



Πτυχιακή εργασία

Τμήμα: Πανεπιστήμιο Θεσσαλίας

Σχολή Θετικών Επιστημών

Τμήμα Πληροφορικής και Τηλεπικοινωνιών

**Κατασκευή ρομποτικού βραχίονα με δυνατότητα ασύρματου
ελέγχου και εκτέλεσης αλγορίθμων ταξινόμησης**

Φοιτητής: Μπουρνάζος Χρήστος

Επιβλέπων καθηγητής: Κολομβάτσος Κωνσταντίνος

Λαμία 2021



Bachelor Thesis

Department: University of Thessaly

School of Science

Department of Computer Science and Telecommunications

**Design of a robotic arm with wireless control capabilities and
execution of sorting algorithms**

Student: Bournazos Christos

Advisor: Kolomvatsos Konstantinos

Lamia 2021

ΕΥΧΑΡΙΣΤΙΕΣ

Σε αυτό το σημείο θέλω να ευχαριστήσω τον καθηγητή μου Κολομβάτσο Κωνσταντίνο, για την στήριξη του καθ' όλη τη διάρκεια της πτυχιακής με τις συμβουλές και την βοήθεια του.

Ακόμα θα ήθελα να ευχαριστήσω την οικογένεια μου για την στήριξή της όλα αυτά τα χρόνια .

ΔΗΛΩΣΗ ΣΥΓΓΡΑΦΕΑ ΠΤΥΧΙΑΚΗΣ ΕΡΓΑΣΙΑΣ

«Με ατομική μου ευθύνη και γνωρίζοντας τις κυρώσεις (1), που προβλέπονται από της διατάξεις της παρ. 6 του άρθρου 22 του Ν. 1599/1986, δηλώνω ότι:

1. Δεν παραθέτω κομμάτια βιβλίων ή άρθρων ή εργασιών άλλων αυτολεξεί χωρίς να τα περικλείω σε εισαγωγικά και χωρίς να αναφέρω το συγγραφέα, τη χρονολογία, τη σελίδα. Η αυτολεξεί παράθεση χωρίς εισαγωγικά χωρίς αναφορά στην πηγή, είναι λογοκλοπή. Πέραν της αυτολεξεί παράθεσης, λογοκλοπή θεωρείται και η παράφραση εδαφίων από έργα άλλων, συμπεριλαμβανομένων και έργων Συμφοιτητών μου, καθώς και η παράθεση στοιχείων που άλλοι συνέλεξαν ή επεξεργάστηκαν, χωρίς αναφορά στην πηγή. Αναφέρω πάντοτε με πληρότητα την πηγή κάτω από τον πίνακα ή σχέδιο, όπως στα παραθέματα.

2. Δέχομαι ότι η αυτολεξεί παράθεση χωρίς εισαγωγικά, ακόμα κι αν συνοδεύεται από αναφορά στην πηγή σε κάποιο άλλο σημείο του κειμένου ή στο τέλος του, είναι αντιγραφή. Η αναφορά στην πηγή στο τέλος π.χ. μιας παραγράφου ή μιας σελίδας, δεν δικαιολογεί συρραφή εδαφίων έργου άλλου συγγραφέα, έστω και παραφρασμένων, και παρουσίασή τους ως δική μου εργασία.

3. Δέχομαι ότι υπάρχει επίσης περιορισμός στο μέγεθος και στη συχνότητα των παραθεμάτων που μπορώ να εντάξω στην εργασία μου εντός εισαγωγικών. Κάθε μεγάλο παράθεμα (π.χ. σε πίνακα ή πλαίσιο, κλπ), προϋποθέτει ειδικές ρυθμίσεις, και όταν δημοσιεύεται προϋποθέτει την άδεια του συγγραφέα ή του εκδότη. Το ίδιο και οι πίνακες και τα σχέδια

4. Δέχομαι όλες τις συνέπειες σε περίπτωση λογοκλοπής ή αντιγραφής.

(1) «Όποιος εν γνώσει του δηλώνει ψευδή γεγονότα ή αρνείται ή αποκρύπτει τα αληθινά με έγγραφη υπεύθυνη δήλωση του άρθρου 8 παρ. 4 Ν. 1599/1986 τιμωρείται με φυλάκιση τουλάχιστον τριών μηνών. Εάν ο υπαίτιος αυτών των πράξεων σκόπευε να προσπορίσει στον εαυτόν του ή σε άλλον περιουσιακό όφελος βλάπτοντας τρίτον ή σκόπευε να βλάψει άλλον, τιμωρείται με κάθειρξη μέχρι 10 ετών.»

ΠΕΡΙΛΗΨΗ

Η τεχνολογική εξέλιξη των τελευταίων δεκαετιών ξεπέρασε τις προσδοκίες και εστίασε στην βελτίωση της καθημερινότητας. Ο άνθρωπος έδωσε βάση στην ανάπτυξη τεχνικών και κλάδων με σκοπό την διευκόλυνση της ζωής του και την αυτοματοποίηση διαδικασιών τόσο στον εργασιακό τομέα όσο και στην διαβίωση. Χαρακτηριστικό παράδειγμα τέτοιου είδους επιστήμης αποτελεί η ρομποτική . Η ρομποτική , είναι κλάδος της τεχνολογίας , που ασχολείται με τη σχεδίαση, την ανάπτυξη και τη μελέτη ρομπότ. Αποτελεί συνδυασμό πολλών άλλων επιστημών, κυρίως της πληροφορικής , της ηλεκτρονικής και της μηχανολογίας. Η επιστήμη αυτή ,έχει κάνει άλματα προόδου και έχει προσφέρει αρκετά τεχνολογικά θαύματα. Τα ρομποτικά συστήματα συνεχώς εξελίσσονται και είναι ήδη μέρος της ζωής μας σε πολλούς τομείς όπως στη βιομηχανία, την ιατρική, τη διασκέδαση και την προσωπική βοήθεια. Με βάση τα παραπάνω , δημιουργήθηκε η παρούσα εργασία , η οποία έχει ως σκοπό την μελέτη ,τον σχεδιασμό και την κατασκευή ενός ρομποτικού βραχίονα ,5 σημείων ελευθερίας με την δυνατότητα εκτέλεσης αλγορίθμων ταξινόμησης και χωρίζεται σε τέσσερα διακριτά τμήματα.

Το πρώτο μέρος , περιλαμβάνει τον σχεδιασμό ,την κατασκευή και τον εξοπλισμό που χρησιμοποιήθηκε ,για την δημιουργία των βασικών τμημάτων του ρομποτικού βραχίονα, καθώς και αναφορά στο λογισμικό που χρησιμοποιήθηκε κατά την διάρκεια.

Το δεύτερο μέρος αποτελείται από αναφορές στον ηλεκτρονικό εξοπλισμό ,όπως μοτέρ και αισθητήρες ,τον μικροελεγκτή ,την τροφοδοσία και το βασικό πρόγραμμα του ελεγκτή που υλοποιήθηκε.

Στο τρίτο τμήμα εξετάζεται το λογισμικό που χρησιμοποιήθηκε για την δημιουργία του προγράμματος ασύρματου ελέγχου του ρομποτικού βραχίονα ,όπως επίσης και η ανάλυση αυτού.

Κλείνοντας ,γίνεται η ενοποίηση των παραπάνω και παρατίθενται τα συμπεράσματα που προκύπτουν και οι προοπτικές εξέλιξης της κατασκευής.

Λέξεις-κλειδιά: ρομποτικός βραχίονας , μικροελεγκτής

ABSTRACT

The technological development of the last decades exceeded the expectations and focused on the improvement of everyday life. Man laid the foundation for the development of techniques in order to facilitate his life and the automation of processes in both the workplace and everyday life. An example of this kind of science is robotics. Robotics is a branch of technology that deals with the design, development and study of robots .It is a combination of many sciences, mainly computer science, electronics and engineering. This science has made huge leaps as far as progress is concerned and has offered many technological marvels. Robotic systems are constantly evolving and are already part of our lives in many areas such as the industry, medicine, entertainment and personal assistance. Based on the above, the present work was created, which aims to study, design and build a robotic arm with five points of freedom and the ability to execute classification algorithms and is divided into four distinct sections.

The first part consists of the design, construction and the equipment used to create the basic parts of the robotic arm, as well as a reference to the software that we used.

The second part consists of references to the electronic equipment, such as motors and sensors, the microcontroller, the power supply and also the basic program of the controller that was implemented.

The third section examines the software used to create the robotic arm wireless control program, as well as an analysis of the software.

Finally we construct the robotic arm , present the conclusion that emerged and the development prospects of the construction.

Keywords: robotic arm , microcontroller

Περιεχόμενα

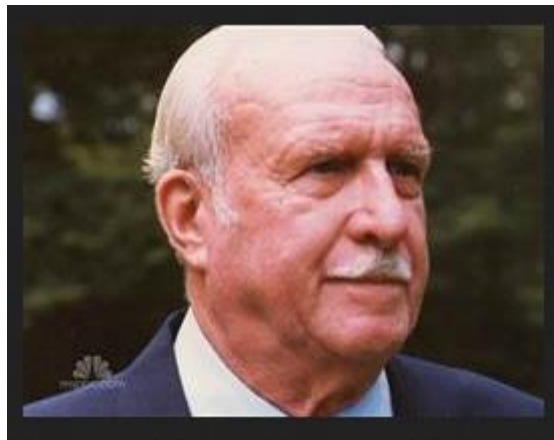
1. Εισαγωγή	1
1.1. Ιστορική αναδρομή της ρομποτικής	1
1.2. Η έννοια του ρομπότ	2
1.3. Τί είναι η Βιομηχανική Ρομποτική	3
1.4. Τμήματα ενός ρομποτικού βραχίονα	4
1.5. Τι είναι το IoT	6
1.6. Κύρια μέρη του IoT	8
1.7. Παραδείγματα IoT	10
2. Κατασκευή και εκτύπωση των τμημάτων του ρομποτικού βραχίονα	13
2.1. Το λογισμικό Solidworks	13
2.1.1. Τι είναι το Solidworks	13
2.1.2. Κατασκευή των στοιχείων του βραχίονα σε Solidworks	14
2.2. Τρισδιάστατη εκτύπωση	25
2.2.1. Τί είναι η τρισδιάστατη εκτύπωση	26
2.2.2. Ο τρισδιάστατος εκτυπωτής	26
2.2.3. Το λογισμικό Ultimaker Cura	27
2.2.4. Εισαγωγή σχεδίων στο Cura	28
2.2.5. Εκτύπωση των σχεδίων του βραχίονα	31
3. Ηλεκτρονικός εξοπλισμός και Βασικό Πρόγραμμα	34
3.1. Εισαγωγή στις πλακέτες IoT	34
3.1.1. Πλακέτες IoT	34
3.1.2. Εισαγωγή στο Arduino	37
3.1.3. Λογισμικό προγραμματισμού του Arduino	40
3.2. Αισθητήρες και μοτέρ	42
3.2.1. Πλακέτα HC – 05	43
3.2.2. Μοτέρ τύπου SG-90	46
3.2.3. Μοτέρ τύπου MG-996R	48
3.2.4. Αισθητήρας χρώματος τύπου TCS3200	49
3.3. Τροφοδοσία	53

3.3.1. Το κύκλωμα στο Arduino.....	54
3.3.2. Τροφοδοσία του ηλεκτρονικού εξοπλισμού.....	55
3.4. Αλγόριθμοι ταξινόμησης.....	55
3.4.1. Τι είναι οι αλγόριθμοι ταξινόμησης	55
3.4.2. Ο αλγόριθμος bubble sort.....	56
3.4.3. Ο αλγόριθμος selection sort.....	57
3.5. Υλοποίηση του προγράμματος σε Arduino.....	59
3.5.1. Κώδικας που εγκαταστάθηκε στο Arduino	59
4. Δημιουργία της εφαρμογής ασύρματου ελέγχου για Android	64
4.1. Το Android studio	64
4.1.1. Τί είναι το Android studio	64
4.1.2. Ανάλυση των τμημάτων μιας εφαρμογής σε Android studio.....	64
4.2. Κατασκευή του Προγράμματος.....	66
4.2.1. Κώδικας της εφαρμογής	66
4.2.2. Δημιουργία αρχείου apk για εγκατάσταση σε Android.....	72
5. Ολοκλήρωση της κατασκευής.....	74
5.1. Συναρμολόγηση των επιμέρους στοιχείων.....	74
5.2. Σύνοψη-Συμπεράσματα-Βελτιώσεις	76
6. Βιβλιογραφία.....	77

1. Εισαγωγή

1.1. Ιστορική αναδρομή της ρομποτικής

Τα πρώτα ρομπότ κατασκευάστηκαν πολλά χρόνια πριν. Ο George Devol ήταν ο πρώτος που αιτήθηκε τα διπλώματα ευρεσιτεχνίας για ρομπότ το 1954 . Ο Devol μαζί με τον Joseph F Engelberger το 1956 , δημιούργησαν την πρώτη εταιρία παραγωγής ρομπότ με το όνομα Unimation. Τα ρομπότ αυτά , που βασίζονταν στο δίπλωμα ευρεσιτεχνίας του Devol , ονομαζόταν επίσης και μηχανές προγραμματισμένων μεταφορών, λόγω της λειτουργίας τους ,που ήταν η μεταφορά αντικειμένων από ένα σημείο σε κάποιο άλλο. Ο μόνος ανταγωνιστής της Unimation ήταν η Cincinnati Milacron Inc.. Αυτό αργότερα άλλαξε, όταν πολλοί Ιαπωνικοί όμιλοι ξεκίνησαν την παραγωγή παρόμοιων βιομηχανικών ρομπότ. [1]



Εικόνα 1.1 Ο George Devol [1]

Αργότερα , ο Victor Scheinman , δημιούργησε το βραχίονα του Στάνφορντ, έναν ηλεκτρικό ρομποτικό άξονα που σχεδιάστηκε για να καταστεί δυνατή η λύση του βραχίονα . Αυτό διεύρυνε τις δυνατότητες χρήσης του ρομπότ σε πιο εξελιγμένες εφαρμογές, όπως η συναρμολόγηση και συγκόλληση. Ο ίδιος, σχεδίασε κι ένα δεύτερο βραχίονα για το εργαστήριο τεχνητής νοημοσύνης του MIT . [2]

Το ενδιαφέρον στη ρομποτική αυξήθηκε στα τέλη του 1970 όταν πολλές εταιρείες των ΗΠΑ εισήλθαν στον τομέα όπως η General Electric. Στις πρωτοπόρες εταιρίες ανήκουν οι Automatrix και η Adept Technology . Αργότερα η Unimation εξαγοράστηκε από την Westinghouse Electric Corporation η οποία με την σειρά της πούλησε την Unimation στην γαλλική Stäubli Faverges SCA .Η Stäubli Faverges SCA έπειτα αγόρασε και το ρομποτικό τμήμα της Bosch .

1.2. Η έννοια του ρομπότ



Εικόνα 1.2 Τα ρομπότ στη βιομηχανία

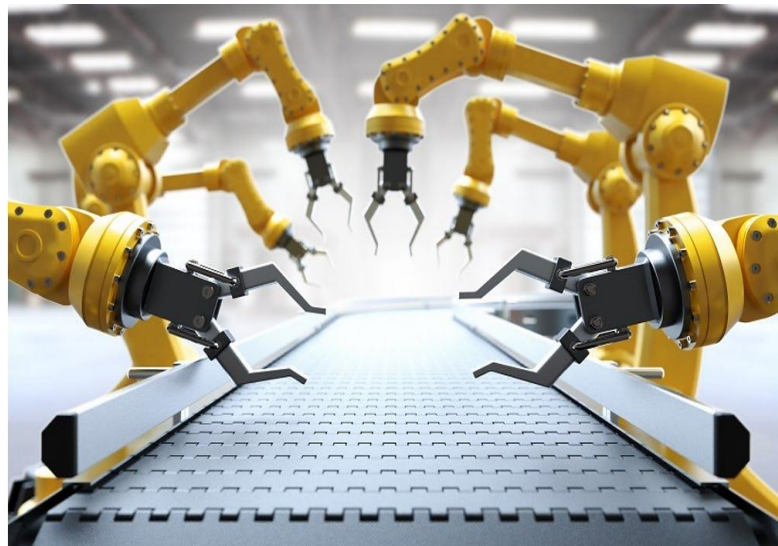
Σύμφωνα με τον ορισμό του Ινστιτούτου Ρομπότ των ΗΠΑ, “ρομπότ είναι μια επαναπρογραμματιζόμενη πολυλειτουργική χειριστική διάταξη, σχεδιασμένη για τη μετακίνηση υλικών, εξαρτημάτων, εργαλείων και εξειδικευμένων διατάξεων, μέσω μεταβλητών και προγραμματισμένων κινήσεων , για την εκτέλεση μιας σειράς εργασιών” . [3]

Τα ρομπότ συγκροτούνται από το μηχανικό και το ηλεκτρικό σύστημα. Στο πρώτο ανήκει το σύστημα κίνησης και στο δεύτερο η μνήμη του , που έχει

δυνατότητες επαναπρογραμματισμού . Υπάρχουν πολλοί τρόποι διάκρισης των ρομπότ. Μία από αυτές είναι η κατηγοριοποίησή τους σε τρεις γενιές. [4]

- I. Στην πρώτη γενιά κατατάσσονται ρομπότ που ελέγχονται από τον άνθρωπο. Οι χειριστές που είναι υπεύθυνοι για την λειτουργία του ρομπότ, τα χρησιμοποιούν για εργασίες όπως η μετακίνηση επικίνδυνων αντικειμένων και ραδιενεργών υλικών.
- II. Η δεύτερη γενιά αποτελείται από τα ρομπότ που διαθέτουν σταθερό πρόγραμμα δράσης και ρομπότ που λαμβάνουν εντολές από κάποιο σύστημα ελέγχου.
- III. Τέλος στην τρίτη γενιά κατατάσσονται ρομπότ που είναι ικανά να αντιλαμβάνονται διάφορες αλλαγές στο περιβάλλον, μέσω των αισθητήρων που διαθέτουν και να εκτελούν εργασίες ανάλογα τις μεταβολές αυτές.

1.3. Τί είναι η Βιομηχανική Ρομποτική



Εικόνα 1.3 Βιομηχανικοί ρομποτικοί βραχίονες

Η Βιομηχανική Ρομποτική αποτελεί έναν κλάδο της Ρομποτικής που ασχολείται με τις εφαρμογές των ρομπότ στην βιομηχανία και έχει εξελιχθεί πολύ

από την στιγμή της δημιουργία του. Η μεγάλη ανάγκη για βιομηχανικά ρομπότ είχε ως αποτέλεσμα , να διαχωριστεί αρκετά από τον κλάδο της Ρομποτικής και αποτελεί πλέον έναν ανεξάρτητο επιστημονικό κλάδο. Υπάρχουν πολλές εταιρίες που εξειδικεύονται αποκλειστικά στην ανάπτυξη βιομηχανικών ρομπότ, όπως επίσης και οργανισμοί οι οποίοι έχουν ως αντικείμενο τους την εξέλιξη και την μελέτη των ρομπότ αυτών. [5]

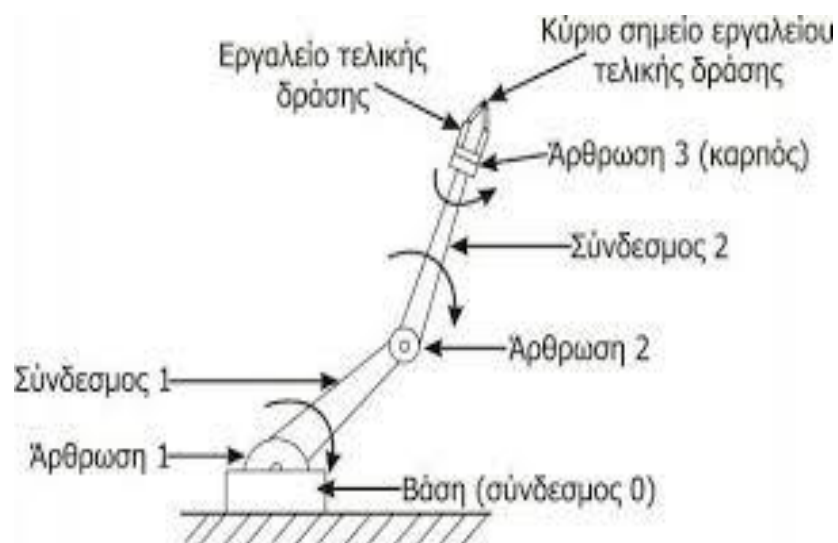
1.4. Τμήματα ενός ρομποτικού βραχίονα

Κάθε τύπος ρομποτικού βραχίονα , χαρακτηρίζεται από τα τμήματα που τον αποτελούν και είναι τα παρακάτω: [6]

- Η βάση του ρομποτικού βραχίονα
 - Οι σύνδεσμοι
 - Οι αρθρώσεις
 - Το εργαλείο τελικής δράσης
 - Το κύριο σημείο του εργαλείου τελικής δράσης
 - Ο βαθμός ελευθερίας
- I. Βάση του ρομποτικού βραχίονα, ονομάζουμε το τμήμα του ρομποτικού βραχίονα που είναι στερεωμένο στο έδαφος ή στο περιβάλλον που το ρομπότ εκτελεί την λειτουργία του. Η βάση είναι ένα σημαντικό τμήμα του ρομποτικού βραχίονα , διότι σε αυτό συνδέεται όλα τα τμήματα των παραπάνω επιπέδων που είναι οι σύνδεσμοι και οι αρθρώσεις.
- II. Οι σύνδεσμοι είναι στερεά σώματα. Αποτελούν τον σκελετό του ρομποτικού βραχίονα.

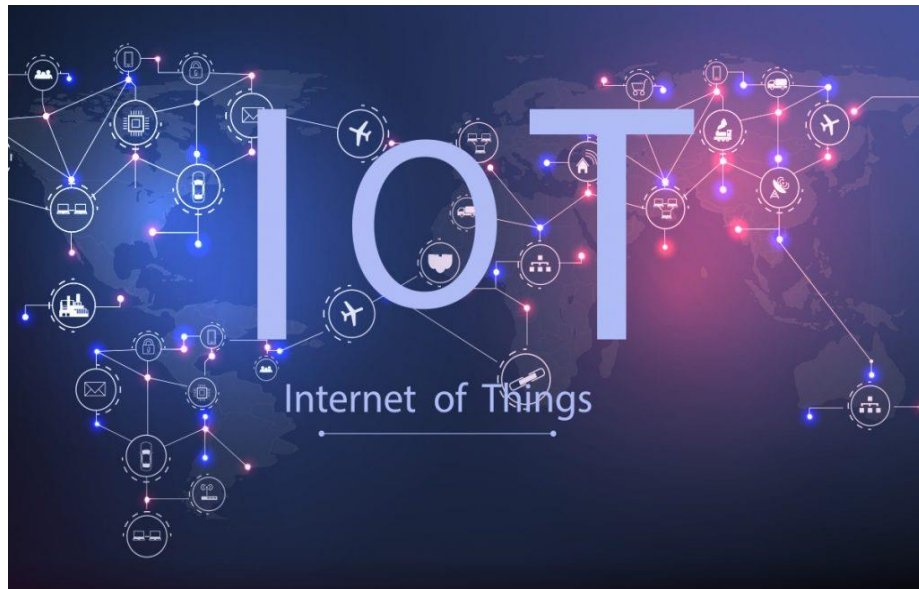
- III. Οι αρθρώσεις είναι μηχανισμοί που επιτρέπουν τη κίνηση μεταξύ των συνδέσμων.
- IV. Ως εργαλείο τελικής δράσης καθίσταται το εργαλείο με το οποίο ο ρομποτικός βραχίονας εκτελεί τις εργασίες που του ανατίθενται. Εργαλείο τελικής δράσης μπορεί να είναι ένα κατσαβίδι , τρυπάνι ,αρπάγη και άλλα.
- V. Κύριο σημείο του εργαλείου τελικής δράσης, ονομάζεται το σημείο του οποίου ,η θέση είναι σημαντική για την σωστή εκτέλεση της εργασίας του ρομποτικού βραχίονα ,όπως για παράδειγμα το σημείο ένωσης των σημείων μιας αρπάγης.
- VI. Τέλος το σύνολο των αρθρώσεων ενός ρομποτικού βραχίονα ,αποτελεί τον βαθμό ελευθερίας αυτού .

Παρακάτω δίνεται μια απεικόνιση ρομποτικού βραχίονα με όλα τα επιμέρους στοιχεία του.



Εικόνα 1.4 Στοιχεία του ρομποτικού βραχίονα [6]

1.5. Τι είναι το IoT



Εικόνα 1.5 IoT

Η έννοια του Internet of Things (IoT) δημιουργήθηκε από τον Kevin Ashton. Ο Ashton ανήκε σε μια ομάδα , που ανακάλυψε τον τρόπο επικοινωνίας των αντικείμενων με το Διαδίκτυο. Το διαδίκτυο των πραγμάτων αποτελεί μια συλλογή από πράγματα στα οποία είναι εγκατεστημένος ηλεκτρονικός εξοπλισμός , λογισμικό και ενεργοποιητές . Το κάθε αντικείμενο της συλλογής , έχει την δυνατότητα σύνδεσης στο διαδίκτυο όπως επίσης και την ικανότητα ανταλλαγής δεδομένων . Οι συσκευές IoT είναι εξοπλισμένες με αισθητήρες , που τις επιτρέπουν να αναπτυχθούν οποιοδήποτε περιβάλλον. [7][8]



Εικόνα 1.6 Το smartwatch

Η έννοια των πραγμάτων όσο αναφορά το IoT , είναι ευρύς και περιλαμβάνει μια ποικιλία φυσικών στοιχείων. Αυτά αποτελούν προσωπικά αντικείμενα και συσκευές που ο κάθε άνθρωπος μπορεί να μεταφέρει καθημερινά , όπως τα έξυπνα τηλέφωνα , τα tablets ,τα smartwatch και γενικά οποιαδήποτε συσκευή που έχει δυνατότητα επεξεργασίας δεδομένων και μπορεί να έχει πρόσβαση στο διαδίκτυο. [9]



Εικόνα 1.7 Το Tablet

Περιλαμβάνει επίσης στοιχεία στο περιβάλλον του ανθρώπου και στις βιομηχανίες που συνδέονται μέσω συσκευής πύλης . Έτσι ,με την παραπάνω έννοια των “πραγμάτων”, στο διαδίκτυο θα μπορεί να συνδεθεί ένας μεγάλος αριθμός συσκευών όπου η κάθε μια θα παρέχει δεδομένα και πληροφορίες και μερικές από αυτές , ακόμη και υπηρεσίες .

Οι συσκευές IoT συλλέγουν δεδομένα από τους αισθητήρες που διαθέτουν . Τα δεδομένα αυτά μπορούν να μοιράζονται μεταξύ άλλων συσκευών με σκοπό είτε την αποστολή τους σε ένα σύννεφο (cloud) είτε να αναλυθούν σε τοπικό επίπεδο.

Συσκευές IoT έχουν την δυνατότητα επικοινωνίας με άλλες IoT συσκευές μέσω ανταλλαγής πληροφοριών . Οι πληροφορίες που στέλνει η μια συσκευή στην άλλη μπορεί να καθορίσει και την συμπεριφορά της κάθε συσκευής σε κάθε περίπτωση . Το μεγαλύτερο τμήμα της εργασίας των συσκευών γίνεται χωρίς την ανθρώπινη παρέμβαση

, παρόλο που οι άνθρωποι μπορούν να αλληλεπιδρούν με τις συσκευές . Παρακάτω παρατίθενται κάποιες εικόνες με συσκευές Ιot.



Εικόνα 1.8 Το smartphone

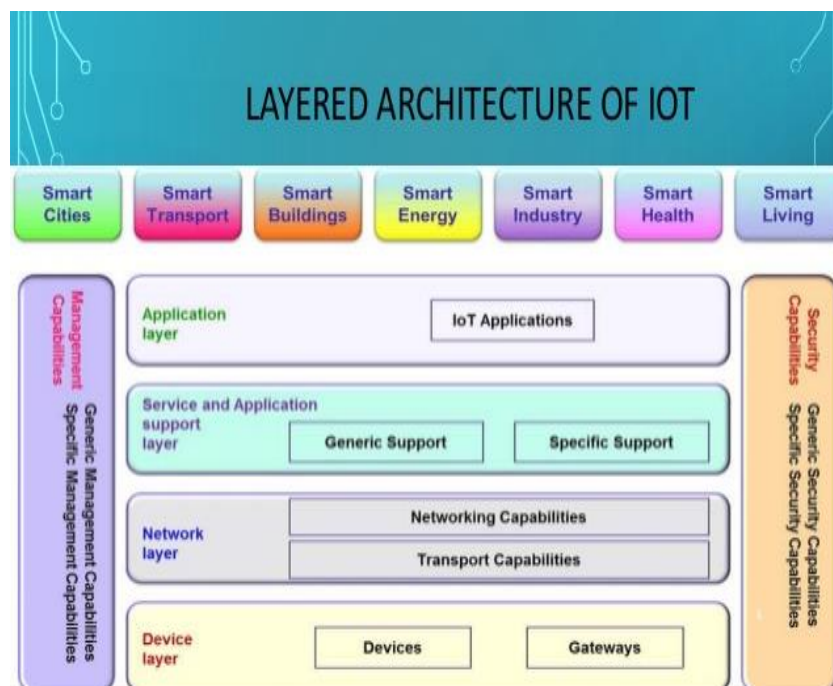


Εικόνα 1.9 Η Alexa της amazon

1.6. Κύρια μέρη του ΙoT

Από πλευράς αρχιτεκτονικής , το διαδίκτυο των πραγμάτων , θα μπορούσε να χωριστεί σε τέσσερα επίπεδα σχεδιασμού , τα οποία αναλύονται παρακάτω . Η εικόνα 2.0

απεικονίζει τα διάφορα επίπεδα σχεδιασμού (Patel et al., 2016)(ITU-T-Y.2060, 2016) .
[10]



Εικόνα 2.0 Τα Επίπεδα του Iot

I. Το χαμηλότερο επίπεδο είναι το επίπεδο συσκευών : Το επίπεδο των devices ή Hardware Layer , αναλαμβάνει την συλλογή των πληροφοριών και τη συνδεσιμότητα του αντικειμένου στο διαδίκτυο .Οι αισθητήρες μπορούν να λάβουν μετρήσεις όπως για παράδειγμα τη θερμοκρασία , τη ταχύτητα, την υγρασία και άλλα . Πολλές φορές , οι αισθητήρες , μπορεί να διαθέτουν κάποια μνήμη, επιτρέποντάς τους και την καταγραφή των μετρήσεων αυτών .

II. Το επίπεδο δικτύου: Το επίπεδο αυτό είναι υπεύθυνο για την συνδεσιμότητα , την πρόσβαση στις υπηρεσίες και για τη μεταφορά των δεδομένων των εφαρμογών όπως επίσης και των πληροφοριών που αφορούν τη διαχείριση και τον έλεγχο τους . Ο μεγάλος όγκος δεδομένων που φορτώνεται στο δίκτυο κάθε φορά , δημιουργεί την ανάγκη για ένα υψηλής απόδοσης δίκτυο. Επίσης υπάρχει η ανάγκη εξυπηρέτησης ενός μεγαλύτερου φάσματος υπηρεσιών και εφαρμογών IoT, όπως οι υπηρεσίες συναλλαγών υψηλής ταχύτητας, εφαρμογές συμβατές με το περιβάλλον και άλλες.

Για τον λόγο αυτό πρέπει να καθίσταται εφικτή η συνεργασία δικτύων με διάφορες τεχνολογίες και πρωτόκολλα πρόσβασης.

III. Το επίπεδο υποστήριξης υπηρεσιών και εφαρμογών: Το επίπεδο αυτό παρέχει δυνατότητες που χρησιμοποιούνται από όλες τις εφαρμογές, όπως για παράδειγμα η επεξεργασία και η αποθήκευση δεδομένων. Σημαντική λειτουργία αυτού του επιπέδου αποτελεί η διαχείριση της δικτυακής κίνησης που δημιουργείται. Πολλές από τις πληροφορίες αυτές απαιτούν επεξεργασία όπως φιλτράρισμα ή δρομολόγηση σε άλλα συστήματα για περαιτέρω επεξεργασία πληροφορίας.

IV. Το επίπεδο εφαρμογών: Το τελευταίο επίπεδο σχεδιασμού περιλαμβάνει τις IoT εφαρμογές. Αυτές είναι τα λογισμικά που εκτελούν συγκεκριμένες εργασίες ανάλογα με την επιθυμία του χρήστη.

1.7. Παραδείγματα IoT

I. Έξυπνοι μετρητές υπηρεσιών ωφελείας : Σε αυτή την κατηγορία, διάφοροι μετρητές IoT , όπως μετρητές ρεύματος και νερού , ενημερώνουν για τα χαρακτηριστικά της κατανάλωσης . Τέτοια χαρακτηριστικά μπορεί να είναι η ποσότητα νερού που καταναλώνεται , οι ώρες χρήσεις μιας ηλεκτρονικής συσκευής και άλλα. Το σύστημα αυτό , αφορά παρόχους υπηρεσιών , υπηρεσίες ύδρευσης αποχέτευσης δήμων και αλυσίδες καταστημάτων λιανικής . Τέλος το σύστημα είναι ικανό για εξορθολογισμό της κατανάλωσης , άμεση επίγνωση βλάβης όπως επίσης και επίγνωση της κατανάλωσης σε πραγματικό χρόνο με απόλυτη ακρίβεια.



Εικόνα 2.1 Έξυπνος μετρητής ρεύματος

II. IoT στις Ιατρικές συσκευές : Στο παράδειγμα αυτό , μια ιατρική συσκευή ή ένα ιατρικό όργανο ,μπορεί να στείλει τις ενδείξεις και τις μετρήσεις του ασθενούς ,μέσω IoT στον θεράποντα ιατρό σε πραγματικό χρόνο και με ακριβή καταγραφή τιμών . Αυτή η εφαρμογή αφορά κυρίως Ιατρικό και νοσηλευτικό προσωπικό κέντρου παροχής υπηρεσιών υγείας και υπηρεσίες παροχής κατ’ οίκον νοσηλείας. Έτσι κερδίζουμε την άμεση και με ακρίβεια καταγραφή των τιμών μετρήσεων των ασθενών σε πραγματικό χρόνο , προσαρμογή της υφιστάμενης θεραπείας χωρίς καθυστέρηση και την βελτίωση των υπηρεσιών υγείας σε ασθενείς με δυσκολία στη μετακίνηση .



Εικόνα 2.2 Έξυπνο ιατρικό όργανο

III. Οι έξυπνες πόλεις : Ίσως το πιο δημοφιλές παράδειγμα IoT στις μέρες μας, είναι οι έξυπνες πόλεις . Στις πόλεις αυτές , το μεγαλύτερο μέρος των συσκευών , είναι ικανό να επικοινωνεί μεταξύ τους και να λαμβάνει αποφάσεις ανάλογα τις περιβαλλοντικές αλλαγές μέσω αισθητήρων . Η απόφαση δηλαδή της μιας συσκευής , μπορεί να επηρεάσει άμεσα την συμπεριφορά μιας άλλης. Για παράδειγμα , οι στύλοι φωτισμού μιας έξυπνης πόλης , αυξάνουν την ισχύ του φωτός που παράγουν , ανάλογα το φυσικό φώς που ανιχνεύουν οι αισθητήρες πως υπάρχει στο περιβάλλον μια δεδομένη στιγμή . Επίσης ένας αισθητήρας που θα μπορούσε να ανιχνεύσει ακραίες καιρικές συνθήκες , θα μπορούσε να ενημερώσει διάφορες συσκευές χρηστών , όπως το smartphone , ώστε οι χρήστες να αποφύγουν την περιοχή εκείνη αν είναι εντός εμβέλειας και υπάρχει κίνδυνος.



Εικόνα 2.3 Το όραμα της έξυπνης πόλης

2. Κατασκευή και εκτύπωση των τμημάτων του ρομποτικού βραχίονα

2.1. Το λογισμικό Solidworks

Στο παρόν κεφάλαιο θα εξεταστεί το λογισμικό Solidworks και οι δυνατότητες που αυτό μπορεί να προσφέρει στον χρήστη.

2.1.1. Τι είναι το Solidworks

Τα τελευταία χρόνια , ο σχεδιασμός προϊόντων αυξάνεται με ραγδαίους ρυθμούς. Η ανάπτυξη νέων τεχνολογιών στον τομέα αυτό , έκανε την πρόσβαση στον κλάδο πιο εύκολη από ποτέ. Ειδικά στις βιομηχανίες , έχει παρατηρηθεί μια στροφή , προς τον σχεδιασμό , με σκοπό όχι μόνο της λειτουργικότητας , αλλά και της ανάπτυξης καινοτόμων προϊόντων. Αυτό είχε ως αποτέλεσμα την δημιουργία προγραμμάτων σχεδιασμού εξαρτημάτων και ανταλλακτικών (3D Modeling) που δίνουν τη δυνατότητα σε οποιονδήποτε να δημιουργήσει τα δικά του σχέδια. Ένα από αυτά τα λογισμικά είναι και το Solidworks το οποίο είναι και αυτό το οποίο θα εξετάσουμε και στο παρόν κεφάλαιο. [11]



Εικόνα 2.4 Solidworks Logo [11]

2.1.2. Κατασκευή των στοιχείων του βραχίονα σε Solidworks

Στο παρόν κεφάλαιο , θα παρουσιαστούν τα σχέδια του ρομποτικού βραχίονα που κατασκευάστηκαν , καθώς επίσης και άλλα σχέδια τα οποία ήταν απαραίτητα για την ολοκλήρωση της εργασίας.

Τα βασικά μέρη του ρομποτικού βραχίονα που δημιουργήθηκαν είναι έξι:

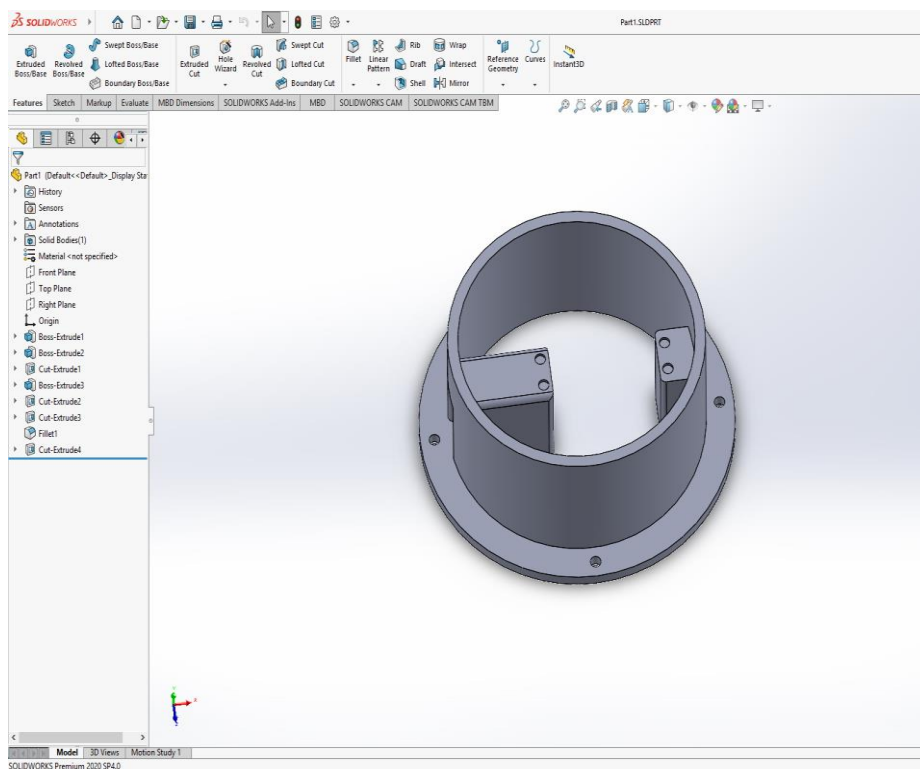
- 1) Η βάση
- 2) Η βάση περιστροφής
- 3) Ο ώμος του βραχίονα
- 4) Ο αγκώνας του βραχίονα
- 5) Ο καρπός του βραχίονα
- 6) Η αρπαγή

Επιπλέον σχέδια που χρησιμοποιήθηκαν στην εργασία είναι δύο:

- 1) Οι κύβοι
- 2) Οι βάσεις των κύβων

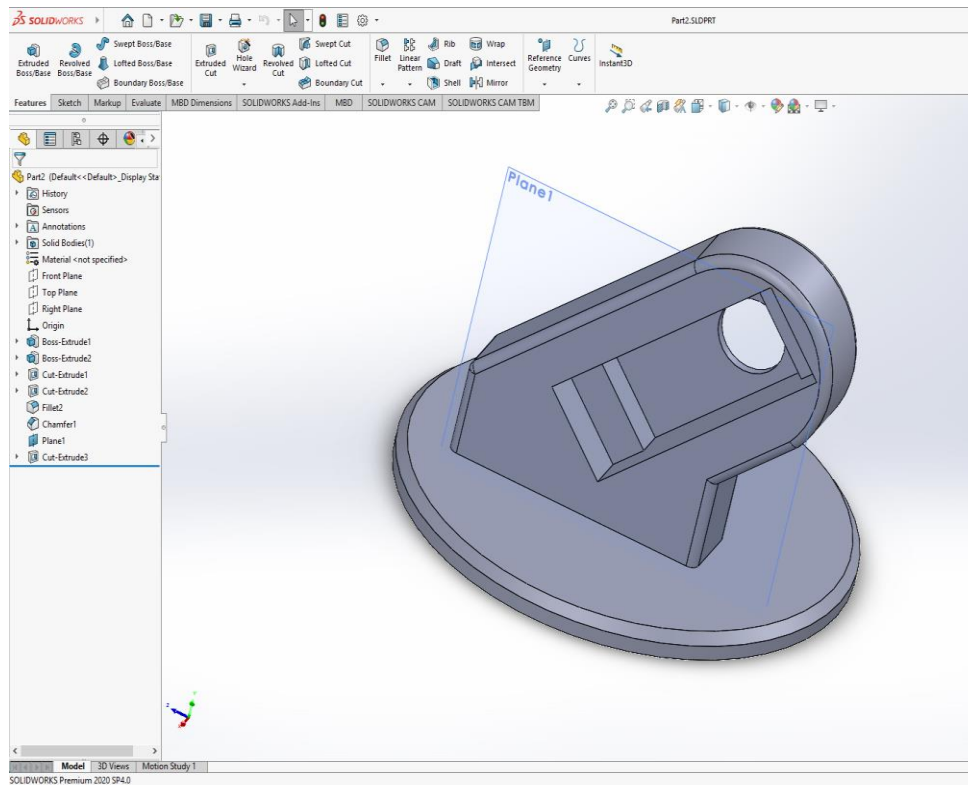
Παρακάτω παρατίθενται εικόνες από το Solidworks που περιλαμβάνουν τα μέρη του βραχίονα και των σχεδίων των βάσεων των κύβων.

Στην παρακάτω εικόνα βλέπουμε την βάση του ρομποτικού βραχίονα . Στο σχέδιο μπορούμε να δούμε όλες τις λεπτομέρειες , όπως τις τρύπες που δημιουργήθηκαν για να βιδωθεί σε σταθερή βάση και τα στηρίγματα για τον ηλεκτρονικό εξοπλισμό που πρόκειται να εγκατασταθεί (μοτέρ).



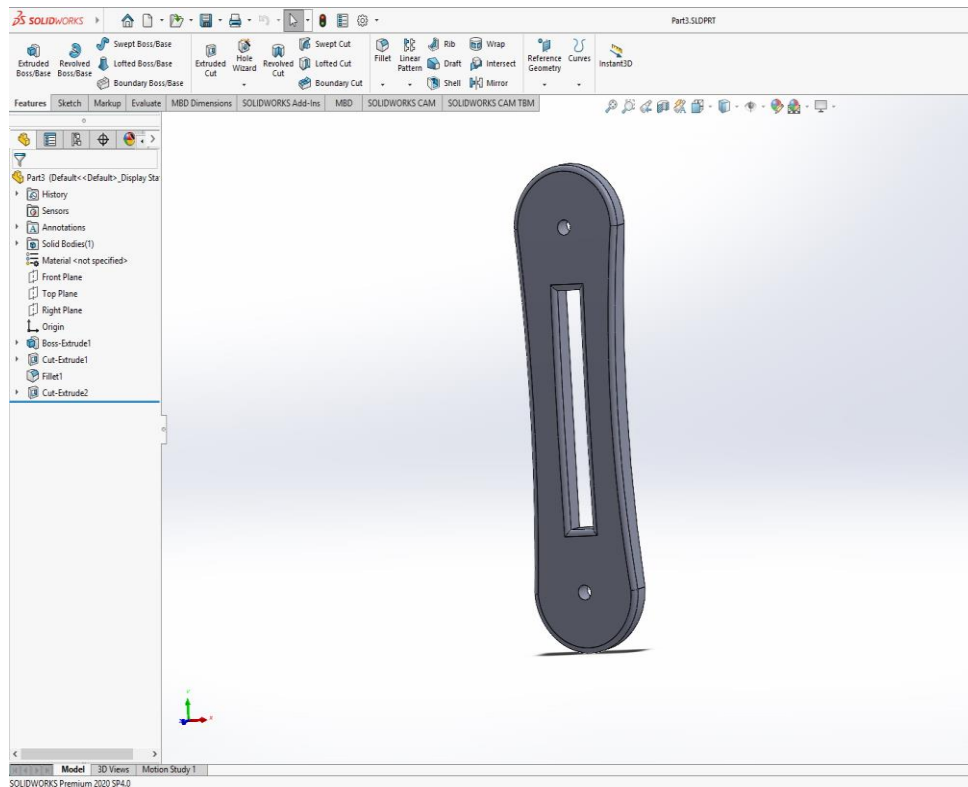
Εικόνα 2.5 Βάση του ρομποτικού Βραχίονα

Παρακάτω βλέπουμε την περιστρεφόμενη βάση , η οποία προσαρμόζεται πάνω από την βάση της εικόνας 2.5 . Αυτό δίνει την δυνατότητα στον ρομποτικό βραχίονα να μπορεί να περιστρέφεται . Επίσης , όπως και στο σχέδιο της εικόνας 2.5 , δημιουργήθηκε χώρος στο σχέδιο για την εγκατάσταση του ηλεκτρονικού εξοπλισμού (μοτέρ) με σκοπό την σχετική κίνηση βάσης περιστροφής και ώμου του βραχίονα .



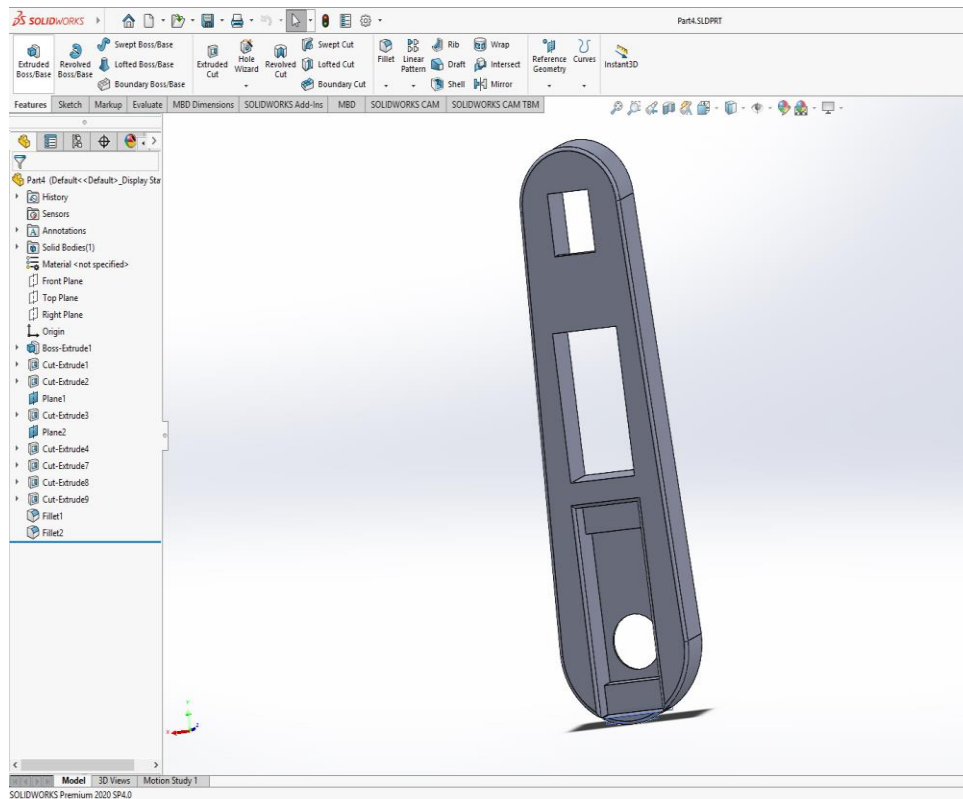
Εικόνα 2.6 Βάση περιστροφής του ρομποτικού Βραχίονα

Στην εικόνα 2.7 απεικονίζεται ο Ώμος του ρομπότ , ο οποίος , το ένα άκρο συνδέεται στην βάση περιστροφής της εικόνας 2.6 , ενώ το άλλο άκρο συνδέεται στο κάτω άκρο της εικόνας 2.8 . Στο σχέδιο , δημιουργήθηκαν και δύο τρύπες διαμέτρου τεσσάρων χιλιοστών , ώστε να γίνουν οι συνδέσεις των τμημάτων με βίδες.



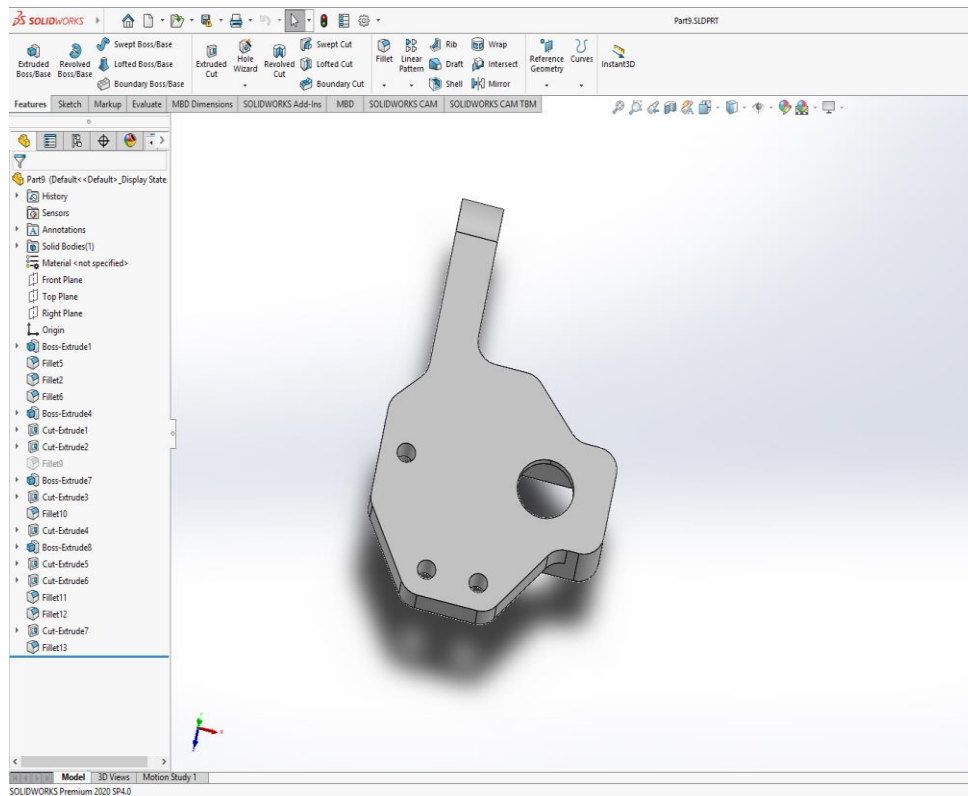
Εικόνα 2.7 Ώμος του ρομποτικού Βραχίονα

Η εικόνα 2.8 δείχνει τον αγκώνα του βραχίονα , όπου το πάνω άκρο συνδέεται με τον καρπό της εικόνας 2.9 . Στο σχέδιο δημιουργήθηκαν βάσεις για την τοποθέτηση δύο μοτέρ ώστε να είναι εφικτή η κίνηση μεταξύ ώμου-αγκώνα και αγκώνα-καρπού.



Εικόνα 2.8 Αγκώνας του ρομποτικού Βραχίονα

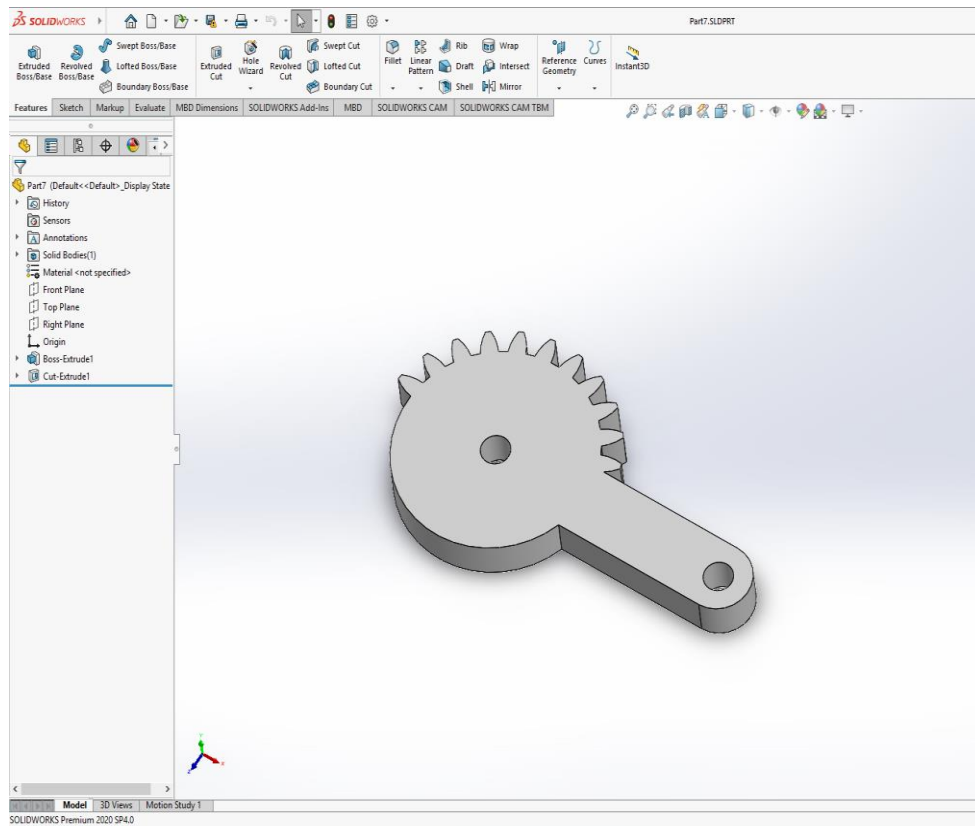
Παρακάτω , ο καρπός της εικόνας 2.9 , συνδέεται με το πάνω μέρος του αγκώνα όπως προαναφέραμε και έχει την δυνατότητα εγκατάστασης μοτέρ στο κάτω μέρος του , για την κίνηση της αρπαγής , όπως επίσης και την εγκατάσταση των επιμέρους εξαρτημάτων της αρπαγής στο πάνω μέρος του.



Εικόνα 2.9 Καρπός του ρομποτικού Βραχίονα

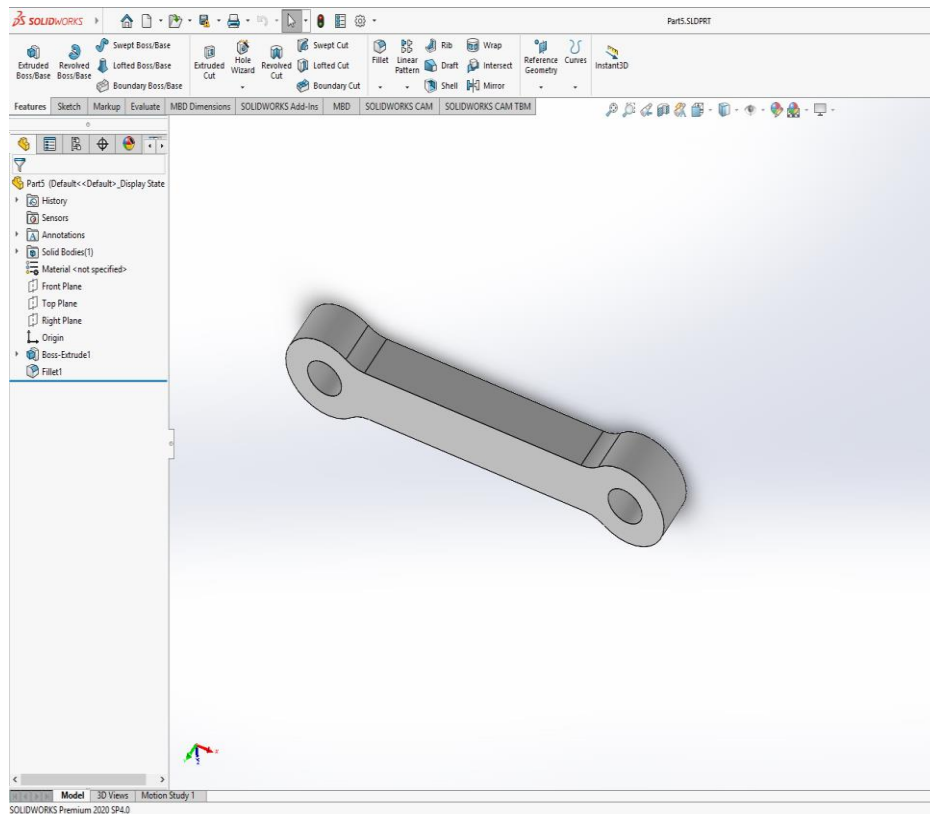
Στις εικόνες παρακάτω φαίνονται τα σχέδια που έγιναν για την ολοκλήρωση της αρπαγής , η οποία αποτελείται από τα γραναζία , τους συνδέσμους γραναζιού-αρπαγής και την αρπαγή.

Η εικόνα 3.0 δείχνει το σχέδιο του γραναζιού της αρπαγής , το οποίο συνδέεται με το μοτέρ του καρπού ώστε η αρπαγή να μπορεί να κλείνει και να ελευθερώνει .Στο σχέδιο έγιναν και τρύπες τεσσάρων χιλιοστών ώστε να γίνει η σύνδεση των τμημάτων της αρπαγής μεταξύ τους.



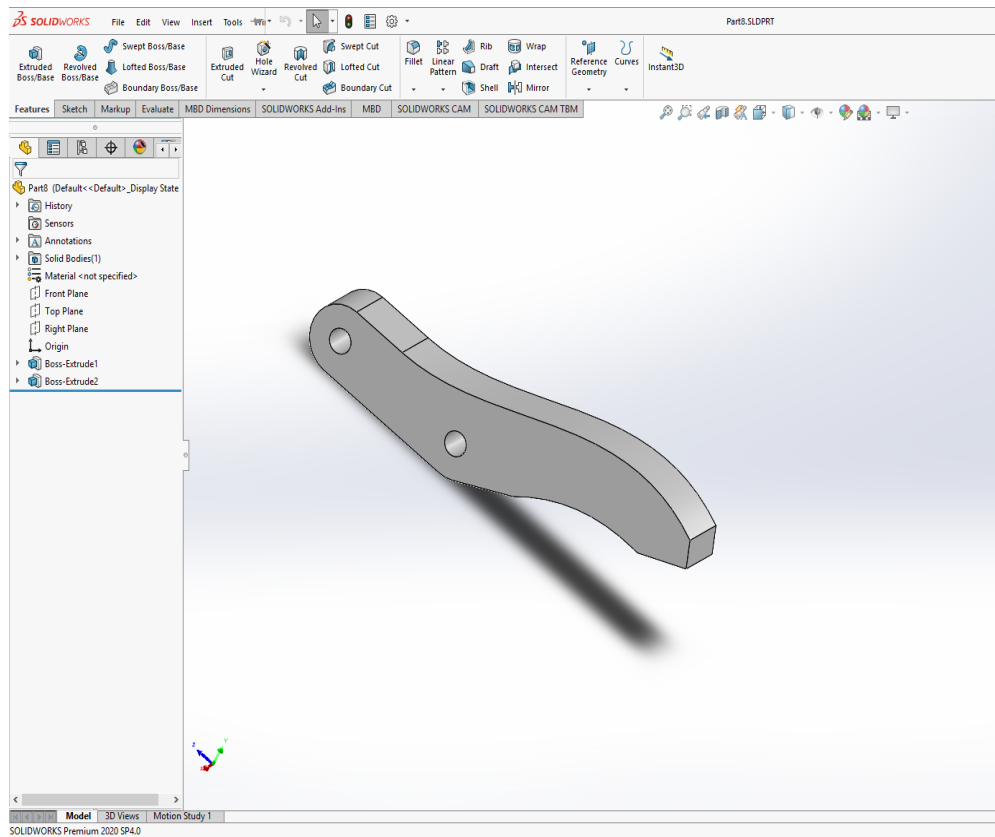
Εικόνα 3.0 Γρανάζι για την αρπαγή

Το άλλο τμήμα της αρπαγής φαίνεται στην εικόνα 3.1 και είναι ο σύνδεσμος του γραναζιού της εικόνας 3.0 με την αρπαγή της εικόνας 3.2. Όπως και στο παραπάνω σχέδιο, δημιουργήθηκαν τρύπες τεσσάρων χιλιοστών για την σύνδεση.



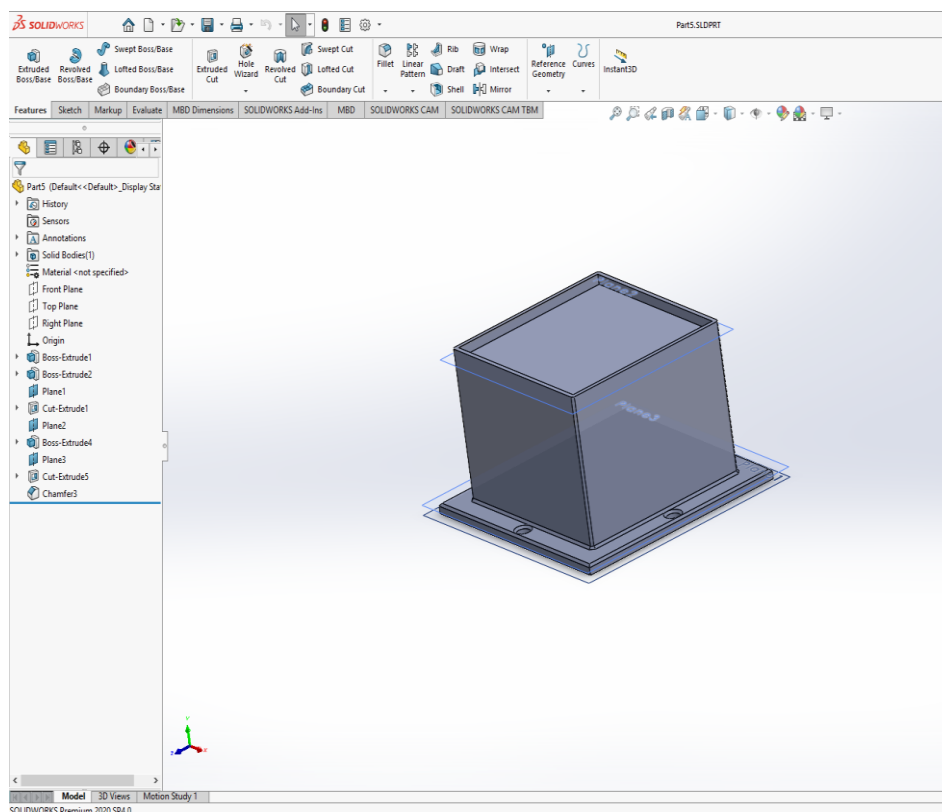
Εικόνα 3.1 Σύνδεσμος γραναζιού – αρπαγής

Τέλος η αρπαγή της εικόνας 3.2 , αποτελεί το τελευταίο τμήμα του συνολικού σχεδίου της αρπαγής , όπου η άκρη της έχει ειδική κατασκευή , ώστε να μπορεί να αρπάξει το επιθυμητό αντικείμενο. Εδώ στην πρώτη τρύπα συνδέεται το γρανάτζι για την αρπαγή ,ενώ στην άλλη συνδέεται ο σύνδεσμος γραναζιού-αρπαγής.



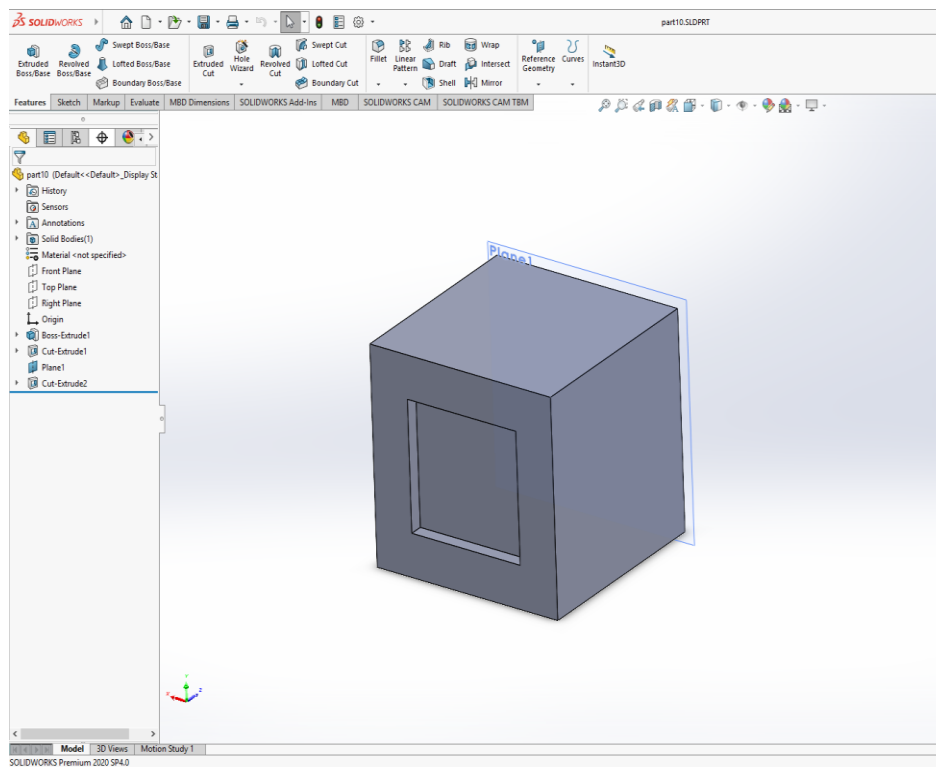
Εικόνα 3.2 Η αρπαγή

Οι βάσεις των κύβων της εικόνας 3.3 δημιουργήθηκαν ώστε να μπορούν να συγκρατούν τους κύβους της εικόνας 3.4 για να μην ξεφεύγουν από την θέση τους . Διαθέτουν τέσσερις τρύπες για την τοποθέτηση σε σταθερή βάση , όπως επίσης και ένα υπερυψωμένο τμήμα στην κορυφή , για την συγκράτηση των κύβων.



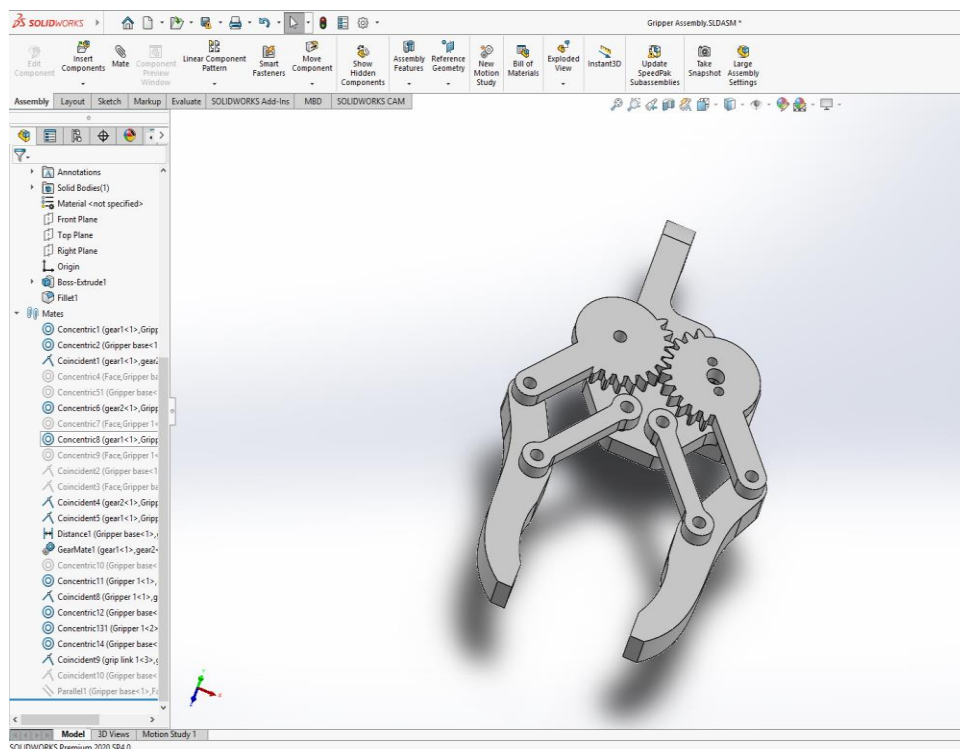
Εικόνα 3.3 Βάση των κύβων

Οι κύβοι της εικόνας 3.4 , τοποθετούνται πάνω στις βάσεις της εικόνας 3.3 . Είναι ειδικά σχεδιασμένοι ώστε η αρπαγή να μπορεί να τους συγκρατήσει εύκολα .



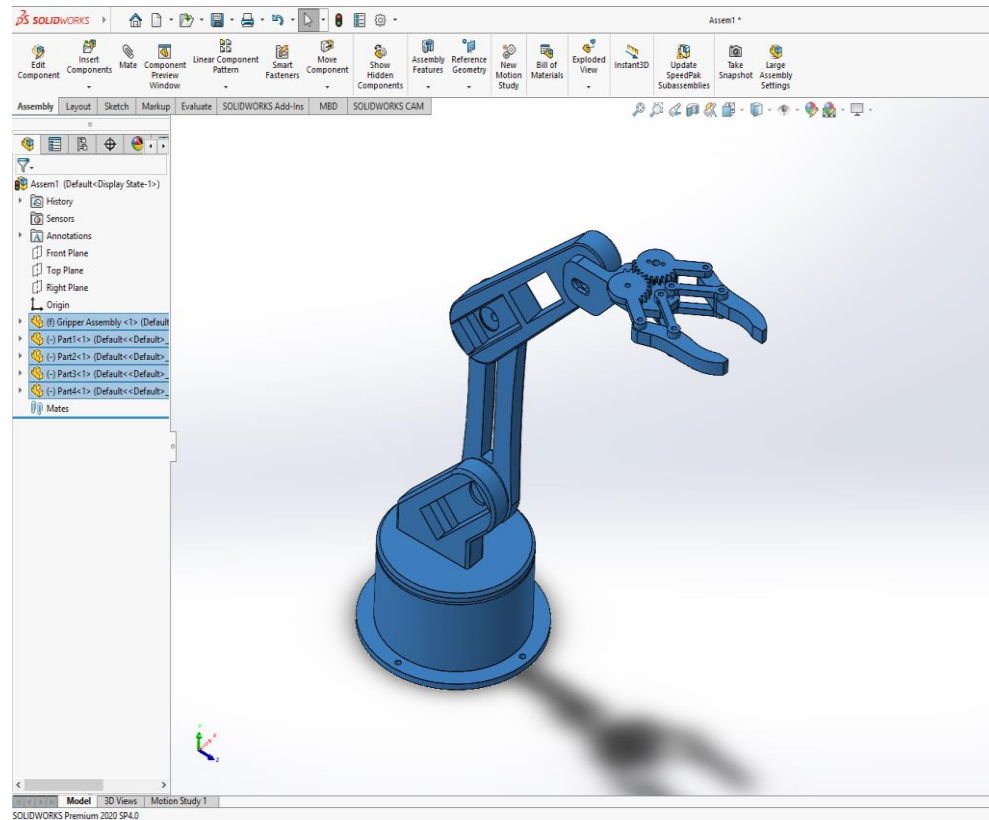
Εικόνα 3.4 Ο κύβος

Το ολοκληρωμένο σχέδιο της αρπαγής φαίνεται παρακάτω στην εικόνα 3.5 , όπου όλα τα επιμέρους κομμάτια ενώθηκαν σε ένα ενιαίο σχέδιο.



Εικόνα 3.5 Ολοκληρωμένο σχέδιο καρπού – αρπαγής

Τέλος μετά την σύνδεση όλων των παραπάνω τμημάτων του βραχίονα , έχουμε το αποτέλεσμα του σχεδίου της εικόνας 3.6 , όπου απεικονίζεται το ολοκληρωμένο σχέδιο .



Εικόνα 3.6 Ολοκληρωμένο σχέδιο του Βραχίονα

2.2. Τρισδιάστατη εκτύπωση

Το κεφάλαιο αυτό θα εστιάσει στην λειτουργία του 3D εκτυπωτή και στην εκτύπωση τμημάτων του βραχίονα με την βοήθεια του λογισμικού Ultimaker Cura.

2.2.1. Τί είναι η Τρισδιάστατη εκτύπωση

Η τρισδιάστατη εκτύπωση ή αλλιώς 3D printing αποτελεί μια μέθοδο , η οποία προσφέρει τη δυνατότητα κατασκευής αντικειμένων μέσω της διαδοχικής πρόσθεσης επάλληλων στρώσεων υλικού. Για την τρισδιάστατη εκτύπωση μπορούν να χρησιμοποιηθούν διάφοροι τύποι υλικών όπως τα κεραμικά και πολυμερή πλαστικά . Σε κάποιες περιπτώσεις χρησιμοποιούνται και υλικά όπως το μέταλλο και το γυαλί . Λόγω της ευκολίας της τρισδιάστατης εκτύπωσης , αρκετοί πιστεύουν ότι στα επόμενα χρόνια, για την κατασκευή πολλών προϊόντων θα χρησιμοποιείται αποκλειστικά η τρισδιάστατη εκτύπωση , αντικαθιστώντας τις παλαιές τεχνικές. [12]

2.2.2. Ο τρισδιάστατος εκτυπωτής

Οι 3D Εκτυπωτές , αποτελούν συσκευές παραγωγής αντικειμένων στις τρεις διαστάσεις από ένα ψηφιακό αρχείο. Στον κλάδο αυτό υπάρχουν πολλοί μέθοδοι κατασκευής. Η πιο γνωστή στην κατηγορία των εκτυπωτών είναι η εναπόθεση υλικού σε διαδοχικές στρώσεις .Η παραπάνω μέθοδος , δημιουργεί συνεχόμενα οριζόντια στρώματα υλικού με σκοπό την κατασκευή του τρισδιάστατου αντικειμένου του χρήστη . Το σχέδιο για την εκτύπωση , γίνεται μέσω μιας κατηγορίας λογισμικού που ονομάζεται CAD. Όσο αναφορά τον τρόπο κατασκευής ,το υλικό που χρησιμοποιεί ο εκτυπωτής , θερμαίνεται μέχρι να λιώσει και ο εκτυπωτής εναποθέτει μια λεπτή στρώση υλικού ώστε να κατασκευαστεί το αντικείμενο. Η κάθε στρώση υλικού που εναποθέτεται , στερεοποιείται αμέσως. Η διαδικασία ξεκινά από την βάση του τρισδιάστατου αντικειμένου και προχωρά προς τα πάνω , μέχρι το ψηφιακό σχέδιο του χρήστη να έχει κατασκευαστεί πλήρως στις τρεις διαστάσεις από τον εκτυπωτή. [13]

Παρακάτω παρατίθεται μια εικόνα 3D εκτυπωτή.



Εικόνα 3.7 3D printer

2.2.3. Το λογισμικό Ultimaker Cura



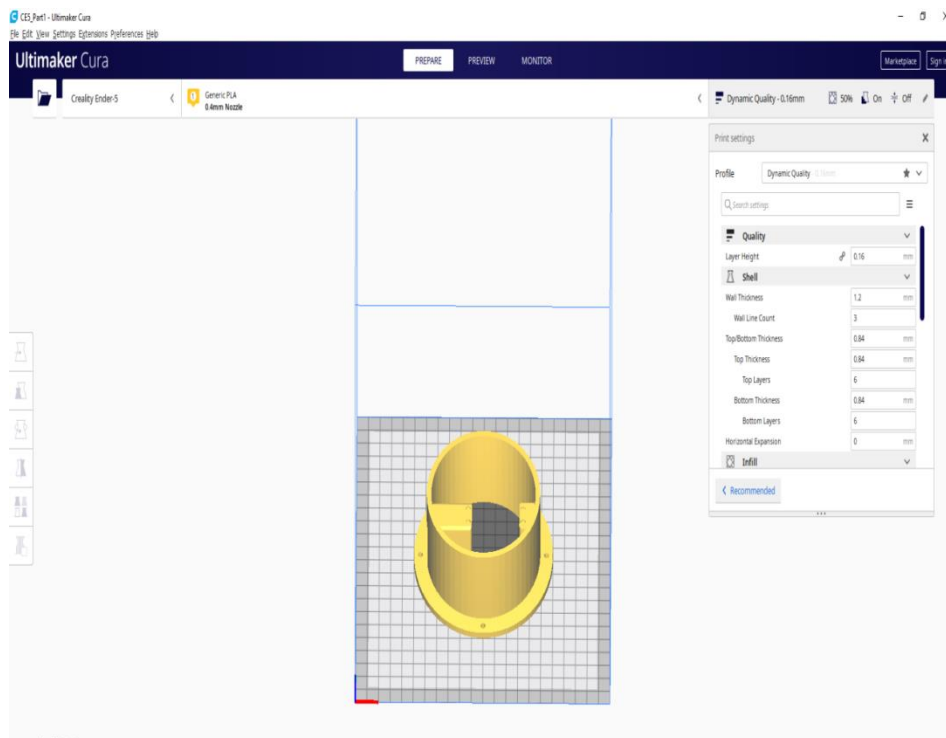
Εικόνα 3.8 Το Logo του Cura [14]

Το Cura , είναι ένα ανοιχτού κώδικα λογισμικό ,το οποίο χρησιμοποιείται για να “τεμαχίσει” σχέδια σε CAD , με σκοπό την εκτύπωση του σχεδίου σε έναν 3D εκτυπωτή. Δημιουργήθηκε από τον David braam , τον οποίο προσέλαβε η εταιρία κατασκευής 3D εκτυπωτών Ultimaker με σκοπό να διατηρήσει το λογισμικό .Το Cura λειτουργεί τεμαχίζοντας το μοντέλο που του δίνεται , σε στρώματα ,δημιουργώντας ένα αρχείο με επέκταση gcode ,το οποίο μπορεί να αναγνωρίσει ένας 3D εκτυπωτής και να το εκτυπώσει. [15]

2.2.4. Εισαγωγή σχεδίων στο Cura

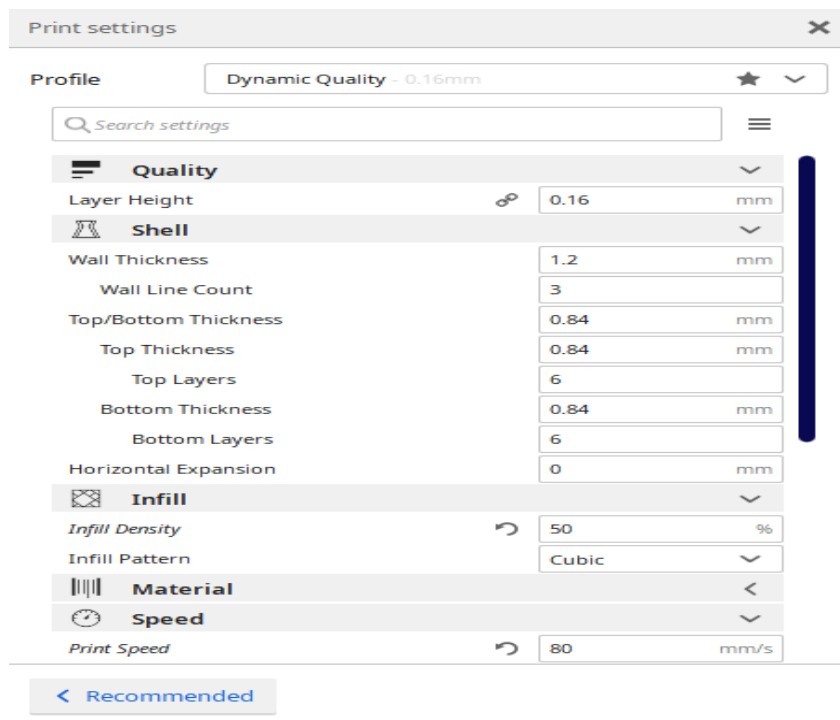
Σε αυτό το κεφάλαιο θα εισάγουμε σχέδια στο Ultimaker Cura . Επίσης θα δούμε τις ρυθμίσεις και τις δυνατότητες που προσφέρει το λογισμικό ,καθώς και τις βασικές ρυθμίσεις που επιλέχθηκαν για την εκτύπωση των σχεδίων της εργασίας.

Τα σχέδια στο Solidworks του κεφαλαίου 2.2 , έχουν επέκταση SLDPRT(solidworks part) , τα οποία μπορούν να εισαχθούν απευθείας στο Cura . Στην παρακάτω εικόνα 3.9, έχουμε εισάγει στο Cura το σχέδιο της βάσης του ρομποτικού βραχίονα.



Εικόνα 3.9 Η βάση του βραχίονα στο Cura

Μερικές από τις βασικές ρυθμίσεις του προγράμματος φαίνονται στην εικόνα 4.0:



Εικόνα 4.0 Ρυθμίσεις του Cura

Στην κατηγορία Quality , έχουμε μόνο μια ρύθμιση , το Layer Height , η οποία καθορίζει πόσο λεπτή θα είναι η στρώση του υλικού της εκτύπωσης κάθε φορά.

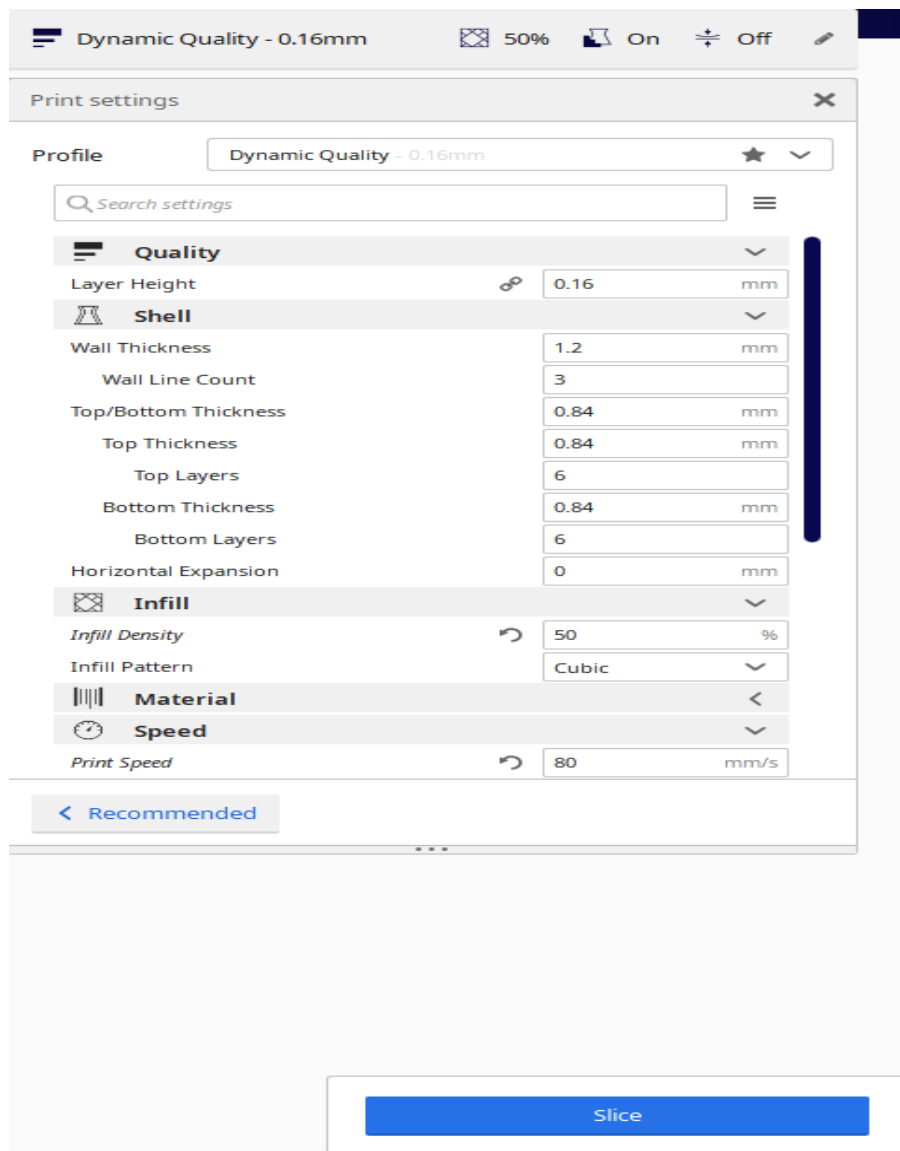
Στην κατηγορία Shell , μπορούμε να καθορίσουμε όλες τις παραμέτρους που αφορούν το κουβούκλιο του σχεδίου , όπως το πάχος και τον αριθμό των στρώσεων .

Στην κατηγορία Infill , η επιλογή Infill density καθορίζει την εσωτερική γέμιση με υλικό ,που θα έχει το σχέδιό μας και η επιλογή Infill pattern , το σχέδιο που θα έχει η εσωτερική γέμιση αυτή.

Τέλος στην κατηγορία Speed υπάρχει η επιλογή Print speed με την οποία ρυθμίζουμε την ταχύτητα της εκτύπωσης.

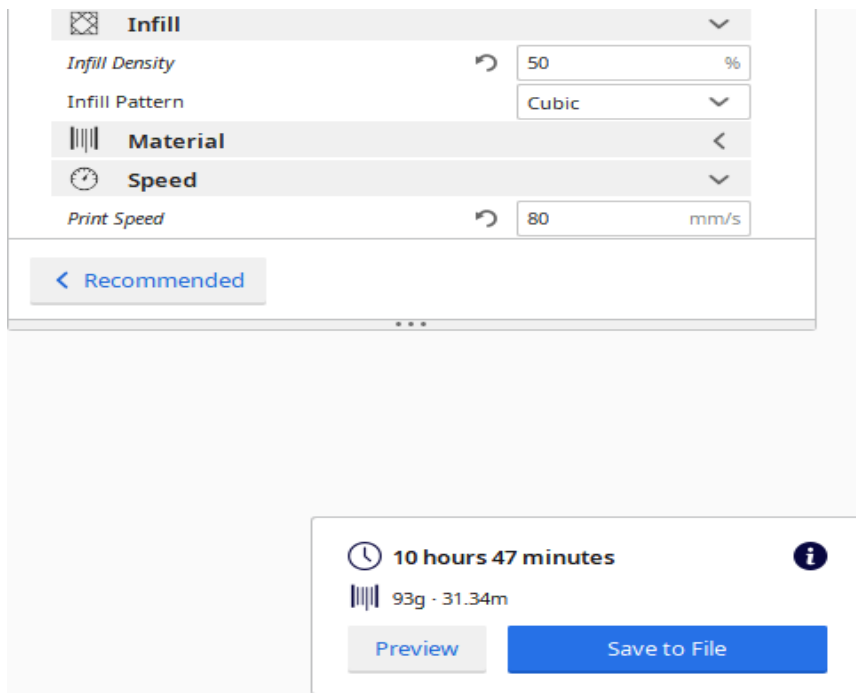
Οι ρυθμίσεις που επιλέχθηκαν για την εκτύπωση όλων των σχεδίων είναι αυτές της εικόνας 4.0.

Αφού επιλέξουμε τις επιθυμητές ρυθμίσεις για την εκτύπωση , επιλέγουμε την επιλογή Slice , όπως φαίνεται στην παρακάτω εικόνα 4.1:



Εικόνα 4.1 Η επιλογή Slice

Με την επιλογή αυτή , το πρόγραμμα θα δημιουργήσει ένα αρχείο με επέκταση gcode και θα εκτιμήσει τον χρόνο που χρειάζεται ο εκτυπωτής για να δημιουργήσει την κατασκευή.

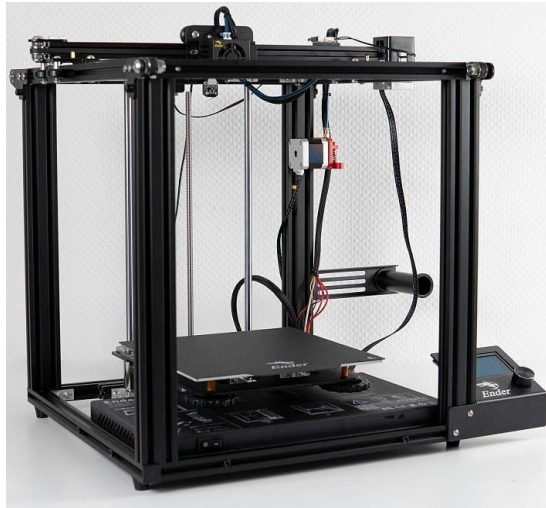


Εικόνα 4.1 Εκτιμώμενος χρόνος κατασκευής

2.2.5. Εκτύπωση των σχεδίων του βραχίονα

Αφού κατασκευάσαμε όλα τα αρχεία gcode , θα τα εκτυπώσουμε στον 3D εκτυπωτή. Ο εκτυπωτής που χρησιμοποιήθηκε για την κατασκευή των κομματιών του ρομποτικού βραχίονα , είναι ο Ender-5 της εταιρίας Creality 3D και το υλικό που χρησιμοποιήθηκε είναι πλαστικό τύπου PLA. [16]

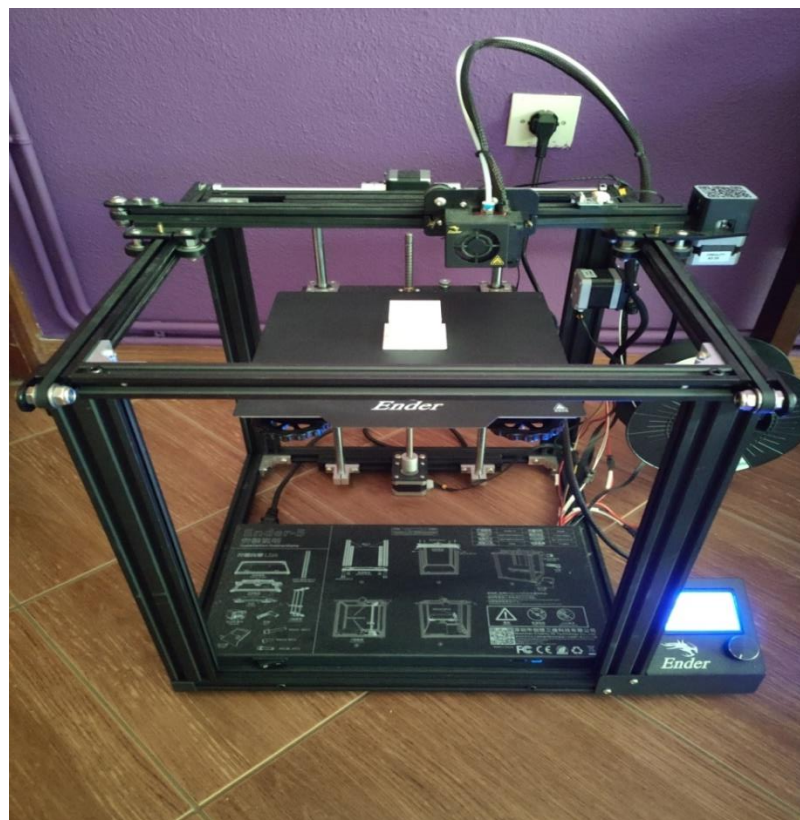
Ο συγκεκριμένος εκτυπωτής , φαίνεται στην εικόνα 4.2 παρακάτω.



Εικόνα 4.2 Ο εκτυπωτής Ender 5 της Creality 3D

Παρακάτω παρατίθενται κάποιες εικόνες από τα εκτυπωμένα σχέδια:

Στην εικόνα 4.3 , βλέπουμε τον 3D εκτυπωτή σε λειτουργία , όπου κατασκευάζει τις βάσεις των κύβων που θα χρειαστούμε.



Εικόνα 4.3 Εκτύπωση των βάσεων των κύβων

Στην εικόνα 4.4 , βλέπουμε το πέρας της κατασκευής των κύβων που θα χρειαστούμε για την λειτουργία της ταξινόμησης του ρομπότ.



Εικόνα 4.4 Εκτύπωση των κύβων

3. Ηλεκτρονικός εξοπλισμός και Βασικό Πρόγραμμα

3.1. Εισαγωγή στις πλακέτες IoT

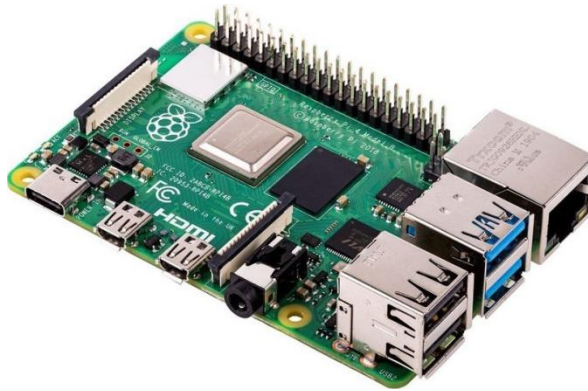
Σε αυτό το κεφάλαιο , θα αναλυθεί ο ηλεκτρονικός εξοπλισμός που χρησιμοποιήθηκε , όπως η πλακέτα IoT , οι αισθητήρες , τα μοτέρ και το πρόγραμμα που τελικά εγκαταστάθηκε στην πλακέτα που χρησιμοποιήσαμε.

3.1.1. Πλακέτες IoT

Σε αυτή τη παράγραφο , θα εξετάσουμε και θα αναλύσουμε κάποιες από τις πιο γνωστές και ευρέως διαδεδομένες πλακέτες IoT , που με τον συνδυασμό αισθητήρων , χρησιμοποιούνται στην ανάπτυξη IoT εφαρμογών.

I. Raspberry Pi :

Η πλακέτα Raspberry Pi , κατασκευάστηκε στο Ηνωμένο Βασίλειο από την εταιρία Raspberry Pi foundation , με σκοπό την εξέλιξη των γνώσεων ατόμων που ασχολούνται με τον τομέα της πληροφορικής , αυτοματισμών και ηλεκτρονικής. Το Raspberry Pi είναι ένας μικροεπεξεργαστής χαμηλού κόστους , μικρού μεγέθους , που έχει την δυνατότητα εκτέλεσης διεργασιών όπως αναγνώριση εικόνας , επεξεργασία κειμένων και άλλα. Στην παρακάτω εικόνα 4.5 απεικονίζεται η νεότερη έκδοση του Raspberry Pi , το Raspberry Pi 4.



Εικόνα 4.5 Το Raspberry Pi 4

Τα χαρακτηριστικά της παραπάνω πλακέτας είναι τα εξής :

1. Διαθέτει τετραπύρηνο επεξεργαστή 64bit (Broadcom BCM2711) ,
χρονισμένο στα 1,5Ghz τύπου Cortex-A72
2. Μνήμη τύπου LPDDR4-2400 SDRAM , που μπορεί να είναι 2Gb , 4Gb ή 8Gb
ανάλογα το μοντέλο.
3. Wifi και Bluetooth
4. Θύρα Ethernet για ενσύρματη σύνδεση στο διαδίκτυο
5. Τέσσερις θύρες usb , δύο τύπου usb 2.0 και δύο τύπου usb 3.0
6. Display port
7. Camera port
8. Hdmi port
9. Θύρα για micro-sd
10. Θύρα για ήχο
11. 40 GPIO pins

II. Intel Edison :

Μια άλλη πλακέτα για την κατασκευή IoT εφαρμογών είναι και η Intel Edison η οποία κατασκευάστηκε από την εταιρία Intel Corporation το 2014. Η πλακέτα έχει δύο

εκδόσεις, μία συμβατή με το Arduino και μία έκδοση σε μικρότερο μέγεθος για την εύκολη ανάπτυξη δοκιμαστικών εφαρμογών. Εξοπλισμένη με το λειτουργικό σύστημα που ανομάζεται Yocto ,που είναι βασισμένο σε Linux , η πλακέτα αυτή είναι κατάλληλη για την κατασκευή IoT εφαρμογών υψηλού επιπέδου.[17]

Η πλακέτα αυτή απεικονίζεται στην εικόνα 4.6.



Εικόνα 4.6 Η πλακέτα Intel Edison

Τα χαρακτηριστικά της παραπάνω πλακέτας είναι τα εξής [18]:

1. Socket 22nm Intel Atom που περιλαμβάνει 2 πυρήνες Atom Silvermont χρονισμένοι στα 500MHz και έναν πυρήνα intel quark χρονισμένος στα 100MHz
2. Wifi και Bluetooth
3. 1 ή 4 GB μνήμης τύπου LPDDR4
4. Συνδέσεις για Usb
5. 3.5 mm θύρα για ήχο
6. Ψηφιακό PDM μικρόφωνο
7. 2.54 mm για Ηχεία
8. Συνδέσεις για βίντεο
9. Υποδοχή για Hdmi
10. Υποδοχή για κάρτα μνήμης τύπου SD
11. Θύρα Ethernet για ενσύρματη σύνδεση στο διαδίκτυο
12. 40 GPIO pins

III. Arduino :

Ίσως η πιο δημοφιλής πλακέτα που επιτρέπει την δημιουργία IoT εφαρμογών ,είναι το Arduino . Το Arduino , δημιουργήθηκε το 2005 από τους Massimo Banzì και David Cueartielles . Οι πρώτες πλακέτες ξεκίνησαν να παράγονται σε ένα μικρό εργοστάσιο, κοντά στο Τορίνο της Ιταλίας. Οι πιο σύγχρονες πλακέτες Arduino , είναι βασισμένες στον μικροελεγκτή Atmel AVR(ATmega328 και ATmega168 στις νεότερες εκδόσεις).Απο το 2005 και έπειτα , δημιουργήθηκαν τουλάχιστον δέκα διαφορετικές εκδοχές του Arduino , η καθε μία με διαφορετικά χαρακτηριστικά και δυνατότητες. [19] Κάποιες απο αυτές τις εκδοχές αποτελεί το Arduino uno , το Arduino nano ,το Arduino mega και άλλα. Στην παρακάτω εικόνα 4.7 απεικονίζεται το Arduino uno.



Εικόνα 4.7 Το Arduino uno

3.1.2. Εισαγωγή στο Arduino

Στο κεφάλαιο αυτό , θα μελετήσουμε τα χαρακτηριστικά της πλακέτας Arduino uno, που είναι και η πλακέτα που χρησιμοποιήθηκε στην εργασία αυτή , σε μεγαλύτερο βάθος και λεπτομέρεια.



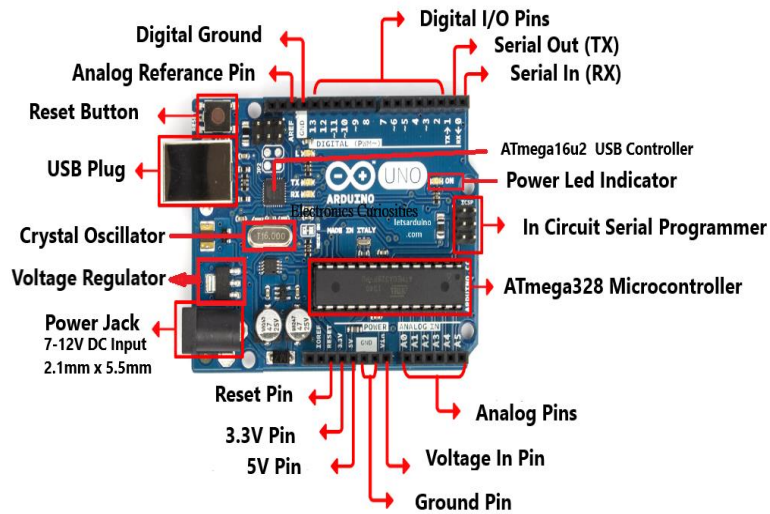
Εικόνα 4.8 Arduino uno με Usb cable

Το Arduino uno , βασίζεται στον μικροεπεξεργαστή ATmega328P . Ο μικροεπεξεργαστής αυτός , είναι 8 bit , 8 Mhz και διαθέτει τρεις διαφορετικές μνήμες [20]:

- I. Η μνήμη Flash : Το Arduino uno , διαθέτει 32 kilobyte , μνήμης flash , όπου τα 30 kilobyte χρησιμοποιούνται για την αποθήκευση των προγραμμάτων των χρηστών, ενώ τα υπόλοιπα 2 , είναι δεσμευμένα από το firmware όπου εγκατέστησε ο κατασκευαστής ,για να είναι δυνατή η εγκατάσταση των προγραμμάτων των χρηστών.
- II. Η μνήμη SRAM: Η μνήμη αυτή είναι απαραίτητη για την αποθήκευση όλων των μεταβλητών του προγράμματος όταν λειτουργεί το Arduino. Το Arduino uno έχει διαθέσιμη μνήμη SRAM 2 kilobyte.
- III. Η μνήμη EEPROM: Η μνήμη αυτή χρησιμοποιείται για εγγραφή ή ανάγνωση δεδομένων από τα προγράμματα.

Το Arduino διαθέτει θύρα USB για την σύνδεση με υπολογιστή και προγραμματισμό.

Η τροφοδοσία του μπορεί να γίνει είτε μέσω του USB , συνεχής τάση 5V, είτε μέσω εξωτερική τροφοδοσίας συνεχούς ρεύματος μέχρι 12V . Ο ρυθμιστής τάσης που περιέχει το Arduino θα μειώσει την οποιαδήποτε τάση από 5-12V σε 5V. Η Παρακάτω εικόνα δείχνει όλα τα στοιχεία του Arduino uno: [20]



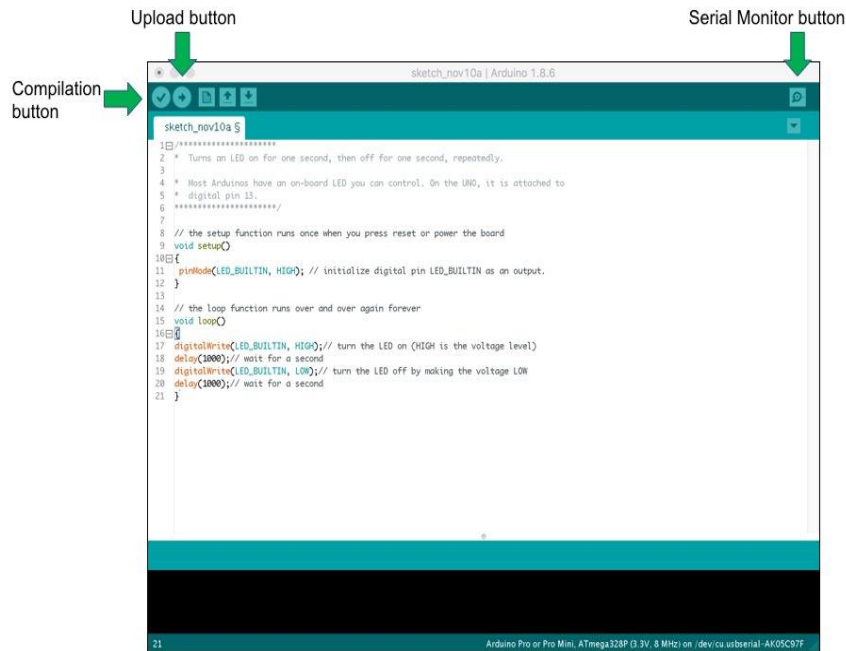
Εικόνα 4.9 Στοιχεία του arduino uno

- 3.3V pin : Τροφοδοτεί με τάση 3,3V .
- 5V pin : Τροφοδοτεί με τάση 5V.
- Reset Pin : Κάνει reset to Arduino σε συνδυασμό με ένα GND ακροδέκτη.
- Ground Pins : Η γείωση.
- Voltage in Pin : Η τάση εισόδου της πλακέτας όταν χρησιμοποιεί εξωτερική πηγή Ενέργειας.
- Power Jack : Η εξωτερική τροφοδοσία.
- Voltage regulator : Ο ρυθμιστής τάσης μειώνει την τάση της εξωτερικής τροφοδοσίας είναι πάνω από 5V.
- USB plug : Η θύρα USB που επιτρέπει σύνδεση με υπολογιστή.
- Reset Button : Κάνει reset to Arduino.
- Analog pins : Αναλογικοί ακροδέκτες που λειτουργούν σαν εισοδοι .
- ATmega Microcontroller : Ο μικροελεγκτής ATmega328.
- Power Led Indicator : Led που είναι ενσωματωμένο στο Arduino , που δείχνει αν υπάρχει τροφοδοσία.
- Digital Ground : Ψηφιακή Γείωση .

- Crystal Oscillator: Ο κρύσταλλος αυτός συγχρονίζει τη λειτουργία του Arduino, αφού ενεργεί ως πηγή ρολογιού στέλνοντας σήματα ON και OFF τα οποία αλλάζουν τη κατάσταση του συστήματος.
- Digital I/O Pins : Ψηφιακοί ακροδέκτες οι οποίοι μπορούν να λειτουργήσουν και σαν είσοδοι και σαν έξοδοι . Ως έξοδοι μπορούν να παράγουν τάση 5V , και να δεχθούν ή να παρέχουν ένταση μέχρι 40mA.
- TX και RX : Ο ακροδέκτης RX είναι για την λήψη δεδομένων προς το Arduino και ο ακροδέκτης TX είναι για την αποστολή δεδομένων από το Arduino.
- Οι ακροδέκτες 2 και 3 μπορούν να χρησιμοποιηθούν και σαν εξωτερικά interrupts.
- Οι ακροδέκτες 12 και 13 μπορούν να χρησιμοποιηθούν και για την επικοινωνία του arduino , με άλλες εξωτερικές συσκευές.
- Οι ακροδέκτες 3, 5, 6, 9, 10, 11 , μπορούν να λειτουργήσουν και ως αναλογικές έξοδοι με το σύστημα PWM (Pulse Width Modulation) για να έχουμε ψηφιακούς παλμούς μεταβλητού πλάτους.

3.1.3. Λογισμικό προγραμματισμού του arduino

Το περιβάλλον προγραμματισμού ενός Arduino είναι ευέλικτο και αρκετά εύκολο στη χρήση. Είναι βασισμένο στη γλώσσα προγραμματισμού java , διαθέτει περιβάλλον για την συγγραφή προγραμμάτων , συντακτική χρωματική σήμανση ,όπως επίσης την σειριακή οθόνη που είναι χρήσιμη για την παρακολούθηση της σειριακής επικοινωνίας ανάμεσα στο Arduino και στον υπολογιστή. Μέσω αυτής μπορούμε να στείλουμε δεδομένα στο Arduino , όπως συμβολοσειρές και χαρακτήρες για την αποσφαλμάτωση προγραμμάτων του χρήστη. Τέλος διαθέτει σημαντικές βιβλιοθήκες με συναρτήσεις καθώς και παραδείγματα χρήσης διάφορων συναρτήσεων. Η γλώσσα προγραμματισμού του Arduino , είναι βασισμένη στη γλώσσα Wiring , που είναι μια παραλλαγή της C , αλλά έχουν πολλά κοινά στοιχεία μεταξύ τους. Στην παρακάτω εικόνα φαίνεται το IDE του περιβάλλοντος προγραμματισμού του Arduino.



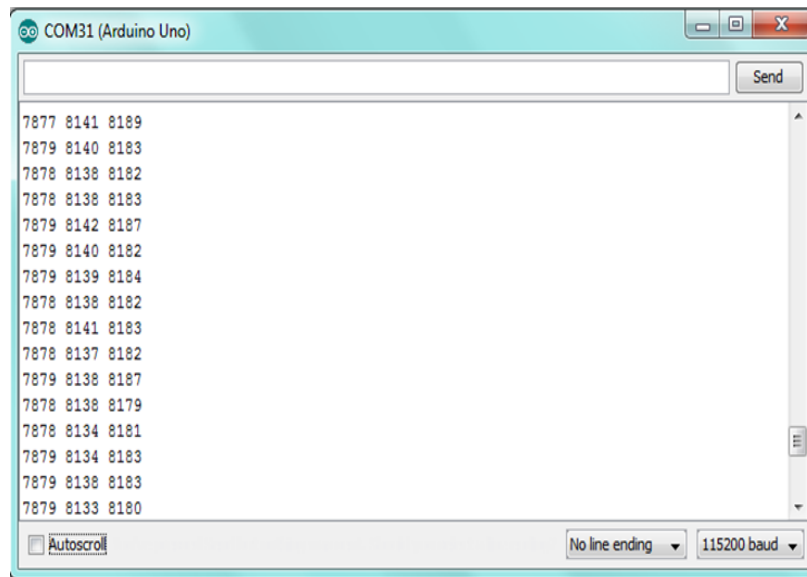
Εικόνα 5.0 Περιβάλλον του Arduino

Το compilation button της εικόνας 5.0 είναι υπεύθυνο για την μεταγλώττιση του προγράμματος του χρήστη.

Το upload button χρησιμοποιείται για να εγκαταστήσει το μεταγλωττισμένο πρόγραμμα στο Arduino .

Το Serial Monitor ανοίγει το παράθυρο επικοινωνίας του υπολογιστή με το Arduino.

Το παράθυρο του Serial Monitor φαίνεται στην εικόνα 5.1



Εικόνα 5.1 Το Serial Monitor του Arduino

Το Serial Monitor μας δίνει την δυνατότητα να λάβουμε και να στείλουμε δεδομένα απο τον υπολογιστή στο Arduino και το αντίστροφα. Η επιλογή Send μας δίνει την δυνατότητα να στείλουμε από τον υπολογιστή μια συμβολοσειρά στο Arduino , και ανάλογα το πρόγραμμα που έχουμε εγκαταστήσει στο Arduino , να περιμένουμε την κατάλληλη αντίδραση-απάντηση από αυτό. Από το παράθυρο αυτό μπορούμε επίσης να ρυθμίσουμε και τον ρυθμό μετάδοσης δεδομένων ανάμεσα στο Arduino και στον υπολογιστή (baud). Τέλος στο κέντρο του Monitor , φαίνονται οι συμβολοσειρές ή αριθμοί που στέλνει ο μικροελεγκτής ή ο υπολογιστής.

3.2. Αισθητήρες και μοτέρ

Στο κεφάλαιο αυτό παρουσιάζεται ο ηλεκτρονικός εξοπλισμός που χρησιμοποιήσαμε , όπως οι αισθητήρες και τα μοτέρ.

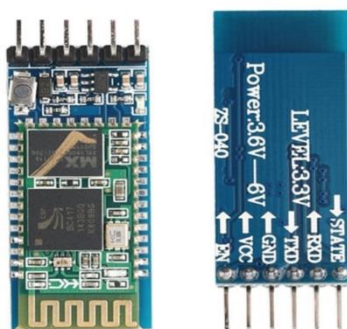
3.2.1. Πλακέτα HC – 05

Υπάρχουν πολλοί τρόποι ασύρματης επικοινωνίας μεταξύ συσκευών ,όπως το Wifi , NRF , Zigbee και Bluetooth.

Το Bluetooth , έχει μέγιστο ρυθμό μεταφοράς δεδομένων 1 Mb/sec , εμβέλεια μέχρι 200 μέτρων και λειτουργεί στο φάσμα συχνοτήτων 2,4Ghz . [21]

Με το Bluetooth μπορούμε να δημιουργήσουμε ένα PAN (personal area network) , με σκοπό τον ασύρματο έλεγχο του ρομποτικού βραχίονα. Έτσι , χρησιμοποιήσαμε την μέθοδο ασύρματης επικοινωνίας με Bluetooth , η οποία είναι αρκετά συνηθισμένη και ταυτόχρονα κατάλληλη για την συγκεκριμένη εργασία.

Για να δημιουργήσουμε ένα κανάλι επικοινωνίας μέσω Bluetooth για δύο συσκευές , αρχικά πρέπει και οι δύο οι συσκευές αυτές να διαθέτουν Bluetooth. Η συσκευή που επιλέξαμε είναι ένα smartphone , που ήδη διαθέτει Bluetooth και η συσκευή που θέλουμε να επικοινωνήσουμε είναι το Arduino , το οποίο δεν έχει . Για το λόγο αυτό πρέπει να συνδέσουμε ένα module στο Arduino ώστε να επιτευχθεί η επικοινωνία. Χρησιμοποιήσαμε την πλακέτα HC-05 Bluetooth module όπως φαίνεται στην εικόνα 5.2.



Εικόνα 5.2 HC-05 Bluetooth Module

Η πλακέτα αυτή , χρησιμοποιεί το πρωτόκολλο επικοινωνίας Bluetooth SPP (Serial Port Protocol) και διαθέτει έξι ακροδέκτες με τις εξής σημάνσεις: [22]

- VCC
- GND
- TXD
- RXD
- STATE
- EN

1. VCC :

Η ένδειξη Vcc , συμβολίζει την τροφοδοσία . Για να μπει σε λειτουργία η πλακέτα , πρέπει ο ακροδέκτης να τροφοδοτηθεί με τάση . Η τάση λειτουργίας είναι τα 3,3V , αλλά μπορεί να τροφοδοτηθεί και με 5V , λόγω της ύπαρξης του ρυθμιστή τάσης της πλακέτας , που μπορεί να μειώσει τα 5V σε 3,3V.

2. GND :

Η ένδειξη Gnd , συμβολίζει την γείωση . Είναι απαραίτητη για κλείσει το κύκλωμα ώστε να τροφοδοτηθεί η πλακέτα σε συνδυασμό με το Vcc.

3. TXD :

Ο ακροδέκτης με την ένδειξη TXD , χρησιμοποιείται για την αποστολή δεδομένων

4. RXD:

Ο ακροδέκτης με την ένδειξη RXD , είναι απαραίτητος για την λήψη δεδομένων

5. STATE:

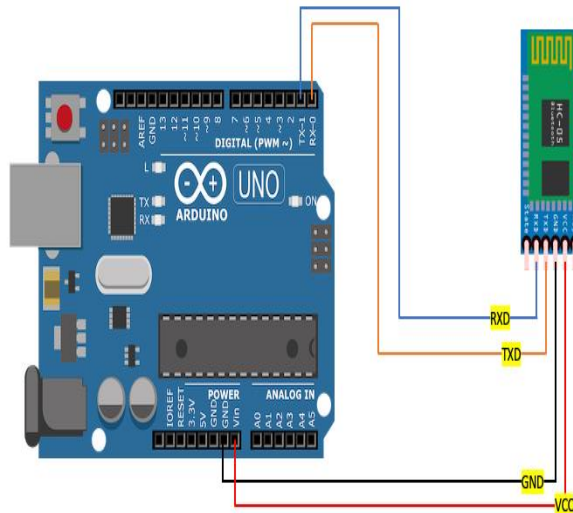
Ο ακροδέκτης με την ένδειξη STATE , μπορεί να χρησιμοποιηθεί απο το Arduino για να γίνει έλεγχος αν υπάρχει ενεργή σύνδεση.

6. EN:

Ο ακροδέκτης με την ένδειξη EN , μπορεί να χρησιμοποιηθεί από το Arduino για την εναλλαγή από Data Mode σε AT command mode και αντίστροφα.

Για την σύνδεση , θα χρησιμοποιήσουμε τέσσερις ακροδέκτες , τους VCC,GND,RXD και TXD. Τους ακροδέκτες STATE και EN δεν θα τους χρησιμοποιήσουμε.

Η σύνδεση φαίνεται στην εικόνα 5.3.



Εικόνα 5.3 Σύνδεση HC-05 – Arduino uno

Ο ακροδέκτης με σήμανση VCC μπορεί να συνδεθεί με τον θηλυκό ακροδέκτη του Arduino με σήμανση 5V , 3,3V ή στον ακροδέκτη Vin αν έχουμε εξωτερική τροφοδοσία.

Ο ακροδέκτης με σήμανση GND συνδέθηκε με τον θηλυκό ακροδέκτη με σήμανση GND.

Ο ακροδέκτης με σήμανση RXD συνδέθηκε με τον θηλυκό ακροδέκτη με σήμανση TXD.

Ο ακροδέκτης με σήμανση TXD συνδέθηκε με τον θηλυκό ακροδέκτη με σήμανση RXD.

3.2.2. Μοτέρ τύπου SG-90

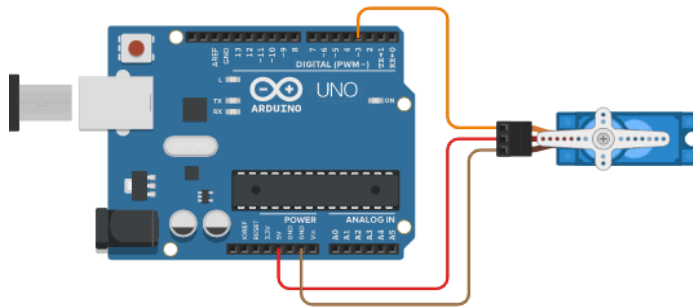
Τα σερβομοτέρ , είναι ειδικά σχεδιασμένα μοτέρ , στα οποία μπορούμε να ρυθμίσουμε την γωνιακή τους περιστροφή με ακρίβεια. Αποτελούνται από τον κινητήρα , τον αισθητήρα θέσης και τον ελεγκτή. Ο κινητήρας ελέγχεται από τον σερβομηχανισμό και διακρίνονται σε DC servo και AC servo , ανάλογα τον τύπο του κινητήρα αν λειτουργεί με συνεχές ρεύμα ή εναλλασσόμενο.

Χρησιμοποιήσαμε δύο σερβομοτέρ τύπου SG-90 , για την λειτουργία της αρπαγής και του καρπού του βραχίονα. Το μοτέρ φαίνεται στην εικόνα 5.4.



Εικόνα 5.4 Σερβομοτέρ τύπου SG-90

Το SG-90 , έχει δυνατότητα περιστροφής , από 0 έως 180 μοίρες και διαθέτει τρεις θηλυκούς ακροδέκτες .Η μέγιστη ροπή του είναι 2,5 kg*cm και η τάση λειτουργίας του είναι από 4,8V ως 6V.[23] Τα καλώδια καταλήγουν στους ακροδέκτες , τα οποία διαθέτουν χρωματική σήμανση για τον διαχωρισμό τους. Στο κόκκινο καλώδιο συνδέεται η τροφοδοσία , το καφέ καλώδιο στην γείωση , ενώ το πορτοκαλί καλώδιο σε έναν ψηφιακό ακροδέκτη του Arduino , για τον έλεγχο της περιστροφής. Η σύνδεσή του με το Arduino φαίνεται στην εικόνα 5.5.



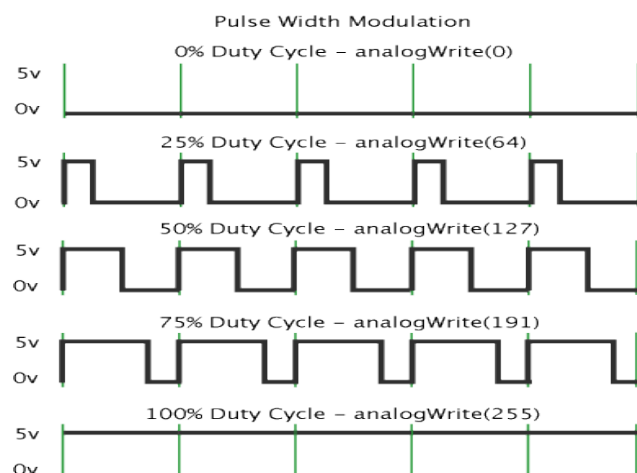
Εικόνα 5.5 Σύνδεση arduino με το μοτέρ SG-90

Το κόκκινο καλώδιο μπορεί να συνδεθεί με τον θηλυκό ακροδέκτη του arduino με σήμανση 5V ή στον ακροδέκτη Vin αν έχουμε εξωτερική τροφοδοσία.

Το καφέ καλώδιο συνδέεται με τον θηλυκό ακροδέκτη με σήμανση GND.

Το πορτοκαλί καλώδιο μπορεί να συνδεθεί σε οποιοδήποτε ψηφιακό ακροδέκτη PWM.

Είναι απαραίτητο το πορτοκαλί καλώδιο να συνδεθεί σε έναν ακροδέκτη PWM. Με τους παλμούς αυτούς, μπορούμε να προσομοιώσουμε ένα αναλογικό σήμα με ψηφιακό τρόπο. Ένας ακροδέκτης PWM, μπορεί να δημιουργήσει έναν τετραγωνικό παλμό για έναν συγκεκριμένο χρόνο. Η παρακάτω εικόνα 5.6 δείχνει την λειτουργία των παλμών αυτών.



Εικόνα 5.6 PWM duty cycles

Στην εικόνα 5.6 , βλέπουμε την συμπεριφορά των παλμών. Στην ουσία ένας τετραγωνικός παλμός λειτουργεί σαν ένας διακόπτης. Στην ανοιχτή λειτουργία του έχουμε τάση 5V , ενώ στην κλειστή 0V. Ο χρόνος για να γίνει αυτή η εναλλαγή ονομάζεται πλάτος του παλμού. Έτσι ανάλογα με το πλάτος του παλμού , μπορούμε να έχουμε και διαφορετική συμπεριφορά στο σερβομοτέρ.

Στο συγκεκριμένο μοτέρ , το duty cycle , κυμαίνεται από 1ms ως 2ms. Στη θέση 90 μοιρών το duty cycle πρέπει να είναι 2ms , στη θέση -90 μοιρών 1ms , στη θέση 0 μοιρών 1.5ms ,ενώ στις ενδιάμεσες θέσεις διαιρείται ανάλογα. [23]

3.2.3. Μοτέρ τύπου MG-996R

Το μοτέρ αυτό ανήκει και αυτό στην κατηγορία των σερβομοτέρ ,όπως και το SG-90. Ο τρόπος λειτουργίας τους είναι παραπλήσιος , με τις διαφορές τους να βρίσκονται στην ποιότητα των υλικών κατασκευής , το μέγεθος του κινητήρα και την διαφορά τους στην ροπή.

Το MG-996R διαθέτει μεγαλύτερη ροπή από το SG-90 , που σημαίνει πως μπορεί να αντέχει μεγαλύτερο φόρτο σε συγκεκριμένη απόσταση από τον άξονα περιστροφής του. Αυτό το καθιστά κατάλληλο για τα τρία σημεία ελευθερίας του βραχίονα που έχουν και τον μεγαλύτερο φόρτο. Έτσι χρησιμοποιήσαμε τρία μοτέρ αυτού του τύπου , ένα για την περιστροφή της βάσης , ένα για την κίνηση του ώμου και τέλος ένα για την κίνηση του αγκώνα. Το μοτέρ το παραθέτουμε στην παρακάτω εικόνα 5.7.



Εικόνα 5.7 Μοτέρ τύπου MG-996R

Οι συνδέσεις με το Arduino είναι ίδιες με το μοτέρ του κεφαλαίου 3.2.2 όπως και ο τρόπος λειτουργίας του. Τα χαρακτηριστικά του μοτέρ αναφέρονται παρακάτω. [24]

Μέγιστη ροπή στα 4.8V : 13kg*cm

Μέγιστη ροπή στα 6V : 15kg*cm

Τάση λειτουργίας : 4.8 - 7.2V.

Τύπος γραναζιών : μεταλλικοί

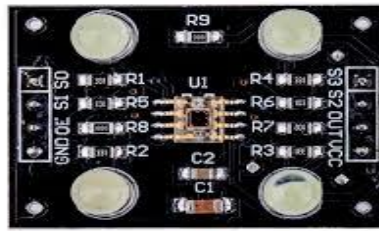
Βάρος : 55g

Ταχύτητα λειτουργίας στα 4.8V : 60 μοίρες / 0.17 sec χωρίς φόρτο

Ταχύτητα λειτουργίας στα 6V : 60 μοίρες / 0.13 sec χωρίς φόρτο

3.2.4. Αισθητήρας χρώματος τύπου TCS3200

Ο αισθητήρας χρώματος TCS 3200 , χρησιμοποιήθηκε για να μπορεί να αναγνωρίσει τα χρώματα των κύβων , ώστε ο ρομποτικός βραχίονας να μπορεί να τα ταξινομήσει .Έχει την δυνατότητα να αναγνωρίσει οποιονδήποτε συνδυασμό χρωμάτων RGB . Διαθέτει έναν πίνακα 8x8 , από φωτοδιόδους , δηλαδή 64 σένσορες χρώματος. Αυτοί οι φωτοδιόδοι είναι καλυμμένοι με 4 διαφορετικά είδη φίλτρων . Οι 16 φωτοδιόδοι έχουν κόκκινο φίλτρο και μπορούν να ανιχνεύσουν την παρουσία του κόκκινου χρώματος. Οι άλλοι 16 φωτοδιόδοι έχουν μπλε φίλτρο και μπορούν να ανιχνεύσουν την ποσότητα του μπλε χρώματος. Το ίδιο ισχύει και για τους υπόλοιπους φωτοδιόδους για το πράσινο χρώμα. Οι τελευταίοι 16 από τους 48 , έχουν καθαρό φίλτρο. Τέλος ο αισθητήρας μπορεί να μετατρέψει την ένταση της κάθε ακτινοβολίας σε συχνότητα , έτσι ανάλογα με τον συνδυασμό των συχνοτήτων των χρωμάτων RGB , μπορούμε να αποφανθούμε για το χρώμα που εντοπίζει ο αισθητήρας. [25] Ο αισθητήρας φαίνεται στην εικόνα 5.8.



Εικόνα 5.8 TCS 3200 Color detector

Ο TCS 3200 , έχει οχτώ ακροδέκτες :

- VCC
- GND
- OUT
- S0
- S1
- S2
- S3
- LED
-

1. VCC :

Η ένδειξη Vcc , συμβολίζει την τροφοδοσία . Για να μπει σε λειτουργία η πλακέτα , πρέπει ο ακροδέκτης να τροφοδοτηθεί με τάση . Η τάση λειτουργίας είναι τα 5V.

2. GND :

Η ένδειξη Gnd , συμβολίζει την γείωση . Είναι απαραίτητη για να κλείσει το κύκλωμα ώστε να τροφοδοτηθεί η πλακέτα σε συνδυασμό με το Vcc.

3. OUT :

Ο ακροδέκτης με την ένδειξη OUT , χρησιμοποιείται για την αποστολή δεδομένων της συχνότητας που διάβασε ο αισθητήρας.

4. S0:

Ο ακροδέκτης με την ένδειξη S0 ,σε συνδυασμό με τον ακροδέκτη S1, είναι απαραίτητοι για την ρύθμιση του ποσοστού της συχνότητας που διαβάζεται.

5. S1:

Ο ακροδέκτης με την ένδειξη S1 ,σε συνδυασμό με τον ακροδέκτη S0, είναι απαραίτητοι για την ρύθμιση του ποσοστού της συχνότητας που διαβάζεται

6. S2:

Ο ακροδέκτης με την ένδειξη S2 ,σε συνδυασμό με τον ακροδέκτη S3, ρυθμίζουν το ποια κατηγορία φωτοδιόδων μπορούν να λειτουργήσουν κάθε φορά.

7. S3:

Ο ακροδέκτης με την ένδειξη S3 ,σε συνδυασμό με τον ακροδέκτη S2, ρυθμίζουν το ποια κατηγορία φωτοδιόδων μπορούν να λειτουργήσουν κάθε φορά.

8. LED:

Ο ακροδέκτης με την ένδειξη LED , θέτει σε λειτουργία τα ενσωματωμένα LED που διαθέτει ο αισθητήρας

Οι πίνακες 1.2 και 1.2 δείχνουν όλους τους συνδυασμούς των S0-S1 και S1-S2:

Πίνακας 1.1 Συνδυασμοί S0-S1

S0	S1	OUTPUT FREQUENCY SCALING
LOW	LOW	Power down
LOW	HIGH	2%
HIGH	LOW	20%
HIGH	HIGH	100%

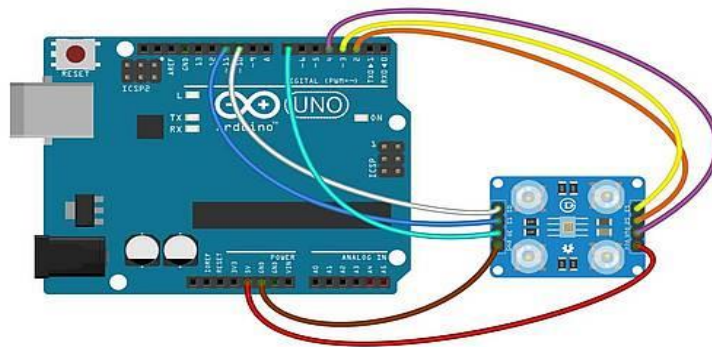
Πίνακας 1.2 Συνδυασμοί S1-S2

S2	S3	PHOTODIODE TYPE
LOW	LOW	RED
LOW	HIGH	BLUE
HIGH	LOW	CLEAR
HIGH	HIGH	GREEN

Η ένδειξη HIGH στους παραπάνω πίνακες συμβολίζει παρουσία τάσης 5V.

Η ένδειξη LOW στους παραπάνω πίνακες συμβολίζει απουσία τάσης .

Η παρακάτω εικόνα 5.9 δείχνει την συνδεσμολογία του αισθητήρα με το Arduino:



Εικόνα 5.9 συνδεσμολογία tcs3200 με το Arduino

Ο ακροδέκτης με την ένδειξη VCC , συνδέεται με την ένδειξη 5V του Arduino ή με την ένδειξη Vin αν υπάρχει εξωτερική τροφοδοσία.

Ο ακροδέκτης με την ένδειξη LED , συνδέεται με την ένδειξη 5V ή με την ένδειξη Vin αν υπάρχει εξωτερική τροφοδοσία για την τροφοδοσία των ενσωματωμένων LED.

Ο ακροδέκτης με την ένδειξη GND , συνδέεται με την ένδειξη της γείωσης GND .

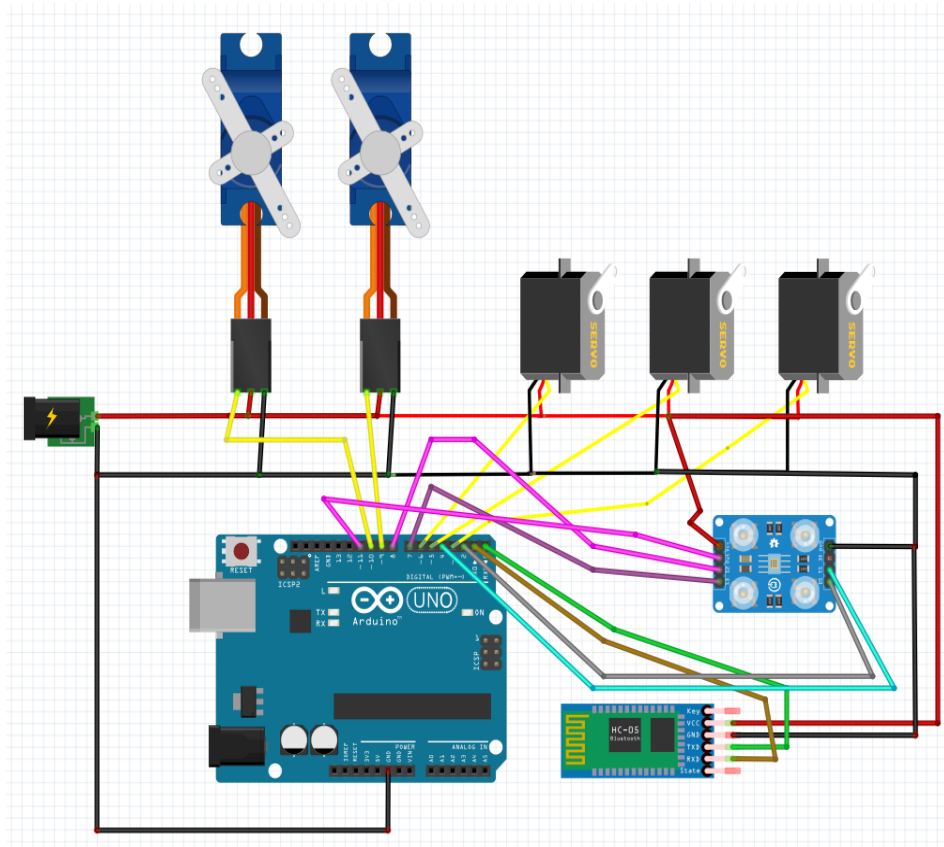
Οι υπόλοιποι ακροδέκτες (OUT,S0,S1,S2,S3) του αισθητήρα μπορούν να συνδεθούν σε οποιοδήποτε ψηφιακό ακροδέκτη που διαθέτει το Arduino.

3.3. Τροφοδοσία

Στο κεφάλαιο αυτό θα δείξουμε την απεικόνιση της συνδεσμολογίας που περιλαμβάνει όλον το ηλεκτρονικό εξοπλισμό , όπως και την πηγή τροφοδοσίας του μικροελεγκτή , των αισθητήρων και των μοτέρ.

3.3.1. Το κύκλωμα στο Arduino

Παρακάτω, εικόνα 6.0 , βλέπουμε την απεικόνιση της συνδεσμολογίας όλου του ηλεκτρικού εξοπλισμού με το Arduino.



Εικόνα 6.0 Συνδεσμολογία στο Arduino

Στην εικόνα 6.0 , βλέπουμε όλον το εξοπλισμό που χρησιμοποιήθηκε , όπως τα μοτέρ τύπου SG-90 και MG-996R , τον αισθητήρα χρώματος , το Arduino , την πλακέτα HC-05, την εξωτερική τροφοδοσία και όλες τις συνδέσεις μεταξύ αυτών.

3.3.2. Τροφοδοσία του ηλεκτρονικού εξοπλισμού

Το Arduino τροφοδοτήθηκε από την θύρα επικοινωνίας με τον υπολογιστή (USB τύπου 3.0 που παράγει τάση 5V) . Η τροφοδοσία όλου του υπόλοιπου ηλεκτρονικού εξοπλισμού , έγινε με την χρήση ενός τροφοδοτικού συνεχούς τάσης 5V και έντασης 8A . Η Εξωτερική πηγή ενέργειας χρησιμοποιήθηκε διότι το Arduino δεν είναι ικανό να παρέχει αρκετή ενέργεια για την λειτουργία όλων των εξαρτημάτων. Το Τροφοδοτικό που επιλέξαμε φαίνεται στην εικόνα 6.1.



Εικόνα 6.1 Τροφοδοτικό 5V 8A DC

3.4. Αλγόριθμοι ταξινόμησης

Στο παρόν κεφάλαιο , θα εξηγήσουμε την έννοια , την λειτουργία και την χρησιμότητα των αλγορίθμων Ταξινόμησης . Θα εξετάσουμε σε βάθος δύο από αυτούς ,οι οποίοι υλοποιήθηκαν και στην συγκεκριμένη εργασία.

3.4.1. Τι είναι οι αλγόριθμοι ταξινόμησης

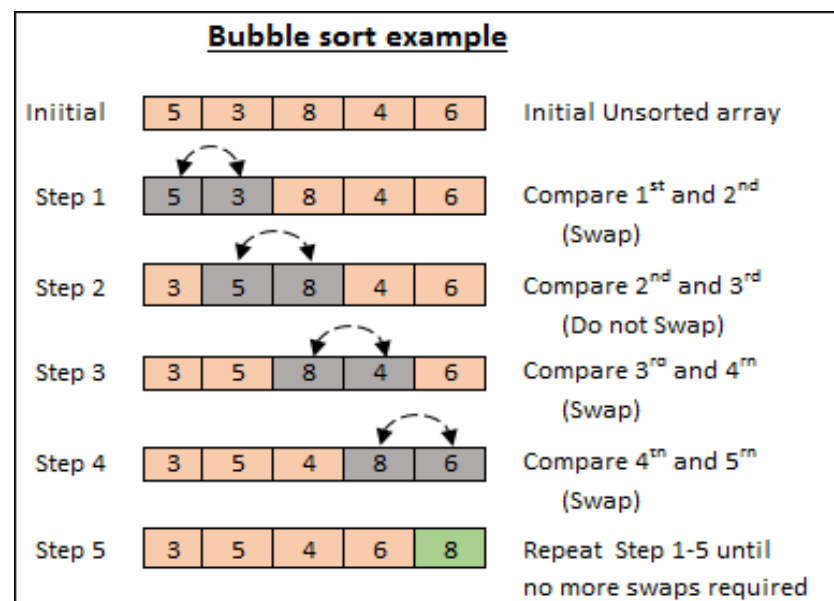
Στην επιστήμη των υπολογιστών , οι αλγόριθμοι ταξινόμησης , αποτελούν κατηγορία αλγορίθμων , που μεταθέτουν στοιχεία μιας ακολουθίας με σκοπό να έχουν μια επιθυμητή σειρά. Η επιθυμητή σειρά μπορεί να έχει πολλές μορφές , όπως αριθμητική αλφαβητική χρωματική και άλλες. Εφόσον είναι αλγόριθμοι , σημαίνει πως ακολουθούν κάποιους συγκεκριμένους κανόνες για να πετύχουν το επιθυμητό

αποτέλεσμα . Βασικό τους χαρακτηριστικό , είναι η πολυπλοκότητά τους , που συμβολίζει τις χρονικές μονάδες που χρειάζονται για να ολοκληρωθεί η ταξινόμηση και ο χώρος που δεσμεύουν. Μερικοί από τους πιο γνωστούς αλγορίθμους είναι οι bubble sort , quick sort ,selection sort , insertion sort , heap sort και merge sort. [26]

3.4.2. Ο αλγόριθμος bubble sort

Ένας από τους αλγορίθμους που επιλέχθηκε να υλοποιηθεί στην παρούσα εργασία είναι ο bubblesort . Στην περίπτωση μας , θα χρησιμοποιηθεί για την ταξινόμηση των κύβων που κατασκευάσαμε στο κεφάλαιο 2 , από τον ρομποτικό βραχίονα . Διαθέτουμε 5 κύβους , οι οποίοι είναι χρωματισμένοι με 5 διαφορετικά χρώματα , τα οποία μπορούν να αναγνωριστούν από τον αισθητήρα χρώματος , με σκοπό τη χρήση του συγκεκριμένου αλγορίθμου για την ταξινόμησή τους από το πιο ανοιχτό χρώμα προς το πιο σκούρο. Παρακάτω δίνεται ένα παράδειγμα της εκτέλεσης του αλγορίθμου bubble sort , με ταξινόμηση αριθμών , από τον μικρότερο στον μεγαλύτερο. Παρόμοια λειτουργεί και στην ταξινόμηση χρωμάτων.

Στην παρακάτω εικόνα 6.2 απεικονίζεται ένας πίνακας αριθμών , 5 στοιχείων , μη ταξινομημένος και τα βήματα εκτέλεσης του αλγορίθμου: [27]



Εικόνα 6.2 Εκτέλεση του bubblesort

Ο αλγόριθμος θα ακολουθήσει συγκεκριμένα βήματα για την ταξινόμηση του παραπάνω πίνακα , τα οποία αναλύονται παρακάτω:

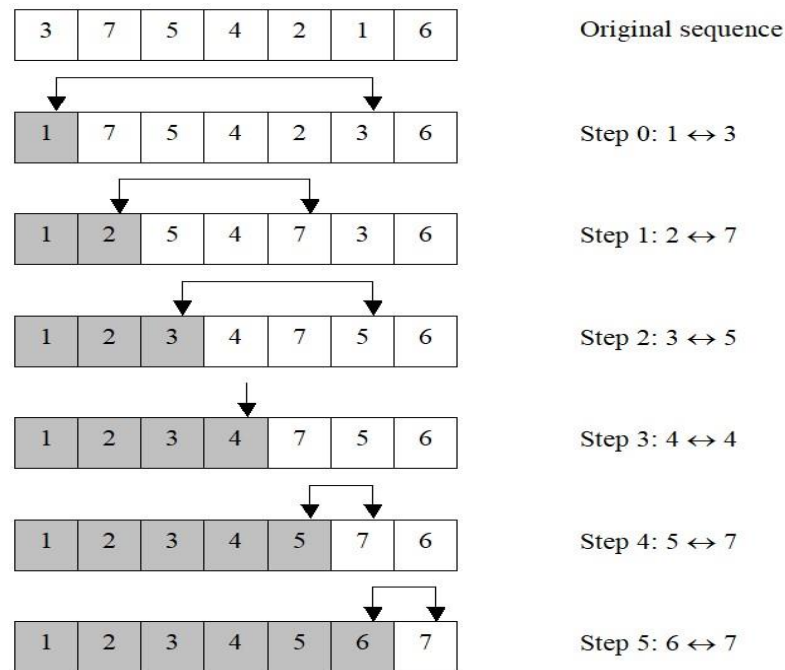
Αρχικά διαβάζονται οι δύο πρώτες θέσεις του πίνακα , που περιέχουν τους αριθμούς 5 και 3 . Ο αλγόριθμος συγκρίνει τους 2 αριθμούς και αν ο πρώτος αριθμός που διαβάστηκε , δηλαδή ο 5 , είναι μεγαλύτερος από τον δεύτερο αριθμό που διαβάστηκε, δηλαδή τον 2 , τότε τους αντιμεταθέτει όπως φαίνεται στην εικόνα 6.2 στο Step 1. Ο αλγόριθμος θα συνεχίσει , συγκρίνοντας τώρα το δεύτερο στοιχείο του πίνακα με το τρίτο . Στο συγκεκριμένο παράδειγμα διαβάζονται οι αριθμοί 5 και 8 , με τον πρώτο να είναι μικρότερος από τον δεύτερο , οπότε δεν θα γίνει αντιμετάθεσή τους, όπως βλέπουμε στο Step 2. Η διαδικασία θα συνεχίσει μέχρι να συγκριθεί και το στοιχείο της προτελευταίας θέσης του πίνακα , με την τελευταία . Αν καθ όλη τη διάρκεια ενός περάσματος του πίνακα , δεν γίνει αντιμετάθεση κανενός στοιχείου , ο αλγόριθμος τελειώνει . Στην περίπτωση όμως που συνέβη αντιμετάθεση , ο αλγόριθμος θα εκτελέσει και άλλο πέρασμα στον πίνακα, μέχρι να υπάρξει ένα πέρασμα που δεν συνέβη καμία αντιμετάθεση όπως φαίνεται στο Step 5.

Η χρονική πολυπλοκότητα του αλγορίθμου είναι τετραγωνική , δηλαδή $O(n^2)$.

Η πολυπλοκότητα σε χώρο του αλγορίθμου είναι σταθερή , δηλαδή $O(1)$.

3.4.3. Ο αλγόριθμος selection sort

Ο δεύτερος αλγόριθμος που επιλέξαμε να υλοποιηθεί είναι ο selection sort . Ο τρόπος λειτουργίας σε σχέση με τον bubblesort είναι διαφορετικός και η λειτουργία του φαίνεται στην εικόνα 6.3. [28]



Εικόνα 6.3 Η εκτέλεση του selection Sort

Ο αλγόριθμος ξεκινά αποθηκεύοντας το στοιχείο της πρώτης θέσης του πίνακα . Έπειτα διαβάζει όλα τα στοιχεία του πίνακα που βρίσκονται δεξιά της θέσης που βρίσκεται το πρώτο στοιχείο που αποθήκευσε , βρίσκει το μικρότερο και αν αυτό είναι και μικρότερο του πρώτου στοιχείου που έχει αποθηκεύσει , τότε τα αλλάζει θέσεις , όπως φαίνεται στο Step 0. Τώρα , γνωρίζουμε πως στην πρώτη θέση του πίνακα , υπάρχει το μικρότερο στοιχείο του πίνακα. Ο αλγόριθμος προχωρά τώρα στην δεύτερη θέση του πίνακα , και επαναλαμβάνει την ίδια διαδικασία .Δηλαδή αποθηκεύει το στοιχείο που βρίσκεται στη δεύτερη θέση του πίνακα , και ελέγχει όλα τα στοιχεία των θέσεων του πίνακα , που βρίσκονται δεξιά από τη θέση αυτή . Βρίσκει το μικρότερο από αυτά ,και τέλος το συγκρίνει με αυτό της δεύτερης θέσης . Αν αυτό της δεύτερης θέσης είναι μεγαλύτερο , τότε τα αντιμεταθέτει , όπως φαίνετε στο Step 1 . Η παραπάνω διαδικασία επαναλαμβάνεται τόσες φορές , όσες και οι θέσεις που έχει ο πίνακας που θέλουμε να ταξινομήσουμε -1. Στην περίπτωσή μας δηλαδή $7-1=6$ φορές. Έπειτα ο αλγόριθμος τελειώνει και ο πίνακας έχει ταξινομηθεί.

Η χρονική πολυπλοκότητα του αλγορίθμου είναι τετραγωνική , δηλαδή $O(n^2)$.

Η πολυπλοκότητα σε χώρο του αλγορίθμου είναι σταθερή , δηλαδή $O(1)$.

3.5. Υλοποίηση του προγράμματος σε Arduino

Η παράγραφος αυτή περιλαμβάνει τα πιο σημαντικά τμήματα του κώδικα του προγράμματος που εγκαταστάθηκαν στο Arduino , για τον έλεγχο του ρομποτικού βραχίονα . Παρατίθενται εικόνες που απεικονίζουν τον κώδικα , καθώς και μια μικρή ανάλυση του κώδικα.

3.5.1. Κώδικας που εγκαταστάθηκε στο Arduino

Στην εικόνα 6.4 παρακάτω , απεικονίζεται ο κώδικας που είναι υπεύθυνος για την κίνηση των μοτέρ.

```
void motorPositions(Servo motor,int pos)//function to move robot depending on motor number and integer
{
    int i;
    int counter = motor.read();
    if(counter < pos)
    {
        for(i=counter;i<pos;i++)
        {
            motor.write(i);
            delay(10);
        }
    }
    else if(counter > pos)
    {
        for(i=counter;i>pos;i--)
        {
            motor.write(i);
            delay(10);
        }
    }
    else{};
}
```

Εικόνα 6.4 Συνάρτηση κίνησης των μοτέρ

Η παραπάνω συνάρτηση κινεί ένα συγκεκριμένο μοτέρ στην επιθυμητή γωνιακή θέση. Δέχεται σαν όρισμα ένα αντικείμενο τύπου Servo , δηλαδή το μοτέρ που θέλουμε να κινήσουμε και ένα όρισμα τύπου int , που δηλώνει το ποιά να είναι η νέα γωνιακή του θέση.

Στην εικόνα 6.5 απεικονίζεται ο κώδικας που καθορίζει την συμπεριφορά του αισθητήρα χρώματος.

```
String checkColour()//check the color from sensor , returns a string of the colour detected
{
    int j;
    for(j=0;j<5;j++)//5 times loop
    {

        int colorMatrix[6]={0,0,0,0,0,0};
        int outputred=0,outputgreen=0,outputblue=0,i;//colour output each time
        Serial.println();//newline to serial monitor

        digitalWrite(S2,LOW);//pinmodes for checking red colour
        digitalWrite(S3,LOW);

        for(i=0;i<6;i++)//6 times loop , stores different values in colorMatrix matrix
        {
            colorMatrix[i] = pulseIn(Out, LOW);
            delay(10);
        }
        Serial.print(" R = ");
        for(i=0;i<6;i++)//here calculating median value of all the values of colorMatrix
        {
            outputred=outputred + colorMatrix[i];
        }
        outputred=int(outputred/6);
        Serial.print(outputred);

        digitalWrite(S2,HIGH);//pinmodes for checking green colour
        digitalWrite(S3,HIGH);

        for(i=0;i<6;i++)
        {
            colorMatrix[i] = pulseIn(Out, LOW);
            delay(10);
        }
        Serial.print(" G = ");
        for(i=0;i<6;i++)
        {
            outputgreen=outputgreen + colorMatrix[i];
        }
        outputgreen=int(outputgreen/6);
        Serial.print(outputgreen);
```

```

digitalWrite(S2,LOW);
digitalWrite(S3,HIGH);

for(i=0;i<6;i++)
{
    colorMatrix[i] = pulseIn(Out, LOW);
    delay(10);
}
Serial.print(" B = ");
for(i=0;i<6;i++)
{
    outputblue=outputblue + colorMatrix[i];
}
outputblue=int(outputblue/6);
Serial.print(outputblue);

if(outputred<=13 && outputgreen<=13 && outputblue<=10)
{
    //Serial.println("WHITE"); debug
    counterMatrix[0]++; //increment a matrix to decide finally the colour
}
else if(outputred<=20)
{
    //Serial.println("RED"); debug
    counterMatrix[1]++; //increment a matrix to decide finally the colour
}
else if(outputblue<=22)
{
    //Serial.println("BLUE"); debug
    counterMatrix[2]++; //increment a matrix to decide finally the colour
}
else if(outputred>=22 && outputred<=44 && outputgreen>=22
&& outputgreen<=39 && outputblue>=23 && outputblue<=30)
{
    //Serial.println("GREEN"); debug
    counterMatrix[4]++; //increment a matrix to decide finally the colour
}
else
{
    //Serial.println("BLACK"); debug
    counterMatrix[3]++; //increment a matrix to decide finally the colour
}
}

```

```

int index;
int maxval=counterMatrix[0];
for(j=0;j<5;j++)//find max value from the matrix
{
    if(counterMatrix[j]>=maxval)
    {
        maxval = counterMatrix[j];
        index=j;//store index of the max value
    }
}

for(int j=0;j<5;j++)//make matrix elements zero
{
    counterMatrix[j]=0;
}

if(index==0)//if index
{
    Serial.println("WHITE");
    return "WHITE";
}
else if(index==1)
{
    Serial.println("RED");
    return "RED";
}
else if(index==2)
{
    Serial.println("BLUE");
    return "BLUE";
}
else if(index==3)
{
    Serial.println("BLACK");
    return "BLACK";
}
else if(index==4)
{
    Serial.println("GREEN");
    return "GREEN";
}
else {
    Serial.println(""); //debug
    return "I"; //debug
}

```

Εικόνα 6.5 Κώδικας αισθητήρα χρώματος

Στα παραπάνω τμήματα Κώδικα , ελέγχουμε το χρώμα των χρωματιστών κύβων ,με τον αισθητήρα χρώματος . Η συνάρτηση , επιστρέφει μια συμβολοσειρά , με το χρώμα που ανίχνευσε ο αισθητήρας.

Τέλος , το παρακάτω τμήμα κώδικα της εικόνας 6.6 , αποτελεί τον κώδικα του αλγορίθμου bubblesort.

```

void bubbleSort()
{
    initialPosition();//initialposition
    while(1)
    {
        for(int j=0;j<number_of_cubes-1;j++)
        {
            gcolposition1 = fixedColourPositions[j];//value from global matrix
            gcolposition2 = fixedColourPositions[j+1];
            medianPosition();
            positioncolour(gcolposition1);
            colourDetected1 = checkColour();//colour at position
            medianPosition();
            positioncolour(gcolposition2);
            colourDetected2 = checkColour();
            motorPositions(motor4,70);//motor 4 parallel
            gcounter = intCompareColours(colourDetected1,colourDetected2);
            if(gcounter==1)
            {
                positiongrab(gcolposition2);
                grab();
                medianPosition();
                positiontemp();
                releasegrip();
                medianPosition();
                positiongrab(gcolposition1);
                grab();
                medianPosition();
                positiongrab(gcolposition2);
                releasegrip();
                medianPosition();
                positiontemp();
                grab();
                medianPosition();

                positiongrab(gcolposition1);
                releasegrip();
                medianPosition();
                bubblesort=1;
            }
            else{};
        }
        if(bubblesort==0)
        {
            break;
        }
        else
        {
            bubblesort=0;
        }
    }
    return;
}

```

```

        positiongrab(gcolposition1);
        releasegrip();
        medianPosition();
        bubblesort=1;
    }
    else{};
}
if(bubblesort==0)
{
    break;
}
else
{
    bubblesort=0;
}
}
return;
}

```

Εικόνα 6.6 Κώδικας Bubblesort

4. Δημιουργία της εφαρμογής ασύρματου ελέγχου για android

4.1. Το Android studio

Σε αυτή την κατηγορία , θα εξηγήσουμε την λειτουργία και τη χρήση του λογισμικού Android studio και θα αναλυθούν τα τμήματα μιας εφαρμογής σε Android.

4.1.1. Τί είναι το Android studio

Το Android studio , αποτελεί ένα προγραμματιστικό περιβάλλον , που χρησιμοποιείται για την κατασκευή εφαρμογών για το λειτουργικό σύστημα Android . Βασίζεται στο λογισμικό της JetBrains' IntelliJ IDEA και είναι διαθέσιμο για Linux , Mac Os και Windows. Το Android αποτελεί λειτουργικό σύστημα σε πολλές σύγχρονες συσκευές , όπως των smartphone και των tablet . Επιλέξαμε αυτό για την δημιουργία της εφαρμογής ασύρματου ελέγχου του βραχίονα , λόγω των υψηλών δυνατοτήτων και ευελιξίας που διαθέτει. [27]

4.1.2. Ανάλυση των τμημάτων μιας εφαρμογής σε Android studio

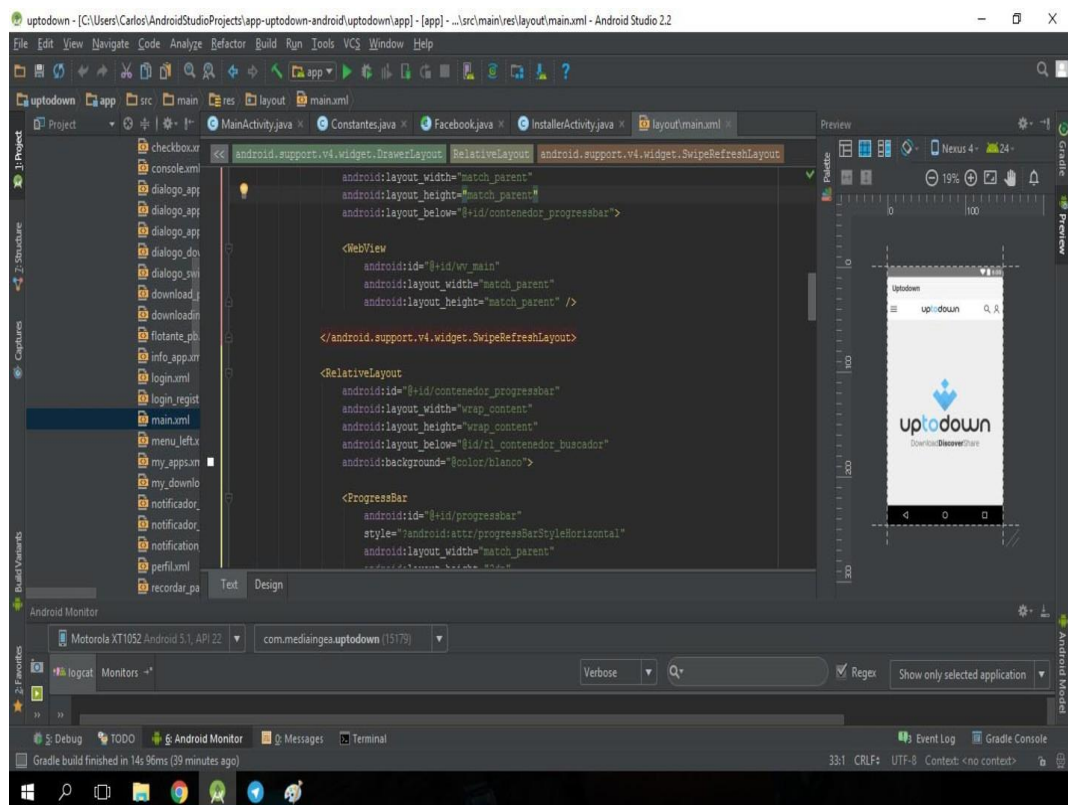
Όλες οι εφαρμογές σε Android , χαρακτηρίζονται από τα τμήματα που τις αποτελούν. Παρακάτω στην εικόνα 6.7 βλέπουμε τους φακέλους που περιλαμβάνονται σε μια εφαρμογή Android.

 .gradle	02-Jan-21 3:37 AM	File folder
 .idea	21-Jan-21 9:28 AM	File folder
 app	02-Jan-21 4:00 AM	File folder

Εικόνα 6.7 Οι φάκελοι μιας εφαρμογής Android

Εμείς θα εστιάσουμε στον φάκελο app που περιέχει βασικά αρχεία της εφαρμογής. Ο φάκελος app περιλαμβάνει τρεις φακέλους. Τον φάκελο Libs, build και src . Ο libs περιέχει όλες τις βιβλιοθήκες που χρησιμοποιήσαμε κατά την διάρκεια κατασκευής της εφαρμογής , ο φάκελος build έχει όλες τις κλάσεις που δημιουργήθηκαν όπως και άλλους πόρους για την σωστή λειτουργία της εφαρμογής. Τέλος , ο φάκελος src περιέχει τον κώδικα της κυρίου προγράμματος όπως και άλλους φακέλους όπως οι drawable , menu , layout , values που περιέχουν εικόνες και συμβολοσειρές. Πιο συγκεκριμένα ο drawable , περιέχει διάφορα εικονίδια και εικόνες που έχει εφαρμογή. Ο φάκελος layout έχει ένα αρχείο που ονομάζεται activity_main.xml. Αυτό το αρχείο καθορίζει την εξωτερική εμφάνιση . Ο φάκελος minipmap έχει τα εικονίδια με επέκταση .png που περιέχονται στην εφαρμογή και τέλος ο φάκελος values έχει τα βασικά αρχεία colors.xml και strings.xml , που είναι υπεύθυνα για το χρώμα και για τις συμβολοσειρές αντίστοιχα. Τέλος κάθε εφαρμογή Android , διαθέτει ένα αρχείο που ονομάζεται AndroidManifest.xml όπου καθορίζει την δομή και τις ανάγκες της εφαρμογής.

Στην παρακάτω εικόνα 6.8 απεικονίζεται το περιβάλλον του Android studio:



Εικόνα 6.8 Το περιβάλλον του Android studio

4.2. Κατασκευή του Προγράμματος

Στο κεφάλαιο αυτό θα παρουσιάσουμε τον κώδικα της εφαρμογής που δημιουργήσαμε ,όπως επίσης και την διαδικασία δημιουργίας του τελικού αρχείου για εγκατάσταση σε Android.

4.2.1. Κώδικας της εφαρμογής

Στο παρόν κεφάλαιο παρατίθενται τμήματα από τους κώδικες της εφαρμογής. Στην εικόνα 6.9 βλέπουμε τμήμα του κώδικα του αρχείου activity_main.xml .

```
1 <?xml version="1.0" encoding="utf-8"?>
2 <androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"
3     xmlns:app="http://schemas.android.com/apk/res-auto"
4     xmlns:tools="http://schemas.android.com/tools"
5     android:layout_width="match_parent"
6     android:layout_height="match_parent"
7
8     android:background="@drawable/sss"
9     android:visibility="visible"
10    tools:context=".MainActivity">
11
12
13    <ImageButton
14        android:id="@+id/Baseleft"
15        android:layout_width="60dp"
16        android:layout_height="60dp"
17        android:layout_marginStart="37dp"
18        android:layout_marginLeft="37dp"
19        android:layout_marginEnd="61dp"
20        android:layout_marginRight="61dp"
21        android:layout_marginBottom="146dp"
22        android:src="@drawable/ic_leftarrow_foreground"
23        app:layout_constraintBottom_toTopOf="@+id/Selectionsortbutton"
24        app:layout_constraintEnd_toStartOf="@+id/Baseleft"
25        app:layout_constraintStart_toStartOf="parent" />
```

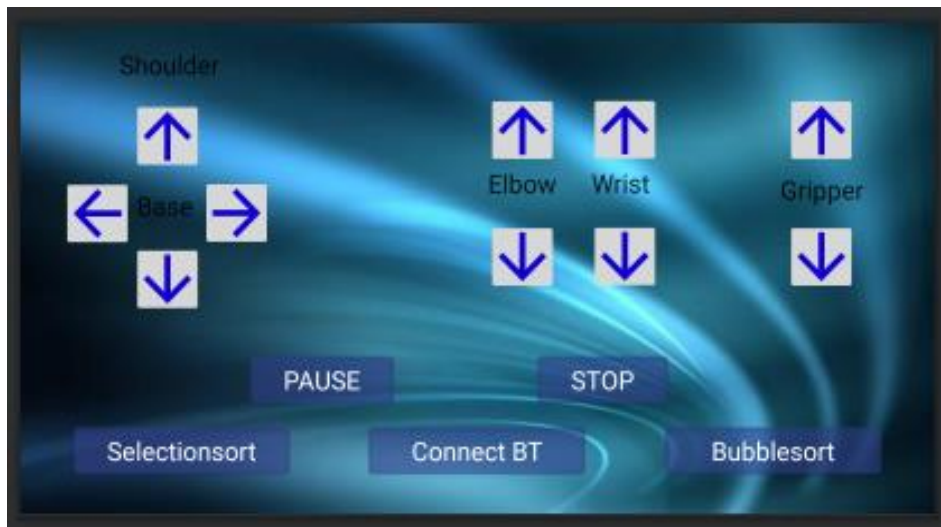
```

192 <Button
193     android:id="@+id/Selectionsortbutton"
194     android:layout_width="188dp"
195     android:layout_height="51dp"
196     android:layout_marginStart="46dp"
197     android:layout_marginLeft="46dp"
198     android:layout_marginBottom="28dp"
199     android:text="Selectionsort"
200     android:textAppearance="@style/TextAppearance.AppCompat.Large"
201     app:backgroundTint="#92323F8C"
202     app:layout_constraintBottom_toBottomOf="parent"
203     app:layout_constraintStart_toStartOf="parent" />
204
205 <TextView
206     android:id="@+id/Shouldertext"
207     android:layout_width="88dp"
208     android:layout_height="45dp"
209     android:layout_marginStart="86dp"
210     android:layout_marginLeft="86dp"
211     android:layout_marginTop="20dp"
212     android:text="Shoulder"
213     android:textAppearance="@style/TextAppearance.AppCompat.Large"
214     app:layout_constraintStart_toStartOf="parent"
215     app:layout_constraintTop_toTopOf="parent" />
216

```

Εικόνα 6.9 activity_main Code

Το τμήμα του κώδικα που παρουσιάζεται στην εικόνα 6.9 , αφορά την εξωτερική εμφάνιση της εφαρμογής . Πιο συγκεκριμένα , έχουμε εισάγει μια εικόνα η οποία χρησιμοποιείται ως φόντο για την εφαρμογή στην γραμμή 8. Στον υπόλοιπο κώδικα της εικόνας , δείχνουμε ενδεικτικά την δημιουργία τριών αντικειμένων τύπου ImageButton , Button και TextView αντίστοιχα. Έπειτα καθορίζουμε την θέση , το μέγεθος και τους περιορισμούς που θα έχει το κάθε αντικείμενο στην εφαρμογή . Δημιουργήθηκαν συνολικά 10 αντικείμενα τύπου ImageButton , 5 τύπου TextView για την δημιουργία κειμένου και 5 τύπου Button. Η Εξωτερική εμφάνιση της εφαρμογής φαίνεται στην εικόνα 7.0.



Εικόνα 7.0 Εξωτερική εμφάνιση της εφαρμογής

Τα κουμπιά στην κατηγορία Gripper , ανοίγουν και κλείνουν την αρπαγή του ρομποτικού βραχίονα . Αυτά στην κατηγορία Elbow,Shoulder και Wrist , κινούν τον αγκώνα τον ώμο και τον καρπό του βραχίονα αντίστοιχα . Στην κατηγορία base , μπορούμε να κινήσουμε την βάση του βραχίονα δεξιά η αριστερά . Στην κάτω περιοχή της εικόνας έχουμε το κουμπί που είναι υπεύθυνο για την σύνδεση μέσω Bluetooth με το arduino , το ConnectBT . Τα bubblesort και selectionsort , είναι για την ταξινόμηση των χρωματιστών κύβων , ενώ τα pause και stop ,το πρώτο παγώνει τον βραχίονα στη θέση που βρίσκεται και το δεύτερο επαναφέρει τον έλεγχο στον χρήστη αν είχε επιλέξει κάποια από τις επιλογές ταξινόμησης.

Στην εικόνα 7.1 παρουσιάζεται ο κώδικας του αρχείου strings.xml.

```

1  <resources>
2      <string name="app_name">RobotArm</string>
3      <string name="connect_bt">Connect BT</string>
4      <string name="selectionsort">Selectionsort</string>
5      <string name="bubblesort">Bubblesort</string>
6      <string name="stop">STOP</string>
7      <string name="pause">PAUSE</string>
8      <string name="shoulder">Shoulder</string>
9      <string name="base"> Base</string>
10     <string name="gripper">Gripper</string>
11     <string name="Wrist">Wrist</string>
12     <string name="elbow">Elbow</string>
13
14 </resources>

```

Εικόνα 7.1 Strings.xml Code

Ο κώδικας της εικόνας 7.1 , παρουσιάζει όλες τις συμβολοσειρές που χρησιμοποιήθηκαν στην εφαρμογή σε αντικείμενα Button ImageView και TextView.

Ακολουθεί ο κώδικας της εικόνας 7.2 που αφορά το αρχείο AndroidManifest.xml.

```
1  <?xml version="1.0" encoding="utf-8"?>
2  <manifest xmlns:android="http://schemas.android.com/apk/res/android"
3      package="com.example.robotarm">
4
5      <uses-permission android:name="android.permission.BLUETOOTH" />
6      <uses-permission android:name="android.permission.BLUETOOTH_ADMIN" />
7
8      <application
9          android:allowBackup="true"
10         android:icon="@mipmap/ic_robot"
11         android:label="RobotArm"
12         android:roundIcon="@mipmap/ic_robot_round"
13         android:supportsRtl="true"
14         android:theme="@style/Theme.RobotArm">
15         <activity android:name=".MainActivity"
16             android:screenOrientation="landscape">
17             <intent-filter>
18                 <action android:name="android.intent.action.MAIN" />
19                 <category android:name="android.intent.category.LAUNCHER" />
20             </intent-filter>
21         </activity>
22     </application>
23
24 </manifest>
```

Εικόνα 7.2 Android Manifest Code

Το αρχείο Android Manifest περιέχει κώδικα που αφορά τις άδειες που θέλουμε να έχει η εφαρμογή .Εδώ δίνουμε την δυνατότητα στο πρόγραμμα στις γραμμές 5 και 6 , με άδεια του χρήστη , να μπορεί να ενεργοποιήσει το Bluetooth της συσκευής. Επίσης καθορίσαμε το εικονίδιο που θέλουμε να έχει η εφαρμογή και θέτουμε την εφαρμογή να ανοίγει σε landscape mode.

Στην εικόνα 7.3 βλέπουμε τμήμα του κώδικα του αρχείου MainActivity.java.

```

158 Selectionsortbutton.setOnTouchListener(new View.OnTouchListener() {
159     @Override
160     public boolean onTouch(View v, MotionEvent event) {
161
162         if (event.getAction() == MotionEvent.ACTION_DOWN && BTON==1)
163         {
164             command = "s";
165
166             try
167             {
168                 outputStream.write(command.getBytes());
169             }
170             catch (IOException e)
171             {
172                 e.printStackTrace();
173             }
174         }
175         else if(event.getAction() == MotionEvent.ACTION_UP && BTON==1)
176         {
177             command = "g";
178             try
179             {
180                 outputStream.write(command.getBytes());
181             }
182             catch (IOException e)
183             {
184                 e.printStackTrace();
185             }
186         }
187
188         return false;
189     }
190 }
191
192 });

```

Εικόνα 7.3 Κώδικας για την λειτουργία των Buttons

Ο Κώδικας της εικόνας 7.3 , αφορά την λειτουργία του Button Selection Sort ,που κάνει ταξινόμηση στους κύβους . Όταν πατάμε το κουμπί Selection sort στην εφαρμογή , το κινητό στέλνει στην πλακέτα HC-05 του Arduino , τον χαρακτήρα s . Το Arduino διαβάζει τον χαρακτήρα που έστειλε το smartphone και ανάλογα τον χαρακτήρα που στέλνει το κάθε Button , έχουμε και την ανάλογη αντίδραση από το Arduino .Στην προκειμένη περίπτωση , όταν το arduino λάβει τον χαρακτήρα s , θα ξεκινήσει την ταξινόμηση των χρωματιστών κύβων με τον αλγόριθμο Selection Sort.

Στην εικόνα 7.4 δείχνουμε τον κώδικα που γράφτηκε για την σύνδεση μέσω Bluetooth.

```

593 public boolean BInitialize()
594 {
595     boolean found = false;
596     Btbutton.setBackgroundColor(Color.TRANSPARENT);
597     BluetoothAdapter bluetoothAdapter = BluetoothAdapter.getDefaultAdapter();
598
599     if(bluetoothAdapter == null) //Checks if the device supports bluetooth
600     {
601         Toast.makeText(getApplicationContext(), text: "Device doesn't support bluetooth", Toast.LENGTH_SHORT).show();
602     }
603
604     if(!bluetoothAdapter.isEnabled()) //Checks if bluetooth is enabled. If not, the program will ask permission from the user to enable it
605     {
606         Intent enableAdapter = new Intent(BluetoothAdapter.ACTION_REQUEST_ENABLE);
607         startActivityForResult(enableAdapter, requestCode: 0);
608
609         try
610         {
611             Thread.sleep( millis: 1000);
612         }
613         catch(InterruptedException e)
614         {
615             e.printStackTrace();
616         }
617     }
618
619     Set<BluetoothDevice> bondedDevices = bluetoothAdapter.getBondedDevices();
620
621     if(bondedDevices.isEmpty()) //Checks for paired bluetooth devices
622     {
623         Toast.makeText(getApplicationContext(), text: "Please pair the device first", Toast.LENGTH_SHORT).show();
624     }
625     else
626     {
627         for(BluetoothDevice iterator : bondedDevices)
628         {
629             if(iterator.getAddress().equals(DEVICE_ADDRESS))
630             {
631                 device = iterator;
632                 found = true;
633                 break;
634             }
635         }

```

```

636     }
637
638     return found;
639 }
640
641 public boolean BConnect()
642 {
643     boolean connected = true;
644
645     try
646     {
647         socket = device.createRfcommSocketToServiceRecord(PORT_UUID); //Creates a socket to handle the outgoing connection
648         socket.connect();
649         Btbutton.setBackgroundColor(Color.GREEN);
650         BTON=1;
651         Toast.makeText(getApplicationContext(),
652             text: "Connection to bluetooth device successful", Toast.LENGTH_LONG).show();
653     }
654     catch(IOException e)
655     {
656         e.printStackTrace();
657         connected = false;
658         BTON=0;
659     }
660
661     if(connected)
662     {
663         try
664         {
665             outputStream = socket.getOutputStream(); //gets the output stream of the socket
666         }
667         catch(IOException e)
668         {
669             e.printStackTrace();
670         }
671     }
672
673     return connected;
674 }

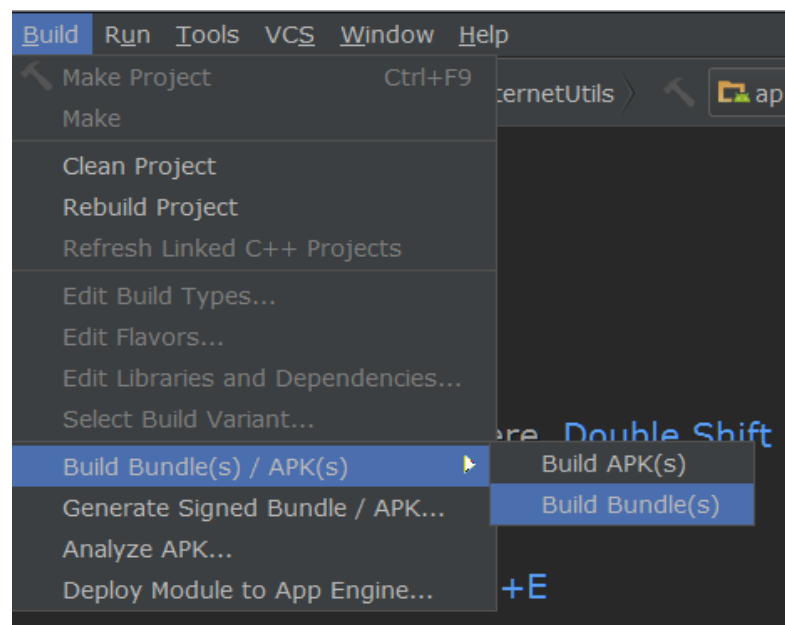
```

Εικόνα 7.4 Κώδικας για σύνδεση μέσω Bluetooth

Ο κώδικας της εικόνας 7.4 , αρχικά ελέγχει αν η συσκευή υποστηρίζει επικοινωνία μέσω Bluetooth . Αν δεν υποστηρίζει τότε εμφανίζει το κατάλληλο μήνυμα. Σε διαφορετική περίπτωση δημιουργεί ένα κανάλι επικοινωνίας ανάμεσα στις δύο συσκευές ,ώστε να μπορούν να ανταλλάσσουν δεδομένα και εμφανίζει μήνυμα αν η διαδικασία ήταν επιτυχής.

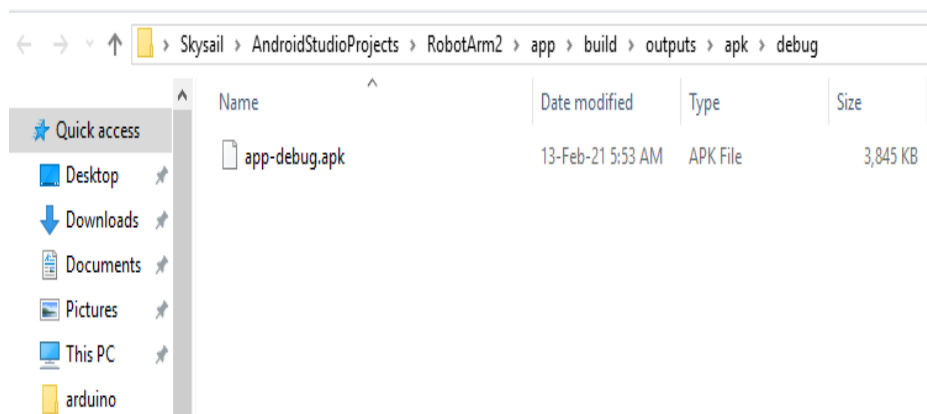
4.2.2. Δημιουργία αρχείου apk για εγκατάσταση σε Android

Αφού έχουμε φτιάξει όλα τα τμήματα του προγράμματος , θα χρησιμοποιήσουμε τον compiler που προσφέρει το Android studio για να κατασκευάσουμε το τελικό αρχείο , το οποίο είναι ένα αρχείο με επέκταση .apk που μπορεί να εκτελεστεί στο λειτουργικό Android . Στην εικόνα παρακάτω 7.5 κατασκευάζουμε το αρχείο με επέκταση .apk.



Εικόνα 7.5 Κατασκευή αρχείου .apk

Η εφαρμογή δημιουργείται όταν επιλέξουμε την επιλογή Build APK(s) της εικόνας 7.5. Έπειτα η εφαρμογή εμφανίζεται στον φάκελο , όπου μπορούμε να την στείλουμε στο smartphone για εγκατάσταση , όπως φαίνεται στην εικόνα 7.6.

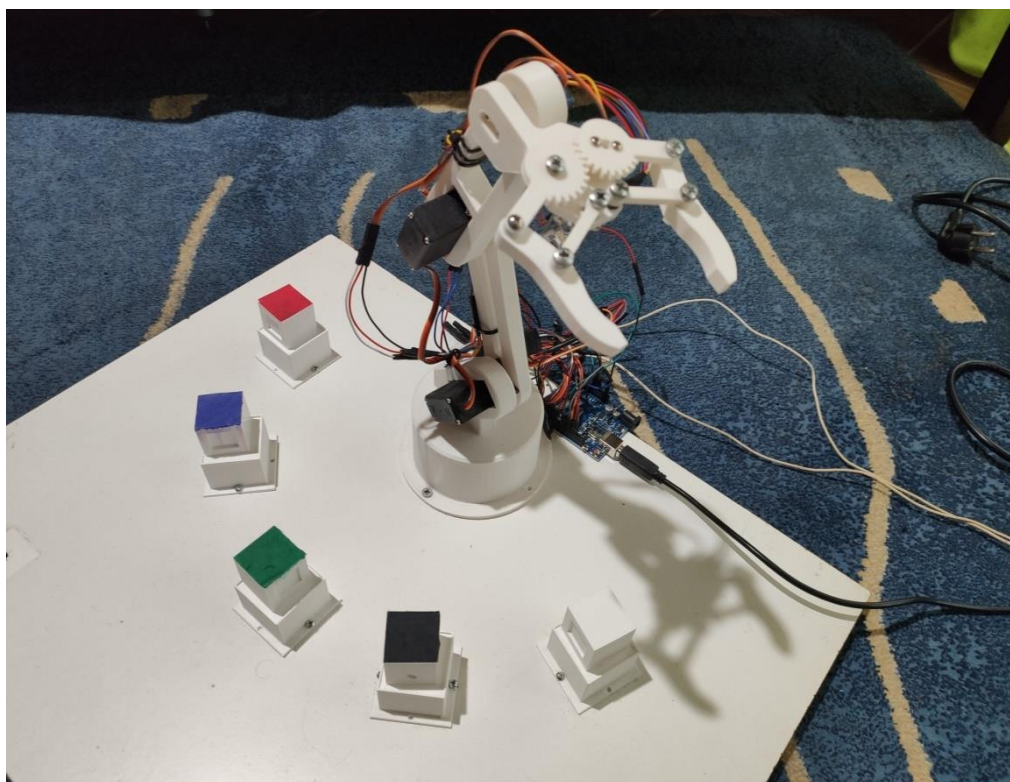


Εικόνα 7.6 Το αρχείο με επέκταση .apk

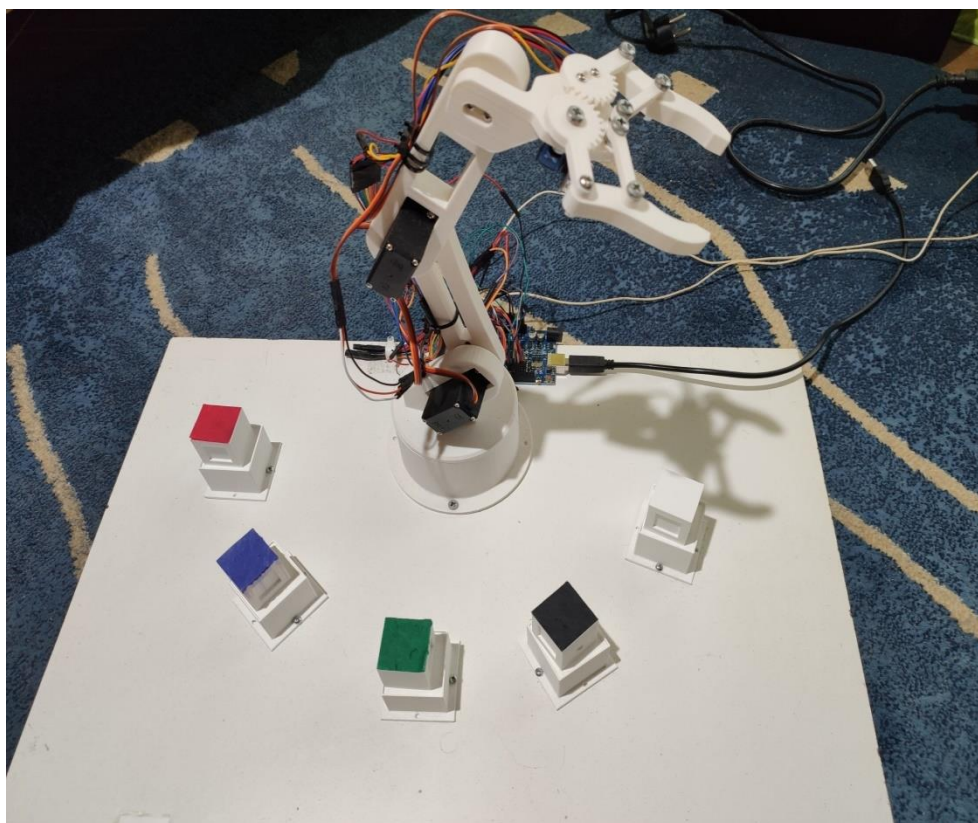
5. Ολοκλήρωση της κατασκευής

5.1. Συναρμολόγηση των επιμέρους στοιχείων

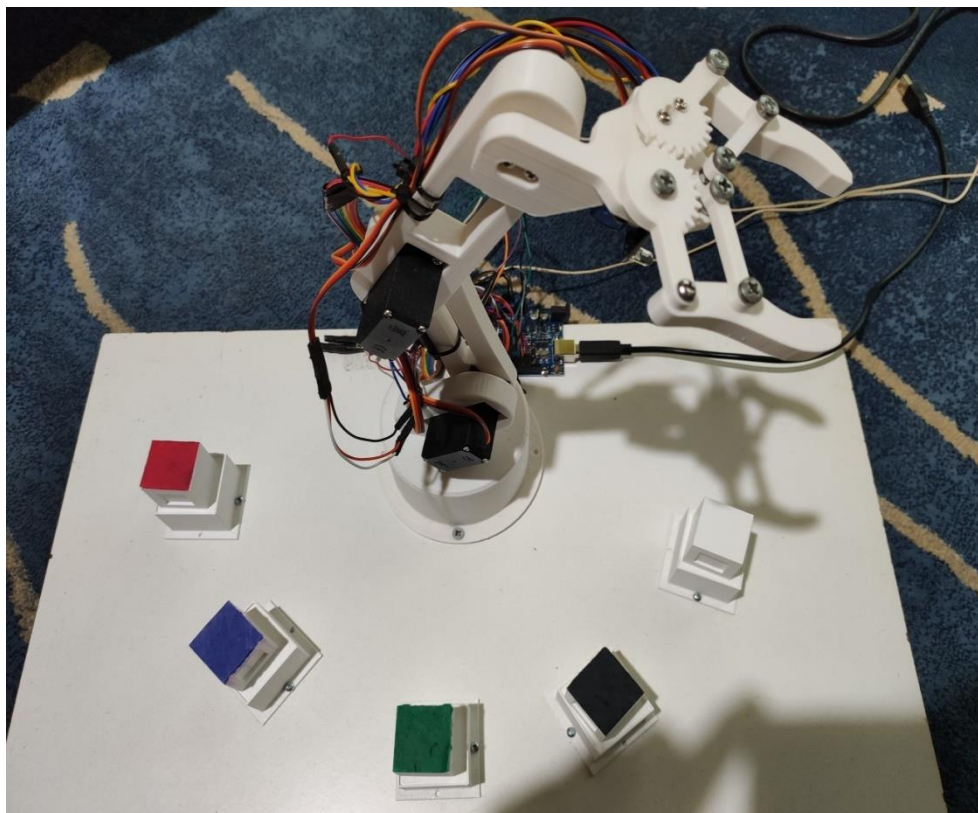
Στο κεφάλαιο αυτό συνδέσαμε όλα τα τμήματα του ρομποτικού βραχίονα που δημιουργήθηκαν στα παραπάνω κεφάλαια σε ένα ενιαίο σχέδιο. Η βάση που χρησιμοποιήθηκε για την στήριξη των σταθερών τμημάτων της εργασίας είναι ξύλινη. Το ολοκληρωμένο σχέδιο της παρούσας εργασίας παρουσιάζεται στις εικόνες 7.7 , 7.8 και 7.9.



Εικόνα 7.7



Εικόνα 7.8



Εικόνα 7.9

Στις παραπάνω εικόνες μπορούμε να διακρίνουμε τον ρομποτικό βραχίονα με όλον τον ηλεκτρονικό εξοπλισμό που εγκαταστάθηκε. Επίσης μπορούμε να διακρίνουμε το Arduino όπως επίσης και τους χρωματιστούς κύβους.

5.2. Σύνοψη-Συμπεράσματα-Βελτιώσεις

Στην παρούσα πτυχιακή εργασία αφού κάναμε μια εισαγωγή στο IoT και στο Arduino , κατασκευάσαμε έναν ρομποτικό βραχίονα με δυνατότητα ασύρματου ελέγχου μέσω bluetooth και εκτέλεσης αλγορίθμων ταξινόμησης , αναλύοντας όλα τα βήματα κάθε φορά , όπως τα σχέδια , το πρόγραμμα στον μικροελεγκτή , το κύκλωμα και το πρόγραμμα για τον έλεγχο στο Android studio.

Όλα τα τμήματα της εργασίας ολοκληρώθηκαν με επιτυχία και μέσω της εφαρμογής μπορούμε να ελέγξουμε όλες τις λειτουργίες του βραχίονα , όπως την οποιαδήποτε κίνηση των συνδέσμων και την έναρξη εκτέλεσης των αλγορίθμων ταξινόμησης.

Η παρούσα εργασία θα μπορούσε να βελτιωθεί σε πολλούς τομείς , διότι η επιστήμη της ρομποτικής από την φύση της εξελίσσεται συνεχώς . Για παράδειγμα θα μπορούσαμε εκτός από τον έλεγχο των κινήσεων από το smartphone , να εγκαθιστούσαμε έναν αισθητήρα αναγνώρισης φωνητικών εντολών , ώστε να μπορούμε να ελέγξουμε τον βραχίονα και μέσω φωνής.

6. Βιβλιογραφία

- [1] https://en.wikipedia.org/wiki/Industrial_robot
- [2] https://en.wikipedia.org/wiki/Robotic_arm
- [3] <http://edurobotics.weebly.com/tauiota-epsilon943nualphaiota.html>
- [4] http://www.grobot.gr/index.php?option=com_content&view=article&id=76
- [5] <https://ecopress.gr/sygchroni-viomichaniki-robotiki-kai-pleonektimata-viomichanikon-robot/>
- [6] http://www.eln.teilam.gr/sites/default/files/lesson08_1.pdf
- [7] https://en.wikipedia.org/wiki/Internet_of_things
- [8] https://ethw.org/Kevin_Ashton
- [9] <https://studycare.gr/ti-einai-to-diadiktyo-ton-pragmaton-iot/>
- [10] <https://www.researchgate.net/scientific-contributions/Keyur-K-Patel-2152250939>
- [11] <https://www.solidworks.com/>
- [12] https://en.wikipedia.org/wiki/3D_printing
- [13] <https://www.3dexpert.gr/main/3dprinters-intro/>
- [14] <https://ultimaker.com/software/ultimaker-cura>
- [15] [https://en.wikipedia.org/wiki/Cura_\(software\)](https://en.wikipedia.org/wiki/Cura_(software))
- [16] <https://www.creality.com/goods-detail/ender-5-3d-printer>
- [17] <https://en.wikipedia.org/wiki/Arduino>
- [18] <https://coral.ai/docs/dev-board/datasheet/#features>
- [19] <https://en.wikipedia.org/wiki/Arduino>
- [20] <https://www.watelectronics.com/arduino-uno-board-tutorial-and-its-applications/>
- [21] <https://en.wikipedia.org/wiki/Bluetooth>

- [22] <https://www.gme.cz/data/attachments/dsh.772-148.1.pdf>
- [23] http://www.ee.ic.ac.uk/pcheung/teaching/DE1_EE/stores/sg90_datasheet.pdf
- [24] https://www.electronicoscaldas.com/datasheet/MG996R_Tower-Pro.pdf
- [25] http://wiki.sunfounder.cc/index.php?title=TCS3200_Color_Sensor_Module
- [26] https://en.wikipedia.org/wiki/Sorting_algorithm
- [27] https://en.wikipedia.org/wiki/Bubble_sort
- [28] https://en.wikipedia.org/wiki/Selection_sort
- [29] https://en.wikipedia.org/wiki/Android_Studio