε να μπορεί να επιτευχθεί το μέγιστο δυνατό αποτέλεσμα. Συνοψίζοντας, το όριο μεταξύ υλικού και λογισμικού είναι φαινομενικά μεταβλητό και πιθανόν να μεταβληθεί με την πρόοδο της τεχνολογίας.

5.1. Η Σημασία των Αλγορίθμων

Όπως έχει ήδη αναφερθεί, για να έρθει σε πέρας μία διεργασία σε έναν υπολογιστή και κατά προέκταση σε ένα υπολογιστικό σύστημα πρέπει:

1. Να σχεδιασθεί ένας αλγόριθμος που να περιγράφει πως θα επιτελεστεί η διεργασία.
2. Να εκφραστεί ο αλγόριθμος σαν πρόγραμμα σε μία κατάλληλη γλώσσα προγραμματισμού.
3. Να εκτελεσθεί από τον υπολογιστή το πρόγραμμα.

Ο ρόλος του αλγορίθμου είναι θεμελιώδης. Χωρίς αλγόριθμο δεν μπορεί να υπάρξει πρόγραμμα και χωρίς πρόγραμμα δεν υπάρχει τίποτα να εκτελεστεί. Βέβαια, οι αλγόριθμοι είναι θεμελιώδης εξεταζόμενοι και από μια άλλη οπτική γωνία: είναι ανεξάρτητοι και από τη γλώσσα στην οποία εκφράζονται και από τον υπολογιστή που τους εκτελεί. Ένα παράδειγμα που μπορεί

να παρατεθεί από την καθημερινή ζωή είναι το ακόλουθο: όλα τα συστήματα κρατήσεων αεροπορικών θέσεων διαθέτουν έναν αλγόριθμο για την κράτηση μίας θέσης. Σε κάθε σύστημα ο αλγόριθμος μπορεί να είναι εκφρασμένος σε διαφορετική γλώσσα προγραμματισμού και να εκτελείται σε διαφορετικό υπολογιστή. Όμως σε όλες τις περιπτώσεις ο αλγόριθμος είναι βασικά ο ίδιος.

Επιστρέφοντας στη σφαίρα της τεχνικής, όλοι οι υπολογιστές (όπως και όλοι οι άνθρωποι) μπορούν να επιτελέσουν παραπλήσια τις ίδιες βασικές λειτουργίες. Οπωσδήποτε υπάρχουν διαφορές, τόσο στη λεπτομέρεια όσο και στην ταχύτητα, αλλά με τη γενική θεώρηση οι ικανότητες όλων των υπολογιστών είναι ίδιες. Ακόμη περισσότερο, αυτές οι ικανότητες είναι αμετάβλητες από τις τεχνολογικές καινοτομίες, καθώς τέτοιες καινοτομίες δεν επηρεάζουν το τι κάνουν οι υπολογιστές, αν και μπορεί να επηρεάζουν την ταχύτητα και την αξιοπιστία με την οποία το κάνουν. Αυτό σημαίνει ότι οι αλγόριθμοι μπορούν να καταστρωθούν και να μελετηθούν ανεξάρτητα από την τεχνολογία της κάθε εποχής και τα αποτελέσματα θα μείνουν έγκυρα σε πείσμα της έλευσης νέων υπολογιστών και νέων γλωσσών προγραμματισμού. Γίνεται εύκολα κατανοητό ότι τόσο ο υπολογιστής όσο και η γλώσσα προγραμματισμού είναι τα μέσα για να εκτελεστεί ο αλγόριθμος και να έρθει σε πέρας η αντίστοιχη διεργασία.

Τα όσα διατυπώθηκαν στην προηγούμενη παράγραφο, δεν έχουν σκοπό σε καμία περίπτωση να υποβαθμίσουν το ρόλο των υπολογιστών και των γλωσσών προγραμματισμού. Οι όποιες εξελίξεις συμβαίνουν στην τεχνολογία δρουν υπέρ των αλγορίθμων καθώς μπορεί να τους κάνει να εκτελούνται ταχύτερα, φθηνότερα και πιο αξιόπιστα. Σαν προέκταση της εξέλιξης έρχεται η μελέτη πεδίων και περιοχών εφαρμογών που στα προηγούμενα χρόνια ήσαν ανέφικτες. Για παράδειγμα μπορεί να αναφερθεί η πρόγνωση του καιρού η οποία βασίζεται στην επίλυση πολύπλοκων εξισώσεων για την ατμόσφαιρα. Οι αλγόριθμοι για την επίλυση αυτών των εξισώσεων είναι γνωστοί εδώ και πολλά χρόνια, αλλά μέχρι την έλευση των υπολογιστών υψηλής ταχύτητας ο χρόνος που χρειαζόταν για την επίλυση τους ήταν τόσο μεγάλος, ώστε ο καιρός είχε περάσει πριν την πρόγνωση του.

Οι γλώσσες προγραμματισμού είναι επίσης σημαντικές, διότι καθορίζουν την ευκολία με την οποία μπορούν να εκφραστούν οι αλγόριθμοι και κατά συνέπεια επηρεάζουν τον ανθρώπινο μόχθο που καταβάλλεται στον προγραμματισμό. Έτσι η ανάπτυξη νέων γλωσσών προγραμματισμού και νέων υπολογιστών είναι μία θεμιτή υπόθεση της επιστήμης των υπολογιστών, αλλά μόνο σαν τρόπος για περισσότερο αποτελεσματική έκφραση και εκτέλεση των αλγορίθμων.

5.2. Χαρακτηριστικά των Αλγορίθμων

Με δεδομένο το ότι οι αλγόριθμοι είναι βασικοί στην επιστήμη των υπολογιστών, στη συνέχεια θα αναλυθούν τα επί μέρους θέματα ή χαρακτηριστικά των αλγορίθμων που μελετούν οι επιστήμονες των υπολογιστών.

Το σημαντικότερο και βασικότερο χαρακτηριστικό είναι η **σχεδίαση (design)** των αλγορίθμων, δηλαδή το πώς σχεδιάζεται ένας αλγόριθμος για τη διεκπεραίωση των απαιτούμενων διεργασιών. Η σχεδίαση ενός αλγορίθμου είναι μία δύσκολη διανοητική δραστηριότητα, πολύ πιο δύσκολη από το να εκφραστεί ένας αλγόριθμος σε πρόγραμμα. Οι επιστήμονες των υπολογιστών προσπαθούν να αντιμετωπίσουν αυτή τη δυσκολία επινοώντας πλαίσια μέσω των οποίων μπορεί να προχωρήσει η διαδικασία σχεδίασης. Πάραυτα η διαδικασία σχεδίασης των αλγορίθμων απαιτεί δημιουργικότητα και διορατικότητα και δεν μπορούν να διατυπωθούν γενικοί κανόνες. Με πιο απλά λόγια, δεν υπάρχει αλγόριθμος για τη σχεδίαση αλγορίθμων.

Η παρατήρηση που διατυπώνεται από την προηγούμενη πρόταση, προκαλεί μερικές ενδιαφέρουσες ερωτήσεις. Υπάρχουν άλλες διεργασίες, για τις οποίες δεν υπάρχει κανένας αλγόριθμος; Τέτοιες διεργασίες αντιστοιχούν σε εκείνες που διαισθητικά θεωρούμε δημιουργικές; Για μία δεδομένη διεργασία, πως μπορεί κάποιος να πει αν υπάρχει ή όχι αλγόριθμος που να την περιγράφει; Η τελευταία ερώτηση είναι ιδιαίτερα σημαντική, επειδή αν μία διεργασία δεν μπορεί να περιγραφεί από έναν αλγόριθμο, δεν υπάρχει περίπτωση να εκτελεστεί ποτέ από υπολογιστή άσχετα από το πόσο μεγάλη, γρήγορη ή ακριβή είναι. Αυτά τα θέματα εντάσσονται στην κατηγορία που ονομάζεται **υπολογισσιμότητα (computability)**.

Αν μία διεργασία μπορεί να περιγραφεί από αλγόριθμο, το επόμενο ερώτημα που απασχολεί είναι πόσο υπολογιστικοί πόροι απαιτούνται για να τον εκτελέσουμε; Συγκεκριμένα, πόσο χρόνο θα πάρει να εκτελέσουμε τον αλγόριθμο και πόση μνήμη θα χρειαστεί; Αν υπάρχουν διάφοροι αλγόριθμοι που περιγράφουν την ίδια διεργασία, ποιός από αυτούς είναι ο “καλύτερος”, με την έννοια της κατανάλωσης των λιγότερων πόρων; Ποιοι είναι οι ελάχιστοι πόροι που απαιτούνται για να διεκπεραιωθεί μία δεδομένη διεργασία, δηλαδή τι πόροι χρησιμοποιούνται για τον καλύτερο δυνατό αλγόριθμο; Μπορεί να υπάρξει ο καλύτερος δυνατός αλγόριθμος; Υπάρχουν διεργασίες όπου ο καλύτερος αλγόριθμος καταναλώνει τόσο πολλούς πόρους, ώστε η εκτέλεση να είναι ανέφικτη, ακόμα και στους μεγαλύτερους και ταχύτερους υπολογιστές. Αυτά τα θέματα, συνολικά αναφέρονται σαν **πολυπλοκότητα (complexity)** των αλγορίθμων.

Ένα άλλο ενδιαφέρον θέμα που απασχολεί τους επιστήμονες είναι η **ορθότητα (correctness)** των αλγορίθμων. Από τη στιγμή που ο σχεδιασμός των αλγορίθμων είναι τόσο δύσκολος, όπως έχει ήδη προαναφερθεί, πως μπορεί κανείς να είναι σίγουρος ότι ένας συγκεκριμένος αλγόριθμος όντως περιγράφει την απαιτούμενη διεργασία; Καθώς οι εφαρμογές των υπολογιστών διευρύνονται, θα παρουσιάζεται ένας αυξανόμενος αριθμός από συνθήκες στις οποίες ένας λανθασμένος αλγόριθμος θα είναι ενδεχομένως ολέθριος.

Συνοψίζοντας την έννοια του αλγορίθμου, γίνεται αντιληπτό ότι είναι η ενοποιός έννοια για όλες τις δραστηριότητες που αντιμετωπίζουν οι επιστήμονες.

6. Δίκτυα Υπολογιστών

Η επικοινωνία μέσω της ανταλλαγής κωδικοποιημένων πληροφοριών είναι κάτι που επιδιώκει ο άνθρωπος από τα πολύ παλιά χρόνια. Όπως για παράδειγμα στην αρχαιότητα, η είδηση για την πτώση της Τροίας μεταδόθηκε στις Μυκήνες μέσα σε μία ημέρα, με τη χρήση ενός δικτύου φρυκτωριών (πυρσών). Παραπλήσια παραδείγματα αποτελούν οι ινδιάνοι της Αμερικής που επικοινωνούσαν με σήματα καπνού και οι ιθαγενείς της Αφρικανικής ζούγκλας οι οποίοι επικοινωνούν με ήχους τυμπάνων.

Όστόσο ο 20ος αιώνας βρίσκει τον άνθρωπο να έχει εξασφαλίσει την απομακρυσμένη επικοινωνία και τη μετάδοση της πληροφορίας μέσω του τηλεφωνικού δικτύου, του ταχυδρομικού δικτύου, των τηλεοπτικών και δορυφορικών δικτύων. Ο ηλεκτρονικός υπολογιστής, αναπόσπαστο κομμάτι της κοινωνίας μας πλέον, κατασκευάστηκε για να επεξεργάζεται και να διαχειρίζεται την πληροφορία οπότε έπρεπε να μπορεί και αυτός να ανταλλάσει πληροφορίες με άλλους ηλεκτρονικούς υπολογιστές.



Σχήμα 24: Χάρτης των φρυκτωριών που αναμετέδωσαν την είδηση για την πτώση της Τροίας από το όρος Ίδη έως τις Μυκήνες.

Ξεκινώντας μία σύντομη ιστορική αναδρομή, στη δεκαετία του 1960 ο ψυχρός πόλεμος μεταξύ ΗΠΑ και Σοβιετικής Ένωσης δημιούργησε την ανάγκη της αδιάκοπτης επικοινωνίας της Αμερικάνικης ενδοχώρας, έστω και αν κάποια επικοινωνιακά κέντρα πλήττονταν από τον εχθρό. Για το λόγο αυτό δημιουργήθηκε το **ARPA (Advanced Research Project Agency)** το οποίο ανέθεσε το 1969 σε ομάδα πανεπιστημίων τη δημιουργία ενός δικτύου με το όνομα ARPAnet. Το εν λόγω δίκτυο, θα συνέδεε μία σειρά από υπολογιστές μεταξύ τους για ερευνητικούς (και όχι μόνο) σκοπούς. Το αποτέλεσμα της δημιουργίας των πανεπιστημίων ήταν, πέρα από τη δημιουργία του δικτύου και τη δημιουργία του πρωτοκόλλου **TCP/IP (Transmission Control Protocol / Internet Protocol)**, που ήταν ανεξάρτητο του λειτουργικού συστήματος και επέτρεπε τη διασύνδεση πολλών διαφορετικών υπολογιστών μεταξύ τους. Το TCP/IP αφού ολοκληρώθηκε, προτάθηκε ως πρότυπο επικοινωνιών το 1973 και υιοθετήθηκε το 1982.

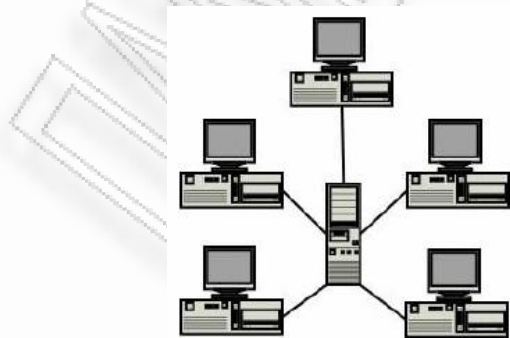
Το ARPAnet σταμάτησε τη λειτουργία του το 1990, καθώς το δίκτυο είχε πλέον σαν αποκλειστικό σκοπό την έρευνα, ενώ άρχισε να διαφαίνεται η στροφή του προς την εμπορική του χρήση. Έτσι το δίκτυο μετονομάστηκε σε Internet συνδέθηκαν πάνω σε αυτό πανεπιστήμια, οργανισμοί και πολυεθνικές εταιρίες. Η μεγάλη έκρηξη της δημοτικότητας του διαδικτύου (Internet) ήρθε το 1993 όταν καθιερώθηκε η υπηρεσία του **Παγκόσμιου Πληροφοριακού Ιστού (World Wide Web –WWW)**. Μέχρι τότε η χρήση του διαδικτύου ήταν αρκετά δύσκολη και απαιτούσε εξειδικευμένες γνώσεις. Ο Παγκόσμιος Πληροφοριακός Ιστός όμως στηριζόταν σε απλές και εύχρηστες εφαρμογές γραφικού περιβάλλοντος επικοινωνίας, με αποτέλεσμα το διαδίκτυο να μπορεί να χρησιμοποιηθεί και από απλούς χρήστες. Σήμερα το διαδίκτυο θεωρείται ο θεμέλιος λίθος για την υλοποίηση της πληροφοριακής επανάστασης και η χρήση του σε πολλές κατηγορίες επαγγελματιών αλλά και σε πολλές δραστηριότητες της καθημερινής μας ζωής είναι πια διαδεδομένη.

6.1. Τα Δίκτυα Υπολογιστών

Γίνεται κατανοητό ότι, η σύνδεση δύο ή περισσότερων αυτόνομων υπολογιστικών συστημάτων μεταξύ τους είναι ένα δίκτυο ηλεκτρονικών υπολογιστών. Έτσι οι χρήστες ενός δικτύου υπολογιστών έχουν τη δυνατότητα: να μοιράζονται κάποιους κοινούς πόρους, να ανταλλάσσουν μηνύματα, να χειρίζονται κοινές εφαρμογές, να έχουν ταυτόχρονη πρόσβαση σε ένα αρχείο δεδομένων.

Η σύνδεση των υπολογιστών μπορεί να επιτευχθεί είτε ενσύρματα (με τη χρήση ομοαξονικού καλωδίου ή UTP 45 ή οπτικών ινών), είτε ασύρματα (με τη χρήση πομποδεκτών ή χρήση δορυφορικών συστημάτων). Ανάλογα με την απόσταση που καλούνται να καλύψουν τα δίκτυα υπολογιστών, διακρίνονται και οι κατηγορίες δικτύων. Οι σημαντικότερες δύο μεγάλες κατηγορίες είναι τα **τοπικά δίκτυα (Local Area Networks – LAN's)** και τα **δίκτυα ευρείας περιοχής (Wide Area Networks – WAN's)**.

Στα **τοπικά δίκτυα** (Σχήμα 25), η διασπορά των υπολογιστών μπορεί να περιλαμβάνει έως και τα όρια μίας μικρής πόλης. Συνήθως η διασύνδεση γίνεται ενσύρματα με τη χρήση ομοαξονικού καλωδίου, χωρίς όμως να αποκλείεται και η ασύρματη διασύνδεση, η οποία κερδίζει συνεχώς έδαφος λόγω κυρίως της μείωσης του κόστους χρήσης εξοπλισμού των ασύρματων πομποδεκτών. Ένα απλοϊκό παράδειγμα μπορεί να αποτελέσει η διασύνδεση των υπολογιστών ενός εργαστηρίου.



Σχήμα 25: Ένα ενδεικτικό τοπικό δίκτυο (Δίκτυο Αστέρα).

Στα πλαίσια όπου του επιτρέπει η αδειοδότηση από το σύστημα, ένας υπολογιστής ενός τοπικού δικτύου, μπορεί: να τυπώσει από το δικτυακό εκτυπωτή, να έχει πρόσβαση στο διαδίκτυο, να επικοινωνήσει με τους υπόλοιπους σταθμούς εργασίας, να ανταλλάξει αρχεία με έναν άλλο σταθμό εργασίας με διαφορετικό χρήστη.

Στα **δίκτυα ευρείας περιοχής** (Σχήμα 26), οι ηλεκτρονικοί υπολογιστές μπορεί να βρίσκονται μέσα σε μία ή περισσότερες πόλεις. Η διασύνδεση επιτυγχάνεται ενσύρματα, χρησιμοποιώντας το τηλεφωνικό δίκτυο ή ασύρματα μέσω συστημάτων επικοινωνίας με δορυφόρο. Παράδειγμα δικτύου ευρείας περιοχής είναι το δίκτυο υπολογιστών μίας τράπεζας. Χάρη σε αυτό, μπορεί να γίνει μία ανάληψη από ένα υποκατάστημα και να ενημερωθούν ταυτόχρονα για αυτή την κίνηση όλα τα υποκαταστήματα της αλυσίδας της τράπεζας. Φυσικά προσφέρονται και πολλές άλλες δυνατότητες που δεν είναι ορατές στον πελάτη.



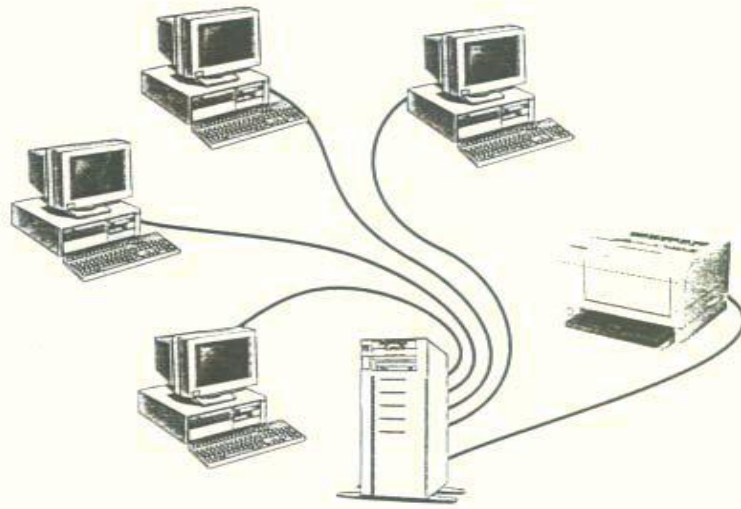
Σχήμα 26: Απεικόνιση δικτύου ευρείας περιοχής

Η διασύνδεση τοπικών δικτύων και δικτύων ευρείας περιοχής, ανεξαρτήτως λειτουργικού συστήματος και αρχιτεκτονικής, δημιουργεί το διαδίκτυο.

6.2. Δομικά Στοιχεία Δικτύου

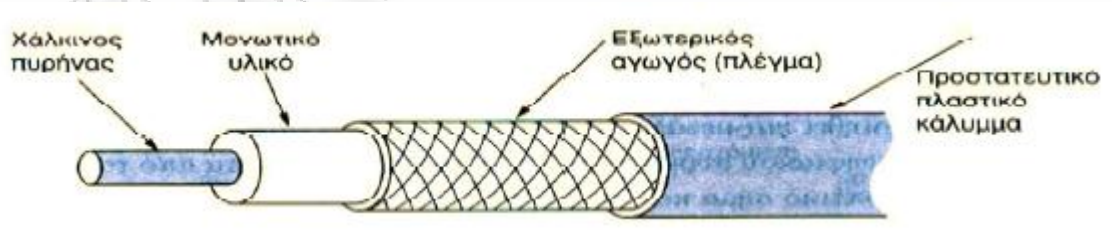
Τα απαραίτητα δομικά στοιχεία για να εγκατασταθεί ένα δίκτυο είναι τα ακόλουθα:

1. **Το Υλικό:** Δηλαδή εννοούνται οι υπολογιστές και οι περιφερειακές συσκευές που θα συνδεθούν στο δίκτυο (Σχήμα 27), τις κάρτες δικτύου, καθώς επίσης και κάποιες συσκευές του εξοπλισμού του δικτύου (το Hub που χρησιμοποιείται για τη διασύνδεση των υπολογιστών ενός δικτύου, το Router που χρησιμοποιείται για τη διασύνδεση δικτύων υπολογιστών κτλ.). Οι υπολογιστές ενός δικτύου μπορούν να διακριθούν σε:
 - A. **Διακομιστές (Server):** Είναι υπολογιστές με αυξημένες δυνατότητες και αρμοδιότητες, που είναι επιφορτισμένοι με την απρόσκοπτη λειτουργία του δικτύου.
 - B. **Σταθμοί εργασίας (Workstations):** Είναι υπολογιστές με δυνατότητες λιγότερες από αυτές των διακομιστών και σε αυτούς εργάζονται οι χρήστες του δικτύου.

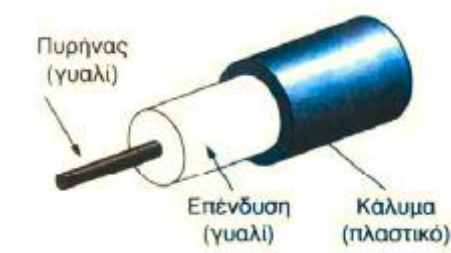


Σχήμα 27: Αναπαράσταση συνδεδεμένων σταθμών εργασίας και μίας περιφερειακής συσκευής (εκτυπωτής), ενός τοπικού δικτύου σε έναν διακομιστή

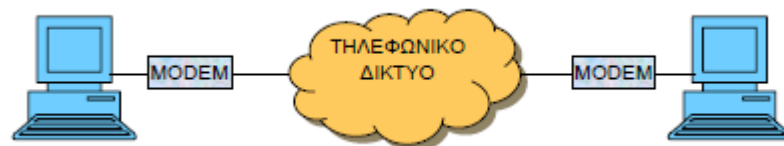
2. **Το λογισμικό:** Το λογισμικό περιλαμβάνει το λειτουργικό σύστημα του δικτύου και τις εφαρμογές που θα χρησιμοποιήσουν οι χρήστες από τους σταθμούς εργασίας.
3. **Το μέσο διασύνδεσης:** Το μέσο διασύνδεσης για τα τοπικά δίκτυα είναι συνήθως κάποιος τύπος καλώδιου χαλκού (Σχήμα 28) (ομοαξονικό, συνεστραμμένου ζεύγους κλπ.) ή ακόμα και οπτικές ίνες (Σχήμα 29). Η σύνδεση μπορεί να επιτυγχάνεται και ασύρματα με τη χρήση πομποδεκτών. Για τα δίκτυα ευρείας περιοχής το μέσο διασύνδεσης είναι το καλώδιο του τηλεφωνικού δικτύου (Σχήμα 30) ή συστήματα επικοινωνίας με δορυφόρο (Σχήμα 31). Σε κάποιες περιπτώσεις χρησιμοποιείται και η οπτική ίνα.



Σχήμα 28: Ομοαξονικό καλώδιο.



Σχήμα 29: Οπτική ίνα.



Σχήμα 30: .Χρησιμοποίηση του τηλεφωνικού δικτύου για διασύνδεση σε ένα δίκτυο ευρείας περιοχής.

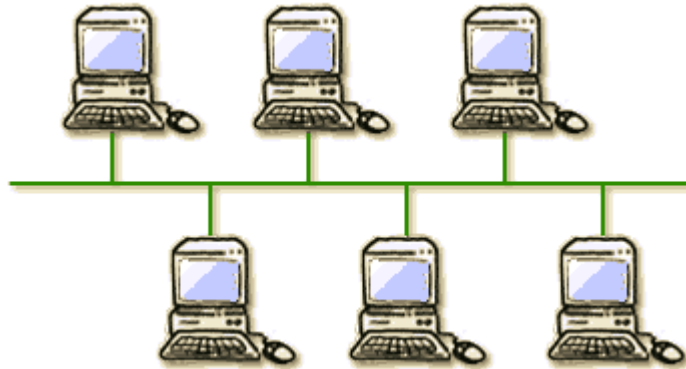


Σχήμα 31: .Χρησιμοποίηση συστημάτων επικοινωνίας με δορυφόρο.

4. **Την τοπολογία:** Ο τρόπος με τον οποίο συνδέονται μεταξύ τους οι υπολογιστές και οι περιφερειακές συσκευές ενός δικτύου ονομάζεται τοπολογία. Οι πιο ευρέως χρησιμοποιούμενες τοπολογίες δικτύων είναι η τοπολογία Αρτηρίας ή “Διαύλου” (“Bus Network”) (Σχήμα 32α), η τοπολογία “Δακτυλίου” (“Ring Network”) (Σχήμα 32β), η τοπολογία “Αστέρα” (“Star Network”) (Σχήμα 32γ) και η τοπολογία “Πλέγματος” (“Mesh Network”) (Σχήμα 32δ).

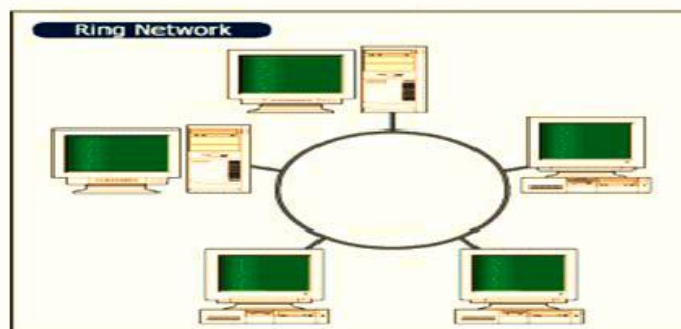
Ακολουθεί μία συνοπτική περιγραφή των τοπολογιών.

Στην **τοπολογία Διαύλου (Bus Network)**, όλα τα μέλη του δικτύου επικοινωνούν μέσω ενός κοινού διαύλου που τα ενώνει σειριακά. Αυτό σημαίνει πως το σήμα μεταδίδεται από μέλος σε μέλος έως ότου να βρεθεί ο σωστός παραλήπτης, ο οποίος τελικά λαμβάνει και το πακέτο. Στα δύο άκρα του διαύλου συνδέονται αντιστάσεις που τον τερματίζουν. Ένα από τα μειονεκτήματα της τοπολογίας αυτής είναι πως σε περίπτωση βλάβης ενός σταθμού "σπάει" ο δίαυλος και το δίκτυο καταρρέει.



Σχήμα 32α: Συνδεσμολογία Αρτηρίας

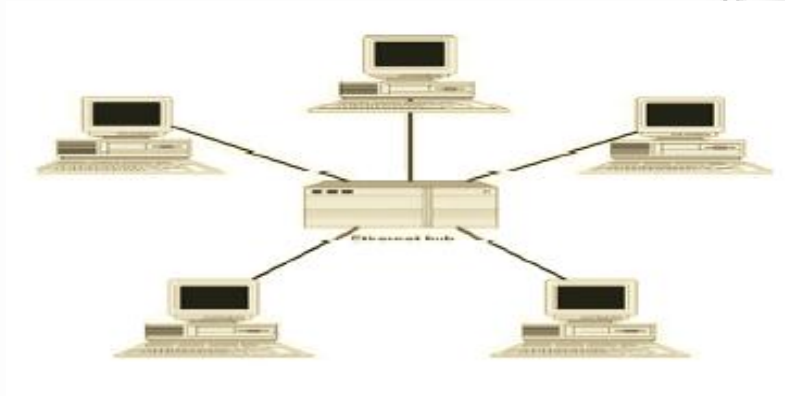
Η **τοπολογία Δακτυλίου (Ring Network)**, είναι παρόμοια της τοπολογίας διαύλου με τη μόνη διαφορά πως αντί για αντιστάσεις στα άκρα του διαύλου, ενώνονται τα δύο ακριανά μέλη του δικτύου μεταξύ τους (το πρώτο και το τελευταίο δηλαδή), σχηματίζοντας έτσι σχήμα δακτυλίου.



Σχήμα 32β: Τοπολογία Δακτυλίου.

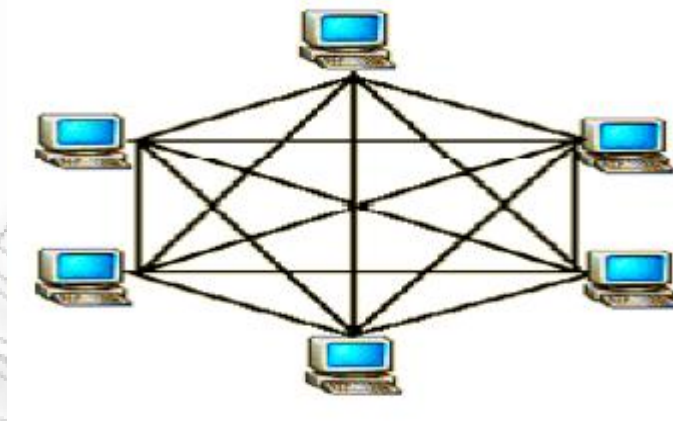
Στην **τοπολογία Αστέρα (Star Network)**, όλοι οι σταθμοί εργασίας συνδέονται άμεσα με έναν κεντρικό υπολογιστή, το Διακομιστή ή Εξυπηρετητή (Server) ή κάποια δικτυακή συσκευή όπως π.χ. έναν δρομολογητή (router), μέσω του οποίου ανταλλάσσουν δεδομένα. Ένα μεγάλο πλεονέκτημα είναι πως το δίκτυο δεν εξαρτάται από την κατάσταση του κάθε σταθμού εργασίας αλλά μόνο από την κατάσταση της κεντρικής μονάδας στην οποία συνδέονται όλα τα μέλη. Το μειονέκτημα είναι πως στην περίπτωση δυσλειτουργίας της κεντρικής μονάδας η

επικοινωνία διακόπτεται εντελώς. Αυτό βέβαια μπορεί να αντιμετωπιστεί με την εγκατάσταση περισσότερων κεντρικών μονάδων έτσι ώστε σε περίπτωση δυσλειτουργίας της μίας να συνεχίζει η άλλη.



Σχήμα 32γ: Τοπολογία Αστέρα.

Η τοπολογία Πλέγματος, χωρίζεται στο **Πλήρες Πλέγμα (Full Mesh)** και **Μερικό Πλέγμα (Partial Mesh)**. Στο **Πλήρες Πλέγμα** κάθε σταθμός εργασίας συνδέεται απευθείας με όλους τους υπόλοιπους σταθμούς, επικοινωνεί δηλαδή άμεσα με καθένα από αυτούς. Στο **Μερικό Πλέγμα**, κάποιοι σταθμοί επικοινωνούν άμεσα με μερικούς ή όλους από τους σταθμούς εργασίας, ενώ κάποιοι άλλοι μόνο με γειτονικούς. Το πλεονέκτημα της παρούσας συνδεσμολογίας είναι πως ακόμα και σε περίπτωση μερικής καταστροφής του μέσου, υπάρχει η δυνατότητα επικοινωνίας δύο σταθμών μέσω εναλλακτικών διαδρομών. Το μειονέκτημα είναι η περιττή καλωδίωση και το κόστος, στην περίπτωση που το μέσο είναι η καλωδίωση.



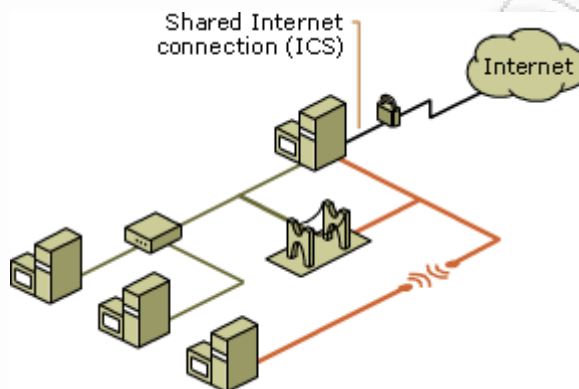
Σχήμα 32δ: Τοπολογία Πλέγματος.

6.3. Συσσκευές Διασύνδεσης

Όπως έχει προαναφερθεί, για τη λειτουργία ενός δικτύου απαιτείται η χρήση ενδιάμεσων περιφερειακών συσκευών, οι οποίες μεταφέρουν πληροφορίες και δεδομένα από ένα δίκτυο ή έναν χρήστη σε ένα ή έναν άλλο. Αυτές οι συσκευές έχουν διάφορα πλεονεκτήματα και μειονεκτήματα. Σε αυτήν την παράγραφο ακολουθεί μία συνοπτική περιγραφή της λειτουργίας τους.

6.3.1. Γέφυρες (Bridges)

Οι γέφυρες (bridges) χωρίζουν ένα σχετικά περιορισμένης έκτασης δίκτυο σε μικρότερα δίκτυα LAN (π.χ. το LAN ενός οργανισμού που βρίσκεται σε ένα κτίριο ή συγκρότημα κτιρίων, στα επιμέρους LAN των διαφόρων τμημάτων του οργανισμού), τα οποία είναι δυνατόν να χρησιμοποιούν διαφορετικά πρωτόκολλα σύνδεσης δεδομένων. Τα τελευταία χρόνια οι γέφυρες έχουν αντικατασταθεί με τους “μεταγωγείς”, υπάρχουν όμως εφαρμογές που τις χρησιμοποιούν ακόμα.



Σχήμα 33: Δικτύωση με γέφυρες, το αποκαλούμενο “bridging networking”

Η γέφυρα δέχεται την ακολουθία των bit του σήματος και συνθέτει τα στοιχειώδη πλαίσια (frames) του επίπεδου σύνδεσης δεδομένων. Στη συνέχεια διαβάζει τη φυσική διεύθυνση (physical address) του προορισμού και αφού ελέγξει τον πίνακα (ή τους πίνακες) φυσικών διευθύνσεων που διαθέτει, στέλνει τα δεδομένα στο κατάλληλο υποδίκτυο LAN. Αν το LAN του πομπού και του δέκτη χρησιμοποιούν διαφορετικό πρωτόκολλο σύνδεσης δεδομένων, η γέφυρα πραγματοποιεί τις απαραίτητες αλλαγές στα δεδομένα.



Σχήμα 34: Συσσκευή “γέφυρα” του εμπορίου.

Οι κυριότερες κατηγορίες γεφυρών είναι:

- **Απλές Γέφυρες (Simple Bridges):** Συνδέουν μόνο δύο υποδίκτυα και χρησιμοποιούν έναν πίνακα φυσικών διευθύνσεων για όλες τις συσκευές και των δύο υποδικτύων. Ο πίνακας αυτός δημιουργείται από ένα διαχειριστή της γέφυρας. Ο διαχειριστής πρέπει να ενημερώνει τον πίνακα κάθε φορά που συνδέεται ή αποσυνδέεται μία συσκευή σε οποιοδήποτε υποδίκτυο.
- **Γέφυρες Πολλαπλών Θυρών (Multiport Bridges):** Αυτές μπορούν να συνδέσουν περισσότερα από δύο υποδίκτυα και περιλαμβάνουν ισάριθμους πίνακες, ως προς τις θύρες εξόδου. Ο κάθε πίνακας περιέχει τις φυσικές διευθύνσεις των συσκευών του υποδικτύου που είναι προσπελάσιμο από την αντίστοιχη θύρα.
- **Διαφανείς Γέφυρες (Transparent Bridges):** Αυτές είναι έξυπνες γέφυρες που έχουν την ικανότητα να μαθαίνουν από μόνες τους τις φυσικές διευθύνσεις των συσκευών, ενημερώνοντας τον πίνακα διευθύνσεων. Για να το κάνουν αυτό διαβάζουν τη διεύθυνση της πηγής κάθε φορά που λαμβάνουν ένα πακέτο (frame), δεδομένου ότι είναι γνωστό για μία γέφυρα το υποδίκτυο στο οποίο βρίσκεται η πηγή. Έπειτα, η διεύθυνση αυτή και τα στοιχεία του υποδικτύου

τοποθετούνται σε έναν πίνακα. Όταν, λοιπόν, σε κάποιο μελλοντικό πλαίσιο διαβαστεί στο πεδίο προορισμού η παραπάνω διεύθυνση, η γέφυρα γνωρίζει που να το στείλει. Αν κάποια διεύθυνση δεν έχει καταγραφεί ακόμη, η γέφυρα στέλνει το αντίστοιχο πλαίσιο σε όλα τα υποδίκτυα. Η διαδικασία αυτή λέγεται “πλημμύρισμα” και κάτω από συγκεκριμένες συνθήκες (π.χ. όταν δύο LAN συνδέονται με δύο ή περισσότερες γέφυρες), μπορεί να οδηγήσει σε φαινόμενα δημιουργίας βρόχων (loops). Για την αποφυγή αυτών των φαινομένων χρησιμοποιούνται κατάλληλοι αλγόριθμοι.

- **Γέφυρες Πηγαίας Δρομολόγησης (Source Routing Bridges):** Το δίκτυο με αυτές τις συσκευές λειτουργεί ως εξής: Στο κάθε πλαίσιο (πακέτο), το πρώτο bit της διεύθυνσης προορισμού δηλώνει το αν η πηγή και ο προορισμός βρίσκονται στο ίδιο LAN. Αν η πηγή και ο προορισμός ανήκουν σε διαφορετικό LAN, η επικεφαλίδα του πλαισίου διαμορφώνεται με τέτοιο τρόπο ώστε να αναφέρεται το μονοπάτι με τις διαδοχικές γέφυρες και τα διαδοχικά LAN που οδηγούν στον προορισμό. Για να επιτευχθεί αυτό, οι γέφυρες που συνδέονται άμεσα με κάποια LAN χαρακτηρίζονται μονομερώς από έναν αριθμό για τα LAN αυτά και το κάθε LAN περιγράφεται από έναν μοναδικό αριθμό. Αυτού του είδους οι γέφυρες εμποδίζουν τη δημιουργία βρόχων.

6.3.2. Δρομολογητές (Routers)

Πρόκειται για ακόμα πιο εξελιγμένες συσκευές, οι οποίες συνδέουν τα δίκτυα ενός διαδικτύου ή τα δίκτυα ενός WAN (Σχήμα 35) που απλώνεται σε μεγάλης έκτασης γεωγραφικές περιοχές. Γενικά, χρησιμοποιούνται σε τέτοια δικτυακά περιβάλλοντα, όπου οι συσκευές αναγνωρίζονται με βάση τη λογική τους διεύθυνση. Στο σχήμα 35α φαίνεται ένας τυπικός δρομολογητής (router) του εμπορίου.



Σχήμα 35: Δικτύωση μέσω δρομολογητή.

Ο δρομολογητής ανακατασκευάζει τα πακέτα δικτύου από τα πλαίσια που λαμβάνει. Έπειτα εξετάζει τη λογική διεύθυνση και ανάλογα δρομολογεί τα πακέτα στο κατάλληλο δίκτυο, ώστε να φτάσουν στον προορισμό. Οι δρομολογητές μπορούν να συνδέουν ταυτόχρονα περισσότερα από δύο δίκτυα και διαθέτουν εξελιγμένο λογισμικό που μπορεί να εξετάζει το βέλτιστο μονοπάτι, κάνοντας χρήση ενός πίνακα δρομολόγησης, για να προωθεί τα πακέτα στον προορισμό. Το βέλτιστο μονοπάτι είναι το μονοπάτι μικρότερου κόστους και τα κριτήρια με τα οποία επιλέγεται το κόστος αυτό, διαφέρουν ανάλογα με τα δίκτυα που συνδέει. Ένα κριτήριο, ίσως το πιο συνηθισμένο, είναι ο αριθμός των βημάτων που απαιτούνται ως τον

προορισμό. Υπάρχουν και άλλα κριτήρια, όπως για παράδειγμα, τα στρατιωτικά δίκτυα όπου επιλέγεται το μονοπάτι με τη δυνατότερη δυνατή ασφάλεια.

Ένας δρομολογητής μπορεί να χρησιμοποιεί το ίδιο μονοπάτι για όλα τα πακέτα της ίδιας εφαρμογής προς τον προορισμό. Μπορεί επίσης να δρομολογεί τα πακέτα της ίδιας εφαρμογής από διαφορετικά μονοπάτια, ανάλογα με τις συνθήκες που επικρατούν στο διαδίκτυο.

Για να αποφεύγονται τα φαινόμενα βρόχων, τα πακέτα φέρουν μία μεταβλητή που ονομάζεται "Χρόνος Ζωής Πακέτου" (TTL–Time To Live ή Packet Lifetime). Η μεταβλητή αυτή εκφράζει ένα μέγιστο όριο βημάτων που αν ξεπεραστεί το πακέτο αυτό απορρίπτεται από τον πρώτο δρομολογητή που θα το συναντήσει. Ο κάθε δρομολογητής αφαιρεί μία μονάδα από το όριο αυτό ώσπου να μηδενιστεί, οπότε και απορρίπτεται το πακέτο.



Σχήμα 35α: Δρομολογητής του εμπορίου

6.3.3. Πύλες (Gateways)

Η πύλη (gateway) είναι η πιο εξελιγμένη συσκευή δικτύωσης. Χρησιμοποιείται για να ενώνει δίκτυα τα οποία λειτουργούν με βάση διαφορετικές αρχιτεκτονικές (π.χ. να ενωθούν δίκτυα αρχιτεκτονικής TCP/IP με δίκτυο αρχιτεκτονικής OSI). Σε πολλές βιβλιογραφίες ως πύλη αναφέρεται η συσκευή που περιγράφεται, στο παρόν, ως δρομολογητής.

6.3.4. Άλλες Συσκευές

Κάποιες άλλες συσκευές δικτύωσης είναι οι ακόλουθες:

- **Δρομολογητές Πολλαπλών Πρωτοκόλλων (Multiprotocol routers):** Πρόκειται για δρομολογητές που συνδέουν δίκτυα, τα οποία χρησιμοποιούν διαφορετικό πρωτόκολλο δικτύου (π.χ. IP του Internet και IPX του Novell). Για να επιτευχθεί αυτό πρέπει να διατηρούν δύο ή περισσότερους διαφορετικούς πίνακες δρομολόγησης.
- **Brouters (Bridge/Router):** Είναι συσκευές που λειτουργούν άλλοτε σαν δρομολογητές και άλλοτε σαν γέφυρες. Δηλαδή, πρώτα προσπαθούν να προωθήσουν τα πακέτα με βάση τη λογική διεύθυνση και αν αποτύχει τα προωθεί με βάση τη φυσική διεύθυνση των πλαισίων που τα απαρτίζουν.
- **Μεταγωγείς (Switches):** Είναι παρόμοιες συσκευές με τις γέφυρες, αλλά πιο αποτελεσματικές. Χρησιμοποιούν ενταμιευτές στους οποίους αποθηκεύουν τα πλαίσια προτού τα προωθήσουν. Άλλοι μεταγωγείς δέχονται τα πλαίσια και τα αποθηκεύουν στον ενταμιευτή εισόδου περιμένοντας να συμπληρωθεί το πακέτο ώστε να το προωθήσουν στην έξοδο και άλλοι στέλνουν το πακέτο σε ένα ενταμιευτή εξόδου. Η τεχνολογία των μεταγωγέων έχει εξελιχθεί και σήμερα αυτοί μπορούν να λειτουργήσουν συνδυαστικά ως δρομολογητές και γέφυρες (brouters) και μάλιστα πολύ πιο γρήγορα από τους κοινούς δρομολογητές.

6.4. Το Πρωτόκολλο Επικοινωνίας TCP/IP

Το TCP/IP είναι το πρωτόκολλο επικοινωνίας για την επικοινωνία ανάμεσα σε υπολογιστές που είναι συνδεδεμένοι στο παγκόσμιο δίκτυο υπολογιστών. Τα αρχικά TCP/IP αναφέρονται σε δύο από τα σημαντικότερα πρωτόκολλα που χρησιμοποιούνται στο Διαδίκτυο, δηλαδή (όπως γίνεται εύκολα αντιληπτό, στο TCP και στο IP.

Τα αρχικά σημαίνουν **Transmission Control Protocol / Internet Protocol**, δηλαδή Πρωτόκολλο Ελέγχου Εκπομπής/Πρωτόκολλο του Internet. Μέσα στο TCP/IP, υπάρχουν πολλά πρωτόκολλα για το χειρισμό της επικοινωνίας των δεδομένων. Αυτά είναι:

- **To UDP (User Datagram Protocol)**, για απλή επικοινωνία ανάμεσα σε εφαρμογές.
- **To ICMP (Internet Control Message Protocol)**, για λάθη και στατιστικές.
- **To DHCP (Dynamic Host Configuration Protocol)**, για δυναμική διευθυνσιοδότηση.

Διαφορετικά πρωτόκολλα επικοινωνίας τα οποία βασίζονται στο TCP/IP.

- **HTTP (Hyper Text Transfer Protocol)**, το πρωτόκολλο αυτό φροντίζει για την επικοινωνία ανάμεσα σε έναν web server και έναν web browser. Χρησιμοποιείται για την αποστολή των αιτήσεων από έναν web Client σε έναν web Server καθώς και για την επιστροφή του περιεχομένου, δηλαδή την εμφάνιση των ιστοσελίδων από το web Server πίσω στο web Client.
- **To HTTPS (Secure HTTP)**, φροντίζει για την ασφαλή επικοινωνία ανάμεσα σε έναν web Server και έναν web browser.
- **To SSL (Secure Sockets Layer)**, χρησιμοποιείται για την κωδικοποίηση (κρυπτογράφηση - encryption) των δεδομένων για να είναι έτσι ασφαλής η μεταφορά τους.
- **To SMTP (Simple Mail Transfer Protocol)**, χρησιμοποιείται για την αποστολή των μηνυμάτων ηλεκτρονικού ταχυδρομείου (e-mails) από το χρήστη σε έναν e-mail Server.

6.5. Η Client - Server Τεχνολογία

Η τεχνολογία των υπολογιστών αναπτύχθηκε βαθμιαία, με τέτοιο τρόπο όπου κάθε καινούργια αρχιτεκτονική έπαιρνε τα πλεονεκτήματα από τις ήδη υπάρχουσες τεχνικές και να εκμεταλλεύονται όλες οι δυνατότητες των υπολογιστών. Σήμερα, οι υπολογιστές είναι μικρότεροι, γρηγορότεροι και φθηνότεροι από ότι παλιότερα. Σαν αποτέλεσμα αυτής της γενικής κατεύθυνσης είναι η διανομή της επεξεργασίας της πληροφορίας αλλά και της ίδιας της πληροφορίας σε όλο το πλήθος αυτών των νέων υπολογιστών.

Ο όρος αρχιτεκτονική συνήθως χρησιμοποιείται για να περιγράψει συστήματα διαχείρισης βάσεων δεδομένων, λειτουργικά συστήματα και άλλους υπολογιστικούς μηχανισμούς λογισμικού και υλικού. Οι αρχιτεκτονικές περιγράφουν πως οι συσκευές και τα λογισμικά πακέτα ταιριάζουν, για να φτιάξουν ένα εύκολο στη χρήση και τη διαχείριση σύνολο.

Η κλασική αρχιτεκτονική αποτελείται από έναν υπολογιστή μεγάλης ισχύος, το Server (που παίζει το ρόλο του οικοδεσπότη-Host) με ένα ή περισσότερα απλά τερματικά (terminals). Οι εφαρμογές ελέγχονται και διανέμονται από αυτόν τον υπολογιστή - "οικοδεσπότη". Σε αυτόν πραγματοποιούνται όλες οι διαχειρίσεις πληροφοριών, η λογική των εφαρμογών και η μορφοποίηση της εμφάνισής τους. Οι χρήστες αλληλεπιδρούν με το κεντρικό σύστημα μέσω των τερματικών τους, τα οποία εμφανίζουν μόνο πληροφορίες. Αυτή είναι η πιο συνηθισμένη αρχιτεκτονική σήμερα.

Ένα καλά οργανωμένο σύστημα που χρησιμοποιεί αυτήν την κλασική αρχιτεκτονική προσφέρει τις ακόλουθες δυνατότητες:

- Ένα υψηλό επίπεδο αξιοπιστίας .
- Κεντρικό έλεγχο και κεντρική διαχείριση των πληροφοριών.
- Ισχυρή διαχείριση των πληροφοριών και δυνατότητα αποθηκεύσεων.

Πάντως, οι κλασικές εφαρμογές περιορίζουν την ευελιξία των τελικών χρηστών. Η διασύνδεση των χρηστών δεν είναι γραφική, κάτι που κάνει το σύστημα δυσκολότερο στη χρήση και σημαίνει ότι ο χρήστης πρέπει να μάθει πως να χρησιμοποιήσει τη γλώσσα του οικοδεσπότη. Επίσης, οι εφαρμογές εξαρτώνται από μια πλατφόρμα, που σημαίνει ότι εάν κάτι συμβεί στον υπολογιστή - "οικοδεσπότη", ο χρήστης δεν μπορεί να χρησιμοποιήσει το σύστημα, έως ότου το σύστημα αρχίσει να επαναλειτουργεί.

Στην Client-Server αρχιτεκτονική, η Client εφαρμογή τρέχει σε έναν πλήρη σταθμό εργασίας. Αυτός ο σταθμός μπορεί να είναι ένας προσωπικός υπολογιστής, ένας UNIX σταθμός εργασίας ή ένας Mac. Η Client εφαρμογή βασίζεται στις υπηρεσίες που προσφέρει ο Server και

επικοινωνούν μέσω πρωτοκόλλων, όπως το πρωτόκολλο του Internet (TCP/IP) ή του Novell (IPX/SPX).

Το περιβάλλον του Client - Server έχει πολλά πλεονεκτήματα σε σχέση με τις κλασικές αρχιτεκτονικές. Η διαχείριση της διασύνδεσης των χρηστών και άλλες επεξεργασίες είναι αποφορτισμένα από τον "οικοδεσπότη", ενώ ο Server ακόμη προσφέρει συγκεντρωμένο έλεγχο των κοινών πόρων. Επειδή ο Client επικοινωνεί με το Server μέσω ενός καθορισμένου συστήματος διασύνδεσης και δεν χρειάζεται να γνωρίζει που ανήκει ο Server ή πως ενεργεί. Ο σταθμός εργασίας τρέχει την εφαρμογή και εμφανίζει τις πληροφορίες στο χρήστη. Μόνο όταν ο Client προσπελάζει πληροφορίες, τότε εγκαθίσταται επικοινωνία με το Server. Ο φόρτος εργασίας μειώνεται δραματικά στον υπολογιστή – "οικοδεσπότη" όσο αυξάνεται η ισχύς κάθε σταθμού εργασίας.

Οι οργανισμοί έχουν να κάνουν με συνεχώς περισσότερα δεδομένα, τα οποία πρέπει να τα διαχειρίζονται και να τα εκμεταλλεύονται στις εργασίες τους. Η αύξηση του όγκου των δεδομένων, σε συνδυασμό με την προσπάθεια των οργανισμών να μειώσουν το κόστος, να αυξήσουν την παραγωγικότητα και να βελτιώσουν τις υπηρεσίες των πελατών (με καλύτερη χρήση πληροφοριών και ταχύτερο χρόνο ανταπόκρισης στους πελάτες ταυτόχρονα), έχουν συμβάλει σε μια ώθηση για δημιουργία και χρήση Client - Server εφαρμογών.

Σε συνδυασμό με τις μεταβαλλόμενες απαιτήσεις των επιχειρήσεων, η ανάπτυξη της τεχνολογίας των Client-Server έχουν οδηγήσει στα ακόλουθα:

1. Πρόοδο στο υλικό.
2. Πρόοδο στο λογισμικό.
3. Πρόοδο στο δίκτυο.

Φυσικά, χωρίς τα Τοπικά Δίκτυα (Local Area Network) και το Διαδίκτυο (Internet) η τεχνολογία των Client-Server δεν θα υπήρχε. Τα δίκτυα προσφέρουν την υποστήριξη της επικοινωνίας που απαιτείται για να συνδεθούν πολλές μηχανές, ακόμα και πολλές πλατφόρμες.

Ένα δίκτυο είναι ένα σύστημα που μεταφέρει πληροφορίες και μηνύματα μεταξύ των επεξεργαστών. Η αρχιτεκτονική ενός δικτύου έχει κανόνες, ή αλλιώς πρωτόκολλα, που καθορίζουν το πως πραγματοποιούνται οι μεταφορές μέσα σε αυτή την αρχιτεκτονική. Διαφορετικά υλικά και λογισμικά μπορούν να επικοινωνούν όσο χρησιμοποιούν τα ίδια τα πρωτόκολλα και τις ίδιες μορφές δεδομένων.

Η πρόοδος στην τεχνολογία των δικτύων επιτρέπει στους σταθμούς εργασίας να συνδέονται σε πολλαπλές πηγές πληροφοριών, δημιουργώντας ένα περιβάλλον βασισμένο στο χρήστη, όπου ο χρήστης έχει άμεση πρόσβαση σε οποιαδήποτε πληροφορία απαιτείται, ανεξάρτητα του που βρίσκεται η πληροφορία αυτή. Η άμεση πρόσβαση σημαίνει ότι ο χρήστης μπορεί να προσπελάσει πληροφορίες από απομακρυσμένες μηχανές χωρίς καν να πρέπει να αλληλεπιδράσει άμεσα με αυτές τις μηχανές ή να γνωρίζει ότι και άλλες μηχανές εμπλέκονται στη διαδικασία. Το δίκτυο μπορεί να είναι μια διάταξη υπολογιστών μέσα σε μια πόλη, σε μια χώρα, ή σε ολόκληρο τον κόσμο.

7. Οι Θύρες Επικοινωνίας του Υπολογιστή

Στο παρόν κεφάλαιο, θα γίνει μία αναφορά στις θύρες επικοινωνίας του Η/Υ, οι οποίες είναι χρήσιμες για τη διασύνδεση του Η/Υ με τις περιφερειακές συσκευές. Περιφερειακές συσκευές νοούνται απλές συσκευές όπως οθόνες, πληκτρολόγια, εκτυπωτές, αλλά μπορεί να είναι και πιο σύνθετες όπως κάρτες δικτύου, modem ή ακόμα και κάποιες συσκευές οι οποίες προορίζονται για περισσότερο σύνθετη και δύσκολη χρήση όπως βιομηχανική χρήση. Χαρακτηριστικές τέτοιες συσκευές αποτελούν, τα αισθητήρια, τα όργανα μετρήσεων ή ακόμη και ρομπωτικοί βραχίονες.

Ανάλογα με τον τρόπο που μεταδίδονται τα bits πληροφορίας (είτε παράλληλα-ταυτόχρονα, είτε σε σειρά) και το ποια θύρα χρησιμοποιούμε για την εφαρμογή που μας ενδιαφέρει, πετυχαίνουμε και ανάλογα αποτελέσματα. Ακολουθεί μία μικρή περιγραφή.

7.1. Σειριακές Θύρες Επικοινωνίας (Serial Ports)

Στους υπολογιστές, η σειριακή θύρα μεταφέρει δεδομένα σε σειρά δηλαδή ένα bit ανά στιγμή (και όχι πολλά bit ταυτόχρονα όπως στην παράλληλη θύρα). Πολλά από τα γνωστά μέσα διασύνδεσης όπως το Ethernet, το FireWire και το USB, όλα αυτά χρησιμοποιούν το σειριακό τρόπο μεταφοράς δεδομένων. Παρόλα αυτά, ο όρος “σειριακή θύρα” έχει επικρατήσει ιστορικά να εννοεί την RS 232, ακόμα και σήμερα όπου ο ρόλος της έχει περιοριστεί σημαντικά.

Σήμερα, σαν μέθοδος διασύνδεσης με τις περιφερειακές συσκευές, χρησιμοποιείται ευρέως η USB και το FireWire. Βέβαια η χρήση της κλασσικής σειριακής θύρας δεν έχει εκλείψει ακόμα εντελώς. Χρησιμοποιείται σε βιομηχανικά συστήματα αυτοματισμών, επιστημονικές αναλύσεις και σε συστήματα Server σαν κονσόλα ελέγχου για να εκτελεί διαγνωστικά εργαλεία. Στον εξοπλισμό δικτύωσης (routers, switches), συχνά χρησιμοποιείται αυτή η κονσόλα ελέγχου για τη διαμόρφωση τους. Επίσης πρέπει να αναφερθεί ότι η σειριακή θύρα λίγες φορές απαιτεί υποστηρικτικό λογισμικό.

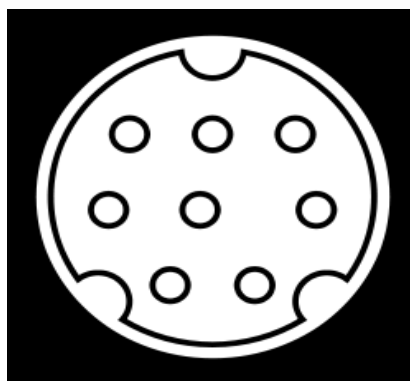
7.1.1. RS 232 (Recommended Standard 232)

Το RS – 232 πρωτοεμφανίστηκε το 1962. Ο Τερματικός Εξοπλισμός Δεδομένων (Data Terminal Equipment – DTE) της εποχής ήταν ηλεκτρομηχανικοί τηλεγραφητές και ο Εξοπλισμός Επικοινωνίας Δεδομένων (Data Communication Equipment - DCE) ήταν τα modem. Όταν τα ηλεκτρονικά τερματικά ξεκίνησαν να χρησιμοποιούνται, συχνά ήταν σχεδιασμένα ώστε να συνεργάζονται με τους παλιούς τηλέτυπους και ως εκ τούτου υποστήριζαν τη συνδεσιμότητα RS – 232. Το 1969 εμφανίστηκε η τελική μορφή του πρότυπου η οποία ήταν προσαρμοσμένη στα ηλεκτρικά χαρακτηριστικά εκείνων των συσκευών. Το πρώτο καιρό της παρουσίας του, υπήρχαν αρκετά προβλήματα συμβατότητας λόγω του ότι, ο κάθε κατασκευαστής ακολουθούσε μια δική του τυποποίηση η οποία είχε κάποιες αποκλίσεις από το αρχικό πρότυπο. Μια συνηθισμένη απόκλιση που εμφανιζόταν ήταν η οδήγηση σημάτων μειωμένης τάσης. Αναφορικά, το πρότυπο απαιτεί από τον πομπό να χρησιμοποιεί τάση +12V ή -12V για να εκπέμψει ένα σήμα και από το δέκτη απαιτεί τάσεις της τάξεως +3V έως -3V για να ξεχωρίσει και να αποκωδικοποιήσει το εκπεμπόμενο σήμα. Μερικοί κατασκευαστές ως εκ τούτου, κατασκεύασαν πομπούς +5V και -5V και τους ονόμασαν “συμβατούς RS-232”.

Αργότερα με την εμφάνιση των προσωπικών υπολογιστών, ξεκίνησε να υπάρχει μία αυστηρότερη τυποποίηση, καθώς πολλές περιφερειακές συσκευές χρησιμοποιούσαν τη συγκεκριμένη συνδεσιμότητα και με βάση το γεγονός ότι οι προσωπικοί υπολογιστές έπρεπε να εξυπηρετούν το μέσο άνθρωπο και όχι ειδήμονες επιστήμονες. Από τα μέσα της δεκαετίας του 1990 μέχρι και τα πιο πρόσφατα χρόνια, η χρήση της RS–232 ήταν τόσο διαδεδομένη όσο της USB. Ακόμα και σήμερα βέβαια χρησιμοποιείται σαν μέθοδος συνδεσιμότητας περιφερειακών των προσωπικών υπολογιστών και βιομηχανικού εξοπλισμού (όπως τα PLC).



Σχήμα 36: Αρσενικό θύρας RS–232. Το συγκεκριμένο πρότυπο είναι που επικράτησε έναντι όλων των άλλων και είναι το DE–9.



Σχήμα 37: Αρσενικό θύρας RS–232. Το συγκεκριμένο πρότυπο είναι το DE–8 και χρησιμοποιήθηκε στους υπολογιστές Mac.

7.1.2. USB (Universal Serial Bus)



Σχήμα 38: Το σήμα κατατεθέν της USB θύρας. Αποτυπώνεται επάνω σε κάθε συσκευή που υποστηρίζει τη συγκεκριμένη συνδεσιμότητα.

Η USB (Universal Serial Bus—Σχήμα 38) συνδεσιμότητα πρωτοεμφανίστηκε το 1996 και αναπτύχθηκε από τη συνεργασία των Hewlett Packard, της Intel, της Lucent (πλέον Alcatel-Lucent), της Microsoft, της NEC και της Philips. Από το 1997 και με τους προσωπικούς υπολογιστές που είχαν λογισμικό Windows 98, άρχισε η υποστήριξη της συγκεκριμένης συνδεσιμότητας.

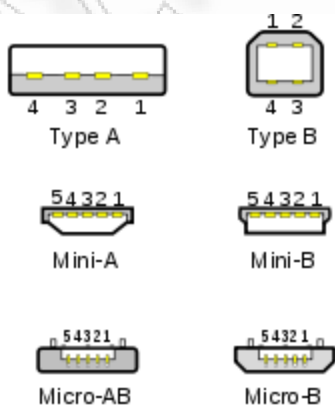
Δεν χρειάστηκε πολύς καιρός έως ότου η δημοτικότητα της USB θύρας να γίνει τόσο μεγάλη, ώστε όλες οι περιφερειακές συσκευές να αντικαταστήσουν τα παλιά πρότυπα σύνδεσης με αυτό το νέο πρότυπο. Σύμφωνα με τα δεδομένα του

έτους 2008, περίπου 2 δισεκατομμύρια συσκευές με συνδεσιμότητα USB, πουλήθηκαν.

Το USB πρότυπο, μέχρι στιγμής έχει παρουσιαστεί σε τρεις διαφορετικές εκδόσεις - ταχύτητες. Το πρώτο ήταν το USB 1.0 που εμφανίστηκε το 1996 (η πρώτη έκδοση) και υποστήριζε ταχύτητα μεταφοράς δεδομένων 1,5 Mbit/sec. Η δεύτερη ανανεωμένη έκδοση του USB, δεν άργησε να έρθει στην αγορά και υποστήριζε συσκευές που απαιτούσαν υψηλές ταχύτητες, όπως οι σκληροί δίσκοι και το μέγεθος των ταχυτήτων ήταν από 1,5 έως 12 Mbit/sec. Η επανάσταση ήρθε τον Απρίλιο του 2000 όπου παρουσιάστηκε η έκδοση 2.0. Ήταν ικανή να μεταφέρει δεδομένα με ταχύτητες έως 480 Mbit/sec και ο εξοπλισμός προσαρμοζόταν σε οποιαδήποτε προηγούμενη έκδοση (1.0 και 1.1).

Πλέον, αναμένεται η έκδοση USB 3.0 η οποία έχει ανακοινωθεί από το Νοέμβριο του 2008 από την Intel που την ανέπτυξε ότι είναι έτοιμη προς εμπορική χρήση. Η δεύτερη ονομασία που φέρει είναι SuperSpeed δίαυλος και υπόσχεται ταχύτητες έως 5 Gbit/sec.

Τα βύσματα της σύνδεσης USB, μπορούν να βρεθούν στο εμπόριο σε διάφορες μορφές. Στο Σχήμα 39 φαίνονται όλα τα USB άκρα που υπάρχουν και απαριθμείται και ο αριθμός των ακίδων που φέρει το καθένα. Η αρχική προδιαγραφή USB είχε σαν βασικά βύσματα τα A και B. Αργότερα στις εκδόσεις 1.1 και 2.0 άρχισε η υποστήριξη των βυσμάτων mini και micro, τα οποία χρησιμοποιούνται για μικρότερες συσκευές όπως PDA's, κινητά τηλέφωνα και ψηφιακές κάμερες.



Σχήμα 39: Τα διάφορα USB βύσματα που υπάρχουν στην αγορά.

7.1.3. FireWire (IEEE 1394 Interface)

Η θύρα FireWire παρουσιάστηκε πρώτη φορά γύρω στο 1995. Αναπτύχθηκε από την Apple με σκοπό να αντικαταστήσει την παράλληλη σύνδεση διαύλου SCSI (Small Computer System Interface - διεπαφή επικοινωνίας του υπολογιστή με διάφορα κομμάτια υλικού όπως σκληρό δίσκο, CD-ROM κλπ). Ο σκοπός χρήσης της θύρας ήταν ως επί το πλείστον πολυμεσικός, δηλαδή για σύνδεση με συσκευές DV Cameras (Digital Video Cameras), αλλά και για χρήσεις υπολογιστικής όρασης και για επαγγελματικά συστήματα ήχου.

Η ταχύτητα μεταφοράς δεδομένων για την εποχή του αλλά και για σήμερα είναι πολύ μεγάλη καθώς κυμαίνεται από 400 έως 3200 Mbit/sec (δηλαδή 50 έως 400 MB/sec) και σαν επιδόσεις ανταγωνίζεται πολύ εύκολα την USB συνδεσιμότητα. Στην αγορά βέβαια δεν βρήκε την πρόπευση ανταπόκριση, καθώς ακόμα και σήμερα συναντιέται μόνο σε ακριβές περιφερειακές συσκευές, σε αντίθεση με την USB συνδεσιμότητα που συναντιέται σε οποιοδήποτε κόστος συσκευή. Όμως κατέχει το δικό της κομμάτι στις επαγγελματικές εφαρμογές και έτσι συναντιέται πολύ συχνά σαν η βασική συνδεσιμότητα στην αεροναυπηγική, στην αυτοκινητοβιομηχανία και σε μερικές εφαρμογές δικτύωσης υπολογιστών.



Σχήμα 40: Τα βύσματα FireWire, 6 και 4 ακίδων αντίστοιχα.

7.1.4. Ethernet

Το Ethernet είναι το συνηθέστερα χρησιμοποιούμενο πρωτόκολλο ενσύρματης τοπικής δικτύωσης υπολογιστών. Αναπτύχθηκε από την εταιρεία Xerox κατά τη δεκαετία του '70 και έγινε δημοφιλές αφότου η Digital Equipment Corporation και η Intel, από κοινού με τη Xerox, προχώρησαν στην προτυποποίησή του το 1980. Το 1985 το Ethernet έγινε αποδεκτό επίσημα από τον οργανισμό IEEE ως το πρότυπο 802.3 για ενσύρματα τοπικά δίκτυα (LAN-Local Area Networks).

Το αρχικό Ethernet επέτρεπε ονομαστικούς ρυθμούς μετάδοσης δεδομένων της τάξης των 3 Mbps (Mega bit per second), μέσω ενός ομοαξονικού καλωδίου στο οποίο συνδέονταν οι επιμέρους υπολογιστές του δικτύου (σύνδεση Token Ring). Τη διασύνδεση αναλάμβανε μία κάρτα δικτύου Ethernet προσαρτημένη σε κάθε κόμβο, με κάθε κάρτα να χαρακτηρίζεται από μία μοναδική, εργοστασιακή 48-bit διεύθυνση MAC (Media Access Control). Σήμερα η σύνδεση Token Ring έχει εγκαταλειφθεί ολοκληρωτικά και οι επιμέρους υπολογιστές του δικτύου συνδέονται ο καθένας σε ανεξάρτητη θύρα ενός Router ή διανομέα (Hub). Έχουν εμφανιστεί νεότερες εκδόσεις του Ethernet οι οποίες χρησιμοποιούν είτε κοινά καλώδια χαλκού με αθωράκιστα (καλώδια UTP – Unshielded Twisted Pair) ή θωρακισμένα (καλώδια STP – Shielded Twisted Pair) συνεστραμμένα ζεύγη αγωγών ή οπτικές ίνες. Αναφορικά, οι διάφορες εκδόσεις Ethernet είναι:

- **Ethernet (10Mbps)**, όπου για τις συνδέσεις με χαλκό χρησιμοποιείται το πρότυπο 10BASE - T και για τις οπτικές ίνες το πρότυπο 10BASE-F (L). Η σύνδεση χαλκού είναι συμβατή με αυτή του Fast Ethernet.
- **Fast Ethernet (100 Mbps)**, όπου για τις συνδέσεις με χαλκό έχει επικρατήσει το πρότυπο 100BASE - TX έναντι των ουσιαστικά εγκαταλελειμμένων 100BASE-T2, 100BASE-T4. Το 100BASE-TX χρησιμοποιεί καλώδια UTP κατηγορίας 5e (CAT-5e) με 2 ζεύγη αγωγών (ένα για αποστολή και ένα για λήψη δεδομένων), σε μήκη μέχρι 100 μέτρα. Πρακτικά, χρησιμοποιούνται καλώδια 4 ζευγών, ώστε να είναι δυνατή η σύνδεση ή η αναβάθμιση σε Gigabit Ethernet (1000BASE-TX). Το αντίστοιχο πρότυπο για τις οπτικές ίνες είναι το 100BASE-FX. Επιπλέον, είναι δυνατή η αυτόματη ανίχνευση κυκλώματος 10BASE-T στην άλλη πλευρά του καλωδίου και η εν συνεχεία υποβάθμιση της ταχύτητας στα 10Mbps (λειτουργία auto-negotiation).

- **Gigabit Ethernet (1 Gbps)**, όπου για τις συνδέσεις με χαλκό έχει επικρατήσει το πρότυπο 1000BASE-T. Το 1000BASE-T χρησιμοποιεί καλώδια UTP κατηγορίας 5e (CAT-5e) με 4 ζεύγη αγωγών. Κάθε ζεύγος μεταφέρει δεδομένα προς τις δύο κατευθύνσεις ταυτόχρονα, ώστε να επιτυγχάνεται η μέγιστη δυνατή ταχύτητα μετάδοσης δεδομένων προς κάθε κατεύθυνση. Ο τρόπος σύνδεσης των ζευγών είναι τέτοιος που επιτρέπει σε μια κάρτα Gigabit Ethernet να μπορεί να ανιχνεύσει την ύπαρξη κυκλώματος Fast Ethernet στην άλλη άκρη του καλωδίου και να αλλάξει αυτόματα το πρωτόκολλό της σε 100BASE-TX (λειτουργία auto-negotiation). Το αντίστοιχο πρότυπο για τις οπτικές ίνες είναι τα 1000BASE-FX.
- **10 Gigabit Ethernet (10Gbps)**.



Σχήμα 41: Το πρότυπο 8P8C (CAT -5e), συχνά αναφερόμενο και σαν RJ 45.

7.2. Η Παράλληλη Θύρα Επικοινωνίας (Parallel Port – IEEE 1284)

Η παράλληλη θύρα επικοινωνίας εμφανίστηκε το 1970 και αποτέλεσε την πρώτη θύρα εκτυπωτή. Σχεδιάστηκε από τους Robert Howard και τον Prentice Robinson της εταιρείας Centronics. Γρήγορα αποτέλεσε και βιομηχανικό πρότυπο, πέρα της οικιακής χρήσης και πολλές εταιρείες με βάση την αρχική ιδέα, παρουσίασαν και τα δικά τους πρότυπα με διαφορετικό αριθμό ακίδων πάνω στη θύρα. Αναφορικά, η VAX χρησιμοποιούσε βύσμα 36 ακίδων (Σχήμα 42), η Texas βύσμα 25 ακίδων και η Data General βύσμα 50 ακίδων. Η IBM παρουσίασε το 1981 τον προσωπικό υπολογιστή της και περιελάμβανε την παράλληλη θύρα όπου μόνο εκτυπωτές με το λογότυπο της IBM μπορούσαν να χρησιμοποιηθούν από τον προσωπικό υπολογιστή της. Η IBM τυποποίησε την παράλληλη θύρα από τη μεριά του υπολογιστή με το βύσμα DB25F (Σχήμα 43) και από τη μεριά του εκτυπωτή με το βύσμα της Centronics. Οι κατασκευάστριες εταιρείες εκτυπωτών υιοθέτησαν αυτήν την τυποποίηση της IBM και έτσι επικράτησε αυτός ο τρόπος συνδεσιμότητας στην αγορά. Πριν την έλευση της USB θύρας, μεγάλος αριθμός περιφερειακών συσκευών χρησιμοποιούσε την παράλληλη θύρα.



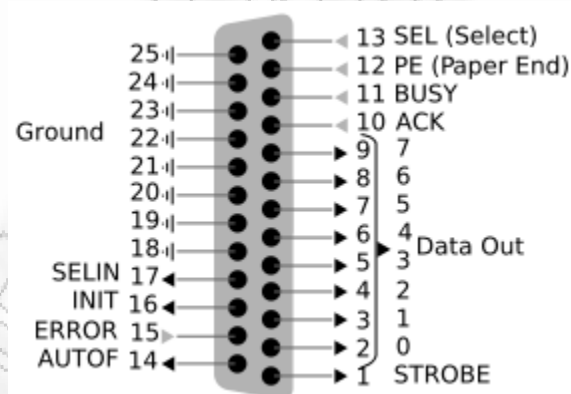
Σχήμα 42: Βύσμα 36 ακίδων (θηλυκό) που χρησιμοποιείτο για βιομηχανική χρήση και στους προσωπικούς υπολογιστές πριν του 1980.

Στην παράλληλη θύρα, η μεταφορά των δεδομένων γίνεται παράλληλα (οχτώ bit ταυτόχρονα), πράγμα που σημαίνει ότι επιτυγχάνονται μεγαλύτερες ταχύτητες σε σχέση με την κλασική σειριακή θύρα επικοινωνίας, όπου η μεταφορά των δεδομένων γίνεται σε σειρά (ένα bit κάθε φορά). Οι 8 γραμμές εξόδου (ακροδέκτες 2 έως 9 – Σχήμα 44) παρέχουν τάση από 2,4 έως 5,5 Volt για το λογικό “1” και από -0,5 έως 0,4 για το λογικό “0”.

Πλέον, η παράλληλη θύρα έχει αντικατασταθεί πλήρως από τα υπόλοιπα σειριακά πρότυπα.



Σχήμα 43: Η παράλληλη θύρα που υιοθετήθηκε από την IBM και επικράτησε στους προσωπικούς υπολογιστές μετά το 1980.



Σχήμα 44: Οι ακροδέκτες της παράλληλης θύρας

Για τον έλεγχο της παράλληλης θύρας από την Visual Basic .NET χρησιμοποιείται το αρχείο βιβλιοθήκης `input32.dll`, το οποίο πρέπει να βρίσκεται αποθηκευμένο στους φακέλους `system32` και `system` των Windows.

8. Περιγραφή της Μεταπτυχιακής Διατριβής

Ο σκοπός της παρούσας μεταπτυχιακής διατριβής είναι να παρουσιάσει τη συνεργασία που μπορεί να επιτευχθεί ανάμεσα στο λογισμικό και το υλικό του υπολογιστή και αυτό παρουσιάζεται με τον έλεγχο ενός ρομποτικού βραχίονα μέσα από τον υπολογιστή. Ο έλεγχος αυτός μπορεί να επιτυγχάνεται είτε **απευθείας**, καθώς ο ρομποτικός βραχίονας είναι συνδεδεμένος στον υπολογιστή και ο χρήστης πληκτρολογεί απευθείας τις εντολές στον υπολογιστή αυτόν, είτε **εξ' αποστάσεως**, καθώς ο χρήστης πληκτρολογεί τις εντολές ελέγχου του ρομποτικού βραχίονα από έναν απομακρυσμένο υπολογιστή από το ρομποτικό βραχίονα. Ο εξ' αποστάσεως έλεγχος μπορεί να γίνεται είτε εντός ενός τοπικού δικτύου, είτε ενός δικτύου που τα όρια του είναι κατά πολύ μεγαλύτερα από αυτά του τοπικού δικτύου (με τη χρήση του Διαδικτύου - Internet).

Αναφορικά με το κομμάτι **λογισμικού (software)**, ο βασικός αλγόριθμος ελέγχου του ρομποτικού βραχίονα (κλάση ελέγχου ρομποτικού βραχίονα – class RobotControl) έχει σχεδιασθεί και γραφτεί από το γράφοντα, ενώ κάποιες προγραμματιστικές τεχνικές όπως οι τεχνικές Server – Client και οι τεχνικές χειρισμού threads, έχουν χρησιμοποιηθεί στην πρωτότυπη τους μορφή όπως προτείνονται να διατυπώνονται σαν έτοιμοι αλγόριθμοι. Βέβαια έχουν γίνει οι όποιες απαραίτητες παρεμβάσεις για την εύρυθμη λειτουργία του κώδικα. Η γλώσσα προγραμματισμού που έχει χρησιμοποιηθεί είναι η Visual Basic .NET 2008.

Στο κομμάτι των **υλικών** που χρησιμοποιήθηκαν (**hardware**), καταρχάς ο ρομποτικός βραχίονας (Σχήμα 45), είναι ένας ρομποτικός βραχίονας - παιχνίδι του εμπορίου. Χρησιμοποιεί για τον έλεγχο του ένα ενσύρματο κοντρόλ ελέγχου και αποτελείται από πέντε συνεχούς ρεύματος κινητήρες (Direct Current Motor–DC Motor), οι οποίοι δίνουν πέντε βαθμούς ελευθερίας στο ρομποτικό βραχίονα. Το ενσύρματο χειριστήριο ελέγχου έχει αφαιρεθεί και πλέον ο έλεγχος του ρομποτικού βραχίονα γίνεται από μία ενδιάμεση πλακέτα 8 ρελέ η οποία συνδέεται στην παράλληλη θύρα του υπολογιστή. Στις παραγράφους που ακολουθούν γίνεται αναλυτική παρουσίαση όλων των μερών και της συνδεσιμότητας, που αποτελούν το όλο εγχείρημα.

8.1. Το Υλικό (Hardware)

8.1.1. Ο Ρομποτικός Βραχίονας

Στην εργασία, όπως έχει ήδη αναφερθεί, χρησιμοποιήθηκε ένας ρομποτικός βραχίονας – παιχνίδι - του εμπορίου, με την επωνυμία Edge robotic arm kit της εταιρίας ThinkGeek, το οποίο αποτελείται από πέντε κινητήρες συνεχούς ρεύματος (Direct Current Motors) και έτσι επιτυγχάνουν πέντε βαθμούς ελευθερίας. Επίσης αποτελείται από ένα ενσύρματο χειριστήριο ελέγχου, το οποίο έχει αφαιρεθεί για τις ανάγκες της εργασίας. Η τροφοδοσία του γίνεται από τέσσερις μπαταρίες μεγέθους 'D' του 1,5 Volt.



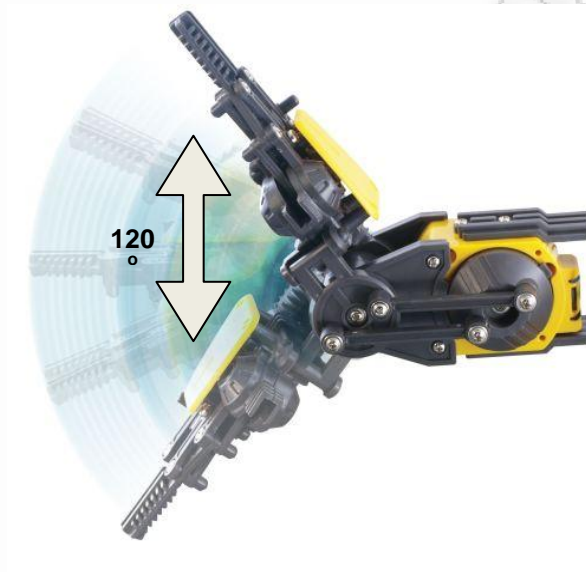
Σχήμα 45: Ρομποτικός βραχίονας

Ακολουθούν τα χαρακτηριστικά λειτουργίας του ρομποτικού βραχίονα.

Ο ρομποτικός βραχίονας που χρησιμοποιείται έχει μία σειρά από ικανότητες και δυνατότητες. Ξεκινώντας θα πρέπει να δοθούν οι διαστάσεις του στο χώρο και οι δυνατότητες κίνησης του σε αυτόν.

Μήκος	22.8 cm
Πλάτος	16 cm
Ύψος	38 cm
Βάρος	658 gr
Μέγιστο Βάρος Ανύψωσης	100 gr

Κίνηση καρπού: 120° (Σχήμα 46)



Σχήμα 46: Κίνηση καρπού ρομποτικού βραχίονα

Κίνηση αγκώνα: 300° (Σχήμα 47)



Σχήμα 47: Κίνηση αγκώνα ρομποτικού βραχίονα

Άνοιγμα αρπάγης: 0 – 4,5 cm (Σχήμα 48)



Σχήμα 48: Άνοιγμα αρπάγης ρομποτικού βραχίονα

Περιστροφική κίνηση βάσης ρομποτικού βραχίονα: 270° (Σχήμα 49)



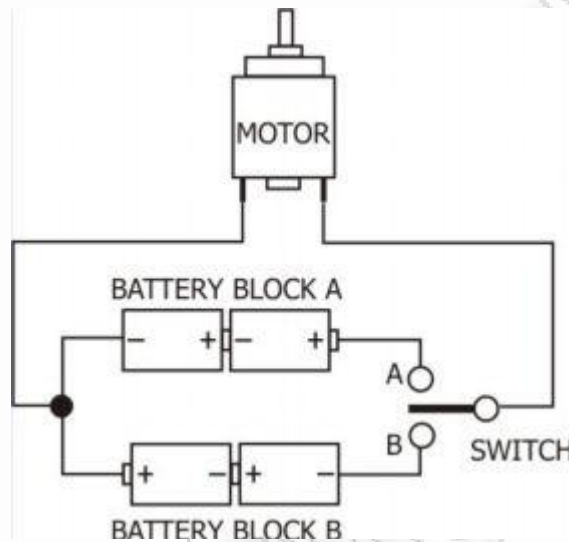
Σχήμα 49: Περιστροφική κίνηση βάσης ρομποτικού βραχίονα

Κίνηση βάσης ρομποτικού βραχίονα: 180° (Σχήμα 50)



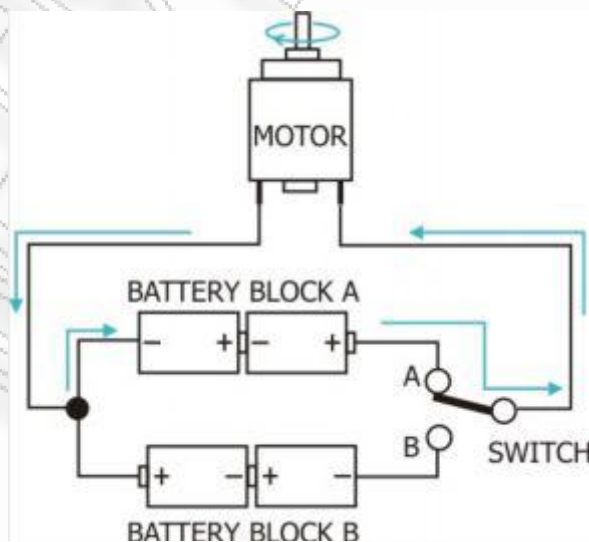
Σχήμα 50: Κίνηση βάσης ρομποτικού βραχίονα

Η ηλεκτρονική καθοδήγηση του ρομποτικού βραχίονα και της κάθε άρθρωσης του γίνεται, όπως έχει ήδη αναφερθεί, από κινητήρες συνεχούς ρεύματος (DC motors). Η περιστροφική κίνηση μπορεί να αλλάξει με αντιστροφή της πολικότητας (+ ή -) στη μπαταρία. Όπως φαίνεται στο παρακάτω σχήμα (Σχήμα 51), ο κινητήρας συνδέεται με δύο συστήματα (block) μπαταριών, στα οποία με τη βοήθεια ενός διακόπτη κλείνει κύκλωμα είτε με το ένα σύστημα (block A), είτε με το άλλο (block B).



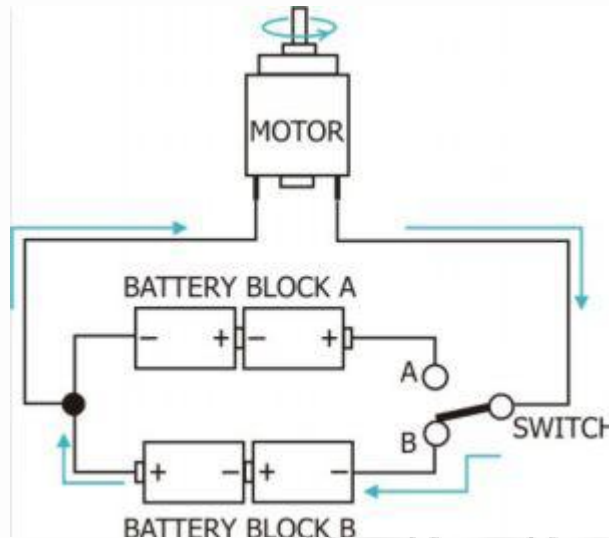
Σχήμα 51: Απεικόνιση σύνδεσης κινητήρα με τα δύο συστήματα μπαταρίας.

Στο σχήμα που ακολουθεί (Σχήμα 52), παρουσιάζεται ο διακόπτης (switch) να βρίσκεται στη θέση A, πράγμα που σημαίνει ότι ο κινητήρας κινείται προς τη μία κατεύθυνση όπου ορίζει το σύστημα μπαταριών A (block A). Επίσης, το ρεύμα διοχετεύεται από το θετικό πόλο της μπαταρίας του συστήματος A και καταλήγει στην άλλη μπαταρία του ίδιου συστήματος, αλλά στον αρνητικό πόλο.



Σχήμα 52: Απεικόνιση σύνδεσης κινητήρα με το block μπαταρίας A.

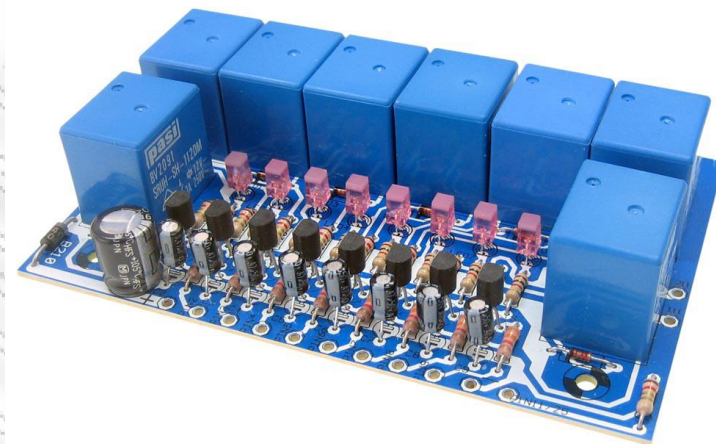
Αντίστοιχα, αν ο διακόπτης πάρει θέση στο σημείο B (Σχήμα 53), κάνει τον κινητήρα να είναι συνδεδεμένος με το σύστημα μπαταριών B (block B). Έτσι η κίνηση του κινητήρα αντιστρέφεται, δηλαδή ο άξονας εκτελεί κίνηση προς την αντίθετη κατεύθυνση.



Σχήμα 53: Απεικόνιση σύνδεσης κινητήρα με το block μπαταρίας B.

8.1.2. Πλακέτα 8 Ρελέ (KEMO B210 kit)

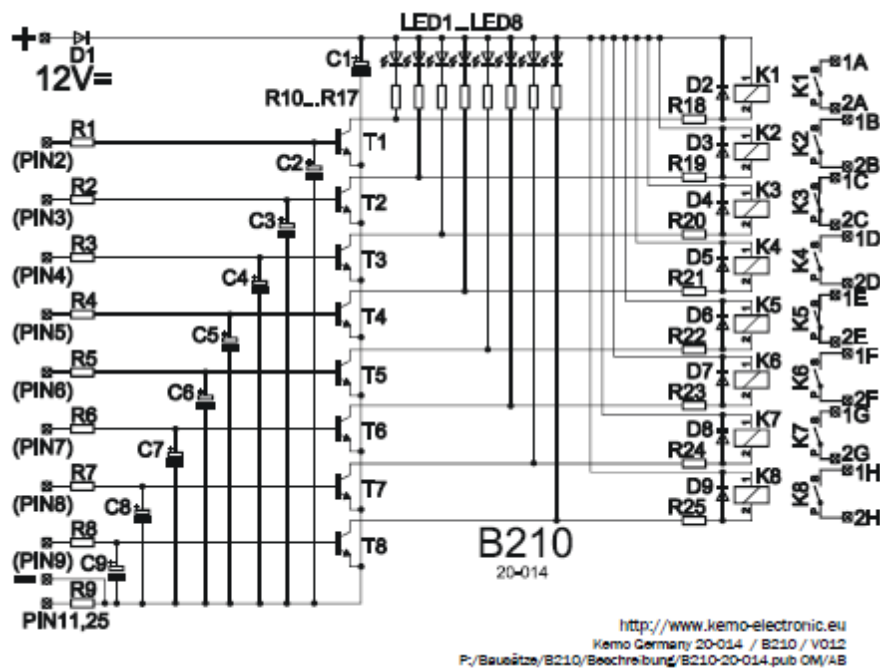
Η εν λόγω πλακέτα (Σχήμα 54), χρησιμοποιείται για να ελέγχει τα ρεύματα που παράγονται από την παράλληλη θύρα του υπολογιστή και να διανέμει το κάθε ρεύμα στον κινητήρα που έχει διαταχθεί από το χρήστη να κινηθεί. Αυτό επιτυγχάνεται με τη βοήθεια του ολοκληρωμένου που υπάρχει και των οχτώ ρελέ τα οποία παίζουν το ρόλο γέφυρας. Δηλαδή, όταν το ρελέ βρίσκεται σε κατάσταση “κανονικής λειτουργίας” (Normally Close) τότε δημιουργείται “κύκλωμα” και εκτελείται η κίνηση. Αντίστροφα, όσα ρελέ δεν δραστηριοποιούνται, βρίσκονται σε κατάσταση “αδράνειας” (Normally Open) και δεν δημιουργείται αγωγίμο κύκλωμα ρεύματος για να κινηθούν οι άλλοι κινητήρες. Για να φαίνεται στην πλακέτα, το ποιος κινητήρας κινείται (και οι αγωγίμες διαδρομές που σχηματίζονται) υπάρχουν ενδεικτικά LED, τα οποία είτε είναι αναμμένα είτε σβηστά για όσο διάστημα διαρκεί κάθε κίνηση κάποιου κινητήρα.



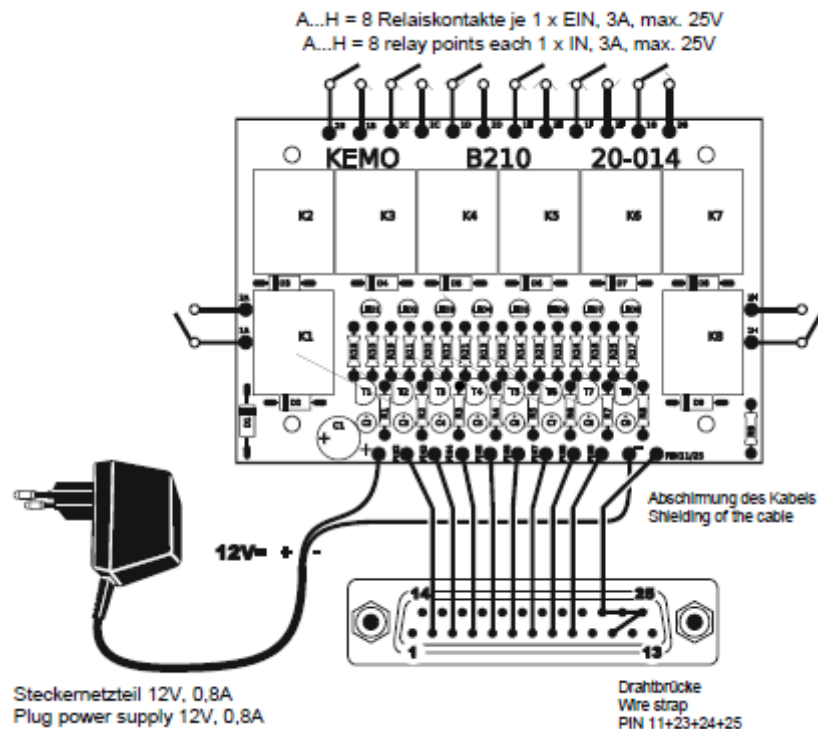
Σχήμα 54: Πλακέτα 8 ρελέ, KEMO B210 kit

Ακολουθεί πίνακας με τα χαρακτηριστικά λειτουργίας της πλακέτας.

Τεχνικά Χαρακτηριστικά	
Τάση Λειτουργίας	12 V / DC max 0.8A
Κανάλια	8
Ρελέ	1 x ON max 3 A, max 25 V το καθένα
Ενδείξεις	Κάθε κανάλι από ένα Led
Διαστάσεις Πλακέτας	108 x 71 mm



Σχήμα 54α: Το ηλεκτρικό κύκλωμα της πλακέτας



Σχήμα 54β: Η συνδεσμολογία της πλακέτας με τις ακίδες του βύσματος της παράλληλης θύρας.

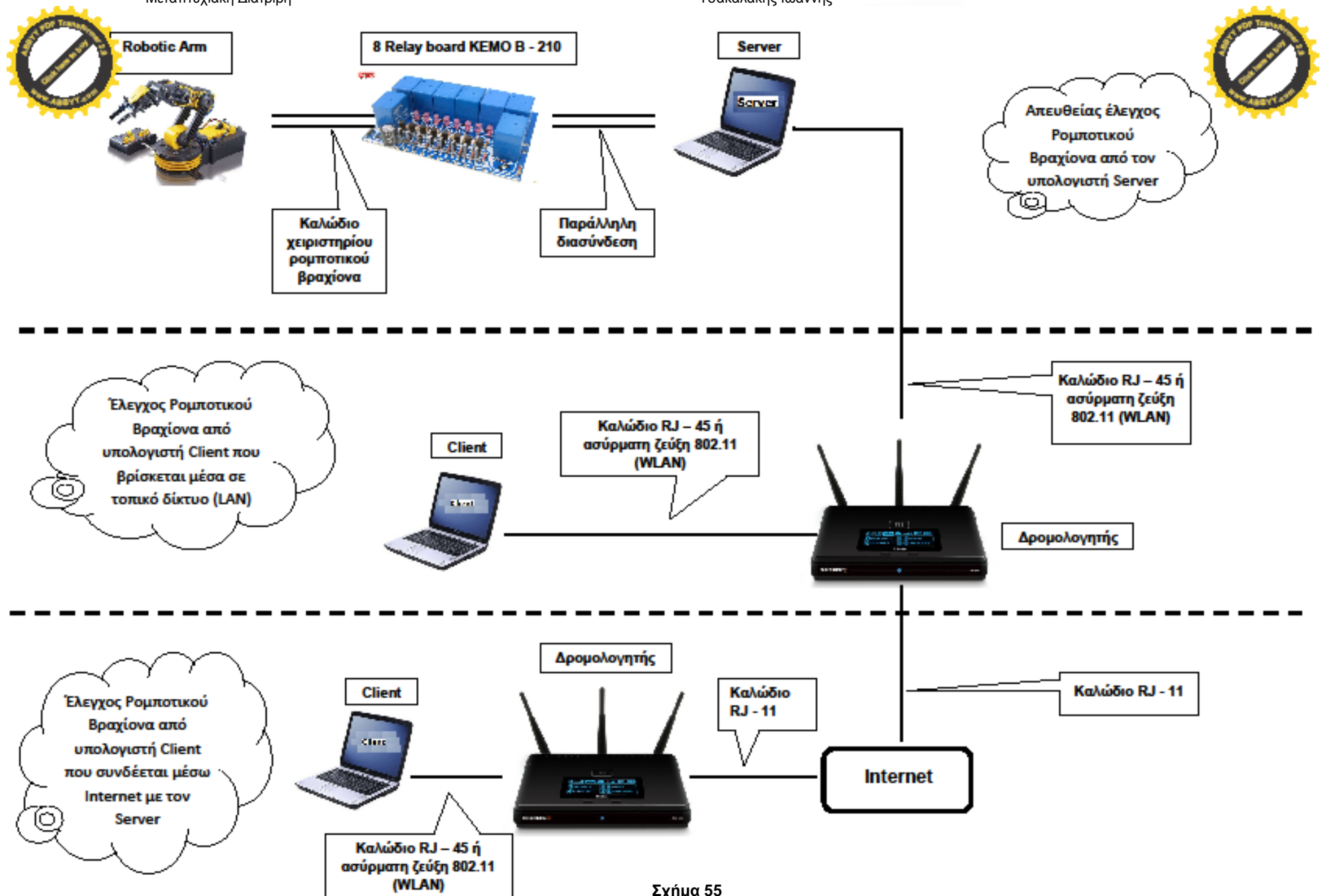
Η πλακέτα συνδέεται με τις ακίδες 2 έως 9 της παράλληλης θύρας. Το λογισμικό, θέτει τις ακίδες είτε σε κατάσταση “1” είτε σε κατάσταση “0”. Αν η αναφερόμενη ακίδα είναι σε κατάσταση “1” τότε και το ανάλογο κανάλι λειτουργεί. Αντίστοιχα αν κάποια ακίδα είναι σε κατάσταση “0” τότε δεν λειτουργεί κάποιο κανάλι. Η ακίδα 2 αντιπροσωπεύει το κανάλι 1, η ακίδα 3 το κανάλι 2 κ.ο.κ..

8.1.3. Η Διασύνδεση του Υλικού

Στην παρούσα ενότητα θα παρουσιαστεί ο τρόπος διασύνδεσης όλου του υλικού, ώστε να επιτευχθεί το τελικό αποτέλεσμα. Στο ακόλουθο σχήμα (Σχήμα 55) υπάρχει μία τέτοια απεικόνιση που παρουσιάζει και τα επίπεδα ελέγχου που παρέχονται από την εν λόγω εργασία. Πρέπει να γίνει αναφορά στο γεγονός ότι η όλη ιδέα της διασύνδεσης του υλικού είναι του γράφοντα. Στο σχήμα παρατηρείται ότι υπάρχει για τον έλεγχο του ρομποτικού βραχίονα, τόσο πρόβλεψη για απευθείας έλεγχο από το χρήστη - χειριστή, πληκτρολογώντας τις εντολές στο Server υπολογιστή ελέγχου του ρομποτικού βραχίονα, τόσο και για εξ' αποστάσεως έλεγχο του ρομποτικού βραχίονα όπου ο χρήστης – χειριστής ελέγχει το ρομποτικό βραχίονα από έναν απομακρυσμένο Client υπολογιστή, που βρίσκεται είτε στα όρια ενός τοπικού δικτύου είτε από μεγαλύτερη απόσταση, όπου ο έλεγχος γίνεται μέσω του διαδικτύου.

Επίσης πρέπει να γίνει αναφορά στο τρόπο διασύνδεσης του ρομποτικού βραχίονα με την πλακέτα με τα οχτώ ρελέ (8 Relay Board). Το εργοστασιακό χειριστήριο ελέγχου του ρομποτικού βραχίονα, έχει απομακρυνθεί και δεν χρησιμοποιείται στην παρούσα εργασία (καθώς ο έλεγχος γίνεται από τον υπολογιστή). Το καλώδιο σύνδεσης του χειριστηρίου με το ρομποτικό βραχίονα, δεν απομακρύνθηκε και χρησιμοποιείται για τη διασύνδεση του ρομποτικού βραχίονα με την πλακέτα των οχτώ ρελέ.

Η πλακέτα των οχτώ ρελέ διαθέτει εργοστασιακή διασύνδεση με τον υπολογιστή μέσω παράλληλης καλωδίωσης.



Σχήμα 55

8.2. Το Λογισμικό (Software)

Το λογισμικό της παρούσας εργασίας έχει γραφεί σε VB 2008 .NET. Η VB 2008 .NET είναι προϊόν της γνωστής εταιρείας λογισμικού Microsoft.

Η καρδιά του όλου εγχειρήματος είναι η κλάση χειρισμού του ρομποτικού βραχίονα. Σε αυτήν παρουσιάζονται, όλες οι δυνατές κινήσεις που μπορούν να εκτελεστούν από το ρομποτικό βραχίονα και επίσης για κάθε κίνηση δημιουργείται και ένα νέο thread, όπου προστατεύει από τυχόν “κολλήματα” του συστήματος.

Όλες οι δυνατές κινήσεις είναι μεταφρασμένες σε αυτή την κλάση σε δυαδικούς αριθμούς (στην κλάση αναφέρονται με τη δεκαδική τους αναπαράσταση, όπως για παράδειγμα η τιμή 130 είναι η κίνηση “OPEN GRIPPER” – “Ανοίγμα Αρπάγης”). Ακολουθεί πίνακας που δείχνει όλες αυτές τις δυνατές κινήσεις (set εντολών) και την αντιστοιχία σε δυαδική και δεκαδική αναπαράσταση

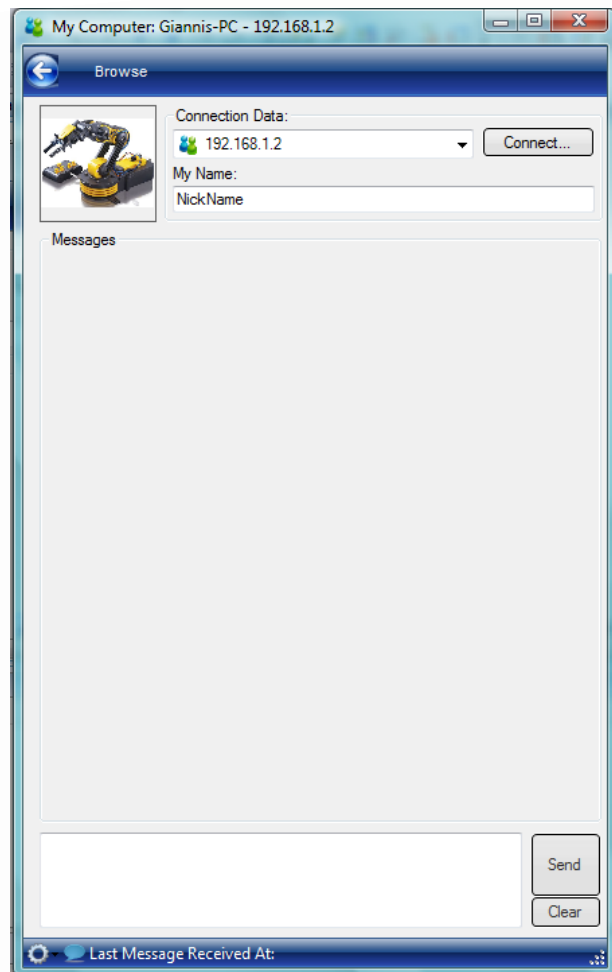
Κίνηση	Εντολή	Δυαδική Αναπαράσταση	Δεκαδική Αναπαράσταση
Ανοίγμα Αρπάγης	OPEN GRIPPER	10000010	130
Κλείσιμο Αρπάγης	CLOSE GRIPPER	00000011	3
Κίνηση Βάσης Αριστερά για	MOVE BASE LEFT FOR	00100001	33
Κίνηση Βάσης Δεξιά για	MOVE BASE RIGHT FOR	10100000	160
Κίνηση Αγκώνα Πάνω για	MOVE ELBOW UP FOR	10111110	136
Κίνηση Αγκώνα Κάτω για	MOVE ELBOW DOWN FOR	00001001	9
Κίνηση Ώμου Πάνω για	MOVE SHOULDER UP FOR	10010000	144
Κίνηση Ώμου Κάτω για	MOVE SHOULDER DOWN FOR	00010001	17
Κίνηση Καρπού Πάνω για	MOVE WRISTL UP FOR	10000100	132
Κίνηση Καρπού Κάτω για	MOVE WRISTL DOWN FOR	00000101	5

Η κάθε εντολή, όπως φαίνεται στην κλάση, είναι ατελής και συμπληρώνεται καθώς ο χρήστης δίνει το χρόνο κίνησης του αντίστοιχου σημείου του βραχίονα (του αντίστοιχου κινητήρα άρθρωσης) που επιθυμεί να κινηθεί. Όπως παρατηρείται, η μόνη εντολή που δεν ακολουθείται από χρονικό προσδιορισμό, είναι η κίνηση ανοίγματος και κλεισίματος της αρπάγης. Αυτή έχει οριστεί μέσα από το πρόγραμμα να είναι στο ένα δευτερόλεπτο (1000 msec).

Η λέξη που σχηματίζεται στη δυαδική της απεικόνιση, φαίνεται στην πλακέτα με τα οχτώ ρελέ (ανάβουν τα led για όσο χρόνο διαρκεί η κίνηση) και η πλακέτα στέλνει ρεύμα στον αντίστοιχο κινητήρα που διατάζεται να κινηθεί.

Στο κεφάλαιο “Παράρτημα” παρουσιάζεται όλη η κλάση κώδικα, η οποία είναι πρωτότυπη και έχει σχεδιαστεί από το γράφοντα. Κομμάτι του λογισμικού είναι επίσης ο προγραμματισμός της φόρμας (Form). Σε αυτό το κομμάτι προγραμματίζονται όλες οι ιδιότητες και ικανότητες που φέρει η φόρμα, καθώς επίσης γίνονται και όλες οι δηλώσεις των αντικειμένων (objects) που χρησιμοποιούνται. Για την επίτευξη του, επίσης χρησιμοποιήθηκαν προγραμματιστικές τεχνικές όπως χειρισμός Threads και τεχνικές Server – Client, όπου ο τρόπος δόμησης τους έχει παραδειγματιστεί από πληθώρα έτοιμων εφαρμογών που υπάρχουν στο διαδίκτυο, αλλά έχει προσαρμοστεί κατάλληλα στις ανάγκες της εργασίας από το γράφοντα. Στο κεφάλαιο “Παράρτημα” παρουσιάζεται όλος ο κώδικας της φόρμας.

Στο σχήμα 56, φαίνεται ο σχεδιασμός της φόρμας, ο οποίος είναι δημιούργημα του γράφοντα και εμπνευσμένος από ανάλογες εφαρμογές του εμπορίου. Πρέπει να τονιστεί ότι έγινε προσπάθεια να υπάρχει στη φόρμα μια διαδραστικότητα και να της δοθεί μία ευχέρεια χειρισμού, ακόμα και από άτομα που δεν έχουν εξειδικευμένες τεχνολογικές σπουδές. Γι’ αυτό το λόγο η φόρμα είναι σχεδιασμένη στα πρότυπα “Δωματίου Συνομιλίας” (Chat Room).

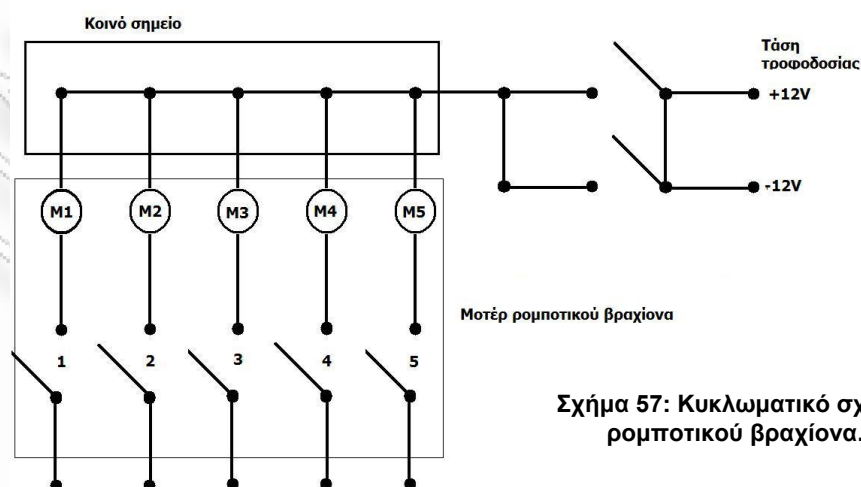


Σχήμα 56: Φόρμα εντολών ρομποτικού βραχίονα.

8.3. Λειτουργία

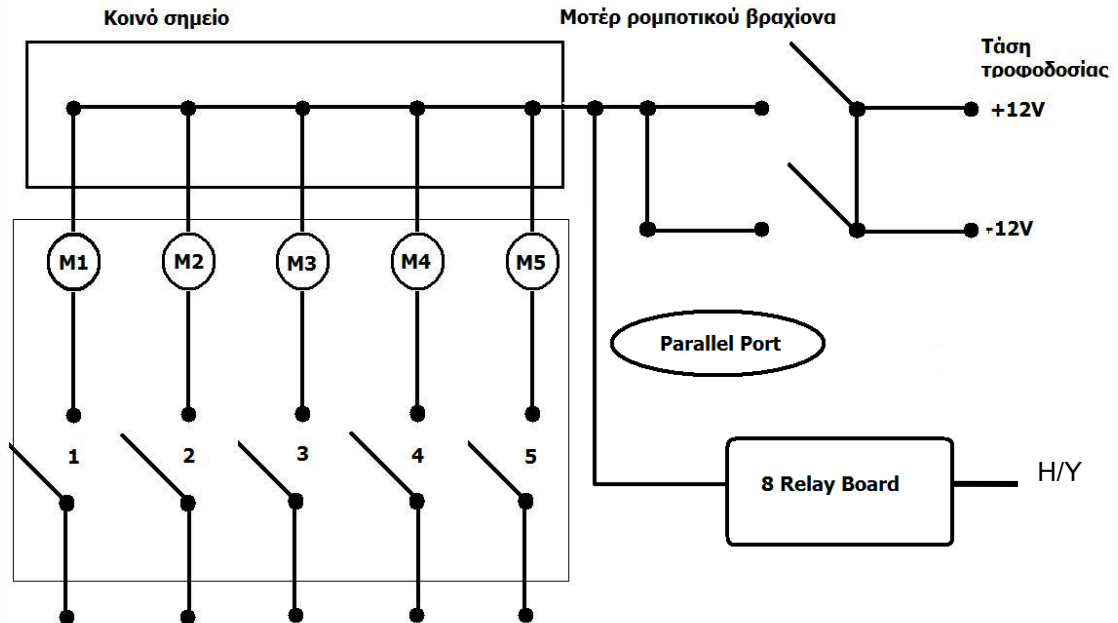
Στο παρόν κεφάλαιο θα δοθεί ένα πλήρες παράδειγμα της όλης λειτουργίας της παρούσας εργασίας. Θα προηγηθεί μία θεωρητική προσέγγιση του εγχειρήματος και του τρόπου όπου ενεργοποιείται ο κάθε κινητήρας για να εκτελεστεί μία κίνηση.

Το ρομποτικό βραχίονα, αν θελήσουμε να αναπαραστήσουμε το κυκλωματικό του σχέδιο θα παρουσιαζόταν σαν του παρακάτω σχήματος (Σχήμα 57).



Σχήμα 57: Κυκλωματικό σχέδιο ρομποτικού βραχίονα.

Εν συνεχεία παρουσιάζεται η σύνδεση του κυκλώματος του βραχίονα με την πλακέτα των οχτώ ρελέ (Σχήμα 58). Η πλακέτα των οχτώ ρελέ, μόλις σχηματίσει τη λέξη του κινητήρα που καλείται να κινηθεί, τη μεταδίδει στο ρομποτικό βραχίονα. Εκείνος, εκτός του ότι ενεργοποιεί τον αντίστοιχο κινητήρα, αν “αντιληφθεί” δεξιόστροφη κίνηση παίρνει τροφοδοσία από το σύστημα μπαταριών που επιτελεί τη δεξιόστροφη κίνηση και το αντίστοιχο γίνεται αν χρειαστεί να επιτελέσει αριστερόστροφη κίνηση (σχετικά κεφάλαιο 9.1.1.).

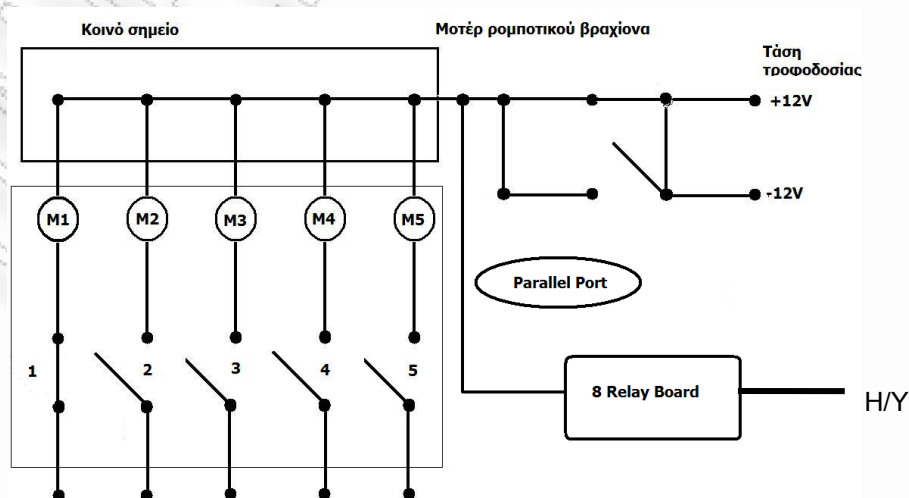


Σχήμα 58: Κυκλωματικό σχέδιο ρομποτικού βραχίονα σε σύνδεση με την πλακέτα των 8 ρελέ

Πρέπει να σημειωθεί ότι, ανά πάσα στιγμή μόνο ένας κινητήρας κίνησης λειτουργεί, οπότε βραχυκυκλώνεται η αντίστοιχη γέφυρα του κινητήρα που λειτουργεί και όλες οι άλλες γέφυρες είναι ανοιχτοκυκλωμένες.

Η διασύνδεση κυκλωματικού σχεδίου ρομποτικού βραχίονα με την πλακέτα οχτώ ρελέ, αποτελεί πρωτότυπη ιδέα και έχει υλοποιηθεί από το γράφοντα.

Αν θελήσουμε να δώσουμε ένα πρώτο απλό παράδειγμα του τρόπου λειτουργίας της πλακέτας θα ήταν το εξής: Αν, για παράδειγμα, δοθεί η εντολή “MOVE BASE LEFT FOR 3”, οι κυκλωματικές αλλαγές σε σχέση με το αδρανές κύκλωμα του σχήματος 58 παρουσιάζονται στο σχήμα 59.



Σχήμα 59: Παράδειγμα εκτέλεσης της εντολής “MOVE BASE LEFT FOR 3”.

Παρατηρείται στο σχήμα, το βραχυκύκλωμα στον αντίστοιχο κινητήρα που κινεί τη βάση, καθώς επίσης παρατηρείται και το βραχυκύκλωμα στην τροφοδοσία που επιτελεί τη δεξιόστροφη κίνηση. Την ίδια στιγμή η πλακέτα με τα οχτώ ρελέ σχηματίζει το δυαδικό αριθμό 00100001 (33 στο δεκαδικό σύστημα) με τα αντίστοιχα led's να μένουν αναμμένα για 3 δευτερόλεπτα.

Στη συνέχεια ακολουθεί ένα πλήρες παράδειγμα λειτουργίας από την πλευρά του χρήστη.

8.3.1. Χρήση με Απευθείας Έλεγχο

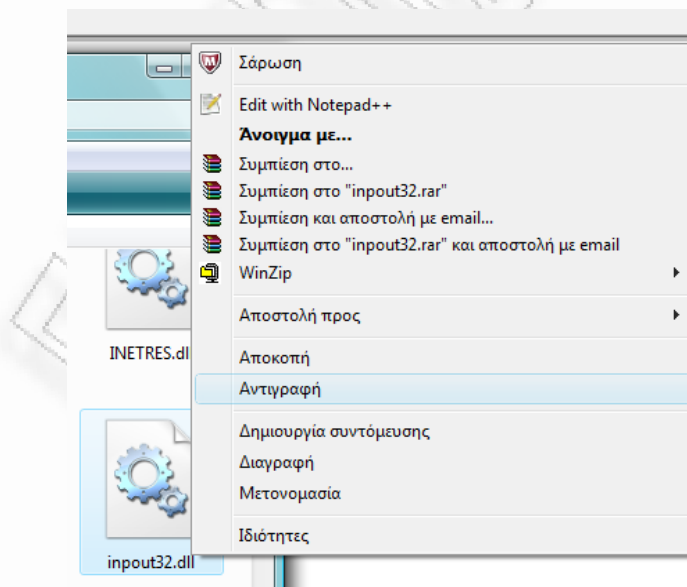
Αρχικά, πριν ξεκινήσει να εκτελείται το πρόγραμμα, πρέπει να γίνει ένας διαχωρισμός σχετικά με τον υπολογιστή όπου εργάζεται ο χειριστής του ρομποτικού βραχίονα. Πιο συγκεκριμένα, αν εργάζεται απ' ευθείας (από το Server υπολογιστή που είναι απευθείας συνδεδεμένος ο ρομποτικός βραχίονας) ή εργάζεται εξ' αποστάσεως (όπου ο υπολογιστής Client συνδέεται στην IP του Server υπολογιστή που είναι συνδεδεμένος ο ρομποτικός βραχίονας και κατόπιν δίνονται οι εντολές χειρισμού του). Σαν πρώτη αναφορά, θα δοθεί ο **απευθείας έλεγχος** του ρομποτικού βραχίονα.

Καταρχάς επιλέγεται με διπλό κλικ ο φάκελος που περιέχει το εκτελέσιμο αρχείο "Robot Arm Project" (Σχήμα 60).

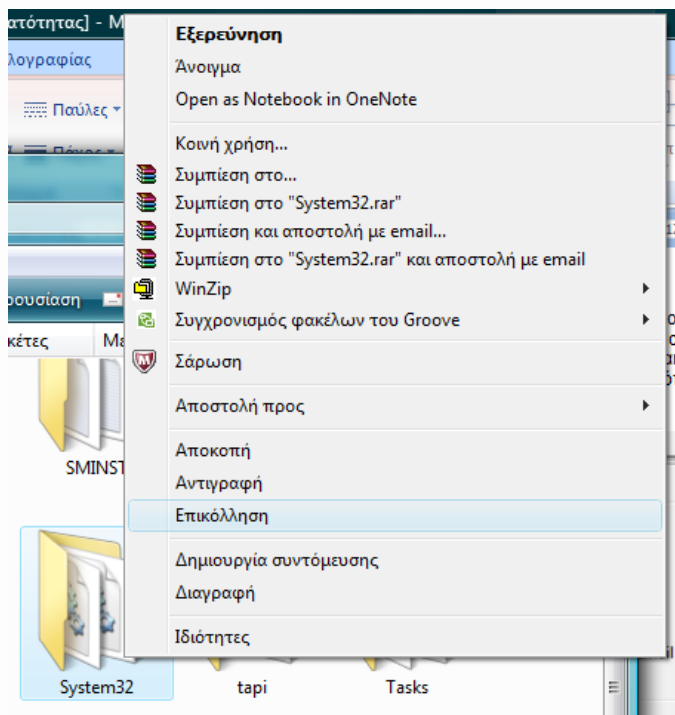


Σχήμα 60: Επιλογή φακέλου

Ο φάκελος περιέχει το αρχείο "Inpout32.dll", όπου είναι αρχείο βιβλιοθήκης χειρισμού της παράλληλης θύρας. Πρέπει να επιλεγθεί και να αντιγραφεί (Σχήμα 61) στους φακέλους Windows→System και Windows→System32 (Σχήμα 62). Το εν λόγω αρχείο διατίθεται στο Internet και απευθύνεται σε όσα λειτουργικά συστήματα **δεν το διαθέτουν**. Τα παλιότερα λειτουργικά συστήματα το διέθεταν.

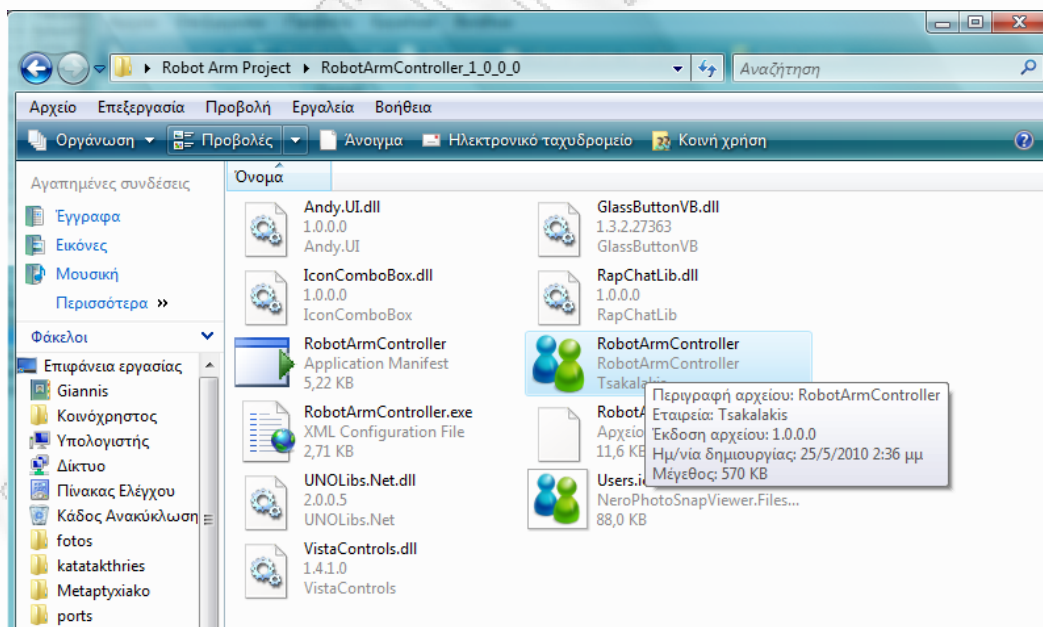


Σχήμα 61: Αντιγραφή αρχείου



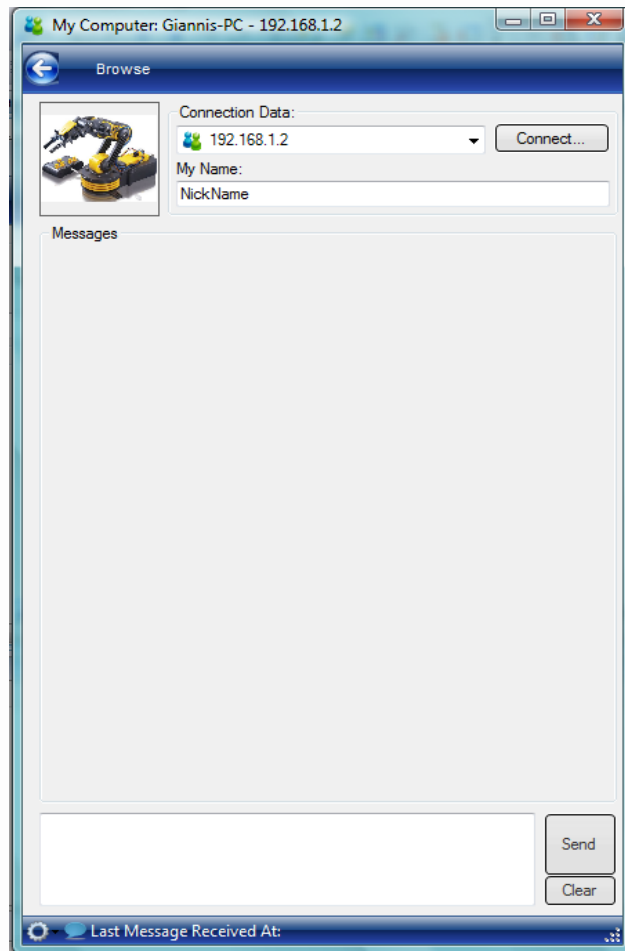
Σχήμα 62: Επικόλληση αρχείου

Αφού έχει εκτελεστεί σωστά το προηγούμενο βήμα, ο χρήστης είναι έτοιμος να εκτελέσει το πρόγραμμα χειρισμού του ρομποτικού βραχίονα, κάνοντας διπλό κλικ στο εκτελέσιμο αρχείο “RobotArmController” (Σχήμα 63).



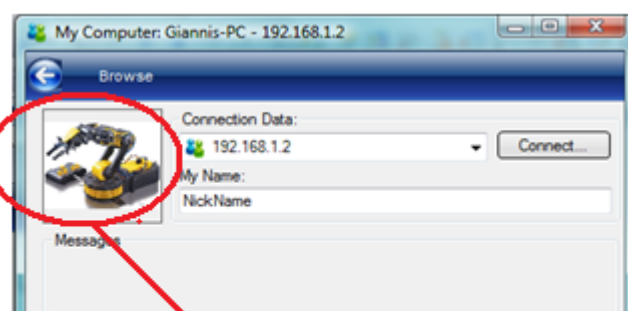
Σχήμα 63: Επιλογή εκτελέσιμου αρχείου αρχείου

Όταν γίνει διπλό κλικ στο εκτελέσιμο αρχείο εμφανίζεται η οθόνη εργασίας (φόρμα – Form) (Σχήμα 64).



Σχήμα 64: Επιφάνεια εργασίας

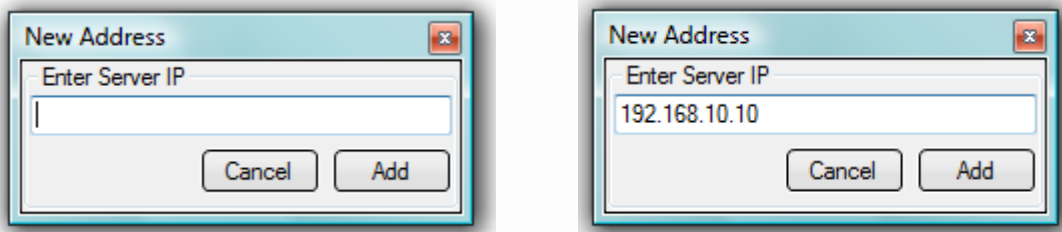
Στην επιφάνεια εργασίας εμφανίζονται διάφορα πεδία τα οποία και επεξηγούνται. Οπτικά το πρώτο που παρατηρείται, είναι το εικονίδιο του ρομποτικού βραχίονα (Σχήμα 65). Επιλέγοντας το και κάνοντας διπλό κλικ εμφανίζεται ένα νέο παράθυρο (Σχήμα 66)



**Εικονίδιο
ρομποτικού
βραχίονα**

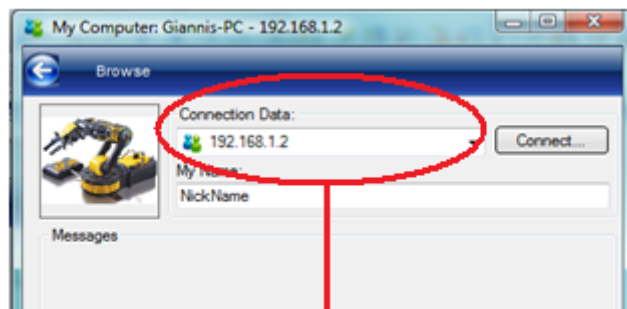
Σχήμα 65: Επιλογή εικονιδίου ρομποτικού βραχίονα

Σε αυτό το πεδίο πληκτρολογείται η IP του Server υπολογιστή όπου είναι συνδεδεμένος ο ρομποτικός βραχίονας και κατόπιν επιλέγεται το κουμπί Add.

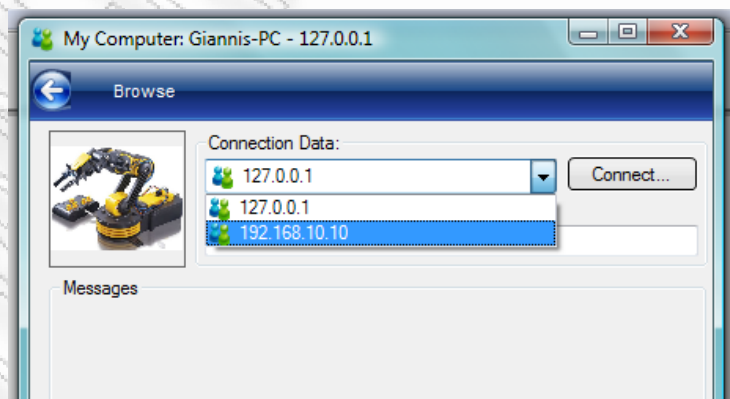


Σχήμα 66: Πληκτρολόγηση IP του Server υπολογιστή

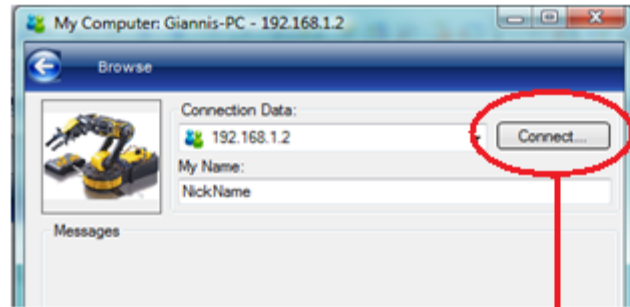
Η IP που πληκτρολογείται εμφανίζεται στο αναδυόμενο μενου ενός άλλου πεδίου. Έστω ότι εργαζόμαστε από Client υπολογιστή και ο Server υπολογιστής διαθέτει IP 192.168.10.10, την πληκτρολογούμε στο πεδίο του σχήματος 66 και εν συνεχεία πατιέται το κουμπί Add. Το μενου "Connection Data" (που φαίνεται στο Σχήμα 67), είναι ένα αναδυόμενο μενου, το οποίο σαν αρχική επιλογή έχει την IP του υπολογιστή που κάθεται ο χρήστης (είτε είναι Client, είτε είναι Server). Η IP που μας ενδιαφέρει βρίσκεται μέσα στις άλλες επιλογές του αναδυόμενου μενου. Απλά επιλέγεται και πατιέται στη συνέχεια το κουμπί "Connect" (Σχήμα 68).



Εμφάνιση των IP



Σχήμα 67: Menu "Connection Data"

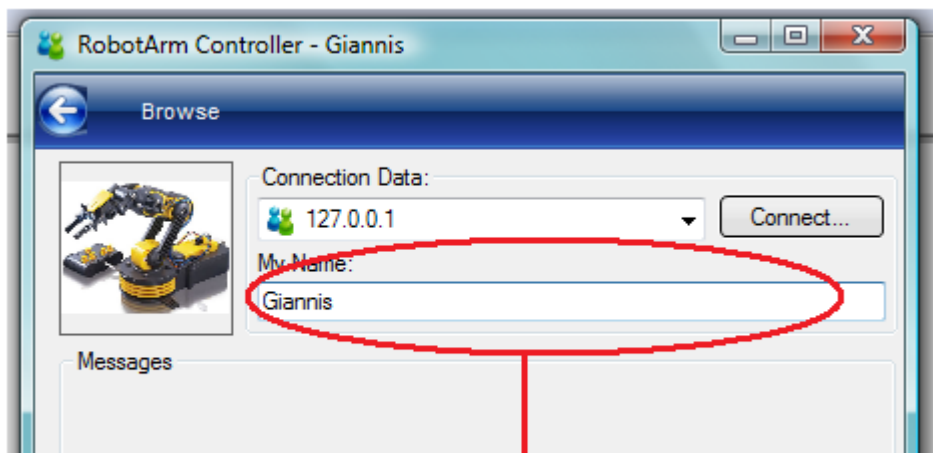


Κουμπι Connect

Σχήμα 68: Κουμπι “Connect”

Στο παρόν κεφάλαιο, όπου ο έλεγχος του ρομποτικού βραχίονα γίνεται απευθείας, δεν μας ενδιαφέρει να συνδεθούμε σε κάποιον απομακρυσμένο υπολογιστή, οπότε μένει ως έχει η IP (δίδεται η localhost) και **δεν επιλέγεται** το κουμπι “Connect”.

Μετά από αυτή τη σύντομη, μα χρήσιμη, αναφορά, σε εμφανές σημείο βρίσκεται και το πεδίο όπου δίδεται το όνομα του χειριστή “My Name” (Σχήμα 69). Αρχική επιλογή του πεδίου είναι “NickName” (ψευδώνυμο), αν δεν πληκτρολογηθεί κάποιο άλλο όνομα. Σε κάθε εντολή που θα πληκτρολογείται στη συνέχεια, αναφέρεται ποιος χειριστής έδωσε μία εντολή ελέγχου, καθώς και την ώρα που πληκτρολόγησε τη συγκεκριμένη εντολή.



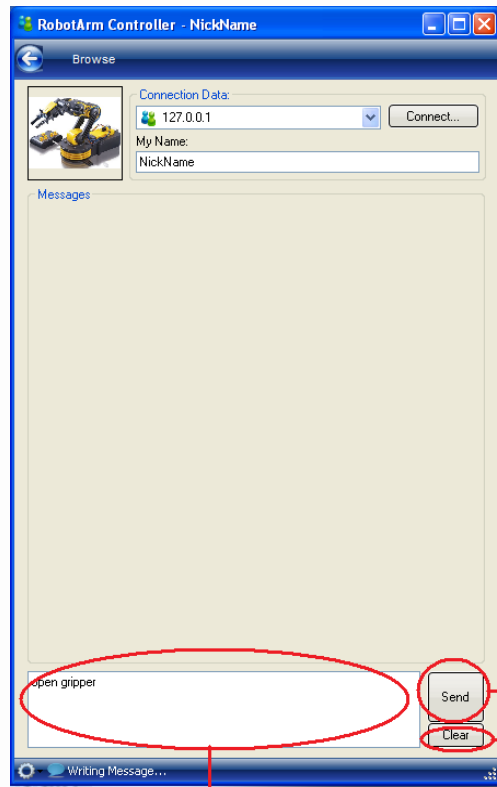
Πεδίο ονομασίας χειριστή

Σχήμα 69: Ονοματοδοσία χειριστή ρομποτικού βραχίονα

Κατόπιν, τα πιο σημαντικά πεδία είναι εκείνα όπου δίνονται (πληκτρολογούνται) οι εντολές και που εμφανίζονται όλα τα μηνύματα εντολών που έχουν δοθεί. Καταρχάς, το πεδίο που δίνονται οι εντολές φαίνεται στο σχήμα 70. Δίπλα από το πεδίο πληκτρολόγησης, υπάρχουν δύο κουμπιά, τα “Send” και “Clear” (Σχήμα 70). Το κουμπι “Send”, προωθεί την εντολή μόλις πληκτρολογηθεί, ώστε να εκτελεστεί. Βέβαια, αντί για το κουμπι “Send”, μόλις διατυπωθεί η εντολή, μπορεί να πατηθεί από το πληκτρολόγιο το πλήκτρο “Enter”. Το κουμπι “Clear” καθαρίζει το πεδίο πληκτρολόγησης.

Μόλις πληκτρολογηθεί μία σωστά διατυπωμένη εντολή στο πεδίο “Messages” (Πεδίο Μηνυμάτων - Σχήμα 71), εμφανίζονται δύο νέες γραμμές οι οποίες αναφέρονται αρχικά και οι δύο, στην ημερομηνία και την ώρα που δέχτηκαν την εντολή. Στην πρώτη γραμμή, στη

συνέχεια, αναφέρεται η εντολή που πληκτρολογήθηκε και στη δεύτερη γραμμή αναφέρεται ότι η εντολή που διατυπώθηκε είναι σωστή συντακτικά (Η αναφορά περί σωστής διατύπωσης της εντολής γίνεται με τη λέξη “Success”). Αν έχει γίνει λανθασμένη σύνταξη εντολής τότε το πεδίο μηνυμάτων εμφανίζεται με κόκκινα υπογραμμισμένα γράμματα και μας πληροφορεί για τη μη σωστή σύνταξη εντολής (Σχήμα 71α). Στα δύο σχήματα που ακολουθούν (Σχήματα 70 και 71), σαν παράδειγμα δίδεται η πληκτρολόγηση της εντολής “OPEN GRIPPER”

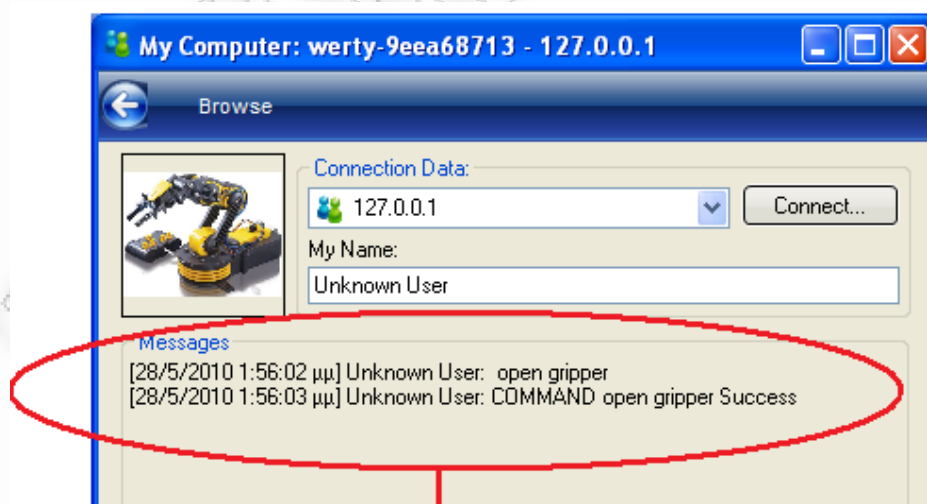


Κουμπι προώθησης εντολών

Κουμπι καθαρισμού πεδίου
πληκτρολόγησης

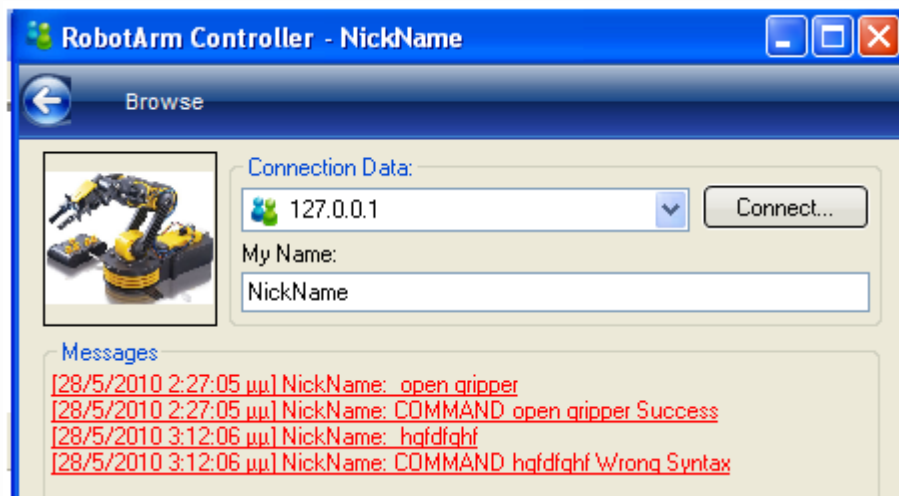
Πεδίο πληκτρολόγησης εντολών

Σχήμα 70: Πεδίο πληκτρολόγησης εντολών και κουμπιά “Send” και “Clear”



Πεδίο μηνυμάτων

Σχήμα 71: Πεδίο μηνυμάτων



Σχήμα 71α: Λάθος σύνταξη εντολής

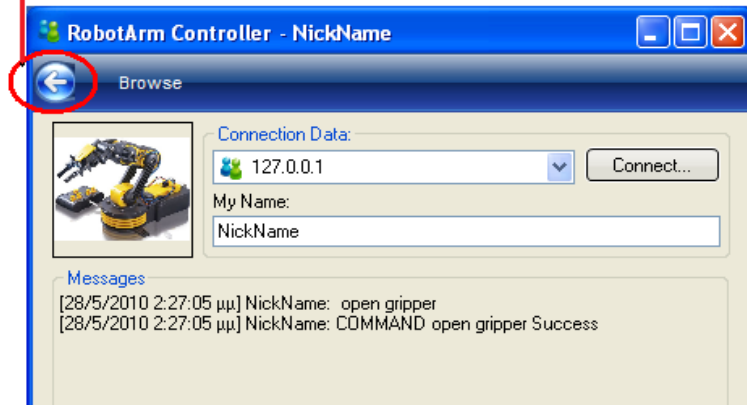
Στη συνέχεια, στο δεξιό μέρος της οθόνης εμφανίζεται ένα απλό οπτικό εφέ το οποίο μας πληροφορεί ότι δόθηκε ένα νέο μήνυμα – εντολή χειρισμού του ρομποτικού βραχίονα (Σχήμα 72). Μετά από διάστημα λίγων δευτερολέπτων εξαφανίζεται.



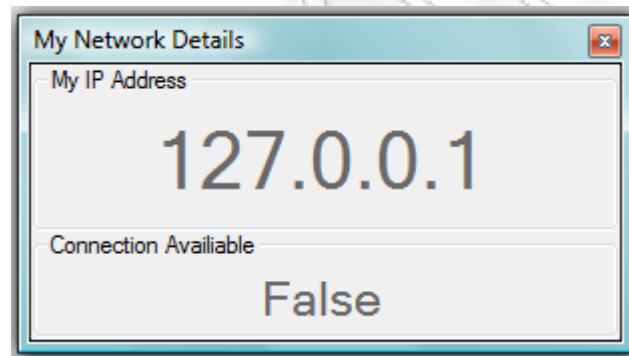
Σχήμα 72: Αναδυόμενο οπτικό εφέ, που πληροφορεί για την εντολή

Επίσης, στην επιφάνεια εργασίας του ρομποτικού βραχίονα, υπάρχει το κουμπί καθαρισμού του πεδίου μηνυμάτων (Σχήμα 73).

Κουμπί καθαρισμού πεδίου μηνυμάτων



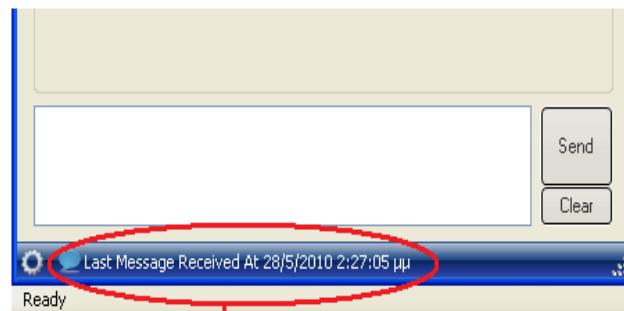
Σχήμα 73



Σχήμα 74: Λεπτομέρειες Δικτύου, οι οποίες εμφανίζονται αφού πατηθεί η επιλογή “Browse”

Αναφορά πρέπει να γίνει και στην επιλογή “Browse” (Σχήμα 73), όπου μόλις επιλεγθεί προβάλλει την εκάστοτε IP του υπολογιστή εργασίας του χρήστη και σαν δεύτερη προβολή εμφανίζει τη διαθεσιμότητα δικτύου (είτε τοπικού είτε Internet σύνδεση). Στην περίπτωση αυτή, όπου ο χειρισμός του ρομποτικού βραχίονα γίνεται από το Server υπολογιστή, σαν IP διεύθυνση θα εμφανίζεται η localhost (127.0.0.1 εδώ συγκεκριμένα) και στη διαθεσιμότητα σύνδεσης εμφανίζεται “False” (“Ψευδής”, δηλαδή δεν υπάρχει διαθέσιμο δίκτυο) (Σχήμα 74).

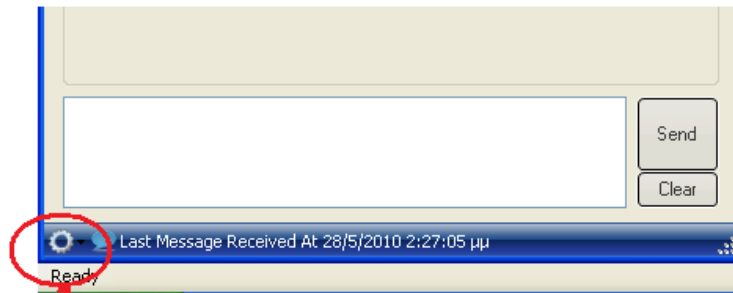
Στο κάτω μέρος της επιφάνειας εργασίας του χειρισμού του ρομποτικού βραχίονα, εμφανίζεται ένα πεδίο το οποίο εμφανίζει το πότε συντάχθηκε η τελευταία εντολή χειρισμού του ρομποτικού βραχίονα (Σχήμα 75)



Σχήμα 75

Πεδίο ενημέρωσης σχετικά με το πότε συντάχθηκε η τελευταία εντολή χειρισμού

Ακριβώς αριστερά από το πεδίο ενημέρωσης, υπάρχει το εικονίδιο ρυθμίσεων της φόρμας εργασίας (Σχήμα 76).

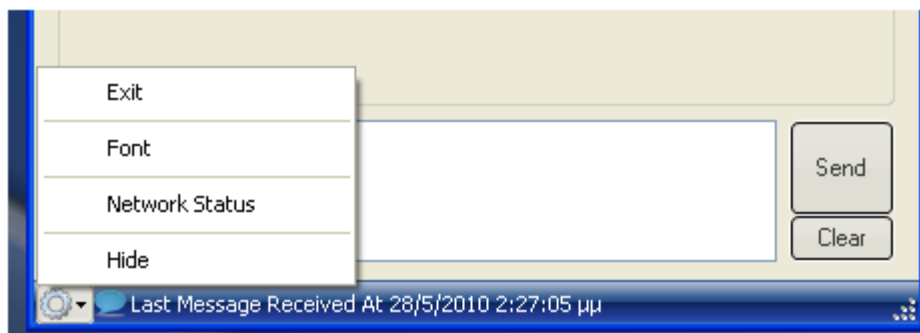


Εικονίδιο Ρυθμίσεων

Σχήμα 76

Όταν επιλεγθεί το συγκεκριμένο εικονίδιο, εμφανίζεται ένα αναδυόμενο menu με διάφορες επιλογές (Σχήμα 77). Αυτές είναι:

- Η επιλογή **“Exit”**, κάνει έξοδο από το πρόγραμμα (τερματισμό).
- Η επιλογή **“Font”**, ορίζει τη γραμματοσειρά που θέλει ο χρήστης να εμφανίζεται στο πεδίο μηνυμάτων.
- Η επιλογή **“Network Status”**, κάνει ό, τι κάνει και η επιλογή **“Browse”** επεξηγήθηκε πιο πριν, δηλαδή εμφανίζει τις λεπτομέρειες του δικτύου (την IP του Server υπολογιστή και το αν υπάρχει διαθέσιμη σύνδεση).
- Η επιλογή **“Hide”** αποκρύπτει την επιφάνεια εργασίας και εμφανίζει ένα βοηθητικό εικονίδιο του προγράμματος στο κάτω δεξιά μέρος της επιφάνειας εργασίας των Windows (Σχήμα 78). Κάνοντας διπλό κλικ επάνω στο συγκεκριμένο εικονίδιο επανέρχεται η επιφάνεια εργασίας χειρισμού του ρομποτικού βραχίονα.



Σχήμα 77: Εμφάνιση αναδυόμενου menu ρυθμίσεων

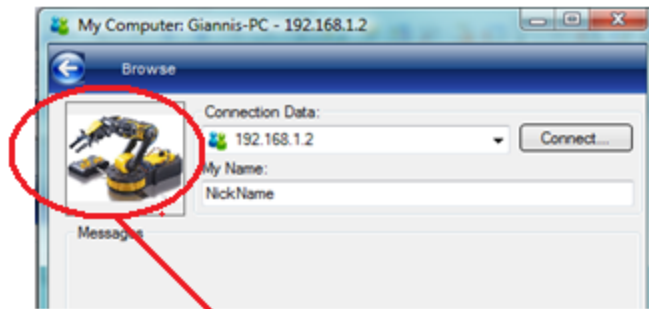


Βοηθητικό εικονίδιο

Σχήμα 78

8.3.2. Χρήση με Εξ' Αποστάσεως Έλεγχο

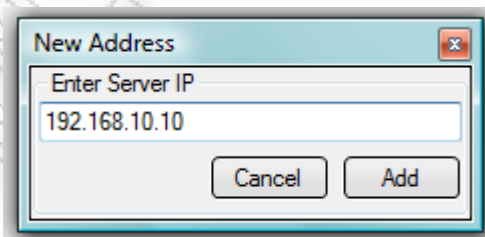
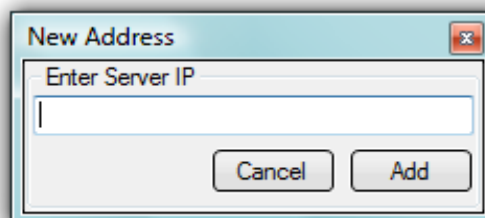
Αφού έχει ελεγχθεί η διασύνδεση του υλικού να έχει γίνει σύμφωνα με το κεφάλαιο 9.1.3. στη σελίδα 47, εκτελείται το εκτελέσιμο αρχείο "RobotArmController" που βρίσκεται στο φάκελο "RobotArmProject" (Σχήματα 63 και 60 αντίστοιχα). Κατόπιν εμφανίζεται η οθόνη εργασίας του ρομποτικού βραχίονα (Σχήμα 64). Το επόμενο σημαντικό βήμα είναι να συνδεθεί ο Client χειριστής στο Server υπολογιστή που είναι συνδεδεμένος ο ρομποτικός βραχίονας. Αυτό γίνεται κάνοντας διπλό κλικ επάνω στο εικονίδιο του ρομποτικού βραχίονα που βρίσκεται στην οθόνη εργασίας επάνω δεξιά, όπως φαίνεται στο σχήμα 79.



Εικονίδιο ρομποτικού βραχίονα

Σχήμα 79: Κάνοντας διπλό κλικ στο εικονίδιο του ρομποτικού βραχίονα

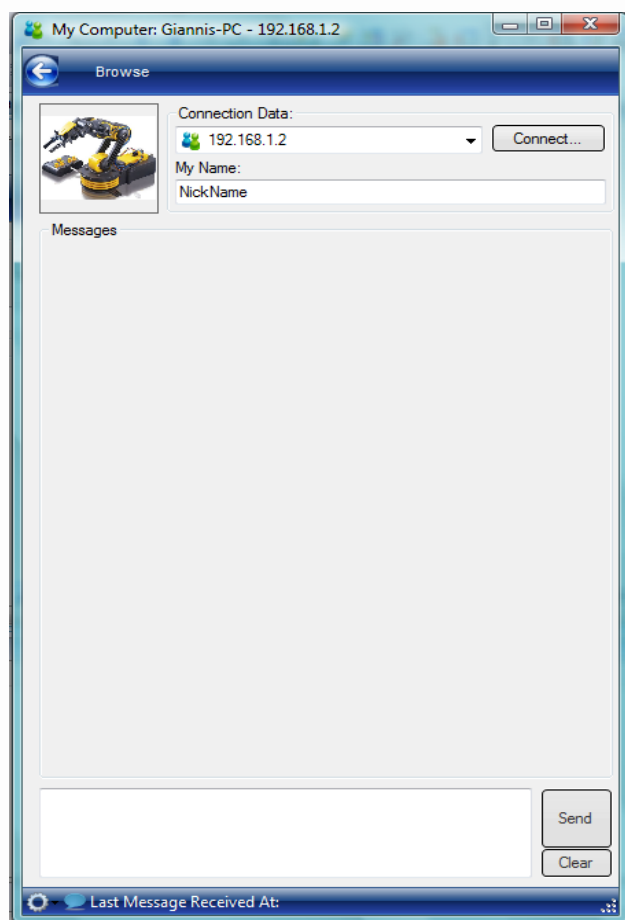
Κάνοντας διπλό κλικ επάνω στο εικονίδιο του ρομποτικού βραχίονα, εμφανίζεται ένα νέο παράθυρο, όπου ζητάει να πληκτρολογηθεί η IP του Server υπολογιστή (Σχήμα 80).



Σχήμα 80: Πληκτρολόγηση IP του Server υπολογιστή

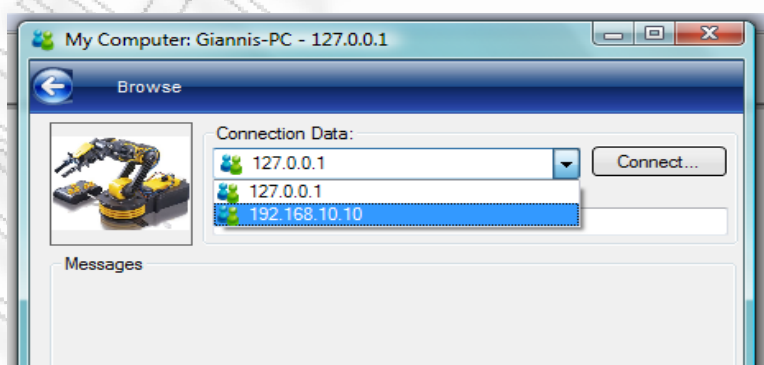
Αφού πληκτρολογηθεί η IP του Server υπολογιστή, ο χειριστής κάνει κλικ στο κουμπί "Add". Έτσι, στο αναδυόμενο menu έχει προστεθεί μία IP. Η χρήση αναδυόμενου menu, γίνεται για το λόγο ότι μπορεί να υπάρχουν περισσότεροι του ενός ρομποτικοί βραχίονες και έτσι ο Client χειριστής έχει την ικανότητα να επιλέξει ποιόν ρομποτικό βραχίονα θα χειριστεί.

Αφού γίνει αυτό το βήμα, επιστρέφουμε στην οθόνη εργασίας και χειρισμού του ρομπωτικού βραχίονα (Σχήμα 81).

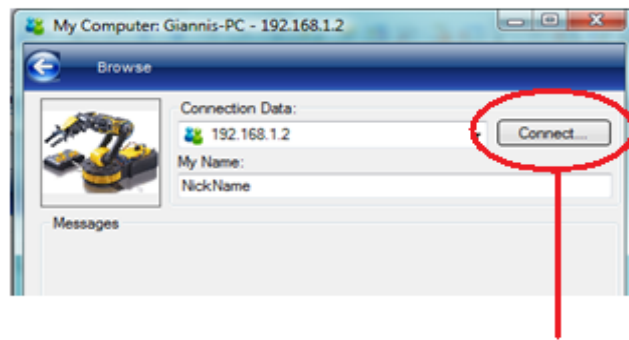


Σχήμα 81: Επιφάνεια εργασίας – χειρισμού

Από το αναδυόμενο μενού "Connection Data", επιλέγεται η IP του Server υπολογιστή (Σχήμα 82) και κατόπιν κάνουμε κλικ στο κουμπί "Connect" (Σχήμα 83).



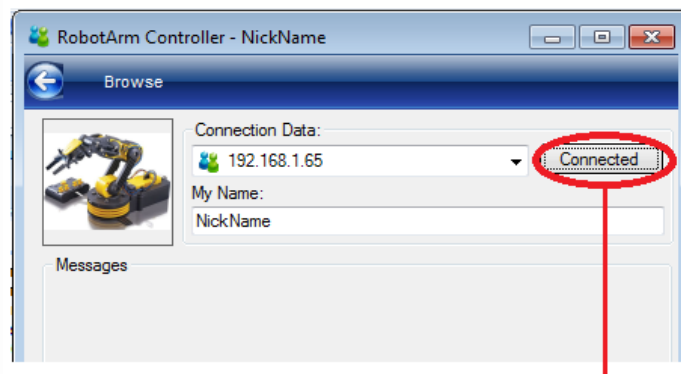
Σχήμα 82: Menu "Connection Data" και επιλογή IP του Server υπολογιστή



Κουμπι Connect

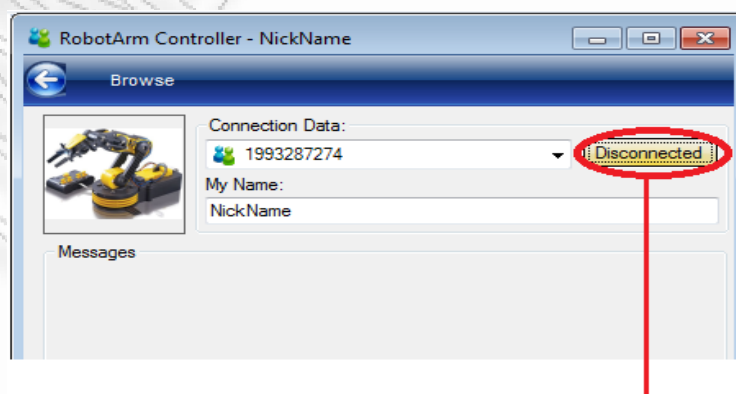
Σχήμα 83: Το κουμπι "Connect", για σύνδεση στην IP

Στη συνέχεια το κουμπι "Connect" μετονομάζεται σε "Connected" (Σχήμα 84), ειδοποιώντας για την ορθή σύνδεση. Οι εντολές χειρισμού του ρομπωτικού βραχίονα πληκτρολογούνται κανονικά, όπως παρουσιάστηκαν στο προηγούμενο κεφάλαιο "Χρήση με Απευθείας Έλεγχο".



Επιτυχής σύνδεση στον Server υπολογιστή

Πρέπει να σημειωθεί ότι, λάθος πληκτρολόγηση της IP του Server υπολογιστή και επιλογή της από το αναδυόμενο menu "Connection Data", έχει σαν αποτέλεσμα το κουμπι "Connect" να μετονομαστεί σε "Disconnected" (Σχήμα 85). Αυτό το γεγονός είναι μία αρχική προειδοποίηση περί λάθους πληκτρολόγησης IP. Αν ο χειριστής, παρόλη την προειδοποίηση που έχει δεχτεί, συνεχίσει και πληκτρολογήσει μία εντολή χειρισμού του ρομπωτικού βραχίονα, το πρόγραμμα θα εμφανίσει exception και ενημερώνει για το λάθος.

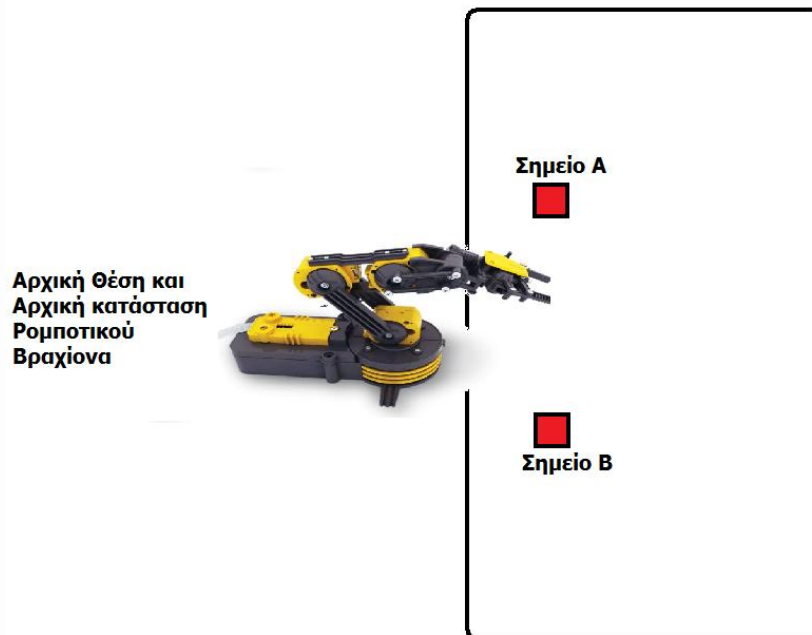


Λάθος πληκτρολόγηση IP

Σχήμα 85: Μετονομασία του κουμπιού "Connect" σε "Disconnected", για να επισημανθεί η λάθος πληκτρολόγηση IP

8.4. Σενάριο Μεταφοράς Αντικειμένου Μέσα στο Χώρο

Έστω ότι υπάρχει η παρακάτω διάταξη χώρου και μέσα στο χώρο υπάρχει ένα αντικείμενο που πρέπει να μετακινηθεί από το σημείο A στο σημείο B (Σχήμα 86).



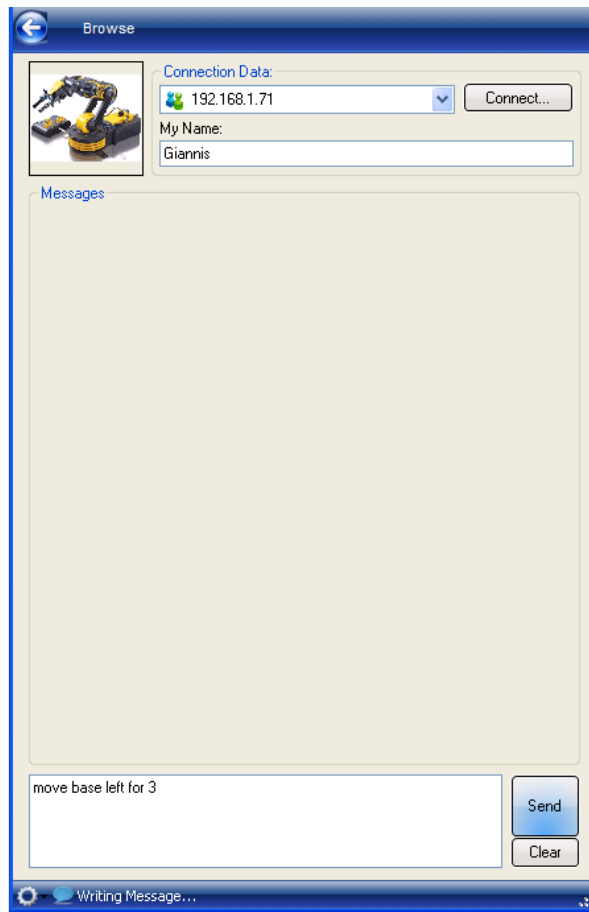
Σχήμα 86: Διάταξη στο χώρο του ρομποτικού βραχίονα

Ο ρομποτικός βραχίονας βρίσκεται σε μία Αρχική Κατάσταση (ή Κατάσταση Ηρεμίας όπως συναντάται η Αρχική Κατάσταση σε πολλές βιβλιογραφίες). Αυτή η κατάσταση υποδηλώνει την αρχική θέση όπου βρίσκονται τα μηχανικά μέρη του ρομποτικού βραχίονα στο χώρο και βεβαίως εννοείται ότι αρχικά βρίσκονται σε ακινησία. Στους επαγγελματικούς ρομποτικούς βραχίονες, η Αρχική Κατάσταση (μπορεί να αναφερθεί και σαν ΑΚ λόγω συντομίας), αποθηκεύεται στην μνήμη του βραχίονα σαν θέση των μηχανικών μερών και διάταξη στο χώρο. Έτσι, κάθε φορά που ο χρήστης καλεί τον ρομποτικό βραχίονα (ενώ βρίσκεται σε θέση πέρα από την αρχική) να μετακινηθεί προς την αρχική του θέση και να αποκτήσει την αρχική του κατάσταση, αυτός χάρη στη μνήμη του μετακινείται χωρίς τη διαμεσολάβηση ηλεκτρολόγησης εντολών από το χρήστη. Ο παρών ρομποτικός βραχίονας, καθότι είναι ρομποτικός βραχίονας – παιχνίδι, δεν παρέχει αυτή την ικανότητα.

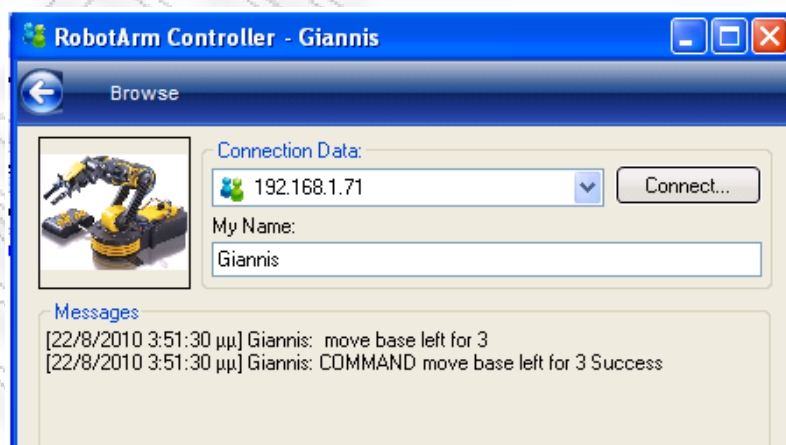
Οπότε σαν ΑΚ στον παρών ρομποτικό βραχίονα, θα υποδηλώνεται μία ιδεατή αρχική κατάσταση σαν αυτή που παρουσιάζεται στην παραπάνω εικόνα και η διαμόρφωση της θα γίνεται πάντοτε από τον χρήστη, προσπαθώντας να προσεγγίσει την εικόνα. Έτσι, σαν πρώτη παρατήρηση, εύκολα κατανοείται ότι η αρπάγη του ρομποτικού βραχίονα σε αυτή τη κατάσταση είναι αρχικά ανοιχτή, οπότε είναι άτοπο και επικίνδυνο για τον βραχίονα να διατυπωθεί σαν πρώτη εντολή χειρισμού η εντολή “OPEN GRIPPER” (άνοιγμα αρπάγης), καθώς ο βραχίονας κινδυνεύει με καταστροφή. Βεβαία, οι επαγγελματικοί ρομποτικοί βραχίονες παρέχουν κάποιες δικλείδες ασφαλείας, όπου εξασφαλίζουν ότι ο βραχίονας δεν κινδυνεύει από τέτοιου είδους αμέλειες.

Για να μετακινηθεί ένα αντικείμενο, όπως στη προκειμένη περίπτωση, από το σημείο A στο σημείο B, θα πρέπει να περάσει από κάποια ενδιάμεσα σημεία (ή ο βραχίονας να κάνει κάποιες “στάσεις”, για να πάρει τη σωστή διάταξη στο χώρο, αποφυγή περιπτώσεων κινήσεων και γενικά οποιοδήποτε προβλήματος). Αρχικά ο ρομποτικός βραχίονας από την ΑΚ θα πρέπει να περιστραφεί η βάση του κατά περίπου 40° αριστερά. Όμως στην διατύπωση των εντολών στη φόρμα εργασίας γίνεται με βάση τα δευτερόλεπτα. Έτσι κατ’ αντιστοιχία, οι 40° αντιστοιχούν σε περίπου 3 δευτερόλεπτα κίνησης της βάσης. Έτσι στη φόρμα εργασίας, ο χρήστης απλά διατυπώνει την εντολή “MOVE BASE LEFT FOR 3” (Σχήματα 87 και 88).

Πρέπει να σημειωθεί πως η αντιστοιχία εύρους κίνησης με τα δευτερόλεπτα κίνησης, είναι εμπειρική.

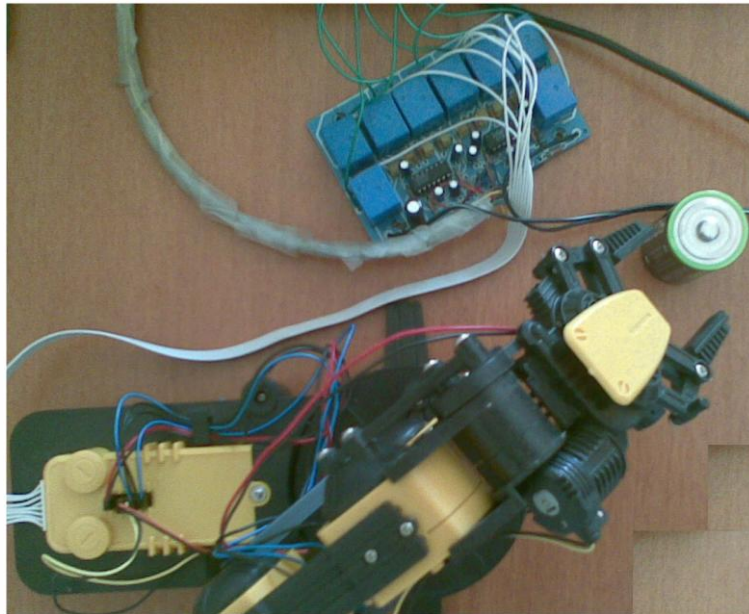


Σχήμα 87: Διατύπωση εντολής

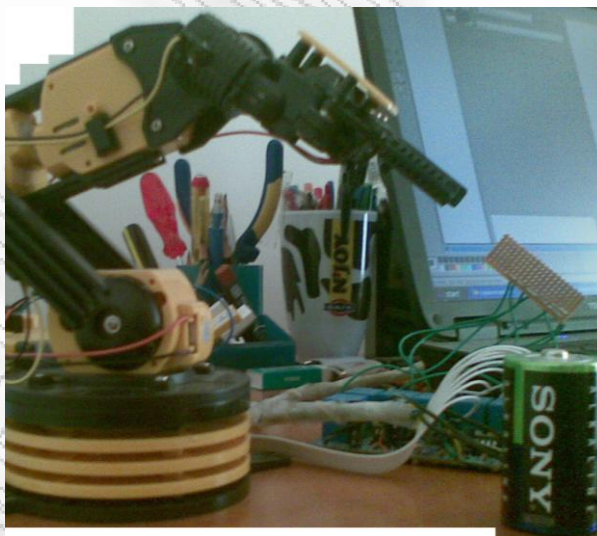


Σχήμα 88: Επιτυχής εκχώρηση εντολής

Έτσι ο ρομποτικός βραχίονας θα μετακινηθεί στο χώρο και θα βρεθεί πάνω ακριβώς από το σημείο A, όπως φαίνεται και στα παρακάτω σχήματα (Σχήματα 89 και 90).



Σχήμα 89: Περιστροφή ρομποτικού βραχίονα κατά 40° από την αρχική του θέση



Σχήμα 90: Πλάγια όψη του ρομποτικού βραχίονα προς το αντικείμενο (μπαταρία) στο σημείο A

Ο βραχίονας για να πιάσει το αντικείμενο, όπως είναι εμφανές, πρέπει να κάνει και μία σειρά από διαδοχικές κινήσεις (Σχήμα 91).



Σχήμα 91: Σειρά διαδοχικών κινήσεων

Αρχικά ο ώμος του ρομποτικού βραχίονα πρέπει να κινηθεί τόσο ώστε να βρεθεί ο βραχίονας στη θέση 1 (Σχήματα 92 και 93).

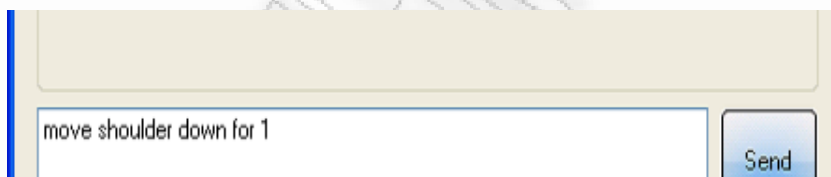


Σχήμα 92: Ο ρομποτικός βραχίονας στη θέση 1



Σχήμα 93: Απεικόνιση της μπροστινής όψης της διάταξης στο χώρο του ρομποτικού βραχίονα και του αντικειμένου (θέση 1)

Πλέον, ο βραχίονας βρίσκεται στη θέση 1, αρκετά κοντά στο αντικείμενο. Επόμενη θέση που πρέπει να βρεθεί είναι η θέση 2 (Σχήμα 91). Διατυπώνεται εντολή ώστε ο ώμος να κινηθεί κατά μικρή κλήση προς τα κάτω (Σχήματα 94, 95, 96, 97), ώστε να απομένει μόνο το κλείσιμο της αρτάνης που θα είναι η 3^η κίνηση.



Σχήμα 94: Διατύπωση εντολής



Σχήμα 95: Επιτυχής εκχώρηση εντολής

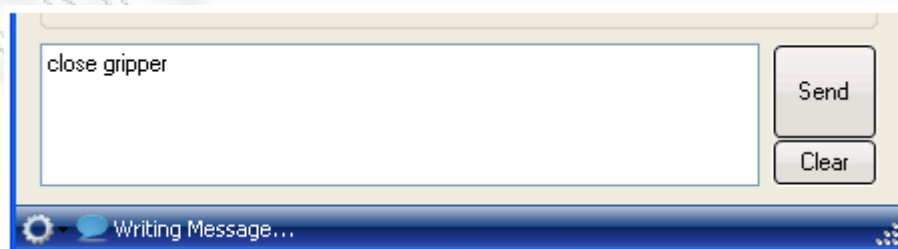


Σχήμα 96: Ο ρομποτικός βραχίονας στη θέση 2

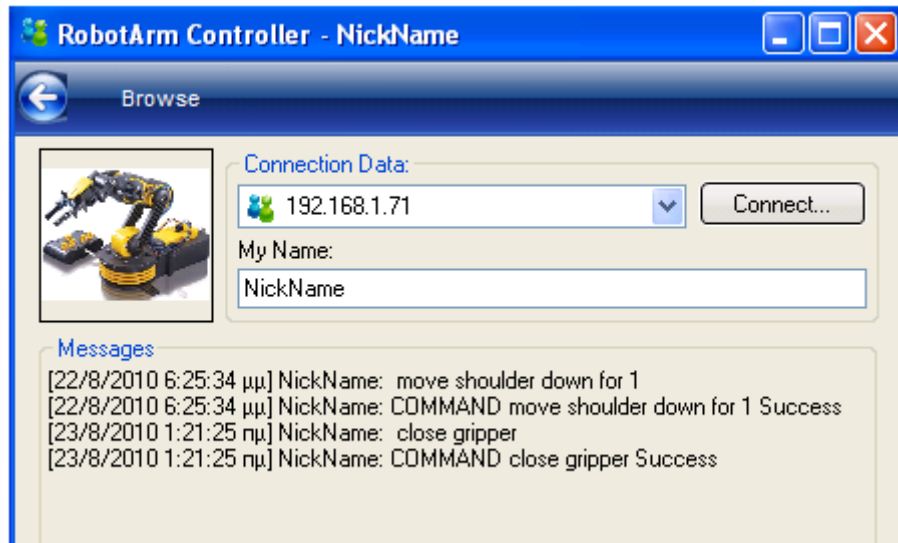


Σχήμα 97: Απεικόνιση της μπροστινής όψης της διάταξης στο χώρο του ρομποτικού βραχίονα και του αντικειμένου (θέση 2)

Όπως έχει ήδη προαναφερθεί στη προηγούμενη παράγραφο, πλέον απομένει μόνο το κλείσιμο της αρπάγης προς το αντικείμενο. Οπότε, πληκτρολογείται η εντολή "CLOSE GRIPPER", ώστε να κλείσει η αρπάγη, εκχωρείται και κατόπιν εκτελείται (Σχήματα 98, 99, 100).



Σχήμα 98: Διατύπωση εντολής



Σχήμα 99: Επιτυχής εκχώρηση εντολής

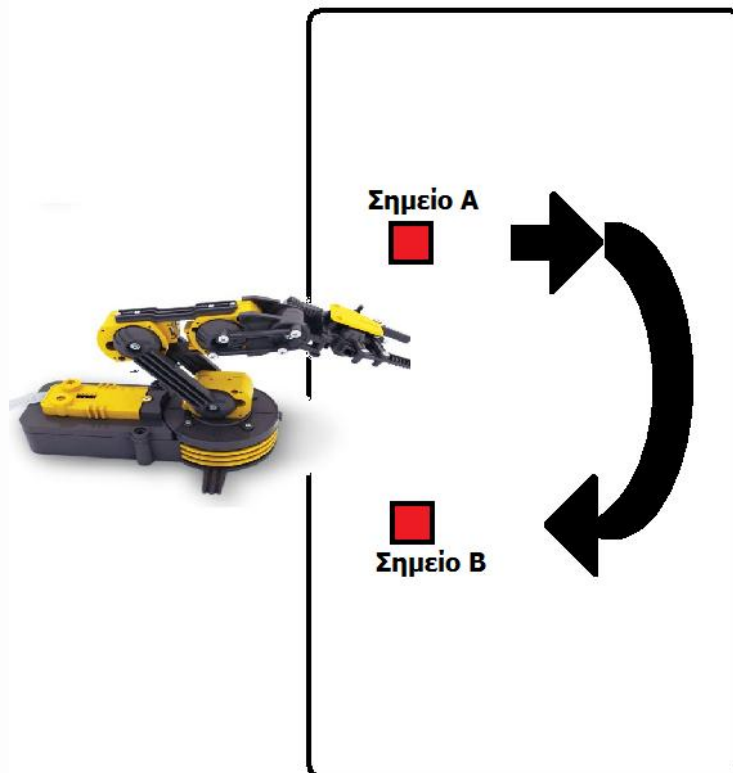
Πλέον ο ρομποτικός βραχίονας πρέπει να μετακινηθεί προς το σημείο B. Το σημείο B απέχει περίπου 100^0 από το σημείο A. Εκφρασμένη αυτή η απόσταση σε χρόνο, είναι περίπου 5 δευτερόλεπτα (Σχήμα 101).



Σχήμα 100: Ο ρομποτικός βραχίονας στη θέση 3 (κλείσιμο αρπάγης)

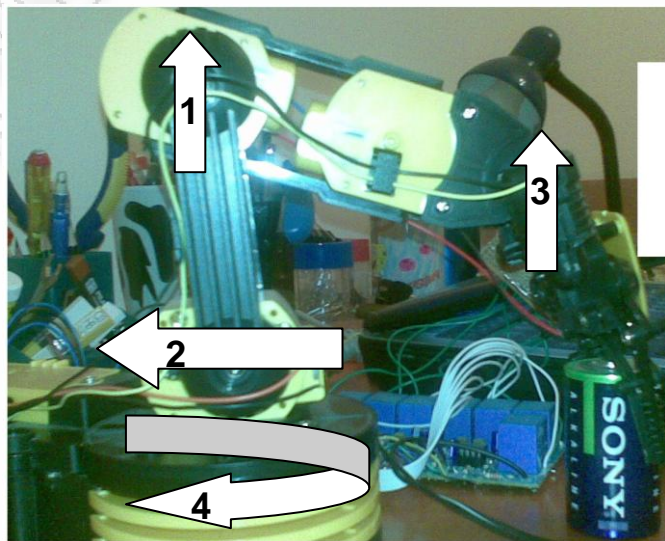
Για να γίνει με επιτυχία η όλη διαδικασία μεταφοράς θα πρέπει ο βραχίονας να ανυψώσει το αντικείμενο σε κάποιο ύψος από την επιφάνεια (Σχήμα 101α). Οπότε, διαδοχικά θα ανυψωθούν: πρώτα ο αγκώνας (Σχήματα 102, 103, 104), μετά ο ώμος (Σχήμα 105, 106, 107), εν συνεχεία ο καρπός (Σχήμα 108, 109, 110) και τέλος θα ακολουθήσει η περιστροφική κίνηση μέχρι το σημείο B (Σχήμα 105). Όταν ο βραχίονας φτάσει πάνω από το σημείο B θα ακολουθήσει η αντίστροφη διαδοχική κινήσεων (κίνηση προς τα κάτω του καρπού, κίνηση προς

τα κάτω του ώμου, κίνηση προς τα κάτω του αγκώνα), ώστε ο βραχίονας να εναποθέσει το αντικείμενο στο σημείο που έχει οριστεί από το χρήστη.



Σχήμα 101: Η κίνηση του ρομποτικού βραχίονα προς το σημείο B

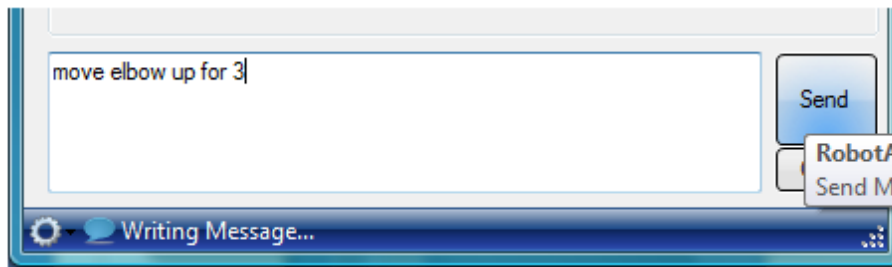
Για λόγους πρακτικούς που εξυπηρετούν την πιο γρήγορη κατανόηση της εργασίας, η μεταφορά του αντικειμένου από το σημείο A στο σημείο B θα γίνει με εξ' αποστάσεως έλεγχο του ρομποτικού βραχίονα (τηλε - έλεγχος ρομποτικού βραχίονα) από απομακρυσμένο υπολογιστή. Έτσι, η πρώτη κίνηση που πρέπει να γίνει από το χρήστη είναι να εκκινήσει την εφαρμογή από τον απομακρυσμένο υπολογιστή (Client) και να περιμένει να εμφανιστεί στην οθόνη του η φόρμα εργασίας. Κατόπιν, πρέπει να αναζητήσει την IP διεύθυνση του υπολογιστή που είναι συνδεδεμένος ο ρομποτικός βραχίονας (Server υπολογιστής). Αυτό μπορεί να γίνει απλά, κάνοντας ένα κλικ στο "Εικονίδιο Ρυθμίσεων" Server υπολογιστή και επιλέγοντας "Network Status" (αναλυτική παρουσίαση υπάρχει στη Σελ. 61 και κεφάλαιο 8.3.2. στη Σελ.62).



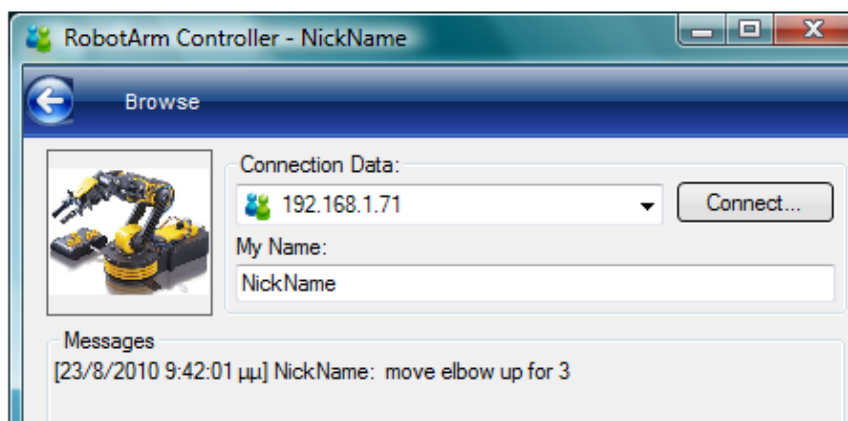
Σχήμα 101^α: Η σειρά των κινήσεων που θα εκτελεστούν από τα μηχανικά μέρη του ρομποτικού βραχίονα

Η IP του Server υπολογιστή είναι 192.168.1.71. Αφού προηγηθεί η σύνδεση του Client υπολογιστή με τον Server (κεφάλαιο 8.3.2. Σελ. 62), ακολουθεί η διαδικασία διατύπωσης των εντολών για έλεγχο του ρομποτικού βραχίονα και μετακίνηση του αντικειμένου από το σημείο A στο σημείο B.

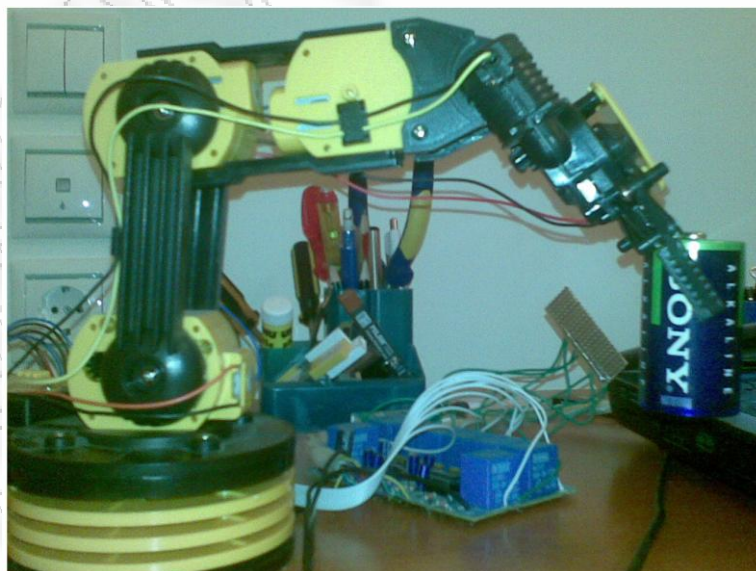
Αρχικά η ανύψωση του αγκώνα:



Σχήμα 102: Διατύπωση εντολής

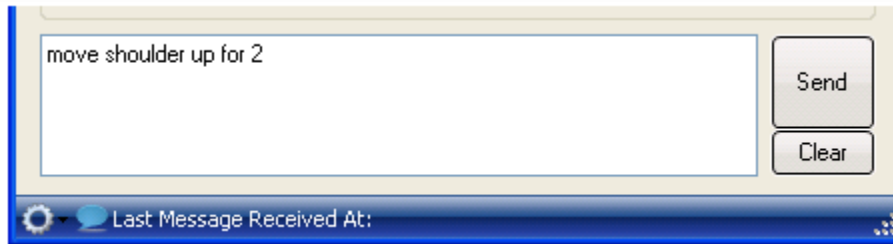


Σχήμα 103: Επιτυχής εκχώρηση εντολής

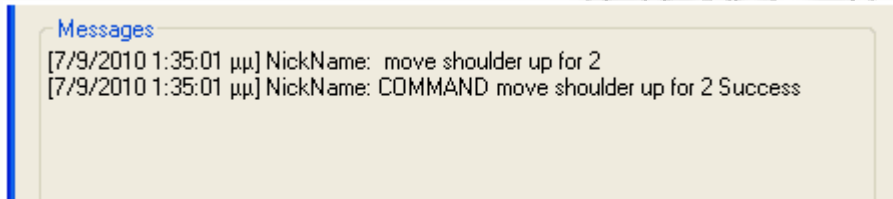


Σχήμα 104: Ανύψωση του αντικειμένου με κίνηση του αγκώνα προς τα επάνω (MOVE ELBOW UP)

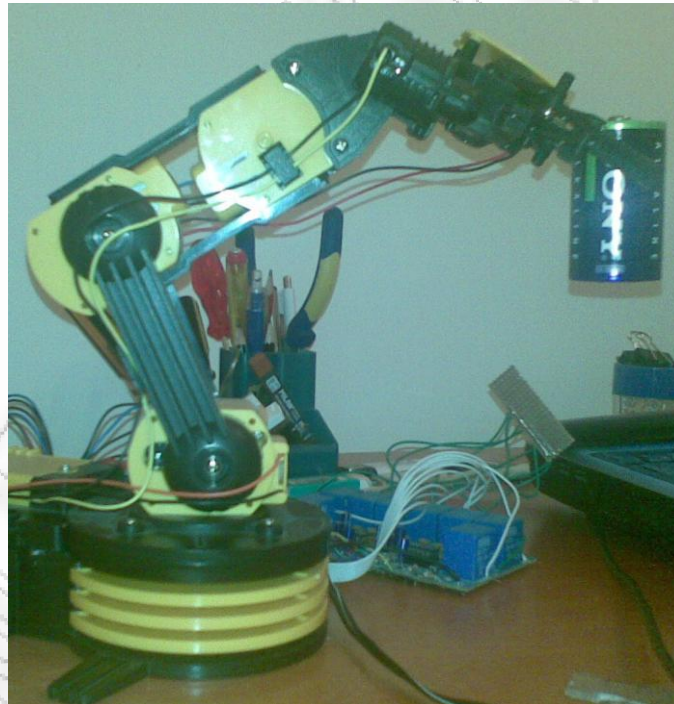
Ακολούθως η ανύψωση του ώμου:



Σχήμα 105: Διατύπωση εντολής

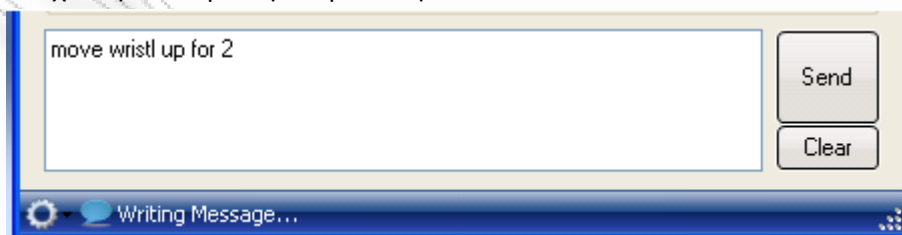


Σχήμα 106: Επιτυχής εκχώρηση εντολής



Σχήμα 107: Ανύψωση του αντικειμένου με κίνηση του ώμου προς τα επάνω (MOVE SHOULDER UP)

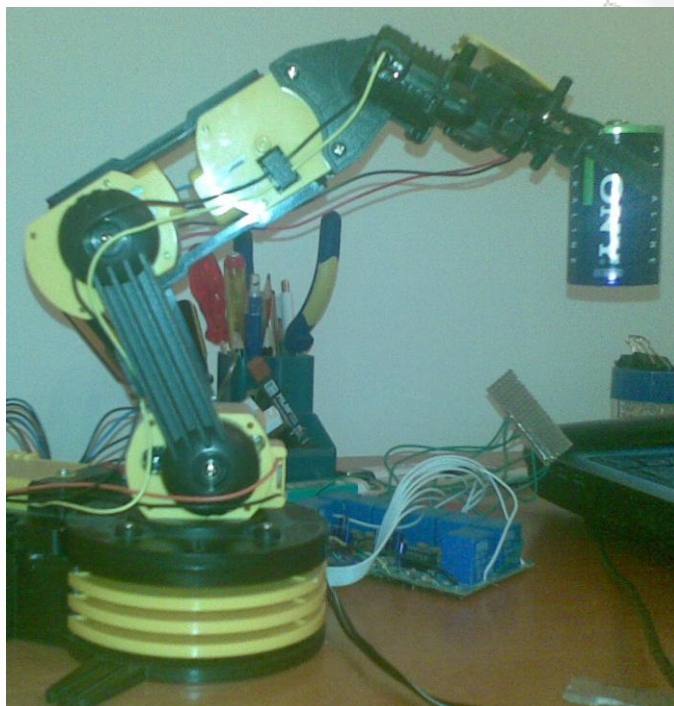
Εν συνεχεία, γίνεται η ανύψωση του καρπού:



Σχήμα 108: Διατύπωση εντολής

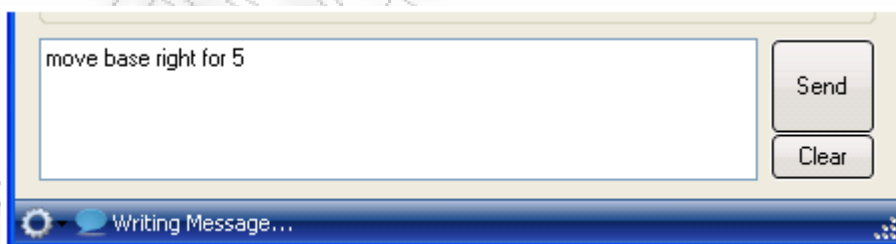
```
[7/9/2010 1:40:30 μμ] NickName: move wristl up for 2  
[7/9/2010 1:40:30 μμ] NickName: COMMAND move wristl up for 2 Success
```

Σχήμα 109: Επιτυχής εκχώρηση εντολής



Σχήμα 110: Ανύψωση του αντικειμένου με κίνηση του καρπού προς τα επάνω (MOVE WRISTL UP)

Μετά γίνεται η περιστροφή της βάσης προς το σημείο που ο βραχίονας θα αφήσει το αντικείμενο:



Σχήμα 111: Διατύπωση εντολής

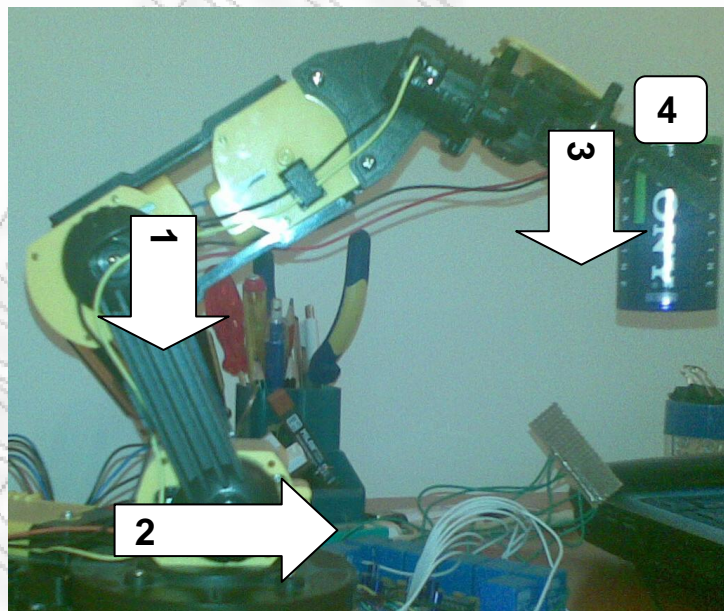
```
Messages  
[7/9/2010 1:40:30 μμ] NickName: move wristl up for 2  
[7/9/2010 1:40:30 μμ] NickName: COMMAND move wristl up for 2 Success  
[7/9/2010 1:44:25 μμ] NickName: move base right for 5  
[7/9/2010 1:44:25 μμ] NickName: COMMAND move base right for 5 Success
```

Σχήμα 112: Επιτυχής εκχώρηση εντολής



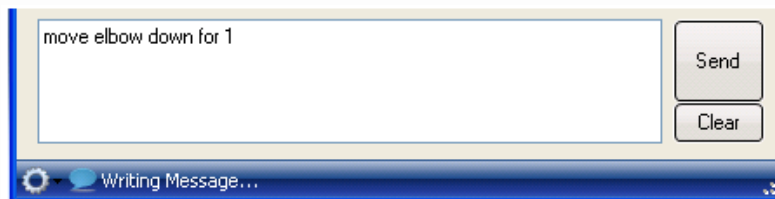
Σχήμα 113: Περιστροφή της βάσης του βραχίονα με κίνηση της βάσης προς τα δεξιά (MOVE BASE RIGHT)

Πλέον, ακολουθεί η διαδικασία εναπόθεσης του αντικειμένου στη θέση προορισμού. Πρώτα θα εκτελεστεί να κατέβει ο αγκώνας (Σχήματα 115, 116, 117), μετά ο ώμος (Σχήματα 118, 119, 120), μετά ο καρπός (Σχήματα 121, 122, 123) και τέλος θα ανοίξει η αρπάγη (Σχήματα 124, 125, 126).



Σχήμα 114: Η σειρά των κινήσεων που θα εκτελεστούν από τα μηχανικά μέρη του ρομποτικού βραχίονα για την εναπόθεση του αντικειμένου

Εκτέλεση κίνησης αγκώνα:



Σχήμα 115: Διατύπωση εντολής

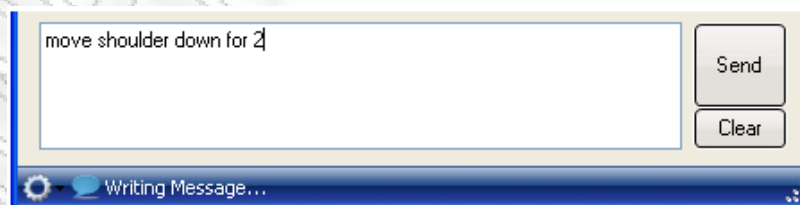
```
[7/9/2010 1:55:36 μμ] NickName: move elbow down for 1  
[7/9/2010 1:55:36 μμ] NickName: COMMAND move elbow down for 1 Success
```

Σχήμα 116: Επιτυχής εκχώρηση εντολής



Σχήμα 117: Κίνηση του αγκώνα του βραχίονα προς τα κάτω (MOVE ELBOW DOWN)

Εκτέλεση κίνησης ώμου:



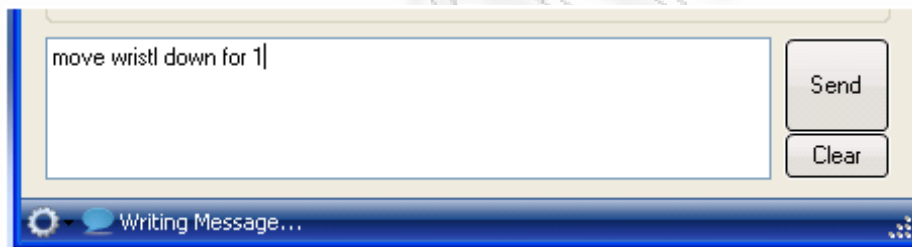
Σχήμα 118: Διατύπωση εντολής

```
[7/9/2010 1:59:22 μμ] NickName: move shoulder down for 2  
[7/9/2010 1:59:22 μμ] NickName: COMMAND move shoulder down for 2 Success
```

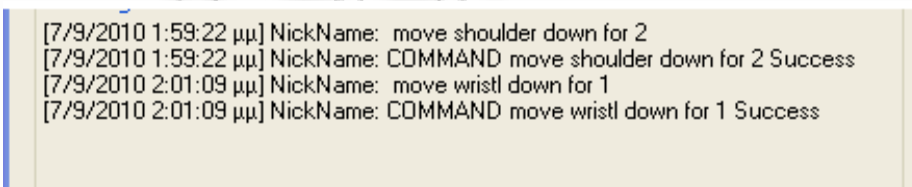
Σχήμα 119: Επιτυχής εκχώρηση εντολής



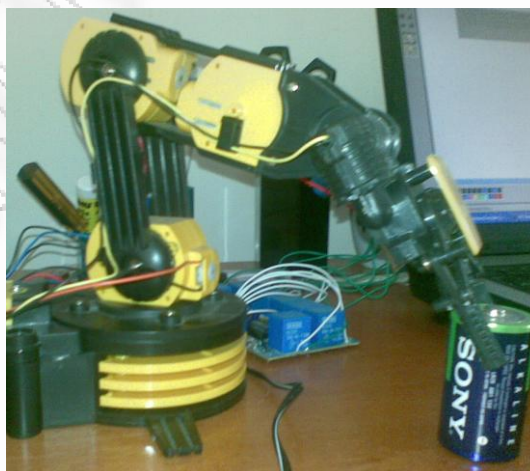
Σχήμα 120: Κίνηση του ώμου του βραχίονα προς τα κάτω (MOVE SHOULDER DOWN)



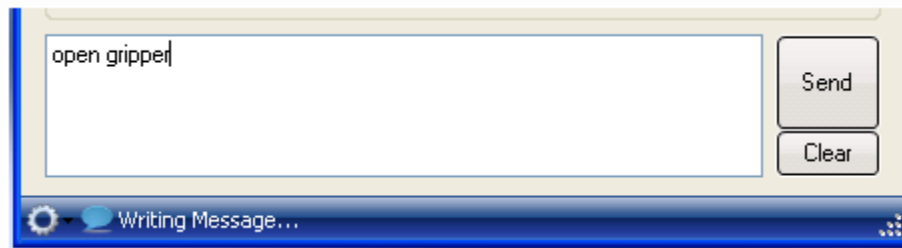
Σχήμα 121: Διατύπωση εντολής



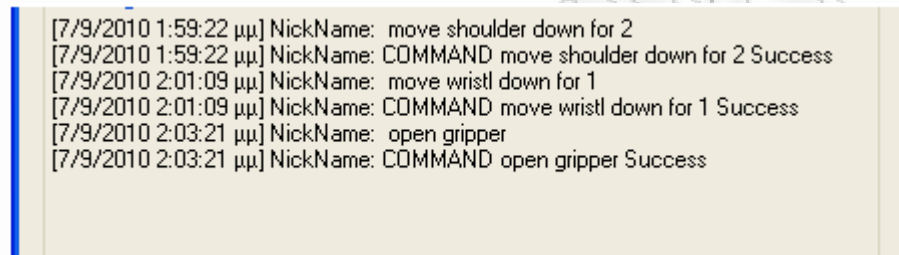
Σχήμα 122: Επιτυχής εκχώρηση εντολής



Σχήμα 123 :Κίνηση του καρπού του βραχίονα προς τα κάτω (MOVE WRISTL DOWN)



Σχήμα 124: Διατύπωση εντολής



Σχήμα 125: Επιτυχής εκχώρηση εντολής



Σχήμα 126: Άνοιγμα της αρπάγης του καρπού του βραχίονα (OPEN GRIPPER)

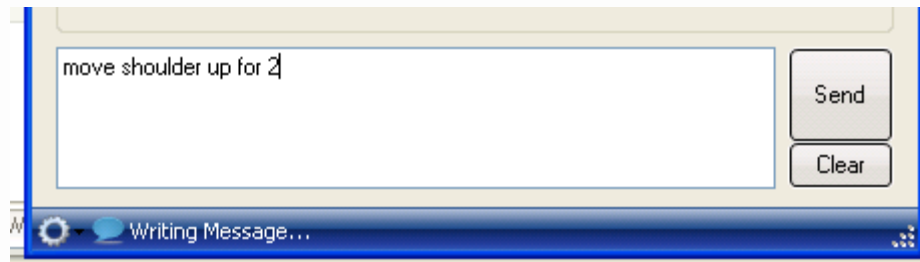
Το αντικείμενο βρίσκεται στο σημείο B.

8.5. Προγραμματισμός – Λειτουργία της Κλάσης

Στο παρόν κεφάλαιο θα παρουσιαστεί ο τρόπος με τον οποίο λειτουργεί η κλάση του ρομποτικού βραχίονα. Η επίδειξη της λειτουργίας θα δοθεί με απλά παραδείγματα κίνησης του βραχίονα.

Η υπεύθυνη συνάρτηση για την εκτέλεση οποιασδήποτε κίνησης του ρομποτικού βραχίονα, είναι η `executeCommand()` όπου έχει αρχική τιμή `False` (Ψευδής, δηλαδή είναι ανενεργή) και αναμένει το χρήστη να πληκτρολογήσει μια εντολή κίνησης. Στο παρακάτω σχήμα (Σχήμα 128) φαίνονται οι μεταβλητές που περιλαμβάνει η κλάση (`Me`), η εντολή χειρισμού που πληκτρολογείται από τον χρήστη (`commandString`) και η ενεργοποίηση εκτέλεσης της εντολής (`executeCommand`).

Έστω ότι πληκτρολογείται και κατοχυρώνεται επιτυχώς η εντολή “move shoulder up for 2” (Σχήμα 127).



Σχήμα 127: Πληκτρολόγηση εντολής “move shoulder up for 2”

Name	Value	Type
Me	{RobotArmController.InputOut32.ClassRobotControl}	RobotArm
commandString	"move shoulder up for 2"	String
executeCommand	False	Boolean

Σχήμα 128: Οι τιμές των μεταβλητών

Αρχικά, το σύστημα θα μετατρέψει την εντολή σε κεφαλαία γράμματα, για σκοπό που εξυπηρετεί μετέπειτα χρήση. Αυτό γίνεται με τη χρήση της συνάρτησης UCase() και αποθηκεύεται εκ νέου στη μεταβλητή commandString (με κεφαλαία γράμματα πλέον - Σχήματα 129 και 130).

```
commandString = UCase (commandString)
```

Σχήμα 129: Μετατροπή σε κεφαλαία

commandString	"MOVE SHOULDER UP FOR 2"	String
---------------	--------------------------	--------

Σχήμα 130

Η διατύπωση της εντολής χωρίζεται σε δύο μέρη. Στο αριστερό μέρος, όπου δίνεται το είδος της κίνησης που καλείται να εκτελέσει ο ρομποτικός βραχίονας (Σχήμα 131) και το δεξιό μέρος, το οποίο περιέχει το χρόνο εκτέλεσης της κίνησης (Σχήμα 132).

MOVE SHOULDER UP
ΑΡΙΣΤΕΡΟ ΜΕΡΟΣ

Σχήμα 131

FOR 2
ΔΕΞΙΟ ΜΕΡΟΣ

Σχήμα 132

Σε αυτό το σημείο πρέπει να προσεχθεί το γεγονός πως οι εντολές που σχετίζονται με την κίνηση της αρπάγης (open gripper, close gripper), δεν ακολουθούνται από χρονικό προσδιορισμό της κίνησης (δηλαδή δεν έχουν δεξιό μέρος) και θεωρούνται ότι είναι εξ' ορισμού ένα δευτερόλεπτο. Αυτό γίνεται για λόγους προστασίας της αρπάγης, καθώς αν κινηθεί πάνω από ένα δευτερόλεπτο υπάρχει κίνδυνος να σπάσει. Παρόμοια συμβαίνει και με τα άλλα μηχανικά μέρη, τους κινητήρες, του βραχίονα, όπου δεν επιτρέπεται να κάνουν κινήσεις άνω των δέκα δευτερολέπτων.

Επανερχόμενοι, πρώτα ελέγχεται το δεξιό μέρος του χρονικού προσδιορισμού της κίνησης. Αν υπάρχει, τότε μετατρέπεται σε 16δικό νούμερο και μεταφέρεται στο χρονόμετρο της κλάσης (Αρχικά το χρονόμετρο της κλάσης είναι '0', Σχήματα 133,134 και 135).

timer	0	Integer
-------	---	---------

Σχήμα 133: Αρχική τιμή timer = 0

timer	2	Integer
-------	---	---------

Σχήμα 134: Τελική τιμή timer = 2

```
If IsNumeric(Right(commandString, 1)) Then
    Me.timer = Convert.ToInt16(Right(commandString, 1))
```

Σχήμα 134^a: Έλεγχος και προγραμματισμός του χρονομέτρου της κλάσης

Ακολούθως, ο κώδικας ελέγχει το αριστερό μέρος της εντολής και το συγκρίνει με το set εντολών που διαθέτει για να αναγνωρίζει το είδος των κινήσεων (Σχήματα 135 και 136), δηλαδή συγκρίνει αν ταυτίζονται τα command String. Για αυτό το λόγο έγινε πρωτίτερα η μετατροπή της εντολής χειρισμού από μικρά σε κεφαλαία γράμματα και πλέον εξασφαλίζεται το γεγονός πως ο χρήστης, μπορεί ελεύθερα να πληκτρολογήει τις εντολές είτε με κεφαλαία είτε με μικρά γράμματα.

```
commandString = Left(commandString, Len(commandString) - 1)
```

Σχήμα 135: Κατοχύρωση εντολής

```
If commandString = "OPEN GRIPPER" Then
    Me.Value = 136
ElseIf commandString = "CLOSE GRIPPER" Then
    Me.Value = 9
ElseIf commandString = "MOVE WRISTL UP FOR " Then
    Me.Value = 5
ElseIf commandString = "MOVE WRISTL DOWN FOR " Then
    Me.Value = 132
ElseIf commandString = "MOVE ELBOW UP FOR " Then
    Me.Value = 3
ElseIf commandString = "MOVE ELBOW DOWN FOR " Then
    Me.Value = 130
ElseIf commandString = "MOVE SHOULDER UP FOR " Then
    Me.Value = 144
ElseIf commandString = "MOVE SHOULDER DOWN FOR " Then
    Me.Value = 17
ElseIf commandString = "MOVE BASE LEFT FOR " Then
    Me.Value = 33
ElseIf commandString = "MOVE BASE RIGHT FOR " Then
    Me.Value = 160
ElseIf commandString = "EXIT" Then
    Me.Value = 0
Else
    Me.Value = -1
End If
```

Σχήμα 136: Set εντολών

Όσο δεν ταιριάζουν οι χαρακτήρες του δοθέντος, από το χρήστη, string με τα κατοχυρωμένα σαν κινήσεις string του βραχίονα, τότε προχωράει την αναζήτηση ο κώδικας. Αν η αναζήτηση δεν ταυτιστεί, τότε στη φόρμα μηνυμάτων εμφανίζεται μήνυμα προς το χρήστη που τον ενημερώνει ότι υπάρχει λάθος σύνταξη εντολής.

Μία σημαντική σημείωση για τον κώδικα είναι ότι συναντιούνται τρεις καταστάσεις. Η κατάσταση “Επιτυχίας” όπου στον κώδικα σημειώνεται με “1”, η κατάσταση “Αποτυχίας” όπου σημειώνεται με “-1” και η κατάσταση “Εξόδου” όπου σημειώνεται με “0”. Σε κάθε “Επιτυχή” κατάσταση δημιουργείται ένα thread (νήμα, Σχήμα 137), όπου μετατρέπει την κίνηση σε μία ανεξάρτητη λειτουργία και την προστατεύει από τυχόν “κολλήματα” που μπορεί να συμβούν στο σύστημα.

```
If (Me.Value <> -1 And Me.Value <> 0) Then
    movementThread = New Threading.Thread(AddressOf move)
    movementThread.Start()
ElseIf (Me.Value = -1) Then
    Return False
ElseIf (Me.Value = 0) Then
    Return False
End If
```

Σχήμα 137: Δημιουργία thread

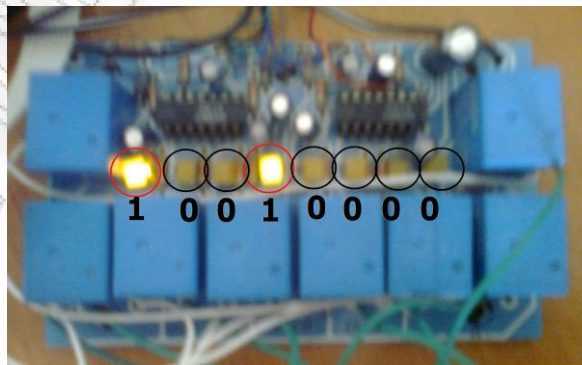
Στη 13^η γραμμή του set εντολών (Σχήμα 136), συναντιέται η εντολή που ταιριάζει με τη δοθείσα από το χρήστη. Παρατηρείται ότι σε κάθε εντολή υπάρχει μία τιμή Value. Αυτή η τιμή είναι η δεκαδική αντιστοιχία της κίνησης, ώστε στη συνέχεια να αποκωδικοποιηθεί σε ρεύματα, να σταλθεί μέσω της παράλληλης θύρας και να ενεργοποιηθούν τα απαιτούμενα ρελέ, από την πλακέτα ελέγχου, που χρειάζονται ώστε να πραγματοποιηθεί η κίνηση από τους κινητήρες του βραχίονα.

Στην εντολή “MOVE SHOULDER UP FOR”, η τιμή Value παίρνει δεκαδική τιμή 144 (ήτοι 10010000 στο δυαδικό σύστημα - Σχήματα 138 και 139). Αυτή η τιμή στέλνεται σαν ρεύμα μέσω της παράλληλης θύρας (port 888 – Σχήμα 138) στην πλακέτα ελέγχου.

PortAddress	888	Short
timer	2	Integer
Value	144	Short

Σχήμα 138: Οι τιμές των μεταβλητών της κλάσης

Η δυαδική αναπαράσταση (σε 8 bit) είναι η αντιστοιχία των 8 ρελέ (Σχήμα 139). Δηλαδή, όταν στην δυαδική αναπαράσταση συναντάται ‘0’, σημαίνει ότι το αντίστοιχο ρελέ είναι σε κατάσταση “Normally Open”, ενώ όταν στη δυαδική αναπαράσταση συναντάται ‘1’, σημαίνει ότι το αντίστοιχο ρελέ είναι σε κατάσταση “Normally Close”.



Σχήμα 139: Αντιστοιχία 8 ρελέ με δυαδική αναπαράσταση 8 bit

8.6. Προοπτικές Εξέλιξης της Εργασίας

Η παρούσα εργασία αποτελεί ένα ερασιτεχνικό εγχείρημα στα πλαίσια μεταπτυχιακής διατριβής. Σκοπός της ήταν να αναδείξει, έστω σε ένα μικρό βαθμό, τις δυνατότητες συνεργασίας λογισμικού και υλικού. Βέβαια, λόγω των περιορισμένων οικονομικών δυνατοτήτων και του περιορισμένου χρόνου (καθώς παρόμοια εγχειρήματα για να καλύψουν όλες τις απαιτήσεις, χρειάζονται συνεργασία πολλών ατόμων, καιρό μελέτης και προεργασίας και πολλές οικονομικές δυνατότητες), εμφανίζονται κάποιες ατέλειες – αδυναμίες του προγράμματος και του ρομποτικού βραχίονα, οι οποίες θα μπορούσαν στο μέλλον, ίσως στα πλαίσια εξέλιξης της εργασίας από το γράφοντα ή από κάποιον άλλο φοιτητή, να απαλειφθούν. Έτσι, σε αυτό το κεφάλαιο γίνεται μία τέτοια αναφορά στις όποιες αδυναμίες παρουσιάζονται.

Αρχίζοντας, πρέπει να γίνει αναφορά στην απαίτηση για επικοινωνία – συνεννόηση μεταξύ Server και Client χειριστών. Αυτό γιατί, οι Client χειριστές δεν έχουν τη δυνατότητα να ξέρουν ποιες κινήσεις έχουν προηγηθεί και υπάρχει σοβαρό ενδεχόμενο να πληκτρολογήσουν μία εντολή η οποία να έχει δοθεί πιο πριν (είτε από κάποιον άλλον Client χειριστή είτε από το Server χειριστή).

Επίσης, ο Client χειριστής δεν είναι σε θέση να γνωρίζει αν πληκτρολόγησε λάθος εντολή και αυτό το γνωρίζει μόνο ο Server υπολογιστής, καθώς εκεί εμφανίζεται το μήνυμα λάθους.

Αναφορά ιδιαίτερη πρέπει να δοθεί στην κατασκευή του ρομποτικού βραχίονα. Η αδυναμία του έγκειται στο γεγονός μη ακρίβειας στην εκτέλεση των κινήσεων, λόγω της χρησιμοποίησης απλών DC κινητήρων. Γι' αυτό το λόγο έχουν προσαρμοστεί οι εντολές χειρισμού με να εκτελούνται με βάση χρονικό προσδιορισμό. Το ιδανικό σενάριο (που όμως απαιτεί κάποια περαιτέρω οικονομική δαπάνη), είναι να μπορούσαν να χρησιμοποιηθούν κάποιοι Servo κινητήρες, οι οποίοι προσδίδουν τεράστια ακρίβεια κινήσεων. Επίσης θα μπορούσε να εξομοιώνεται κάθε φορά η κίνηση του ρομποτικού βραχίονα με τη χρήση ενός flash player, δείχνοντας στο χειριστή τη θέση και την κίνηση του ρομποτικού βραχίονα στο χώρο. Εννοείται πως θα υπήρχε κατάργηση του χρονικού προσδιορισμού των κινήσεων και αυτές θα δίδονταν προς εκτέλεση από έναν ψηφιακό πίνακα ελέγχου.

Μία πρόσθετη λειτουργία που θα μπορούσε να υπάρχει στο πρόγραμμα είναι εκείνη του χρονοπρογραμματισμού των κινήσεων. Δηλαδή, σε ένα αρχείο κειμένου, ο χειριστής να διατυπώνει τις κινήσεις που θέλει να εκτελεστούν (πρέπει να έχει προηγηθεί μελέτη των κινήσεων) και τη συγκεκριμένη ώρα εκτέλεσης της κάθε κίνησης.

Αν καλύπτονταν όλες αυτές οι ατέλειες που προαναφέρθηκαν και επιπλέον προσαρμοζόντουσαν κάποιες κάμερες επάνω στο ρομποτικό βραχίονα, θα μπορούσε το όλο εγχείρημα να αποτελέσει ένα ιδανικό τηλεματικό σύστημα, το οποίο με την κατάλληλη προσαρμογή και επέκταση του προγράμματος θα μπορούσε να εξάγει αποτελέσματα και μετρήσεις.

Δεν θα ήταν υπερβολή επίσης να αναφερθεί ότι το πρόγραμμα ελέγχου του ρομποτικού βραχίονα από τη μεριά του Client, θα μπορούσε να ελεγχθεί από ένα κινητό τηλέφωνο (Smartphone), είτε μέσω μηνύματος, όπου θα διατυπωνόταν με συγκεκριμένη μορφή και θα αποστελλόταν σε ένα GSM modem όπου θα είχε ενσωματωμένο ο ρομποτικός βραχίονας, είτε με απευθείας έλεγχο μέσω ασύρματης Bluetooth συνδεσιμότητας. Όλο αυτό το προαναφερθέν σενάριο θα μπορούσε να αποτελέσει ιδανική σκέψη αν προσαρμόζονταν κατάλληλα και κάμερες επίβλεψης.

Κάτι άλλο που θα μπορούσε όχι απαραίτητα να αλλαχθεί, αλλά να ληφθεί υπόψη στην περαιτέρω βελτίωση του προγράμματος, θα ήταν η αλλαγή της διασύνδεσης από παράλληλη σε USB. Από τη μεριά του γράφοντος αυτό δεν μπόρεσε να πραγματοποιηθεί καθώς για τον προγραμματισμό της USB θύρας απαιτούνται πολλές και εξειδικευμένες γνώσεις πάνω στον προγραμματισμό της, ενώ στην παράλληλη θύρα δεν συμβαίνει το ίδιο. Στο διαδίκτυο, υπάρχουν πολλά προγράμματα – παραδείγματα χειρισμού της παράλληλης θύρας και σε γενικές γραμμές, για εργαστηριακές επιδείξεις η χρησιμοποίηση της, λόγω της ευκολίας της, την καθιστά ιδανική. Όμως πλέον όλο και λιγότερο συναντιέται η παράλληλη θύρα για τους λόγους που έχουν αναφερθεί στο κεφάλαιο 7 (“Οι Θύρες Επικοινωνίας του Υπολογιστή”).

Όλες αυτές οι ιδέες που αναφέρθηκαν για περαιτέρω εξέλιξη της παρούσας εργασίας αποτελούν υποκειμενική άποψη του γράφοντα. Όπως συμβαίνει με όλες τις επιστήμες, έτσι και στις πληροφορικές επιστήμες, μπορούν να αναπτυχθούν πολλά σκεπτικά και ιδέες.

9. Επίλογος

Σε αυτό το κεφάλαιο θα περιγραφεί μία συνοπτική, στα όρια του επιτρεπτού, ανάλυση για τα κοινωνικά θέματα που προκύπτουν από τη συνεχόμενη και ταχύτατα αναπτυσσόμενη χρήση των υπολογιστών. Γιατί, το πιο σημαντικό και ενδιαφέρον αποτέλεσμα της επανάστασης των υπολογιστών, είναι η επίδραση τους στη ζωή των ανθρώπων και στη δομή της κοινωνίας.

Ένα τέτοιο χαρακτηριστικό παράδειγμα αποτελεί το παρόν νομισματικό σύστημα, όπου η επανάσταση των υπολογιστών οδηγεί σε αυτό που έχει ονομασθεί από τους επιστήμονες του χώρου “κοινωνία χωρίς μετρητά” (cashless society), καθώς και τα χρήματα (με τη μορφή μετρητών) ίσως καταστούν άχρηστα.

Τα γραφεία μίας επιχείρησης είναι μία άλλη περιοχή στην οποία η επανάσταση των υπολογιστών θα μπορούσε να έχει μεγάλη επίδραση. Έχει συντελεστεί ήδη μία γρήγορη απομάκρυνση από τις κοινές γραφομηχανές προς όφελος “συστημάτων επεξεργασίας κειμένου” (word-processing systems). Τα συστήματα αυτά είναι ουσιαστικά “γραφομηχανές” συνδεδεμένες (ή ενσωματωμένες) σε υπολογιστή και στο οποίο μπορεί να αποθηκευτεί ολόκληρο το κείμενο όπου πληκτρολογείται. Ο υπολογιστής επιτρέπει να διορθωθούν τα λάθη και να αναθεωρηθεί το ήδη υπάρχον κείμενο, χωρίς να είναι ανάγκη να πληκτρολογηθεί από την αρχή. Επίσης, πολλές εργασίες που γίνονται από διευθυντικά στελέχη όπως η συγκέντρωση πληροφοριών και η εξαγωγή προβλέψεων, είναι επαναλαμβανόμενες και υπάρχει ανάγκη για αυτόματα συστήματα διαχείρισης πληροφοριών τα οποία θα βοηθούν στη λήψη αποφάσεων. Το αυτοματοποιημένο γραφείο του μέλλοντος, θα μπορέσει να γίνει πολύ καλύτερο από το γραφείο με τη σημερινή έννοια.

Το σπίτι είναι μία άλλη περιοχή όπου ουσιαστικές αλλαγές είναι δυνατές με τη βοήθεια του αυτοματισμού. Οι εφαρμογές των υπολογιστών στις οικιακές συσκευές και στην οικιακή ψυχαγωγία είναι πραγματικά απεριόριστες. Το “έξυπνο σπίτι”, όπως αποκαλείται το αυτοματοποιημένο σπίτι, θα μπορεί να ρυθμίζεται σύμφωνα με τις συνήθειες του ιδιοκτήτη του (να έχει έτοιμο ζεστό νερό, θερμοκρασία σπιτιού σύμφωνα με αυτή που ζητά ο ιδιοκτήτης τη δεδομένη χρονική στιγμή) και θα μπορεί να καλύψει τις πρώτες ανάγκες του, όπως την κάλυψη τροφίμων δίνοντας παραγγελία στο super market της περιοχής και κανονίζοντας τις πληρωμές λογαριασμών του νοικοκυριού με μία απλή εντολή στην τράπεζα.

Θα είναι επίσης δυνατόν στο μέλλον, να κατασκευαστούν αυτοματοποιημένα ρομπότ – υπηρέτες του σπιτιού, τα οποία θα καθαρίζουν, θα γυαλίζουν, θα σιδερώνουν, αν και αυτό το γεγονός φαντάζει ακόμα κάπως μακρινό προς το παρόν. Παρ’ όλα αυτά, βιομηχανικά ρομπότ με σχετικά απλά και αυστηρά καθορισμένα καθήκοντα, απασχολούνται με ολοένα αυξανόμενους αριθμούς και μπορεί τελικά να αυτοματοποιήσουν τελείως όλες τις γραμμές μαζικής παραγωγής, καταργώντας ακόμη και όση ανθρώπινη παρέμβαση έχει απομείνει σε αυτές.

Μέχρι τις αρχές της δεκαετίας του 1950 υπήρχαν δυοίωνες προβλέψεις ότι η επανάσταση των υπολογιστών θα επέφερε δραματική μείωση των ωρών εργασίας και μαζική ανεργία μέσα σε λίγες δεκαετίες. Αυτές οι προβλέψεις δεν έχουν επαληθευτεί μέχρι σήμερα, τουλάχιστον όχι στην έκταση που αρχικά είχε προβλεφτεί. Οπωσδήποτε ο αυτοματισμός έχει προκαλέσει μία σταθερή αύξηση στην παραγωγικότητα, αλλά αυτό κατά μεγάλη πιθανότητα έχει συνδυαστεί με την αυξημένη κατανάλωση, παρά με περισσότερο ελεύθερο χρόνο. Αν και το συνολικό επίπεδο απασχόλησης δεν έχει αλλάξει δραματικά εξαιτίας του αυτοματισμού, έχει συντελεστεί **εργασιακή μετατόπιση**, σαν αποτέλεσμα μετάβασης σε νέες περιοχές απασχόλησης.

Φυσικά, πολλοί άνθρωποι περιμένουν από την εργασία τους περισσότερα από το να έχουν απλά μία αμοιβή. Σε αυτό το γεγονός εμπλέκονται παράγοντες, όπως η ηθική ικανοποίηση, μία επιτυχημένη επαγγελματική καριέρα, ικανοποίηση της επιθυμίας για κοινωνική άνοδο, άσκηση εξουσίας σε άλλους, έλεγχος πάνω σε γεγονότα και υποταγή στο πιεστικό κοινωνικό κατεστημένο. Ο αυτοματισμός μπορεί να επηρεάσει όχι μόνο την ποσότητα και την κατανομή της εργασίας, αλλά και αυτούς τους μη υλικούς παράγοντες. Γενικά, η φύση των υπολογιστών οδηγεί κάποιον στο συμπέρασμα ότι τα πληκτικά και γεμάτα επαναλήψεις

καθήκοντα θα αυτοματοποιηθούν, όπως και μερικά επαγγέλματα με συνθήκες δυσμενείς για τον άνθρωπο. Ως παραδείγματα αναφέρονται οι μειώσεις στα υπαλληλικά επαγγέλματα που προκλήθηκαν από τους επεξεργαστές κειμένου και τον υπόλοιπο εξοπλισμό εργασίας δεδομένων, η αντικατάσταση εργασιών χειρισμού μηχανών από εξοπλισμό αυτόματου ελέγχου και η εξαφάνιση ορισμένων ειδικών κατηγοριών επαγγελματιών, όπως οι στοιχειοθέτες στα τυπογραφεία. Από την άλλη πλευρά, θα δημιουργηθούν κάποια νέα επαγγέλματα που προϋποθέτουν κυρίως ένα επίπεδο δημιουργικότητας και ευφυΐας, απαραίτητο σε όσους ασχολούνται με υπολογιστές. Οι νέες απαιτήσεις για τεχνικούς και προγραμματιστές ηλεκτρονικών υπολογιστών είναι προφανή παραδείγματα. Είναι λογικό κάποιες ανωμαλίες να είναι αναμενόμενες. Για παράδειγμα, σε πλήρως βιομηχανικούς χώρους είναι απαραίτητο να προσλαμβάνονται ειδικά εκπαιδευμένοι χειριστές για τις περιπτώσεις άμεσης ανάγκης τις οποίες ο υπολογιστής ελέγχου αδυνατεί να αντιμετωπίσει.

Η εξαφάνιση των παλιών επαγγελμάτων και η δημιουργία νέων μπορεί ωστόσο να έχει επίδραση στο επίπεδο εκπαίδευσης και ικανότητας που απαιτούνται για το εργατικό δυναμικό. Άνθρωποι που βρίσκουν ιδιαίτερα δύσκολη την απόκτηση νέων ικανοτήτων και την προσαρμογή σε νέα καθήκοντα μπορεί να επηρεαστούν διαφορετικά από εκείνους που θεωρούν πρόκληση την απόκτηση νέων ικανοτήτων. Μελέτες δείχνουν ότι το εργατικό δυναμικό είναι πιο μορφωμένο από αυτό των περασμένων δεκαετιών, όσον αφορά το επίπεδο τυπικής εκπαίδευσης. Το επίπεδο των γνώσεων που απαιτείται μετά την εισβολή των υπολογιστών φαίνεται να εξαρτάται πολύ από κάθε συγκεκριμένη εφαρμογή και καμία ξεκάθαρη τάση δεν είναι, προς το παρόν τουλάχιστον, ευδιάκριτη.

Άλλη πολύ σημαντική επίδραση των υπολογιστών στην κοινωνία, είναι η διαρκής συγκέντρωση, αποθήκευση και αποδοτική χρησιμοποίηση πληροφοριών. Συχνά αναφέρεται μία έντονη σύγκρουση μεταξύ περιπτώσεων όπου τέτοιες πρακτικές είναι λογικές και επιθυμητές και άλλων όπου το βασικό δικαίωμα του πολίτη στη ζωή παραβιάζεται. Όπως και σε άλλες συγκρουόμενες απαιτήσεις στην κοινωνία μας, είναι απαραίτητη η ισορροπία ανάμεσα στην απεριόριστη χρήση των πληροφοριών από τη μία πλευρά και στην απόλυτη μυστικότητα από την άλλη.

Πάντως η ολική αντικατάσταση των ανθρώπων από μηχανές, θα μπορούσε να αποτελέσει ένα από τα πιθανά σενάρια για το τέλος της ανθρώπινης κοινωνίας. Οι απόγονοι του ανθρώπινου είδους σε μία κοινωνία “μηχανών” θα είναι “ηλεκτρονικά” παιδιά για τα οποία ο άνθρωπος “γονιός” θα “μοχθεί” για την εξέλιξή τους, η οποία εξέλιξη θα έγκειται στο σχεδιασμό και τη βελτίωση της κατασκευής τους. Προς το παρόν, ο διαχωρισμός ανάμεσα στον άνθρωπο και τη μηχανή θα παραμείνει σαφής, καθώς οι υπολογιστές θα εκτελούν συνηθισμένες και επαναληπτικές νοητικές εργασίες και οι άνθρωποι με τις φανερές τους ικανότητες για δημιουργία, καινοτομία και πρωτοτυπία δε θα γίνουν ποτέ περιττοί.

10. Παράρτημα

Ακολουθεί το κομμάτι κώδικα της κλάσης (ClassRobotControl.vb)

```
Option Strict Off
Option Explicit On

Module InputOut32

    Public Class ClassRobotControl
        Public confirmExec As Boolean = False
        Public Value As Short = 0
        Public PortAddress As Short = &H378S
        Public timer As Integer = 0
        Public movementThread As Threading.Thread

        Public Declare Function Inp Lib "inpout32.dll" Alias "Inp32" (ByVal
        PortAddress As Short) As Short
        Public Declare Sub Out Lib "inpout32.dll" Alias "Out32" (ByVal
        PortAddress As Short, ByVal Value As Short)

        Public Function initRobot () As Boolean
            Me.openPort(0)
        End Function

        Public Function executeCommand(ByVal commandString As String) As
        Boolean
            Try
                commandString = UCase(commandString)

                If IsNumeric(Right(commandString, 1)) Then
                    Me.timer = Convert.ToInt16(Right(commandString,1))
                    commandString = Left(commandString, Len(commandString) - 1)
                End If

                If commandString = "OPEN GRIPPER" Then
                    Me.Value = 130
                ElseIf commandString = "CLOSE GRIPPER" Then
                    Me.Value = 3
                ElseIf commandString = "MOVE WRISTL UP FOR " Then
                    Me.Value = 132
                ElseIf commandString = "MOVE WRISTL DOWN FOR " Then
                    Me.Value = 5
                ElseIf commandString = "MOVE ELBOW UP FOR " Then
```

```
Me.Value = 136
ElseIf commandString = "MOVE ELBOW DOWN FOR " Then
Me.Value = 9
ElseIf commandString = "MOVE SHOULDER UP FOR " Then
Me.Value = 144
ElseIf commandString = "MOVE SHOULDER DOWN FOR " Then
Me.Value = 17
ElseIf commandString = "MOVE BASE LEFT FOR " Then
Me.Value = 33
ElseIf commandString = "MOVE BASE RIGHT FOR " Then
Me.Value = 160
ElseIf commandString = "EXIT" Then
Me.Value = 0
Else
Me.Value = -1
End If

If (Me.Value <> -1 And Me.Value <> 0) Then
movementThread = New Threading.Thread(AddressOf move)
movementThread.Start()
ElseIf (Me.Value = -1) Then
Return False
ElseIf (Me.Value = 0) Then
Return False
End If
Catch ex As Exception
MsgBox(ex.Message)
End Try

Return True
End Function

Private Function openPort(ByVal value As Short) As Boolean
Out(Me.PortAddress, value)
Me.confirmExec = True
End Function

Public Sub move(ByVal sender As System.Object)
Me.openPort(Me.Value)
If Me.Value = 130 Or Me.Value = 3 Then
System.Threading.Thread.Sleep(1000)
Else
System.Threading.Thread.Sleep(Me.timer * 1000)
End If
Me.openPort(0)
End Sub
```

```
Private Function readPort() As String
Return CStr(Inp(Me.PortAddress))
End Function
End Class
End Module
```

Ακολουθεί το κομμάτι κώδικα της φόρμας (Form1.vb)

```
Imports System.Net.Sockets
Imports System.Threading
Imports System.IO
Imports System
Imports System.Net.Dns
Imports System.Drawing.Bitmap
Imports CustomControls.IconComboBox
Imports UNOLibs.Net.ClientClass
Imports RapChatLib.RapChatLib

Public Class Form1

#Region "Declarations"

    Dim Listener As New TcpListener(65535)
    Dim Client As New TcpClient
    Dim Message As String = ""
    Dim Listener1 As New TcpListener(65534)
    Dim Client1 As New TcpClient
    Dim Message1 As String = ""
    Dim IPAdd As String
    Dim clnt As New UNOLibs.Net.ClientClass
    Dim clnt2 As New UNOLibs.Net.ClientClass
    Dim WithEvents server As UNOLibs.Net.ServerClass
    Dim WithEvents server2 As UNOLibs.Net.ServerClass

    Dim robot As ClassRobotControl = New ClassRobotControl

#End Region

#Region "Form"

    'auth h synarthsh einai gia thn arxikopoihsh tou programmatos kai
    ths plaketas (kata to fortwma form load)

    Private Sub Form1_Load(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Load
```



```

robot.initRobot()

Me.Location = My.Settings.WindowPosition      'topo8ethsh formas
Me.Size = My.Settings.WindowSize            ' mege8os
txtSound.Text = My.Settings.Audio           'hxos
txtName.Text = My.Settings.Username         'onoma
PicClient.ImageLocation = My.Settings.DisplayPic 'image
picMypic.ImageLocation = My.Settings.DisplayPic 'image
frmSettings.chkSound.CheckState = My.Settings.ChkSound
'chksound

server = New UNOLibs.Net.ServerClass(65533, True,
My.Computer.FileSystem.SpecialDirectories.MyDocuments & "\")
server2 = New UNOLibs.Net.ServerClass(65532, True, "C:\")
Dim ListThread As New Thread(New ThreadStart(AddressOf
Listening)) 'Creates the thread
ListThread.Start() 'Starts the thread

'gia thn euresh ths ip
Dim shostname As String
shostname = System.Net.Dns.GetHostName
Console.WriteLine("Your Machine Name = " & shostname)
'Call Get IPAddress
Console.WriteLine("Your IP = " & GetIPAddress())

Me.Text = "My Computer: " & shostname & " - " & GetIPAddress()

StatusStrip1.Text = ("My Computer: " & shostname & " - " &
GetIPAddress())
NotifyIcon1.Visible = True
Dim me1 As New IconComboItem
Dim ico As New Icon(My.Application.Info.DirectoryPath &
"\Users.ico")
me1.DisplayText = GetIPAddress()
me1.ItemImage = ico
cmbAddress.Items.Add(me1)
My.Application.SaveMySettingsOnExit = True
cmbAddress.SelectedIndex = 0
End Sub

Private Sub Form1_FormClosed(ByVal sender As System.Object, ByVal
e As System.Windows.Forms.FormClosedEventArgs) Handles
 MyBase.FormClosed
    NotifyIcon1.Visible = False
End Sub

'kleisimo formas

```

```
Private Sub Form1_FormClosing(ByVal sender As System.Object, ByVal
e As System.Windows.Forms.FormClosingEventArgs) Handles
MyBase.FormClosing
    Dim offlinestring As String = " Has gone Offline. You can No
longer Message This User."
    If RichTextBox1.Text.Contains(offlinestring) Then
        End
    Else
        'txtmessage.Text = txtName.Text & " Has gone Offline. You
can No longer Message This User."
        btnSend.PerformClick()

        My.Settings.WindowPosition = Me.Location
        My.Settings.WindowSize = Me.Size
        My.Settings.Username = txtName.Text
        My.Settings.DisplayPic = PicClient.ImageLocation
        My.Settings.Audio = txtSound.Text
        My.Settings.Save()

    End If
End Sub

#End Region

#Region "Functions"

Public Shared Function GetIPAddress() As String
    Dim oAddr As System.Net.IPAddress
    Dim sAddr As String
    With
System.Net.Dns.GetHostByName(System.Net.Dns.GetHostName())
        oAddr = New System.Net.IPAddress(.AddressList(0).Address)
        sAddr = oAddr.ToString
    End With
    GetIPAddress = sAddr
End Function

Private Sub Listening()
    Listener.Start()
    Listener1.Start()
End Sub

#End Region

#Region "System Programming"
```

```

Private Sub NotifyIcon1_MouseDoubleClick(ByVal sender As System.Object, ByVal e As System.Windows.Forms.MouseEventArgs) Handles NotifyIcon1.MouseDoubleClick
    Me.Show()
End Sub

Private Sub Timer1_Tick(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Timer1.Tick

    If Listener.Pending = True Then
        Message = ""
        Client = Listener.AcceptTcpClient()

        Dim Reader As New StreamReader(Client.GetStream())
        While Reader.Peek > -1
            Message += Convert.ToChar(Reader.Read()).ToString
        End While

        Dim stringLength() As String = Split(Message, ":", 2)

        If robot.executeCommand(stringLength(1).Trim()) Then
            RichTextBox1.ForeColor = Color.Black
            RichTextBox1.Text += "[" & Date.Now & "]" + stringLength(0).Trim() + ": COMMAND " + stringLength(1).Trim() + " Success" + vbCrLf
        Else
            RichTextBox1.ForeColor = Color.Red
            RichTextBox1.Text += "[" & Date.Now & "]" + stringLength(0).Trim() + ": COMMAND " + stringLength(1).Trim() + " Wrong Syntax" + vbCrLf
        End If

        If frmSettings.chkSound.CheckState = CheckState.Unchecked
Then
            NotifyIcon1.BalloonTipTitle = ("New Message")
            NotifyIcon1.BalloonTipText = Message
            NotifyIcon1.BalloonTipIcon = ToolTipIcon.Info
            frmMessage.Show()
            frmMessage.lblMessage.Text = Message
            lblStat.Text = ("Last Message Received At " & My.Computer.Clock.LocalTime)
        Else
            NotifyIcon1.BalloonTipTitle = ("New Message")
            NotifyIcon1.BalloonTipText = Message
            NotifyIcon1.BalloonTipIcon = ToolTipIcon.Info
            frmMessage.Show()
            frmMessage.lblMessage.Text = Message
            lblStat.Text = ("Last Message Received At " & My.Computer.Clock.LocalTime)
        End If
    End If
End Sub

```

```

        End If
    End If

    If Listener1.Pending = True Then
        Message1 = ""
        Client1 = Listener1.AcceptTcpClient()

        Dim Reader1 As New StreamReader(Client1.GetStream())
        While Reader1.Peek > -1
            Message1 = Message1 +
            Convert.ToChar(Reader1.Read()).ToString
        End While
        PicClient.Image = StringToBitmap(Message1)
    End If
End Sub

Private Sub btnSend_Click(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles btnSend.Click
    If txtName.Text = "" Or cmbAddress.Text = "" Then
        MessageBox.Show("All Fields must be Filled", "Error
Sending Message", MessageBoxButtons.OK, MessageBoxIcon.Error)
    Else
        'robot.executeCommand(txtmessage.Text)

        btnImage.PerformClick()
        Try
            Client = New TcpClient(cmbAddress.Text, 65535)
            Dim Writer As New StreamWriter(Client.GetStream())

            Writer.Write(txtName.Text & ": " & txtmessage.Text)
            Writer.Flush()
            RichTextBox1.AppendText("[ " & Date.Now & " ] " &
txtName.Text & ": " & txtmessage.Text & vbCrLf)
            RichTextBox1.SelectionStart = RichTextBox1.TextLength
            RichTextBox1.Focus()
            RichTextBox1.ScrollToCaret()

            txtmessage.Text = ""
            txtmessage.Focus()
        Catch ex As Exception
            Console.WriteLine(ex)
            Dim Errorresult As String = ex.Message
            MessageBox.Show(Errorresult & vbCrLf & vbCrLf &
"Please Review Client Address", "Error Sending Message",
MessageBoxButtons.OK, MessageBoxIcon.Error)
        End Try
    End If
End Sub

```

```
Private Sub btnConnect_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnConnect.Click
    If cmbAddress.Text.Length < 4 Then
        MessageBox.Show("Please Enter a Valid Address", "Error",
        MessageBoxButtons.OK, MessageBoxIcon.Error)
    Else
        btnConnect.Text = "Connecting"
        Dim pingresult As String =
        My.Computer.Network.Ping(cmbAddress.Text)
        If pingresult = "True" Then
            btnConnect.Text = "Connected"
        Else
            btnConnect.Text = "Disconnected"
        End If
    End If
End Sub

Private Sub btnClear_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnClear.Click
    txtmessage.Text = ""
    lblStat.Text = "Last Message Received At: "
End Sub

Private Sub txtmessage_TextChanged(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles txtmessage.TextChanged
    Dim imagepath As String = My.Application.Info.DirectoryPath &
"\Message.png"
    Dim img As Image = Image.FromFile(imagepath)
    lblStat.Image = img
    lblStat.Text = "Writing Message..."

    If txtmessage.Text <> "" Then
        btnClear.Enabled = True
        btnSend.Enabled = True
    Else
        btnClear.Enabled = False
        btnSend.Enabled = False
    End If
End Sub

Private Sub cmbAddress_TextChanged(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles cmbAddress.TextChanged
    btnConnect.Text = "Connect..."
End Sub
```

```
Public Function StringToBitmap(ByVal sImageData As String) As
Bitmap
    Try
        Dim ms As New
MemoryStream(Convert.FromBase64String(sImageData))
        Dim bmp As Bitmap = Bitmap.FromStream(ms)
        Return bmp
    Catch ex As Exception
        Return Nothing
    End Try
End Function

Private Sub ImageButton2_Click(ByVal sender As System.Object,
ByVal e As System.EventArgs)
    btnConnect.PerformClick()
End Sub

Private Sub ImagebtnConnect_Click(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles ImageButton1.Click
    cmbAddress.Text = ""
    txtName.Text = "NickName"
    txtmessage.Text = ""
    RichTextBox1.Text = ""
End Sub

Private Sub btnImage_Click(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles btnImage.Click
    clnt2.SendMessage(cmbAddress.Text, 65532, TextBox1.Text)
End Sub

Private Sub txtName_TextChanged(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles txtName.TextChanged
    Me.Text = ("RobotArm Controller - " & txtName.Text)
End Sub

#Region "Menu Bottom"
Private Sub HideToolStripMenuItem_Click(ByVal sender As
System.Object, ByVal e As System.EventArgs) Handles
HideToolStripMenuItem.Click
    Me.Hide()
End Sub

Private Sub NetworkStatusToolStripMenuItem_Click(ByVal sender As
System.Object, ByVal e As System.EventArgs) Handles
NetworkStatusToolStripMenuItem.Click
    btnBrowse.PerformClicked()
End Sub
```

```
Private Sub SettingsToolStripMenuItem_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
    frmSettings.Show()
End Sub

Private Sub FontToolStripMenuItem_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles FontToolStripMenuItem.Click
    fntFont.ShowDialog()
    If DialogResult.OK Then
        RichTextBox1.Font = fntFont.Font
    End If
End Sub

Private Sub ExitToolStripMenuItem_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles ExitToolStripMenuItem.Click
    End
End Sub
#End Region

Private Sub btnBrowse_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnBrowse.Click
    frmMyData.Show()
End Sub

Private Sub txtmessage_Leave(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles txtmessage.Leave
    lblStat.Text = "Last Message Received At:"
End Sub

Private Sub PicClient_DoubleClick(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles PicClient.DoubleClick
    frmNewAddress.Show()
End Sub

Private Sub OnDiagnosticMessage(ByVal Args As String) Handles server.DiagnosticMessage
    RichTextBox1.Text += (txtName.Text & " - " & Args) + vbCrLf
    'My.Computer.Audio.Play(My.Application.Info.DirectoryPath &
    "\Sounds\Notify.wav")
End Sub

Private Sub OnIncomingMessage(ByVal Args As UNOLibs.Net.ServerClass.InMessEvArgs) Handles server2.IncomingMessage
    PicClient.Image = StringToBitmap(Args.message)
End Sub
```

```
Private Sub FontColourToolStripMenuItem_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
    clrFont.ShowDialog()
    If DialogResult.OK Then
        RichTextBox1.ForeColor = clrFont.Color
    Else
        RichTextBox1.ForeColor = Color.Black
    End If
End Sub

Private Sub ExitToolStripMenuItem2_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
    End
End Sub

Private Sub ShowToolStripMenuItem1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
    Me.Show()
End Sub

Private Sub SettingsToolStripMenuItem1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
    frmSettings.Show()
End Sub

Private Sub SettingsToolStripMenuItem2_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
    frmSettings.Show()
End Sub

Private Sub RichTextBox1_MouseMove(ByVal sender As System.Object, ByVal e As System.Windows.Forms.MouseEventArgs) Handles RichTextBox1.MouseMove
    If RichTextBox1.SelectionLength > 0 Then
        cmsmCut.Enabled = True
        cmsmCopy.Enabled = True
    Else
        cmsmCut.Enabled = False
        cmsmCopy.Enabled = False
    End If
End Sub

Private Sub SelectAllToolStripMenuItem_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles SelectAllToolStripMenuItem.Click
    RichTextBox1.SelectAll()
End Sub
```



```
End Sub

#End Region

Private Sub AboutRobotArmControllerToolStripMenuItem_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
    Process.Start(My.Application.Info.DirectoryPath & "\About RobotArmController\About RobotArmController.exe")
End Sub

End Class
```

11. Βιβλιογραφία

11.1. Ιστορικές Πηγές

- **Ο μύθος του Δαίδαλου**

www.nikaria.gr/pages/Δαίδαλος.html

- **Αυτόματοι μηχανισμοί Ηφαίστου και Δαίδαλου**

<http://thenetwar.com/2009/10/3189/>

- **Το ρομπότ Τάλως**

<http://el.shvoong.com/books/mythology-ancient-literature/1746502-το-ρομπότ-τάλως/>

- **Αυτόνομα, ευφυή ρομποτικά συστήματα**

<http://www.akazoo.gr/Forum/Forum.aspx?node=forum&g=posts&t=661>

- **Ισαάκ Ασίμωφ**

http://el.wikipedia.org/wiki/Ισαάκ_Ασίμωφ

- **Οι τρεις νόμοι της ρομποτικής**

http://el.wikipedia.org/wiki/Τρεις_νόμοι_της_ρομποτικής

11.2. Πηγές Σχετικές με τις Θύρες Επικοινωνίας του Υπολογιστή

- **Η παράλληλη θύρα**

http://en.wikipedia.org/wiki/Parallel_port

- **Η USB θύρα**

<http://en.wikipedia.org/wiki/Usb>

- **Η σειριακή θύρα (γενικές πληροφορίες)**

http://en.wikipedia.org/wiki/Serial_port

- **Η θύρα RS-232**

http://en.wikipedia.org/wiki/RS_232

- **Η θύρα FireWire**

<http://en.wikipedia.org/wiki/FireWire>

- **Ethernet**

<http://en.wikipedia.org/wiki/Ethernet>

- **Η USB 3.0, Άρθρο του μηνιαίου περιοδικού “PC WORLD”, τεύχους Αυγούστου 2009**

http://www.pcw.gr/Article/PCs-Notebooks/USB3_high_speed_trasfer_data_system_on_chip/236-4514.html

- **Δημιουργία σημάτων εξόδου στην παράλληλη θύρα του PC, ΚΕΝΤΡΟ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΝΕΩΝ ΤΕΧΝΟΛΟΓΙΩΝ ΗΛΕΙΑΣ (ΚΕΠΛΗΝΕΤ ΗΛΕΙΑΣ)**

<http://dide.ilei.sch.gr/keplinet/programming/parallel.php>

- **Οι θύρες επικοινωνίας στον υπολογιστή (γενικές πληροφορίες)**

<http://www.texnologia.gr/index.asp?mod=articles&id=136>

11.3. Πηγές Σχετικές με τα Δίκτυα Υπολογιστών

- **Τοπικά Δίκτυα Υπολογιστών**

Σημειώσεις μαθήματος “Εισαγωγή στην Πληροφορική”, ΣΤΕΦ Τμήμα Πληροφορικής, ΑΤΕΙ Θεσσαλονίκης, Ηλιούδης Χρήστος, 2006

- **Το Internet**

Σημειώσεις μαθήματος “Εισαγωγή στο Internet”, Πανεπιστήμιο Μακεδονίας, Τμήμα Οικονομικών και Κοινωνικών Σπουδών, Κέντρο Υπολογιστών και Δικτύων, 2007

- **Τι είναι ο Client-Server;**

Σημειώσεις μαθήματος “Δίκτυα Υπολογιστών”, Τμήμα Αυτοματισμού, ΑΤΕΙ Πειραιά, Αγγελόπουλος Ιωάννης, 2004

- **Client-Server (γενικές πληροφορίες)**

<http://en.wikipedia.org/wiki/Client-server>

- **Γενικές πληροφορίες για τα δίκτυα υπολογιστών**

“Computer Networks, fourth edition,2003”, του συγγραφέα Andrew Tanenbaum, εκδότης: Pearson Education Inc, ISBN:0130661023

11.4. Πηγές Σχετικές με την Τηλεματική

- **Γενικές πληροφορίες για την Τηλεματική**

<http://egnatia.ee.auth.gr/~aalexioy/telematique.htm>

11.5. Άρθρα και Γενικές Πληροφορίες για τη Ρομποτική

- **Ρομπότ - δασκάλα**

<http://www.protothema.gr/article/?aid=32375>

- **Τα ρομπότ στην υπηρεσία της ιατρικής**

<http://www.protothema.gr/greece/article/?aid=11915>

- **Έξυπνα ρομπότ στις διαταγές μας**

<http://www.tovima.gr/default.asp?pid=46&ct=50&artId=269906&dt=23/08/2009>

- **Πότε κατασκευάστηκε το πρώτο ρομπότ**

<http://www.focusmag.gr/articles/view-article.rx?oid=185275>

- **Ρομπότ βοηθό από το Stanford University**

<http://medgreece.gr/2008/ένα-βήμα-πιο-κοντά-στα-ρομπότ-βοηθούς-απο-το-Stanford-University>

- **Ρομποτικό σύστημα διαχείρισης απορριμμάτων στη Θεσσαλονίκη**

http://portal.kathimerini.gr/4dcgi/_w_articles_kathbreak_1_03/07/2008_239602

- **What is robotic surgery?**

<http://www.topbits.com/robotic-surgery.html>

- **Ρομποτική (γενικές πληροφορίες)**

<http://en.wikipedia.org/wiki/Robotics>

- **Τα Lego Mindstorms (γενικές πληροφορίες και εφαρμογές τους)**

http://en.wikipedia.org/wiki/Lego_Mindstorms

- **Ο ρομποτικός βραχίονας της εργασίας**

http://www.smarttoys.gr/product_info.php?products_id=3585

11.6. Πληροφορίες για την Πλακέτα ΚΕΜΟ Β-210

<http://www.kemo-electronic.de/en/Transformer-Dimmer/Switches/Kits/B210-8-Channel-relay-board.php>

11.7. Πηγές Σχετικές με την Επιστήμη των Υπολογιστών

- **“ΕΙΣΑΓΩΓΗ ΣΤΗΝ ΕΠΙΣΤΗΜΗ ΤΩΝ ΥΠΟΛΟΓΙΣΤΩΝ”,**

συγγραφέας Σιδερίδης Αλέξανδρος, Αυτοέκδοση του συγγραφέα, ISBN: 960858471X

- **“Εισαγωγή στη σύγχρονη επιστήμη των υπολογιστών”,**

των συγγραφέων Lister, Andrew M., εκδότης: Δίαυλος, ISBN: 9607140656

- **“ΕΙΣΑΓΩΓΗ ΣΤΗ ΡΟΜΠΟΤΙΚΗ”,**

συγγραφέας CRAIG. J.JOHN, εκδότης: Τζιόλας, ISBN: 9604181602

11.8. Πηγές Προγραμματισμού στη Visual Basic 2008 .NET

- <http://www.vbdotnetheaven.com>

- <http://www.vbtutor.net/vb2008/vb2008tutor.html>

- <http://www.developerfusion.com/tag/vb.net/>
- **“Professional Visual Basic 2008”**,
των συγγραφέων Bill, Evjen, Billy, Hollis, Bill, Sheldon, Kent, Sharkey, εκδότης: John Wiley and Sons Ltd, ISBN: 0470191368
- **“VISUAL BASIC 2008 ΒΗΜΑ ΒΗΜΑ”**,
συγγραφέας HALVORSON MICHAEL, εκδότης: ΚΛΕΙΔΑΡΙΘΜΟΣ, ISBN: 9604611496
- **“Visual Basic 2008 for Dummies”**,
συγγραφέας Bill, Sempf, εκδότης: John Wiley and Sons Ltd, ISBN: 0470182385