



**ΤΕΧΝΟΛΟΓΙΚΟ ΕΚΠΑΙΔΕΥΤΙΚΟ  
ΙΔΡΥΜΑ ΚΡΗΤΗΣ  
ΤΜΗΜΑ ΗΛΕΚΤΡΟΝΙΚΗΣ**

**ΣΧΕΔΙΑΣΜΟΣ, ΑΝΑΠΤΥΞΗ ΚΑΙ ΥΛΟΠΟΙΗΣΗ  
ΡΟΜΠΟΤΙΚΟΥ ΟΧΗΜΑΤΟΣ ΚΑΘΕΤΟΥ  
ΑΠΟΠΡΟΣΓΕΙΩΣΕΩΣ ΜΕ ΤΕΣΣΕΡΙΣ  
ΡΟΤΟΡΕΣ**

Τριμελής Επιτροπή:

Ε. Δοϊτσίδης

Επίκουρος Καθηγητής (επιβλέπων)

Γ. Φουσκιτάκης

Επίκουρος Καθηγητής

Ε. Κόκκινος

Επίκουρος Καθηγητής

υπό

Ηλία Νιώρα

Χανιά, 2012

## **ΕΥΧΑΡΙΣΤΙΕΣ**

Η σελίδα αυτή μπορεί να παραλειφθεί.

## ΠΕΡΙΛΗΨΗ ΠΤΥΧΙΑΚΗΣ

Στη παρούσα εργασία παρουσιάζεται η ανάπτυξη μια πρωτότυπης εναέριας ρομποτικής πλατφόρμας, ικανής να χρησιμοποιηθεί τόσο για την κατανόηση βασικών θεμάτων που σχετίζονται με την ρομποτική, τον έλεγχο, τις τηλεπικοινωνίες και την μηχανική, όσο και για να αποτελέσει μια πλατφόρμα πειραματισμού για την διεξαγωγή βασικών ερευνητικών πειραμάτων.

Στα πλαίσια της εργασίας πραγματοποιήθηκε ο σχεδιασμός της τόσο σε επίπεδο υλικού (hardware), όσο και σε επίπεδο λογισμικού (software). Στη συνέχεια πραγματοποιήθηκε η ολοκλήρωση της προτεινόμενης κατασκευής και δοκιμάστηκε η λειτουργικότητα της. Ο σχεδιασμός και η υλοποίηση πραγματοποιήθηκαν με στόχο τη βέλτιστη σχέση κόστους – απόδοσης. Η υλοποίηση της συγκεκριμένης πλατφόρμας πραγματοποιήθηκε στο τμήμα Ηλεκτρονικής του Τ.Ε.Ι Κρήτης.

## ΠΕΡΙΕΧΟΜΕΝΑ

ΣΧΕΔΙΑΣΜΟΣ, ΑΝΑΠΤΥΞΗ ΚΑΙ ΥΛΟΠΟΙΗΣΗ ΡΟΜΠΟΤΙΚΟΥ ΟΧΗΜΑΤΟΣ ΚΑΘΕΤΟΥ ΑΠΟΠΡΟΣΓΕΙΩΣΕΩΣ ΜΕ ΤΕΣΣΕΡΙΣ ΡΟΤΟΡΕΣ.....	i
ΚΕΦΑΛΑΙΟ 1 .....	1
ΕΙΣΑΓΩΓΗ .....	1
1.1 ΑΝΤΙΚΕΙΜΕΝΟ ΤΗΣ ΕΡΓΑΣΙΑΣ .....	1
1.2 ΔΟΜΗ ΤΗΣ ΕΡΓΑΣΙΑΣ .....	1
ΚΕΦΑΛΑΙΟ 2 .....	2
ΒΙΒΛΙΟΓΡΑΦΙΚΗ ΕΠΙΣΚΟΠΗΣΗ .....	2
2.1 ΙΣΤΟΡΙΚΗ ΑΝΑΔΡΟΜΗ.....	2
2.2 ΕΙΔΗ ΕΛΙΚΟΠΤΕΡΩΝ .....	3
2.3 ΧΡΗΣΕΙΣ ΜΗ ΕΠΑΝΔΡΟΜΕΝΩΝ ΕΛΙΚΟΠΤΕΡΩΝ .....	6
2.4 ΕΜΠΟΡΙΚΑ ΔΙΑΘΕΣΙΜΑ ΕΛΙΚΟΠΤΕΡΑ ΜΕ ΤΕΣΣΕΡΙΣ ΡΟΤΟΡΕΣ.....	6
ΚΕΦΑΛΑΙΟ 3 .....	9
ΣΧΕΔΙΑΣΗ ΚΑΙ ΚΑΤΑΣΚΕΥΗ .....	9
3.1 ΚΑΤΑΣΚΕΥΗ ΠΛΑΙΣΙΟΥ .....	9
3.2 ΠΛΑΚΕΤΑ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΥ.....	11
3.3 ΚΕΝΤΡΙΚΗ ΠΛΑΚΕΤΑ ΕΛΕΓΧΟΥ .....	16
3.4 ΚΙΝΗΤΗΡΕΣ ΚΑΙ ΕΛΕΓΚΤΕΣ ΤΑΧΥΤΗΤΑΣ .....	26
3.5 ΜΠΑΤΑΡΙΑ.....	32
3.6 ΑΣΥΡΜΑΤΗ ΕΠΙΚΟΙΝΩΝΙΑ (RF) .....	33
3.7 ΑΙΣΘΗΤΗΡΙΑ.....	36
ΚΕΦΑΛΑΙΟ 4 .....	38
ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΣ .....	38
4.1 ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΣ ΤΟΥ ΜΙΚΡΟΕΠΕΞΕΡΓΑΣΤΗ .....	38
4.2 ΠΡΟΓΡΑΜΜΑΤΙΖΟΝΤΑΣ ΤΟΝ ΜΙΚΡΟΕΠΕΞΕΡΓΑΣΤΗ.....	38
4.3 ΑΝΑΛΥΣΗ ΛΟΓΙΣΜΙΚΟΥ .....	42
ΚΕΦΑΛΑΙΟ 5 .....	44
ΤΟ ΠΡΟΤΕΙΝΟΜΕΝΟ ΕΝΑΕΡΙΟ ΟΧΗΜΑ .....	44
5.1 Η ΤΕΛΙΚΗ ΜΟΡΦΗ ΤΟΥ ΠΡΟΤΕΙΝΟΜΕΝΟΥ ΟΧΗΜΑΤΟΣ.....	44

5.2 ΓΡΑΦΙΚΟ ΠΕΡΙΒΑΛΛΟΝ ΔΕΙΠΑΦΗΣ .....	47
5.3 ΕΝΔΕΙΚΤΙΚΟ ΣΧΗΜΑ ΕΛΕΓΧΟΥ .....	47
ΚΕΦΑΛΑΙΟ 6 .....	49
ΣΥΜΠΕΡΑΣΜΑΤΑ – ΜΕΛΛΟΝΤΙΚΕΣ ΕΠΕΚΤΑΣΕΙΣ.....	49
6.1 ΣΥΜΠΕΡΑΣΜΑΤΑ .....	49
6.2 ΜΕΛΛΟΝΤΙΚΕΣ ΕΠΕΚΤΑΣΕΙΣ .....	49
ΒΙΒΛΙΟΓΡΑΦΙΑ .....	51
ΠΑΡΑΡΤΗΜΑ .....	52
Πρόγραμμα κεντρικής πλακέτας ελέγχου :.....	52
Πρόγραμμα πλακέτας προγραμματισμού για ασύρματη επικοινωνία :.....	57
Πρόγραμμα συλλογής δεδομένων από επιταχυνσιόμετρο : .....	58
Πρόγραμμα συλλογής δεδομένων από γυροσκόπιο : .....	60

# ΚΕΦΑΛΑΙΟ 1

## ΕΙΣΑΓΩΓΗ

### 1.1 ΑΝΤΙΚΕΙΜΕΝΟ ΤΗΣ ΕΡΓΑΣΙΑΣ

Σκοπός της παρούσας εργασίας, ήταν η ανάπτυξη μια πρωτότυπης εναέριας ρομποτικής πλατφόρμας, ικανής να χρησιμοποιηθεί τόσο για την κατανόηση βασικών θεμάτων που σχετίζονται με την ρομποτική, τον έλεγχο, τις τηλεπικοινωνίες και την μηχανική, όσο και για να αποτελέσει μια πλατφόρμα πειραματισμού για την διεξαγωγή βασικών ερευνητικών πειραμάτων.

Στα πλαίσια της εργασίας πραγματοποιήθηκε ο σχεδιασμός της τόσο σε επίπεδο υλικού (hardware), όσο και σε επίπεδο λογισμικού (software). Στη συνέχεια πραγματοποιήθηκε η ολοκλήρωση της προτεινόμενης κατασκευής και δοκιμάστηκε η λειτουργικότητα της. Ο σχεδιασμός και η υλοποίηση πραγματοποιήθηκαν με στόχο τη βέλτιστη σχέση κόστους – απόδοσης. Η υλοποίηση της συγκεκριμένης πλατφόρμας πραγματοποιήθηκε στο τμήμα Ηλεκτρονικής του Τ.Ε.Ι Κρήτης.

### 1.2 ΔΟΜΗ ΤΗΣ ΕΡΓΑΣΙΑΣ

Στο 1<sup>ο</sup> κεφάλαιο παρουσιάζεται το αντικείμενο της παρούσας εργασίας και παρουσιάζεται η δομή της. Στο 2<sup>ο</sup> κεφάλαιο γίνεται μια ιστορική αναδρομή της εξέλιξης των διαφόρων τύπων ελικοπτέρων, παρουσιάζονται συνοπτικά τα ιδιαίτερα χαρακτηριστικά της κάθε μιας κατηγορίας και γίνεται μια σύντομη αναφορά στα κυριότερα εμπορικά μικρού μεγέθους εναέρια οχήματα με τέσσερις ρότορες.

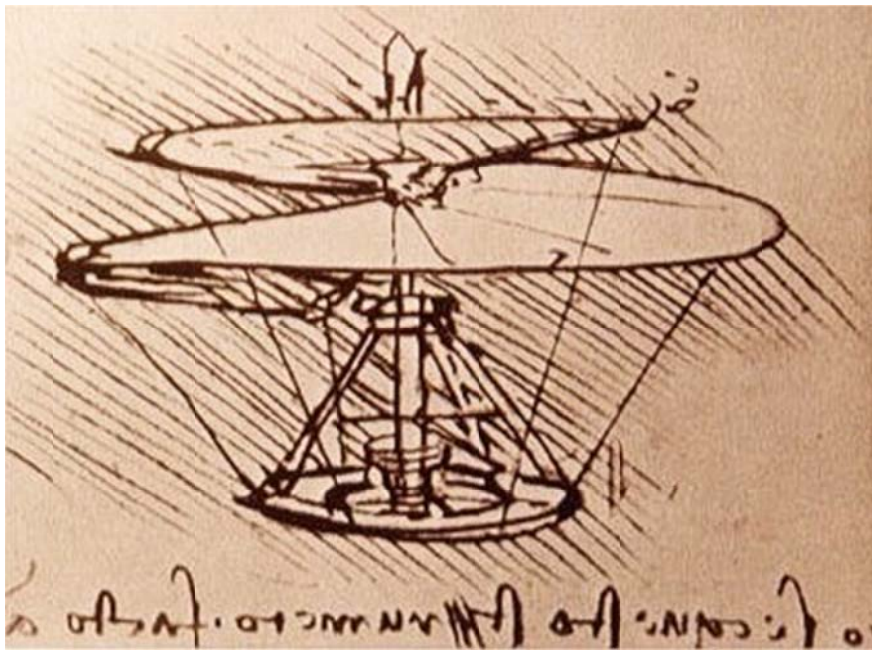
Στο 3<sup>ο</sup> κεφάλαιο παρουσιάζονται τα δομικά μέρη που αποτελούν το πρωτότυπο εναέριο όχημα, και γίνεται μια αναλυτική περιγραφή του τρόπου λειτουργίας τους. Στο 4<sup>ο</sup> κεφάλαιο παρουσιάζεται ο τρόπος ανάπτυξης και το λογισμικό που τελικά αναπτύχθηκε στα πλαίσια της συγκεκριμένης εργασίας. Στο 5<sup>ο</sup> κεφάλαιο παρουσιάζεται το ολοκληρωμένο πρωτότυπο και το γραφικό περιβάλλον διεπαφής που αναπτύχθηκε για τον έλεγχο του. Τέλος στο 6<sup>ο</sup> κεφάλαιο παρουσιάζονται συνοπτικά τα επιτεύγματα της εργασίας καθώς δίνονται και κάποιες ιδέες για μελλοντικές προεκτάσεις.

## ΚΕΦΑΛΑΙΟ 2

### ΒΙΒΛΙΟΓΡΑΦΙΚΗ ΕΠΙΣΚΟΠΗΣΗ

#### 2.1 ΙΣΤΟΡΙΚΗ ΑΝΑΔΡΟΜΗ

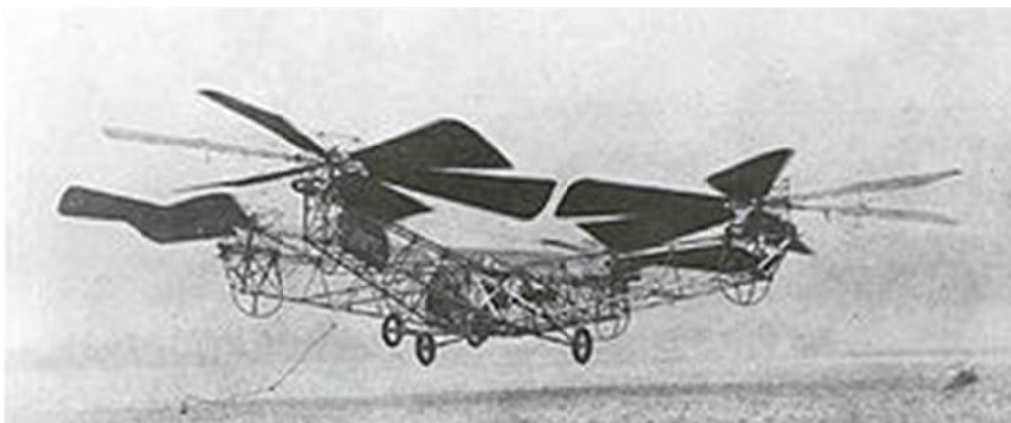
Η πρώτη αναφορά σε εναέρια οχήματα συναντάται στη Κίνα το 400 π. Χ όπου κατασκευάζονταν μικρά αεροπλάνα από μπαμπού. Το 1480 μ.Χ ο Leonardo da Vinci παρουσιάζει τα σχέδια ενός πρωτότυπου ιπτάμενου οχήματος, που αναφέρεται ως ιπτάμενο κατσαβίδι. Αυτό παρουσίαζε αρκετά κοινά χαρακτηριστικά με τα σύγχρονα ελικόπτερα.



Σχήμα 2.1: Leonardo da Vinci το 1480 : ιπτάμενο κατσαβίδι

Η πρώτη σύγχρονη έρευνα που πραγματοποιήθηκε σχετικά με ελικόπτερα με πολλαπλούς έλικες έγιναν το 1920 από τον Etienne Oehmichen. Μια από τις κατασκευές του είχε τα βασικά χαρακτηριστικά ενός σύγχρονου ελικοπτερου με περισσότερους από δύο έλικες. Η συγκεκριμένη συσκευή μετά από βελτιώσεις κατόρθωσε να παραμείνει αρκετά λεπτά σε πτήση και να διανύσει απόσταση μεγαλύτερη του ενός χιλιομέτρου.

Μια ακόμα προσπάθεια έγινε από τους Dr. George de Bothezat και Ivan Jerome που κατασκεύασαν αεροσκάφος σε σχήμα Χ. Αυτό έφτανε σε ανώτατο ύψος 5m, και πραγματοποίησε από το 1922 έως το 1923 περισσότερες από 100 πτήσεις.



Σχήμα 2.2 : De Bothezat 1923 :Το πρώτο ελικόπτερο τύπου Χ

Τα οχήματα αυτά παρουσίαζαν πληθώρα προβλημάτων. Η τεχνολογία της εποχής δεν επέτρεπε την γρήγορη εξέλιξη των ελικοπτέρων με τέσσερις κινητήρες, γι'αυτό και το επόμενο βήμα έγινε αρκετά αργότερα -μετά το 1950- όπου και κατασκευάστηκαν τέτοιου τύπου ελικόπτερα για στρατιωτικούς και πολιτικούς σκοπούς.

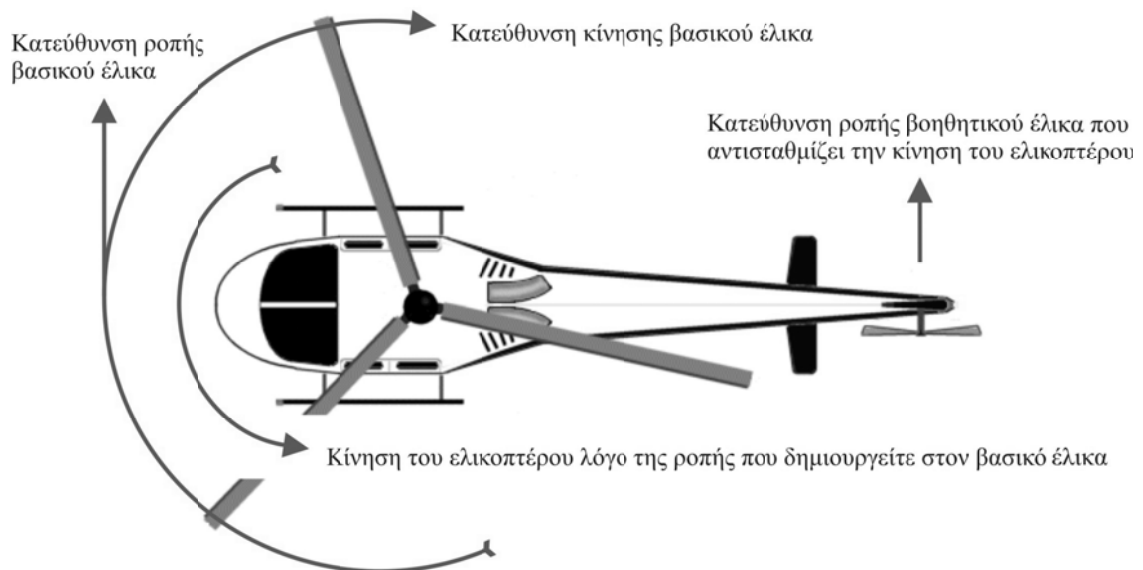
Η εξέλιξη αυτών των μελετών είχε σαν αποτέλεσμα την κατασκευή μη επανδρωμένων ελικοπτέρων (UAV) που χαρακτηρίζονται από τη σταθερότητα πτήσης και τις υψηλές ταχύτητες, άλλο ένα πλεονέκτημα είναι το πολύ μικρό μέγεθος και την χαμηλή κατανάλωση ενέργειας των UAV.

## 2.2 ΕΙΔΗ ΕΛΙΚΟΠΤΕΡΩΝ

Τα είδη ελικοπτέρων χωρίζονται σε δύο βασικές κατηγορίες: α) τα ελικόπτερα που χρησιμοποιούν έλικες οριζόντιους και κάθετους ως προς το επίπεδο ανύψωσης και β) τα ελικόπτερα που χρησιμοποιούν μόνο οριζόντιους έλικες ως προς το επίπεδο ανύψωσης.

Τα ελικόπτερα που χρησιμοποιούν οριζόντιο αλλά και κάθετο έλικα ως προς το επίπεδο ανύψωσης αποτελούνται από έναν κεντρικό έλικα, τοποθετημένο οριζόντια στο επίπεδο ανύψωσης που δημιουργεί την απαραίτητη ώθηση ώστε να ανυψωθεί το ελικόπτερο. Η περιστροφική κίνηση του κεντρικού έλικα δημιουργεί ροπή αντίθετης κατεύθυνσης στον κορμό του ελικοπτέρου. Προκειμένου να εξαλείφει η δύναμη αυτή τοποθετείται ένας βοηθητικός έλικας στην ούρα του ελικοπτέρου που περιστρέφεται κάθετα ως προς το επίπεδο ανύψωσης (Σχήμα 2.3).





Σχήμα 2.3 : Οι δυνάμεις που ασκούν οι δύο έλικες σε ελικοπτερο με οριζόντιο και κάθετο έλικα.

Το δεύτερο είδος ελικοπτέρων χρησιμοποιεί μόνο έλικες οριζόντιους ως προς το επίπεδο ανύψωσης. Στην περίπτωση αυτή και οι δύο έλικες παράγουν την ίδια προωθητική δύναμη στρεφόμενοι όμως προς την αντίθετη κατεύθυνση με αποτέλεσμα να αποτρέπεται η δημιουργία ανεπιθύμητης ροπής, καθώς η ανεπιθύμητη ροπή του ενός έλικα εξουδετερώνεται από την ανεπιθύμητη ροπή του άλλου έλικα. Στο δεύτερο είδος ελικοπτέρων οι έλικες μπορεί να στρέφονται στον ίδιο άξονα όπως φαίνεται στο Σχήμα 2.4 η σε διαφορετικούς άξονες όπως παρουσιάζετε στο Σχήμα 2.5.



Σχήμα 2.4 : Ομοαξονικό ελικόπτερο (παρουσιάζεται ο τρόπος αλληλεξουδετέρωσης των δύο αντίρροπων δυνάμεων)



Σχήμα 2.5 : Ελικόπτερο με δύο έλικες τοποθετημένους σε διαφορετικούς άξονες (παρουσιάζεται ο τρόπος αλληλεξουδετέρωσης των δύο αντίρροπων δυνάμεων)

Ανάλογη λειτουργία με τα ελικόπτερα με δύο έλικες έχουν και αυτά με περισσότερους έλικες όπως αυτό που θα μελετηθεί στην παρούσα εργασία.

### **2.3 ΧΡΗΣΕΙΣ ΜΗ ΕΠΑΝΔΡΟΜΕΝΩΝ ΕΛΙΚΟΠΤΕΡΩΝ**

Οι χρήσεις των μη επανδρωμένων ελικοπτέρων μπορούν να χωριστούν σε 2 μεγάλες κατηγορίες, ανάλογα με τις εφαρμογές που χρησιμοποιούνται σήμερα. Αναλυτικά αυτές παρουσιάζονται στις [1-4] και συγκεκριμένα χωρίζονται σε πολιτικές και στρατιωτικές εφαρμογές.

*Πολιτικές Εφαρμογές:*

- Παρακολούθηση αγωγών πετρελαίου
- Επιτήρηση γραμμών μεταφοράς ηλεκτρικού ρεύματος
- Έλεγχος κυκλοφορίας αυτοκινητοδρόμων
- Μέσα Μαζικής ενημέρωσης
- Αποστολές έρευνας και διάσωσης
- Παρακολούθηση φυσικών καταστροφών και περιβαλλοντικών μετρήσεων
- Έρευνα για φυσικούς πόρους
- Πυροπροστασία – επιτήρηση δασών
- Αλιεία
- Χρήσεις τοπογραφικών εφαρμογών
- Γεωργία

*Στρατιωτικές εφαρμογές:*

- Ανίχνευση
- Στόχευση από απόσταση
- Απεικόνιση πεδίου μάχης – ανίχνευση ελευθέρων σκοπευτών
- Έρευνα για νάρκες
- Ανίχνευση χημικών, πυρηνικών και βιολογικών όπλων

### **2.4 ΕΜΠΟΡΙΚΑ ΔΙΑΘΕΣΙΜΑ ΕΛΙΚΟΠΤΕΡΑ ΜΕ ΤΕΣΣΕΡΙΣ ΡΟΤΟΡΕΣ**

Υπάρχουν αρκετά εμπορικά διαθέσιμα ελικόπτερα με τέσσερις ή και παραπάνω ρότορες, παρουσιάζονται στη συνέχεια [5].

Η εταιρία AscTec κατασκευάζει το Hornet που είναι ένα όχημα με τέσσερις ρότορες και με συνολική διάμετρο 28 εκατοστά είναι το μικρότερο διαθέσιμο εμπορικά όχημα. Έχει

δυνατότητα να μεταφέρει επιπλέον φορτίο της τάξης των 50 γραμμαρίων και παρουσιάζεται στο σχήμα 2.6



*Σχήμα 2.6 : Το όχημα AscTec Hornet*

Η εταιρία AscTec κατασκευάζει και το εναέριο όχημα Eagle με 8 ρότορες και μπορεί να μεταφέρει φορτίο μέχρι και 1.5 κιλά. Σημαντικό στοιχείο της συγκεκριμένης κατασκευής είναι η δυνατότητα να συνεχίσει την λειτουργία της ακόμα και αν χάσει κάποιο ρότορα εξαιτίας μηχανικής βλάβης. Το συγκεκριμένο όχημα παρουσιάζεται στο σχήμα 2.7.



*Σχήμα 2.7 : Το όχημα AscTec Eagle*

Η εταιρία Draganflyer κατασκευάζει το ομώνυμο εναέριο όχημα, που φέρει οπτικό αισθητήρα. Το συγκεκριμένο όχημα παρουσιάζεται στο σχήμα 2.8.



*Σχήμα 2.8 : Το όχημα Draganflyer*

Η εταιρία MicroDrones κατασκευάζει το μοντέλο MD4-200, που φέρει οπτικό αισθητήρα. Το συγκεκριμένο όχημα παρουσιάζεται στο σχήμα 2.9.



*Σχήμα 2.9 : Το όχημα MD4-200*

## ΚΕΦΑΛΑΙΟ 3

### ΣΧΕΔΙΑΣΗ ΚΑΙ ΚΑΤΑΣΚΕΥΗ

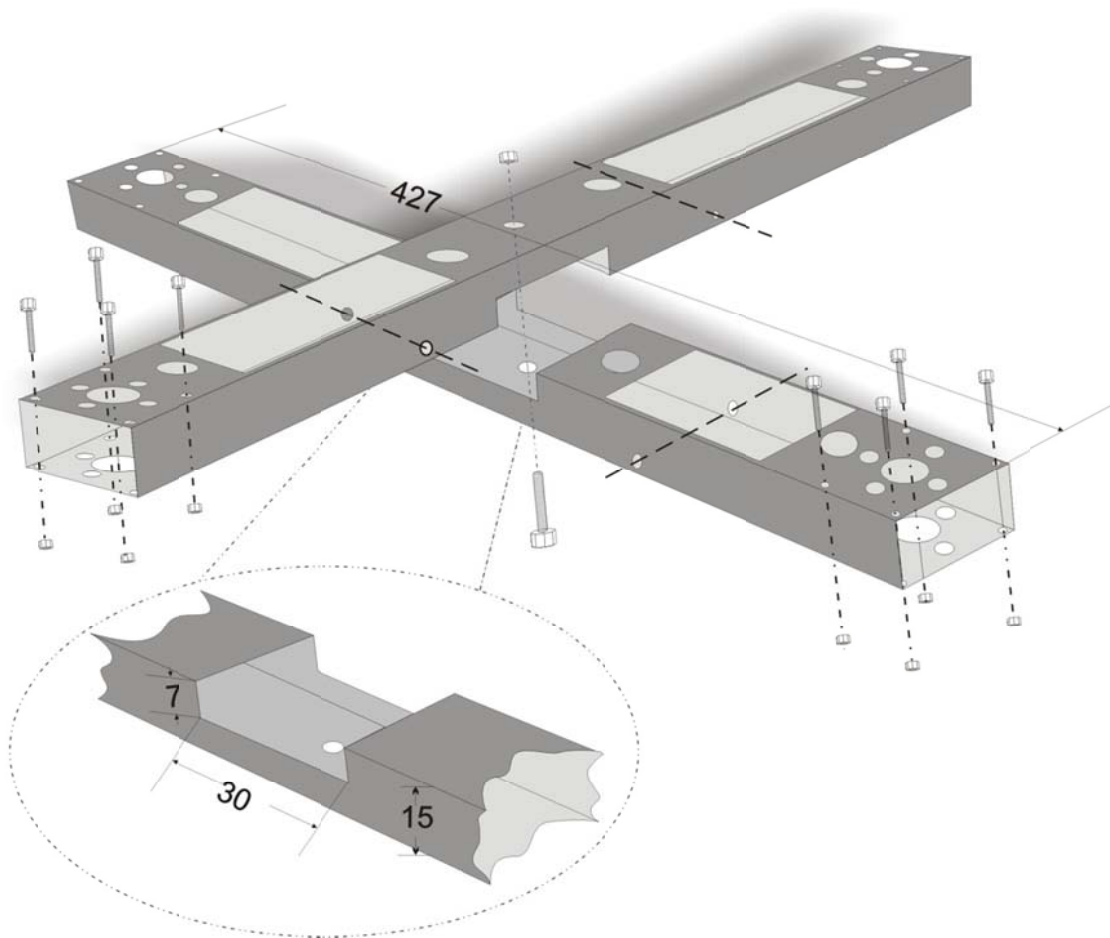
#### 3.1 ΚΑΤΑΣΚΕΥΗ ΠΛΑΙΣΙΟΥ

Στα πλαίσια της παρούσας εργασίας, σχεδιάστηκε, αναπτύχθηκε και κατασκευάστηκε πρωτότυπο εναέριο ρομποτικό όχημα. Αρχικά έγινε σχεδιασμός του πλαισίου, που χρησιμοποιήθηκε σαν βάση προσαρμογής των διαφόρων δομικών στοιχείων που κατασκευάστηκαν ή χρησιμοποιήθηκαν.

Για τον σχεδιασμό του πλαισίου πραγματοποιήθηκε έρευνα για να αποφασιστεί το υλικό κατασκευής με κύρια χαρακτηριστικά την μεγάλη αντοχή και το χαμηλό βάρος. Το υλικό που επιλέχθηκε ήταν το αλουμίνιο καθώς είναι αρκετά ανθεκτικό, έχει χαμηλό βάρος και είναι εύκολο να το προμηθευτούμε σε πολλές διαφορετικές διαστάσεις. Σημαντικός παράγοντας είναι και το κόστος το οποίο δεν είναι ιδιαίτερα υψηλό.

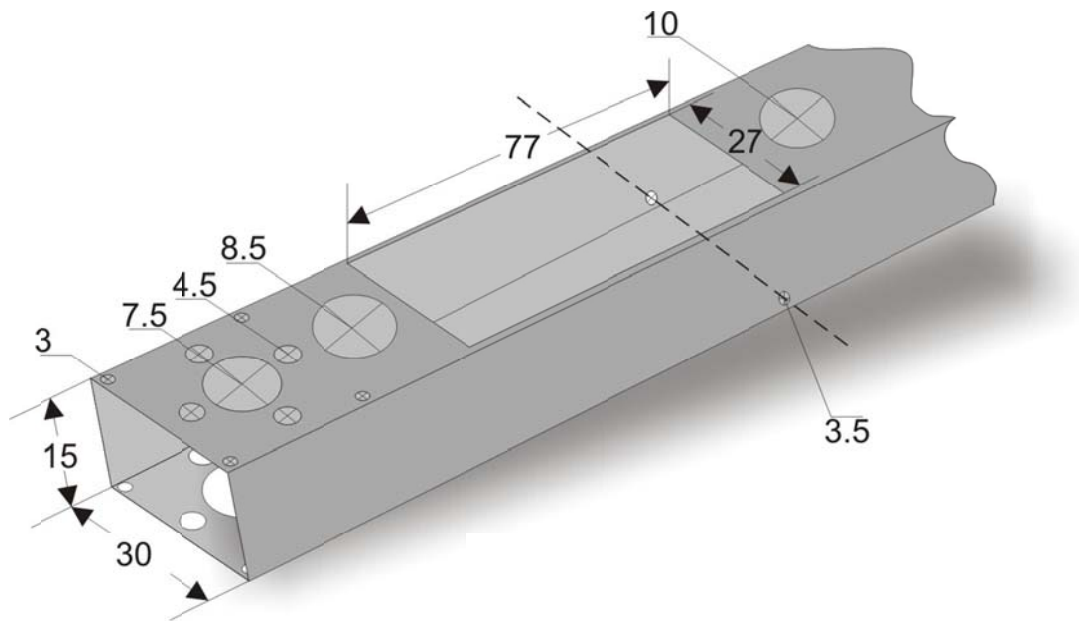
Για την κατεργασία του αλουμινίου χρησιμοποιήθηκαν τρυπάνια κατάλληλα για την διάτρηση αλουμινίου, λίμες για την αφαίρεση υλικού καθώς και φρέζα για αφαίρεση υλικού με μεγάλη ακρίβεια. Όλες οι αναγκαίες κατεργασίες έγιναν στο μηχανουργείο του τμήματος Ηλεκτρονικής του ΤΕΙ Κρήτης.

Το τελικό σχέδιο του αλουμινένιου πλαισίου έχει σταυροειδές σχήμα, ώστε στο κάθε άκρο να προσαρμόζονται οι βάσεις των κινητήρων, ενώ στο κέντρο του στηρίζεται η πλακέτα ελέγχου. Το αλουμινένιο πλαίσιο παρουσιάζεται αναλυτικά στο σχήμα που ακολουθεί.



Σχήμα 3.1 :Το πλαίσιο στήριξης της προτεινόμενης κατασκευής.

Στο σχήμα παρουσιάζονται οι τέσσερις υποδοχές όπου προσαρμόζονται οι κινητήρες, καθώς και οι αντίστοιχες υποδοχές για τις βίδες στήριξης. Η κεντρική βίδα είναι υπεύθυνη για τη σωστή συναρμογή των δύο ράβδων καθώς και για την προσαρμογή της πλακέτας ελέγχου. Οι τέσσερις πλευρές του αλουμινένιου πλαισίου είναι συμμετρικές μεταξύ τους και έχουν τη μορφή που φαίνετε παρακάτω.



Σχήμα 3.2: Διατομές οπών αλουμινένιου πλαισίου.

Οι οπές στις οποίες στηρίζονται οι βάσεις των κινητήρων έχουν διάμετρο 3mm. Οι οπές με διάμετρο 4.5mm έχουν δημιουργηθεί έτσι ώστε να επιτευχθεί καλύτερη ροή του αέρα με αποτέλεσμα την καλύτερη ψύξη του κινητήρα. Η οπή με διάμετρο 7.5mm δημιουργήθηκε έτσι ώστε να περνά ο άξονας του κινητήρα, ενώ οι οπές των 8.5mm και 10mm δημιουργήθηκαν για να περάσουν τα καλώδια και για μείωση βάρους. Τέλος αφαιρέθηκε υλικό σε σχήμα ορθογωνίου έτσι ώστε να τοποθετηθεί το ηλεκτρονικό σύστημα ελέγχου ταχύτητας (ESC) το οποίο σταθεροποιήθηκε με τη βοήθεια των οπών με διατομή 3.5mm.

### 3.2 ΠΛΑΚΕΤΑ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΥ

Για τις ανάγκες της εργασίας επιλέχθηκε κατάλληλος μικροεπεξεργαστής με τη βοήθεια του οποίου ελέγχουμε τις στροφές των τεσσάρων κινητήρων αλλά και επικοινωνούμε ασύρματα με ηλεκτρονικό υπολογιστή.

Μετά από έρευνα που πραγματοποιήθηκε καταλήξαμε στον 8-Bit μικροεπεξεργαστή ATmega328p της εταιρίας AVR διότι καλύπτει όλες απαιτήσεις της εργασίας και παρέχει και επιπλέον δυνατότητες που μπορεί να χρησιμοποιηθούν στο μέλλον. Συγκριτικά στοιχεία του συγκεκριμένου μικροεπεξεργαστή σε σχέση με άλλους εμπορικά διαθέσιμους παρατίθενται στον πίνακα που ακολουθεί.



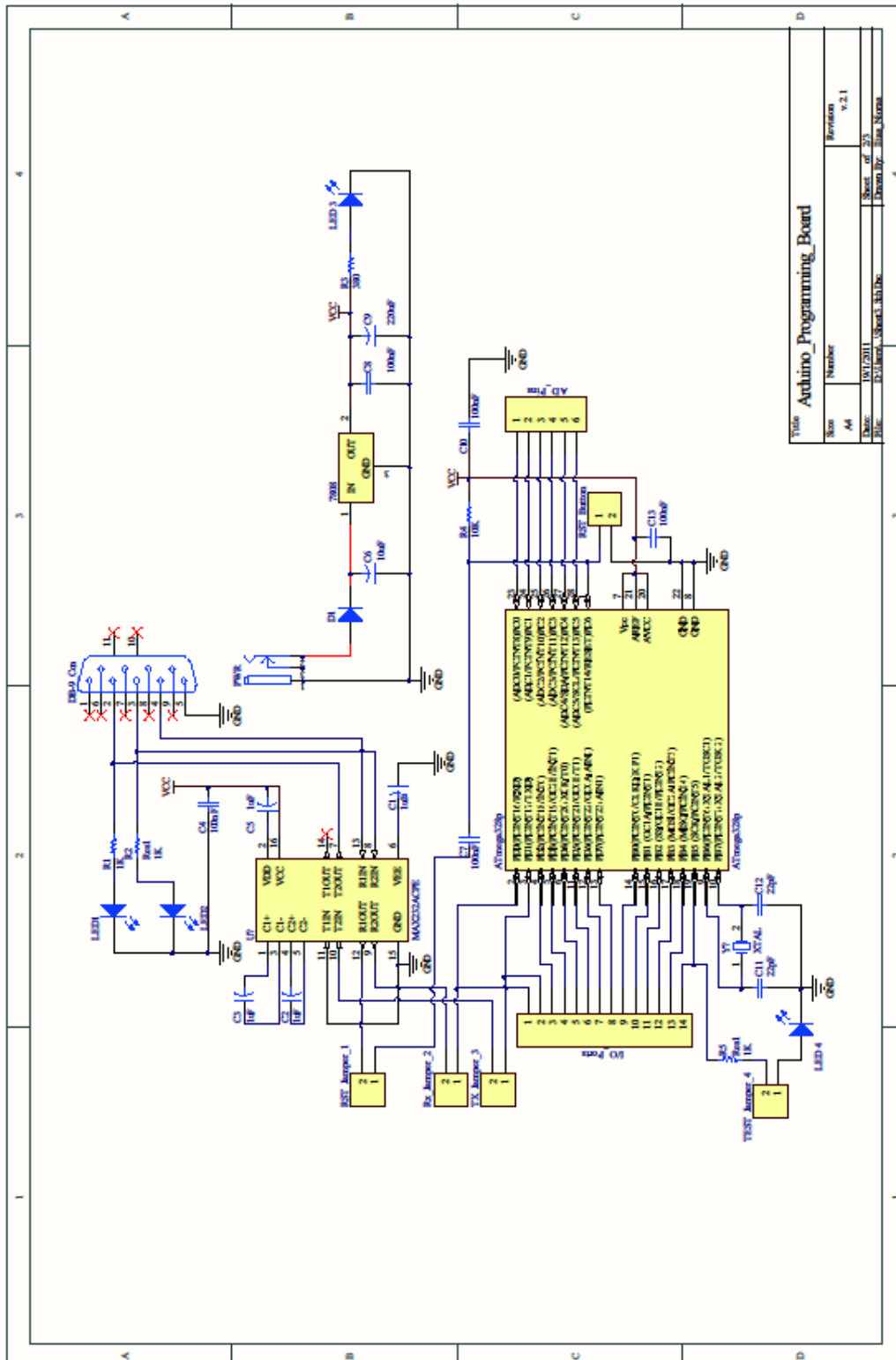
Arduino	Diecimila	Uno	Mega	Mega2560
Processor (ATmel) AVR	ATmega168	ATmega328	ATmega1280	ATmega2560
KB Flash	16	32	128	256
KB EEPROM	0.5	1	4	4
KB SRAM	1	2	8	8
Digital I/O Pins (40 mA)	14	14	54	54
Analog Input Pins	6	6	16	16

*Πίνακας 3.1 Λίστα δυνατοτήτων επεξεργαστών που υποστηρίζουν λογισμικό Arduino.*

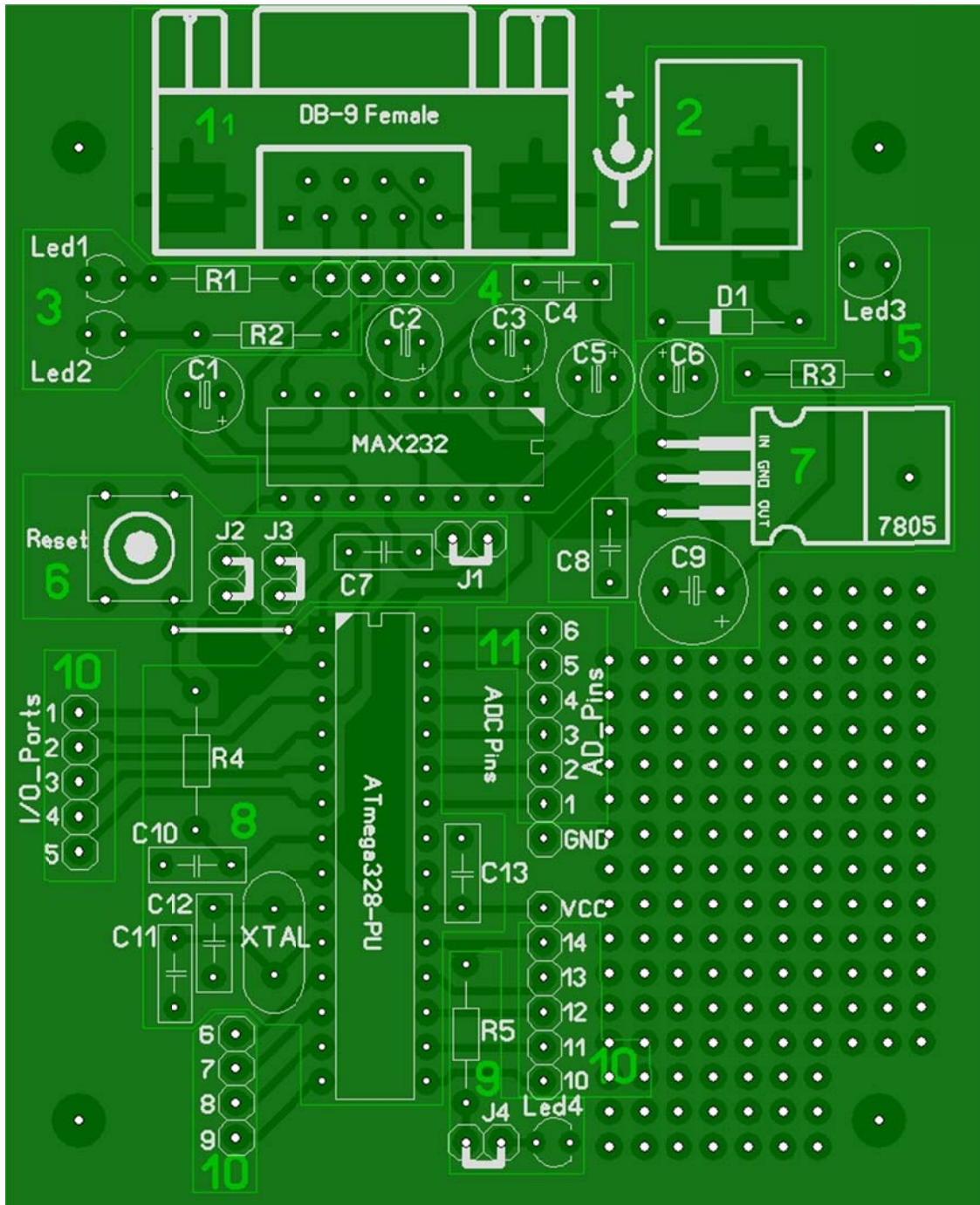
Τα βασικά χαρακτηριστικά του μικροεπεξεργαστή που επιλέχθηκε είναι ότι, διαθέτει έξι κανάλια που παράγουν παλμικό σήμα διαμορφωμένο κατά πλάτος (PWM), δύο 8-bit χρονόμετρα/μετρητές και ένα 16-bit χρονόμετρο/μετρητή, 32K bites μνήμη προγραμματισμού, επαφές που μετατρέπουν το αναλογικό σήμα σε ψηφιακό (AD converter) και δύο επαφές για εξωτερικές παρεμβολές (interrupts). Το σημαντικότερο πλεονέκτημα του είναι ότι έχει την δυνατότητα, με τη βοήθεια της κατάλληλης πλακέτας προγραμματισμού, να προγραμματιστεί χρησιμοποιώντας ως πηγαίο πρόγραμμα εκκίνησης Arduino (boot loader Arduino) και γλώσσα προγραμματισμού Arduino, που είναι φιλική προς το χρήστη και διατίθεται ελεύθερα στο διαδίκτυο[[www.Arduino.cc](http://www.Arduino.cc)].

Για τον προγραμματισμό του μικροεπεξεργαστή είναι απαραίτητη η χρήση κατάλληλης πλακέτας προγραμματισμού, συμβατής και με τον μικροεπεξεργαστή Atmega328p αλλά και με τη γλώσσα προγραμματισμού Arduino. Στο εμπόριο υπάρχουν πληθώρα από πλακέτες προγραμματισμού από τις οποίες θα μπορούσαμε να επιλέξουμε την κατάλληλη, ωστόσο θέλοντας να μειώσουμε το κόστος σχεδιάστηκε και κατασκευάστηκε πλακέτα προγραμματισμού που βοήθησε στη εξοικείωση με τη γλώσσα και το περιβάλλον προγραμματισμού.

Υπάρχουν αρκετές εμπορικά διαθέσιμες πλακέτες αλλά στα πλαίσια της εργασίας επιλέχθηκε η ανάπτυξη μια πλακέτας προσαρμοσμένης στις δικές μας ανάγκες. Στο σχήμα 3.3 παρουσιάζεται το σχηματικό της πλακέτας προγραμματισμού, ενώ στο σχήμα 3.4 προκειμένου ο αναγνώστης να κατανοήσει καλύτερα τις λειτουργίες της γίνεται διαχωρισμός της σε λειτουργικές περιοχές, που με την σειρά τους αναλύονται.



Σχήμα 3.3 :Σχηματικό πλακέτας προγραμματισμού



Σχήμα.3.4 :Περιοχές πλακέτας προγραμματισμού

Η πλακέτα προγραμματισμού αποτελείται από έντεκα περιοχές όπως παρουσιάζετε στο Σχήμα 3.4 . Οι περιοχές αυτές αναλύονται παρακάτω.

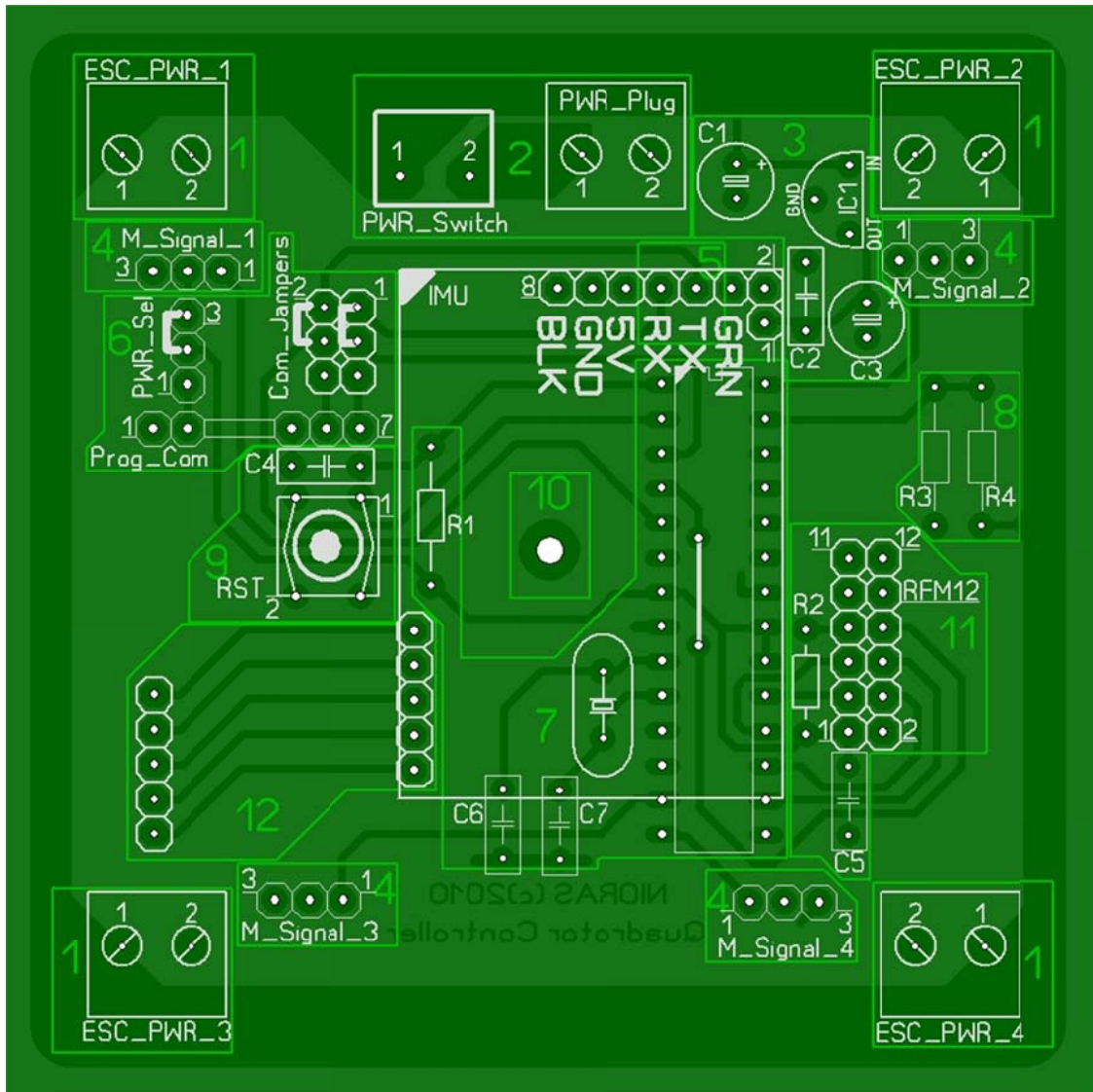
1. Στην περιοχή 1 έχει τοποθετηθεί υποδοχή για σειριακή επικοινωνία. Η υποδοχή DB-9 δίνει την δυνατότητα ενσύρματης επικοινωνίας με ηλεκτρονικό υπολογιστή για την μεταφορά δεδομένων από και προς την πλακέτα προγραμματισμού.
2. Στη περιοχή 2 λαμβάνουμε ρεύμα από εξωτερική πηγή τροφοδοσίας όπως μπαταρία ή τροφοδοτικό. Η διάδος ασφαλείας D1 αποκόπτει το ρεύμα σε περίπτωση που το κύκλωμα τροφοδοτηθεί με λανθασμένη πολικότητα.
3. Η περιοχή 3 περιλαμβάνει δύο φωτοдиодους που ενημερώνουν τον χρήστη πότε γίνεται αποστολή ή λήψη δεδομένων από τη σειριακή θύρα. Οι δύο φωτοдиодοι είναι 3mm και χρησιμοποιούμε της αντιστάσεις R1 και R2 με τιμές 1K Ohm για να πετύχουμε τη σωστή λειτουργία τους.
4. Στην περιοχή 4 γίνεται η μετατροπή του πρωτοκόλλου της σειριακής θύρας (RS-232) σε πρωτόκολλο που μπορεί να αναγνωρίσει ο μικροεπεξεργαστής ATmega328p (TTL). Η μετατροπή επιτυγχάνεται με τη βοήθεια του ολοκληρωμένου κυκλώματος MAX232 και με τη βοήθεια των ηλεκτρολυτικών πυκνωτών C1 1μF, C2 1μF, C3 1μF και C5 1μF και με την κατάλληλη συνδεσμολογία. Ο πυκνωτής C4 100nF είναι πυκνωτής εξομάλυνσης για την τροφοδοσία του ολοκληρωμένου MAX232.
5. Η περιοχή 5 ενημερώνει τον χρήστη αν το κύκλωμα λαμβάνει σωστή τροφοδοσία ρεύματος από την πηγή. Κατά την ορθή πόλωση του κυκλώματος η φωτοдиодος 3 φωτοβολεί ενώ εάν το κύκλωμα τροφοδοτηθεί με λανθασμένης πολικότητας τάση η φωτοдиодος δεν φωτοβολεί ενημερώνοντας τον χρήστη ότι το κύκλωμα δεν λαμβάνει ρεύμα. Για τη σωστή λειτουργία της φωτοдиодου των 5mm χρησιμοποιούμε την αντίσταση R3 380Ohm.
6. Στην περιοχή 6 βρίσκονται οι βραχυκυκλωτήρες που ενώνουν τη σειριακή θύρα με τον μικροεπεξεργαστή καθώς και το κουμπί εξωτερικής επανεκκίνησης. Οι βραχυκυκλωτήρες J1, J2 και J3 αν αφαιρεθούν αποκόπτουν την σειριακή επικοινωνία της πλακέτας προγραμματισμού με τον ηλεκτρονικό υπολογιστή, ενώ το κουμπί RESET δίνει την δυνατότητα στον χρήστη να πραγματοποιήσει εκκίνηση του μικροεπεξεργαστή ανά πάσα στιγμή το επιθυμεί. Ο πυκνωτής C7 100nF εξομαλύνει την τάση με την οποία θα τροφοδοτηθεί ο μικροεπεξεργαστής.
7. Στην περιοχή 7 έχει τοποθετηθεί κύκλωμα κατάλληλο ώστε να μετατρέπει την τάση που λαμβάνουμε από την εξωτερική τροφοδοσία σε τάση κατάλληλη για τα στοιχεία του κυκλώματος. Το ολοκληρωμένο 7805 έχει την δυνατότητα να σταθεροποιεί την τάση στην έξοδο του στα 5V αν στην είσοδο του λάβει τάση από 6V έως και 18V, αυτό το πετυχαίνει με την βοήθεια των δύο ηλεκτρολυτικών πυκνωτών C6 10μF και C9 220μF. Ο πυκνωτής C8 100nF εξομαλύνει την τάση με την οποία θα τροφοδοτηθεί ο μικροεπεξεργαστής.

8. Στην περιοχή 8 βρίσκεται ο μικροεπεξεργαστής και το κύκλωμα χρονισμού. Για την σωστή λειτουργία του μικροεπεξεργαστή είναι απαραίτητη η χρήση ενός κρυστάλλου που θα συγχρονίζει με κατάλληλους παλμούς τα ρολόγια του. Αυτό επιτυγχάνετε με τη χρήση του κρυστάλλου XTAL 16MHz και δύο πυκνωτών C11 και C12, 22pF ο καθένας. Η αντίσταση R4 10KOhm είναι τοποθετημένη για τη σωστή λειτουργία της εξωτερικής επανεκκίνησης, ενώ οι πυκνωτές C10 και C13 των 100nF εξομαλύνουν την τάση που τροφοδοτήστε ο μικροεπεξεργαστής.
9. Η περιοχή 9 αποτελεί ένδειξη λειτουργίας. Η φωτοдиодος 4 είναι κατάλληλα συνδεδεμένη, μέσω της αντίστασης R5 1K Ohm, στην επαφή 14 του μικροεπεξεργαστή. Όταν τοποθετήσουμε για πρώτη φορά έναν μικροεπεξεργαστή με πηγαίο πρόγραμμα Arduino στο κύκλωμά θα πρέπει η φωτοдиодος να αναβοσβήνει προς ένδειξη σωστής λειτουργίας. Η φωτοдиодος αυτή μπορεί να χρησιμοποιηθεί και στο μέλλον με διάφορους τρόπους από τον χρήστη με σκοπό την εξοικείωση του με το προγραμματιστικό περιβάλλον, Σε περίπτωση που δεν είναι επιθυμητή η χρήση της φωτοδόδου 4 μπορεί να απενεργοποιηθεί αφαιρώντας τον βραχυκυκλωτήρα J4.
10. Στην περιοχή 10 παρουσιάζονται οι επαφές εισόδου/εξόδου. Οι επαφές αυτές μπορούν να χρησιμοποιηθούν είτε σαν είσοδος είτε σαν έξοδος από τον χρήστη, με την βοήθεια του κατάλληλου λογισμικού είναι δυνατή η χρήση των επαφών αυτών για την οδήγηση κινητήρων, ρύθμιση της φωτεινότητας φωτοδόδων, ασύρματης επικοινωνίας με άλλες συσκευές και πληθώρα από άλλες εφαρμογές.

### 3.3 ΚΕΝΤΡΙΚΗ ΠΛΑΚΕΤΑ ΕΛΕΓΧΟΥ

Στη παράγραφο που ακολουθεί θα αναλυθεί η κεντρική πλακέτα ελέγχου που σχεδιάστηκε και κατασκευάστηκε. Η πλακέτα αυτή είναι τοποθετημένη στο κέντρο της κατασκευής και σε αυτή είναι τοποθετημένος ο μικροεπεξεργαστής που με το κατάλληλο λογισμικό επιτρέπει τον έλεγχο των τεσσάρων κινητήρων αλλά και την ασύρματη επικοινωνία με ένα σταθμό βάσης. Στο σχήμα 3.3. παρουσιάζεται το σχηματικό της κεντρικής πλακέτας ελέγχου. Ο σχεδιασμός έγινε λαμβάνοντας υπόψη το αλουμινένιο πλαίσιο που σχεδιάστηκε. Κατά τη φάση του σχεδιασμού του κυκλώματος έγινε προσπάθεια να αφαιρεθούν όλα τα κομμάτια του κυκλώματος που δεν ήταν απαραίτητα. Κύριος λόγος ήταν η προσπάθεια διατήρησης του βάρους σε όσο το δυνατόν χαμηλότερα επίπεδα.





Σχήμα 3.6: Περιχές κεντρικής πλακέτας ελέγχου

Τα εξαρτήματα που χρησιμοποιήθηκαν στην κεντρική πλακέτα ελέγχου παρουσιάζονται αναλυτικά στον πίνακα 3.1.

Λίστα Εξαρτημάτων:	
R1,R2,R4	10KΩ
R3	8.2KΩ
C1	10μF / 35Volts
C3	220mF / 16Volts

C2,C4,C5	100nF
C6,C7	22pF
XTAL	16MHz
IC1	7808
IC2	Atmega328p
Jack1	Wire jacks
PWR	Power switch

Πίνακας 3.2 Λίστα εξαρτημάτων πλακέτας ελέγχου

Η κεντρική πλακέτα ελέγχου αποτελείται από δεκατρία επιμέρους τμήματα (Σχήμα 3.3), με διαφορετικές επιμέρους λειτουργίες που θα αναλυθούν στη συνέχεια.

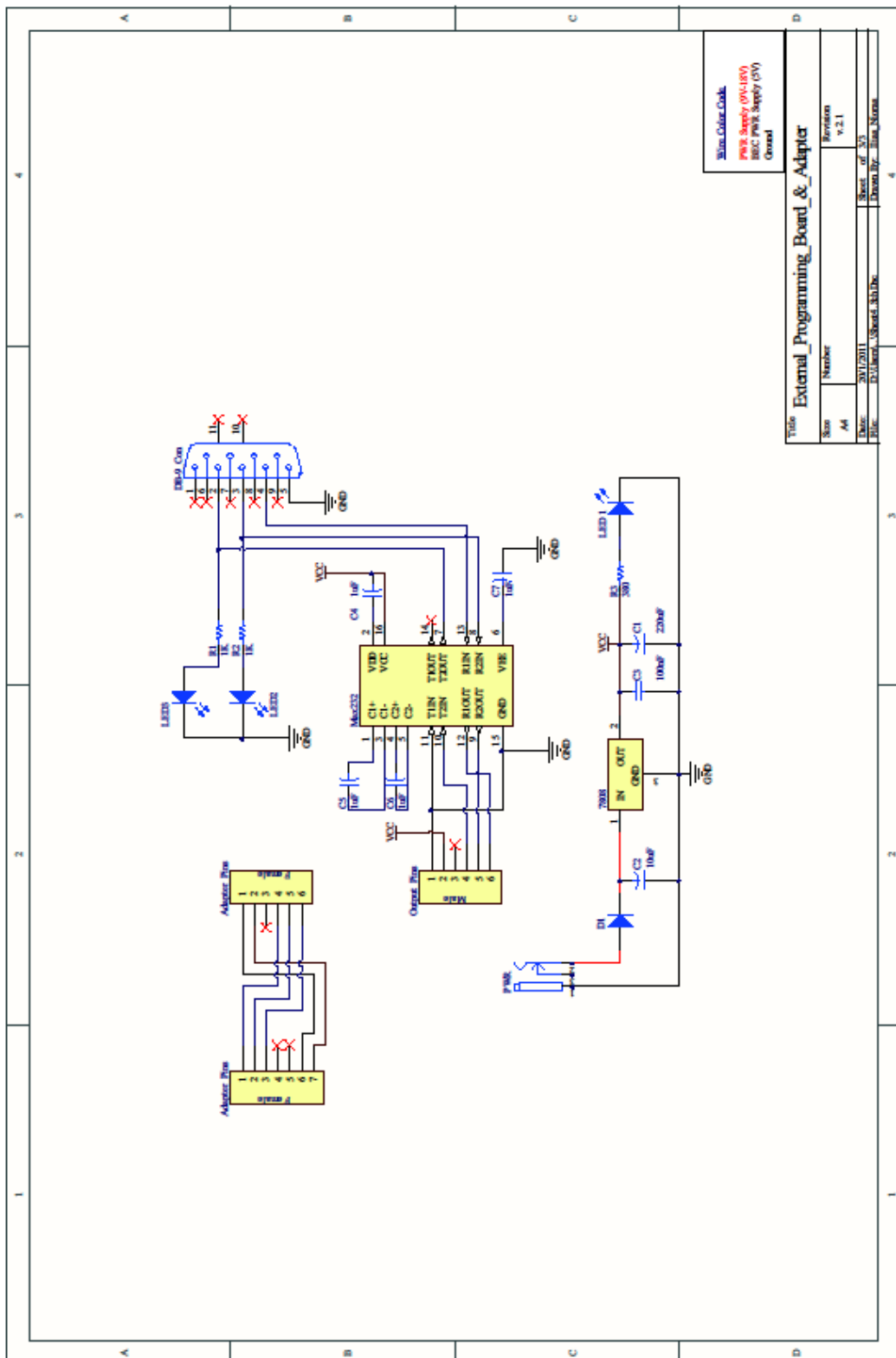
1. Η περιοχή 1 αποτελείται από τέσσερις υποδοχές όπου συνδέονται οι τροφοδοσίες των τεσσάρων ηλεκτρονικών ελεγκτών ταχύτητας (ESCs). Ο ένας ακροδέκτης βρίσκεται στη γείωση ενώ ο άλλος στην τροφοδοσία της μπαταρίας που αντιστοιχεί σε 11.1Volts ή στον θετικό ακροδέκτη του τροφοδοτικού που αντιστοιχεί σε 12Volts. Οι αγωγοί είναι αρκετά μεγάλοι, καθώς τα ρεύματα που τροφοδοτούν τα ESCs και κατ'επέκταση τους κινητήρες είναι πολύ μεγάλα.
2. Η περιοχή 2 αποτελείται από δύο εξαρτήματα, το ένα είναι μια υποδοχή ανάλογη αυτών που χρησιμοποιήθηκαν για τη σύνδεση των ESCs και ενώνει την μπαταρία ή το τροφοδοτικό με την πλακέτα ελέγχου. Το δεύτερο εξάρτημα είναι ένας απλός διακόπτης που ρυθμίζει την τροφοδοσία της πλακέτας ελέγχου.
3. Στη περιοχή 3 υπάρχει το κύκλωμα σταθεροποίησης τάσης. Αυτό είναι υπεύθυνο για την μετατροπή της τάσης που λαμβάνεται από την μπαταρία ή το τροφοδοτικό σε τάση κατάλληλη για την πλακέτα των αισθητήρων. Η σταθεροποίηση της τάσης επιτυγχάνετε με δύο πυκνωτές C1 και C3 που είναι ηλεκτρολυτικοί πυκνωτές με τιμές 10μF στα 35Volts και 220μF στα 16Volts αντίστοιχα, ο πυκνωτής C2 είναι πυκνωτής σύζευξης και έχει τιμή 100nF. Τέλος χρησιμοποιούμε το ολοκληρωμένο 78L33 (IC1) που είναι ένας σταθεροποιητής τάσης στα 3.3Volts.
4. Στη περιοχή 4 υπάρχουν οι υποδοχές ελέγχου των κινητήρων με διαμόρφωση εύρους παλμού (Pulse Width Modulation ή PWM). Ο μικροεπεξεργαστής με τη βοήθεια του κατάλληλου λογισμικού και σύμφωνα με τις ενδείξεις των αισθητήριων αποστέλλει σε κάθε μία από τις τέσσερις αυτές υποδοχές ένα σήμα PWM το οποίο στη συνέχεια λαμβάνετε από τα ESCs και αναλόγως κινούνται οι κινητήρες. Όταν το ESC τροφοδοτηθεί με σωστή τάση επιστρέφει σταθερή τάση 5Volts. Την ιδιότητα αυτή την εκμεταλλευόμαστε και λαμβάνουμε την τάση που επιστρέφουν τα ESCs ώστε να τροφοδοτήσουμε όλα τα εξαρτήματα που λειτουργούν με τάση εισόδου 5 Volts όπως είναι ο μικροεπεξεργαστής και το σύστημα ασύρματης επικοινωνίας.



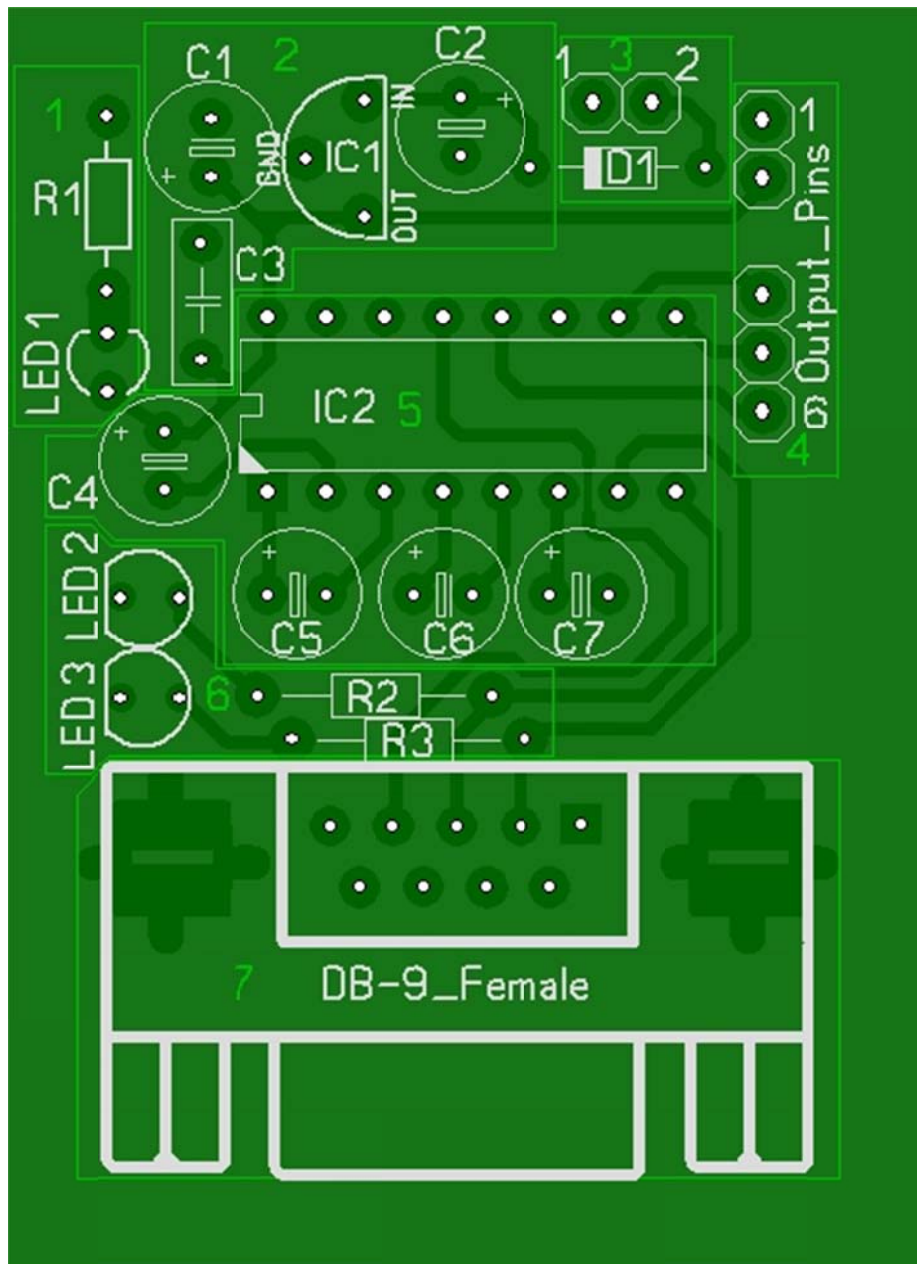
5. Στη περιοχή 5 υπάρχουν οι υποδοχές επικοινωνίας με την πλακέτα αισθητηρίων. Οι υποδοχές αυτές συνδέουν την πλακέτα των αισθητηρίων με τις υποδοχές της σειριακής θύρας του μικροεπεξεργαστή. Το Tx του μικροεπεξεργαστή ενώνετε με το Rx της πλακέτας αισθητηρίων και το Rx του μικροεπεξεργαστή με το Tx της πλακέτας αισθητηρίων αντίστοιχα.
6. Στη περιοχή 6 υπάρχουν οι υποδοχές προγραμματισμού του μικροεπεξεργαστή. Οι υποδοχές που παρουσιάζονται στο κομμάτι αυτό χρησιμοποιούνται για τον προγραμματισμό του μικροεπεξεργαστή με τη χρήση ειδικού μετατροπέα και βοηθητικής πλακέτας προγραμματισμού. Ο μετατροπέας συνδέεται στις πέντε υποδοχές κατάλληλα ενώ ταυτόχρονα οι βραχυκυκλωτήρες ενώνουν τις κάτω υποδοχές κατά τον προγραμματισμό. Στην περίπτωση αυτή η κεντρική πλακέτα ελέγχου λαμβάνει τροφοδοσία από βοηθητική πλακέτα προγραμματισμού και η σειριακή θύρα του μικροεπεξεργαστή συνδέετε με ηλεκτρονικό υπολογιστή. Μετά τον προγραμματισμό της κεντρικής πλακέτας ελέγχου αποσυνδέουμε τον μετατροπέα και την βοηθητική πλακέτα προγραμματισμού και μετακινούμε τους βραχυκυκλωτήρες στις πάνω θέσεις, έτσι ώστε η κεντρική πλακέτα ελέγχου να λαμβάνει τροφοδοσία από τα ESCs και η σειριακή θύρα του μικροεπεξεργαστή συνδέετε με την πλακέτα των αισθητηρίων.
7. Στη περιοχή 7 είναι τοποθετημένος ο μικροεπεξεργαστής που είναι υπεύθυνος για τον έλεγχο των τεσσάρων κινητήρων και την ασύρματη επικοινωνία. Ο μικροεπεξεργαστής λαμβάνει πληροφορίες μέσω της σειριακής θύρας που διαθέτει είτε από την πλακέτα αισθητηρίων (αυτόνομη λειτουργία της κατασκευής), είτε από τον ηλεκτρονικό υπολογιστή κατά τον προγραμματισμό όπως έχει αναφερθεί στην έκτη περιοχή του κυκλώματος. Για να επιτευχθεί ο συγχρονισμός των ρολογιών του μικροεπεξεργαστή έχει τοποθετηθεί κρύσταλλος 16MHz σε συνδυασμό με δύο πυκνωτές C6 και C7 χωρητικότητας 22pF, η αντίσταση R1 χρησιμοποιείται για να επιτευχθεί σωστή πόλωση της τροφοδοσίας του μικροεπεξεργαστή.
8. Στη περιοχή 8 γίνεται ο έλεγχος της τάσης τροφοδοσίας. Το κύκλωμα αυτό λαμβάνει σαν είσοδο την τάση τροφοδοσίας από την μπαταρία που μέσω ενός διαιρέτη τάσης οδηγείται σε μία από τις αναλογικές εισόδους του μικροεπεξεργαστή. Η είσοδος αυτή έχει τη δυνατότητα μετατροπής του αναλογικού σήματος σε ψηφιακό (A/D converter). Στην περίπτωση που η μπαταρία εξασθενίσει ο μικροεπεξεργαστής ενημερώνει τον χειριστή και σταδιακά σβήνει τους κινητήρες, προστατεύοντας την μπαταρία. Ο διαιρέτης τάσης αποτελείται από δύο αντιστάσεις R2 10KΩ και R3 8.2KΩ.
9. Στη περιοχή 9 βρίσκεται το κύκλωμα επανεκκίνησης. Το κύκλωμα αυτό δίνει τη δυνατότητα στο χειριστή να πραγματοποιήσει εξωτερικά επανεκκίνηση του μικροεπεξεργαστή πατώντας ένα κουμπί. Ο πυκνωτής C4 με τιμή 100nF φιλτράρει την πληροφορία επανεκκίνησης που λαμβάνει ο μικροεπεξεργαστής όταν πατηθεί το κουμπί RST ή όταν πραγματοποιηθεί επανεκκίνηση από τον ηλεκτρονικό υπολογιστή κατά τον προγραμματισμό.
10. Στη περιοχή 10 βρίσκεται το σημείο στήριξης της κεντρικής πλακέτας ελέγχου. Στο κέντρο της πλακέτας ελέγχου δημιουργήσαμε οπή τέτοια ώστε η βίδα που συνδέει τις δύο αλουμινένιες ράβδους, που αποτελούν το πλαίσιο να περνά μέσα από την οπή αυτή και σταθεροποιεί την κεντρική πλακέτα ελέγχου πάνω στο σασί

- της κατασκευής. Κατά τον σχεδιασμό αποφύγαμε την τοποθέτηση των αγωγών κοντά στο σημείο στήριξης. Τοποθετήθηκε κατάλληλο μονωτικό υλικό ανάμεσα στο αλουμινένιο σασί και την κεντρική πλακέτα προγραμματισμού ώστε να αποφευχθούν τυχόν βραχυκυκλώματα και να μειωθούν οι κραδασμοί που δημιουργούν οι κινητήρες κατά τη λειτουργία τους.
11. Στη περιοχή 11, βρίσκεται το κύκλωμα ασύρματης επικοινωνίας. Είναι υπεύθυνο για την ασύρματη μεταφορά της πληροφορίας που στέλνει ο χειριστής από τον ηλεκτρονικό υπολογιστή προς τη κεντρική πλακέτα ελέγχου και αντίστροφα. Αποτελείτε από την αντίσταση R4 με τιμή 10KΩ και έναν πυκνωτή φιλτραρίσματος 100nF. Στη διπλή σειρά υποδοχών τοποθετήθηκε το κύκλωμα της ασύρματης επικοινωνίας. Όταν ο μικροεπεξεργαστής δεχτεί σήμα στην υποδοχή πέντε τότε ενεργοποιείται η ασύρματη επικοινωνία και αποστέλλετε η λαμβάνετε πληροφορία.
  12. Στη περιοχή 12 βρίσκονται υποδοχές που μπορούν να χρησιμοποιηθούν είτε σαν είσοδοι είτε σαν έξοδοι και συνδέονται πάνω στην πλακέτα αισθητηρίων ώστε να χρησιμοποιηθούν στο μέλλον για την βελτίωση των δυνατοτήτων τις κατασκευής.

Για τον προγραμματισμό του μικροεπεξεργαστή που βρίσκεται πάνω στην κεντρική πλακέτα ελέγχου κατασκευάστηκε μια βοηθητική πλακέτα προγραμματισμού Σχήμα 3.4. Η δυνατότητα αυτή απαιτεί την τοποθέτηση των βραχυκυκλωτήρων στη σωστή θέση. Η βοηθητική πλακέτα προγραμματισμού αποτελείτε από περιοχές ίδιες με αυτές της πλακέτας προγραμματισμού που έχουν αναλυθεί σε προηγούμενη παράγραφο (βλέπε παράγραφο 3.2).



Σχήμα 3.7 :Σχηματικό βοηθητικής πλακέτας προγραμματισμού

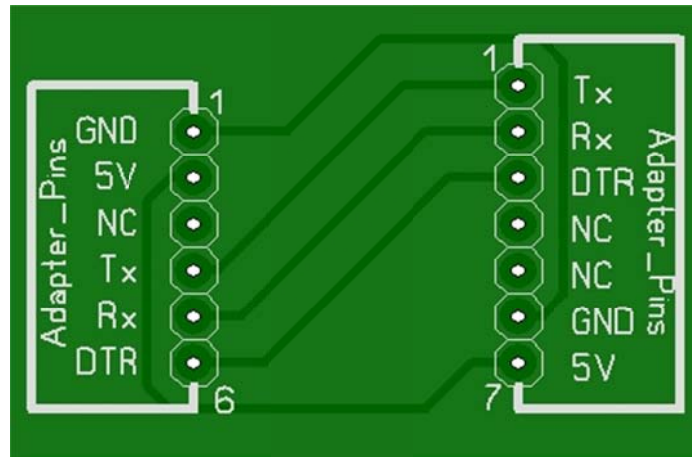


Σχήμα 3.8: Περιοχές βοηθητικής πλακέτας προγραμματισμού

Λίστα Εξαρτημάτων:	
C1	22uF
C2	10uF
C4,C5,C6,C7	1μF / 35Volts
C3	100nF
D1	
R1	380Ω
R2,R3	1KΩ
LED1	5mm (Green)
LED2,LED3	3mm (Red)
IC1	7805
IC2	MAX232
DB-9	Serial female

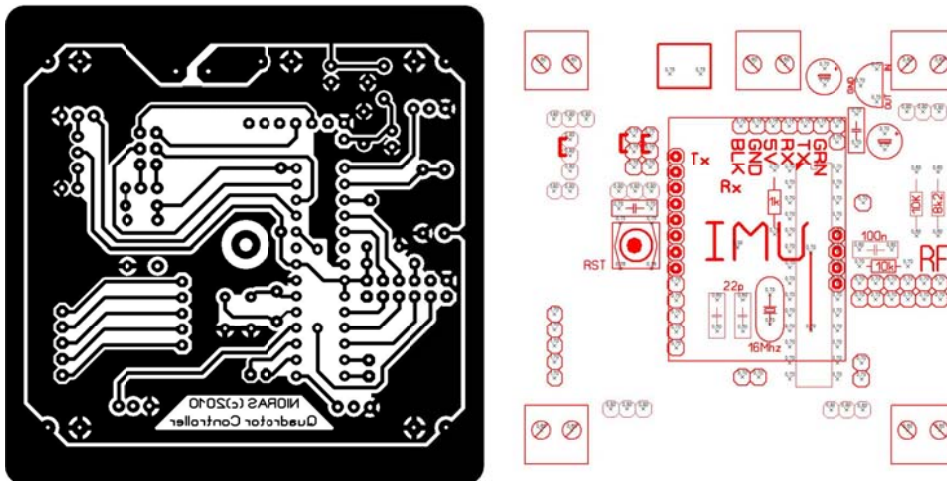
Πίνακας 3.3.Λίστα εξαρτημάτων βοηθητικής πλακέτας προγραμματισμού

Η βοηθητική πλακέτα προγραμματισμού είναι ένα κύκλωμα ενσύρματης επικοινωνίας μεταξύ της κεντρικής πλακέτας ελέγχου και του ηλεκτρονικού υπολογιστή. Κατά τον προγραμματισμό συνδέουμε κατάλληλα τις υποδοχές της βοηθητικής πλακέτας προγραμματισμού Output\_Pins (Βλέπε περιοχή 4, Σχήμα 3.8 Output\_Pins) στις υποδοχές για επικοινωνία προγραμματισμού Prog\_Com της κεντρικής πλακέτας ελέγχου (Βλέπε περιοχή 6, Σχήμα 3.3, Prog\_Com. Τοποθετούμε τον βραχυκυκλωτήρα PWR\_Sel στη θέση ένα (1) και δύο (2) C έτσι ώστε η κεντρική πλακέτα ελέγχου να τροφοδοτηθεί με ρεύμα από την βοηθητική πλακέτα προγραμματισμού τέλος θα πρέπει να τοποθετήσουμε τους βραχυκυκλωτήρες Com\_Jampers στις θέσεις τέσσερα (4), έξι (6) και τρία (3), πέντε (5) (Βλέπε περιοχή 6, Σχήμα 3.3,Com\_Jampers ) ώστε να αποκόψουμε την επικοινωνία της κεντρικής πλακέτας ελέγχου από την πλακέτα αισθητηρίων και να τη συνδέσουμε με την σειριακή θύρα. Για την σωστή επικοινωνία μεταξύ κεντρικής πλακέτας ελέγχου και βοηθητικής πλακέτας προγραμματισμού σχεδιάστηκε και κατασκευάστηκε κατάλληλος προσαρμογέας που παρουσιάζετε παρακάτω (Σχήμα 3.5).

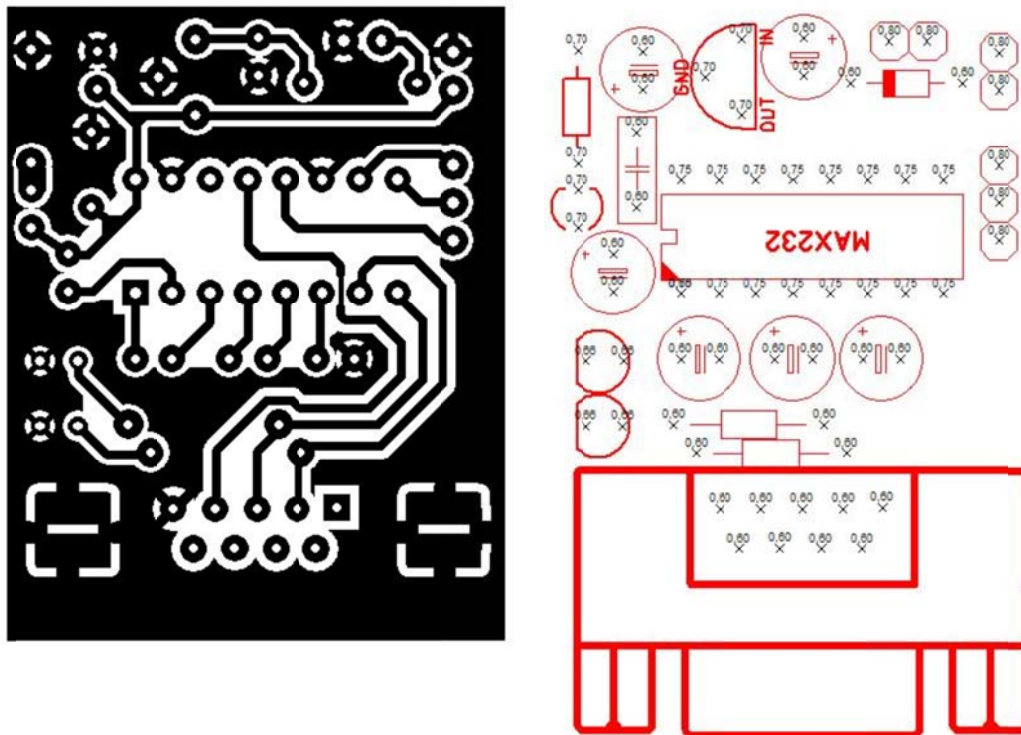


Σχήμα 3.9: Προσαρμογέας

Στο σχήμα διακρίνονται έξι υποδοχές στα αριστερά, που συνδέονται με την εξωτερική πλακέτα προγραμματισμού. Στα δεξιά έχουν τοποθετηθεί επτά υποδοχές που συνδέονται κατάλληλα με την κεντρική πλακέτα ελέγχου. Η υποδοχή DTR (Data Terminal Ready) χρησιμοποιείται κατά τη λειτουργία του προγραμματισμού ενώ οι υποδοχές που έχουν σημειωθεί με NC (Non Connection) δεν συνδέονται πουθενά.. Με τον διαχωρισμό της κεντρικής πλακέτας ελέγχου από την εξωτερική πλακέτα προγραμματισμού επιτυγχάνεται η μείωση του βάρους αλλά κυρίως την μείωση της κατανάλωσης ρεύματος στην κεντρική πλακέτα ελέγχου, καθώς δεν τροφοδοτείται συνεχώς το ολοκληρωμένο MAX232.



Σχήμα 3.10: Κεντρική Πλακέτα Ελέγχου (σηματικό)



Σχήμα 3.11: Βοηθητική πλακέτα προγραμματισμού (σχηματικό)

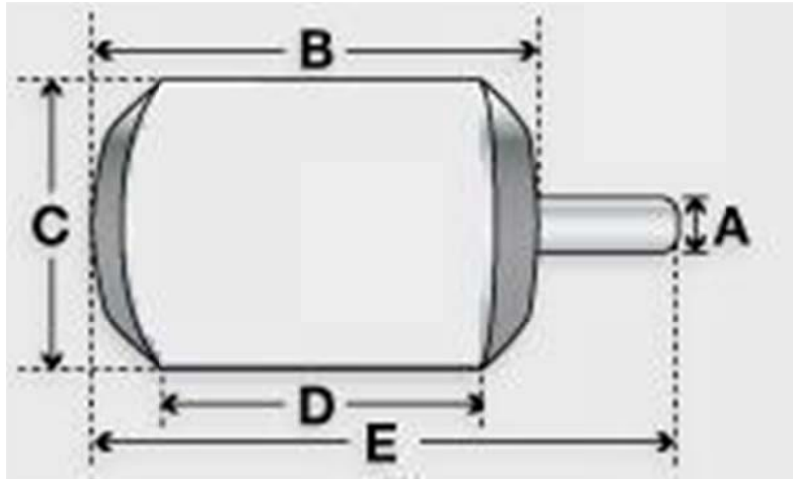
### 3.4 ΚΙΝΗΤΗΡΕΣ ΚΑΙ ΕΛΕΓΚΤΕΣ ΤΑΧΥΤΗΤΑΣ

Οι κινητήρες που επιλέχθηκαν και τοποθετήθηκαν στα τέσσερα άκρα της κατασκευής (όπως έχει σημειωθεί στο κεφάλαιο 3.1) είναι της εταιρίας Turnigy. Συγκεκριμένα επιλέχθηκε ο κινητήρας 2205 50turn 1088kv 8A Outrunner. Τα βασικά του χαρακτηριστικά παρουσιάζονται στον πίνακα 3.4

Χαρακτηριστικά Κινητήρα	
KV	1088 (rpm/V)
Βάρος	30 (g)
Ρεύμα λειτουργίας	6~8 (A)
Τάση λειτουργίας	11.1 (V)
Μέγιστο στιγμιαίο ρεύμα	8 (A)
Προτεινόμενη μπαταρία	1000mAh ανά κινητήρα τριών κελιών
Μέγεθος άξονα (A)	3.175 (mm)
Μήκος (B)	22 (mm)
Διάμετρος (C)	28 (mm)

Μήκος κλωβού (D)	9 (mm)
Ολικό μήκος (E)	37 (mm)

Πίνακας 3.4 Εργοστασιακά χαρακτηριστικά κινητήρων



Σχήμα 3.12 : Διαστάσεις κινητήρα



Εικόνα 3.13 : Κινητήρας Turnigy 2205 50turn 1088kv 8A Outrunner

Ο κινητήρας αυτός επιλέχτηκε διότι έχει χαμηλό ρεύμα λειτουργίας σε σχέση με τους αντίστοιχους κινητήρες της κατηγορίας του και επιτυγχάνει αρκετά μεγάλη προωθητική δύναμη σε χαμηλές στροφές που προτείνετε για κατασκευές ελικοπτέρων.

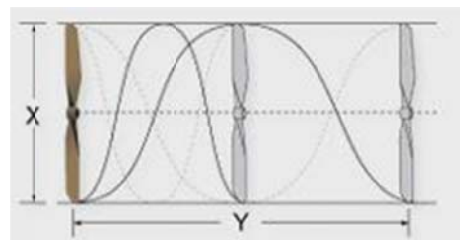


Στους κινητήρες αυτούς τοποθετήθηκαν κατάλληλοι έλικες που έχουν τη δυνατότητα να παράγουν και αυτοί με τη σειρά τους μεγάλη προωθητική δύναμη σε χαμηλές στροφές. Η επιλογή έγινε με μεγάλη προσοχή διότι υπάρχει μεγάλη ποικιλία από διαφορετικούς έλικες στην αγορά. Έπειτα από έρευνα καταλήξαμε στο συμπέρασμα ότι έπρεπε να χρησιμοποιηθούν έλικες με περισσότερα από δύο πτερύγια, με μήκος περίπου είκοσι εκατοστά. Οι έλικες που χρησιμοποιήθηκαν παρουσιάζονται παρακάτω.



Σχήμα 3.14 : Έλικες τριών πτερυγίων

Χαρακτηριστικά Έλικα	
Μήκος (X)	200 (mm)
Ανοιγμα (Y)	100 (mm)
Πτερύγια	3



Πίνακας 3.5: Εργοστασιακά χαρακτηριστικά έλικας

Οι έλικες αυτοί συνοδεύονται από κατάλληλους δακτυλίους που τοποθετούνται στο κέντρο ώστε να εφαρμόζουν με ακρίβεια στον άξονα του κινητήρα.

Για τις ανάγκες της κατασκευής αγοράστηκαν συνολικά τέσσερις έλικες με τα ίδια χαρακτηριστικά, δύο εκ των οποίων παράγουν προωθητική δύναμη προς τα κάτω

στρεφόμενοι δεξιόστροφα και δύο που παράγουν προωθητική δύναμη προς τα κάτω στρεφόμενοι αριστερόστροφα, διότι η κατασκευή υπάγεται στην δεύτερη κατηγορία ελικοπτέρων που προαναφέραμε στο κεφάλαιο 2.2 Είδη Ελικοπτέρων. Για να πετύχουμε ροπή προς την ίδια κατεύθυνση στρεφόμενοι αντίρροπα οι έλικες θα πρέπει να έχουν, ανά δύο, διαφορετική κλίση σε κάθε πτερύγιο ως προς τον άξονα περιστροφής που παρουσιάζετε στην εικόνα παρακάτω.

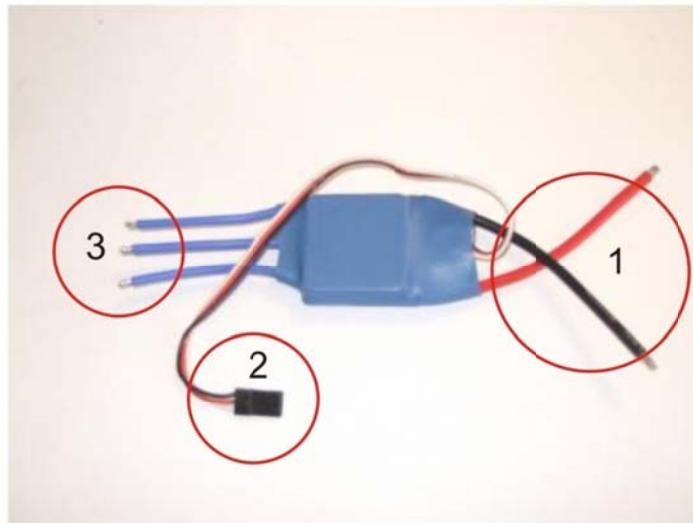


Σχήμα 3.15 : Αριστερόστροφη και δεξιόστροφη έλικα. Παρατηρούμε την διαφορά κλίσης στα πτερύγια τις κάθε έλικας ως προς τον άξονα περιστροφής.

Για τον έλεγχο των τεσσάρων κινητήρων χρησιμοποιήθηκαν τέσσερις ηλεκτρονικοί ελεγκτές ταχύτητας (Electronic Speed Controller), ένας σε κάθε κινητήρα. Οι ηλεκτρονικοί ελεγκτές ταχύτητας τροφοδοτούν τους κινητήρες με ρεύμα από την μπαταρία ή το τροφοδοτικό, λαμβάνουν από τον μικροεπεξεργαστή σήμα έτσι ώστε αυξάνουν η μειώνουν την ταχύτητα των κινητήρων. Στο σχήμα που ακολουθεί παρουσιάζονται: 1) τα καλώδια τροφοδοσίας, 2) τα καλώδια σήματος και 3) τα καλώδια εξόδου. Τα χαρακτηριστικά των ηλεκτρονικών ελεγκτών ταχύτητας παρουσιάζονται στον παρακάτω πίνακα.

Χαρακτηριστικά	
Βάρος	11 g
Ρεύμα λειτουργίας	10A
Μέγιστο Ρεύμα λειτουργίας	13A
Συμβατότητα με μπαταρίες Li-Poly	2-3 κελιών
Συμβατότητα με μπαταρίες Ni-MH	4-10 κελιών
Ένταση ρεύματος επιστροφής	1A

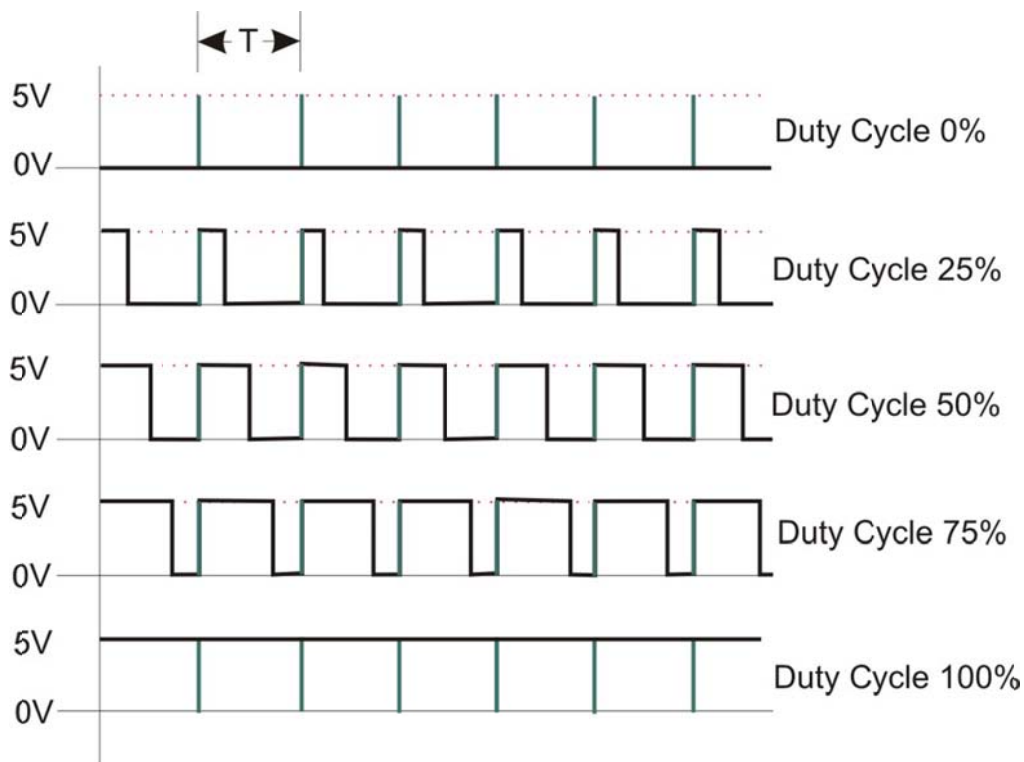
Πίνακας 3.6 Χαρακτηριστικά Ηλεκτρονικών ελεγκτών ταχύτητας



Σχήμα 3.16 Electronic speed controller.

Στην περιοχή ένα (1) του σχήματος έχουν σημειωθεί τα καλώδια τροφοδοσίας των ηλεκτρονικών ελεγκτών ταχύτητας που με τη σειρά τους συνδέονται στην κεντρική πλακέτα ελέγχου στην περιοχή ένα (1) (Βλέπε Σχήμα 3.3). Από τους αγωγούς αυτούς περνά το ρεύμα τροφοδοσίας των κινητήρων.

Στην περιοχή δύο (2) έχει σημειωθεί ένα καλώδιο με τρεις επαφές μία γείωση που αντιστοιχεί στο μαύρη επαφή, μία επιστροφή τάσης (BEC) που αντιστοιχεί στην κόκκινη επαφή και μία άσπρη που αντιστοιχεί στην επαφή σήματος. Το καλώδιο αυτό τοποθετείτε στις επαφές που έχουν σημειωθεί στην κεντρική πλακέτα ελέγχου (Βλέπε Σχήμα 3.6, Περιοχές 4) έτσι ώστε η άσπρη επαφή σήματος να συνδεθεί με την επαφή ένα (1) και η μαύρη επαφή γείωσης να συνδεθεί με την επαφή τρία (3). Το σήμα που στέλνει ο μικροεπεξεργαστής στην άσπρη επαφή σήματος των ηλεκτρονικών ελεγκτών ταχύτητας (Σχήμα 3.6 Περιοχή 4) είναι σήμα παλμού διαμορφωμένο κατά πλάτος (Pulse Width Modulation). Τα χαρακτηριστικά του σήματος συναρτήσκει του χρόνου είναι το πλάτος και η συχνότητα του παλμού. Οι περισσότεροι ESCs, όπως και αυτός που χρησιμοποιήσαμε έχουν σαν συχνότητα λειτουργεί τα 50Hz δηλαδή ο χρόνος μιας περιόδου είναι 2m sec. Το πλάτος του παλμού μέσα στον χρόνο μιας περιόδου χαρακτηρίζετε από τον κύκλο καθήκοντος (Duty Cycle), όσο μεγαλύτερος είναι ο κύκλος καθήκοντος του παλμού τόσο πιο γρήγορα θα κινηθεί ο κινητήρας. Ο κύκλος καθήκοντος εκφράζει το χρόνο που ο παλμός βρίσκεται σε κατάσταση υψηλής στάθμης σε χρόνο μιας περιόδου και παρουσιάζετε σε επί τις εκατό ποσοστό. Στο Σήμα 3.12 που ακολουθεί παρατηρούμε τον παλμό που δέχονται τα ESCs από τον μικροεπεξεργαστή και την αντιστοιχία τους σε κύκλους καθήκοντος.



Σχήμα 3.17 : Διαμόρφωση παλμού κατά πλάτος (PWM)

Για κύκλο καθήκοντος μηδέν τις εκατό ο παλμός είναι μηδενικός, που αντιστοιχεί σε μηδενική ροπή στους κινητήρες. Για κύκλο καθήκοντος πενήντα τις εκατό ο παλμός βρίσκεται σε κατάσταση υψηλής στάθμης για χρονικό διάστημα 1m sec, δηλαδή μισής περιόδου και σε κατάσταση χαμηλής στάθμης για το άλλο μισό της περιόδου, και οι κινητήρες αποδίδουν το μισό της ισχύος. Τέλος για κύκλο καθήκοντος εκατό τις εκατό ο παλμός βρίσκεται μόνιμος σε υψηλή στάθμη για ολόκληρο το χρόνο μιας περιόδου, στην κατάσταση αυτή ο κινητήρας αποδίδει την μέγιστη ροπή.

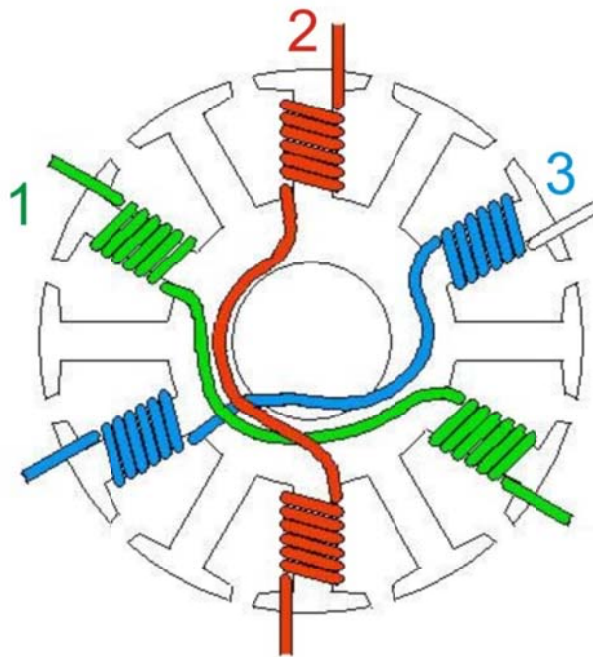
Το κόκκινο καλώδιο επιστροφής τάσης, επιστρέφει σταθερή τάση 5V όταν τροφοδοτηθούν οι ηλεκτρονικοί ελεγκτές ταχύτητας με σωστά. Την ιδιότητα αυτή εκμεταλλευόμαστε παίρνοντας την τάση αυτή για να τροφοδοτήσουμε την κεντρική πλακέτα ελέγχου μέσω του ενός από τους τέσσερις ηλεκτρονικούς ελεγκτές ταχύτητας (Βλέπε Σχήμα 3.3, Περιοχή 4, M\_Signal\_1).

Το μαύρο καλώδιο γείωσης ενώνετε στην επαφή τρία (3) και καταλήγει στη γείωση της κεντρικής πλακέτας ελέγχου, έτσι ώστε η γείωση να είναι ίδια και για τους τέσσερις ηλεκτρονικούς ελεγκτές ταχύτητας και για την κεντρική πλακέτα ελέγχου.

Στην περιοχή τρία (3) βρίσκονται οι επαφές εξόδου των ηλεκτρονικών ελεγκτών ταχύτητας που συνδέονται στις επαφές εισόδου των κινητήρων. Το μικρό διαμορφωμένο κατά πλάτος σήμα που στέλνει ο μικροεπεξεργαστής στην είσοδο των ηλεκτρονικών

ελεγκτών ταχύτητας εμφανίζετε σαν πολύ ισχυρό σήμα, κατάλληλο να κινήσει τους κινητήρες, στις τρεις επαφές εξόδου.

Κάθε ένα από τα τρία καλώδια εξόδου τροφοδοτούν διαδοχικά με διαφορά φάσης 0, 120 και 240 τα τρία τυλίγματα του κινητήρα (Βλέπε Σχήμα 3.18), με τον τρόπο αυτό πετυχαίνουμε πολύ μεγαλύτερη ροπή στους κινητήρες με μικρότερες καταναλώσεις ενέργειας.



Σχήμα 3.18 Τυλίγματα στον στάτη του κινητήρα

### 3.5 ΜΠΑΤΑΡΙΑ

Για να τροφοδοτήσουμε την κατασκευή με κατάλληλη ενέργεια είναι απαραίτητο να τοποθετηθεί κατάλληλη μπαταρία έτσι ώστε να μπορούν να λειτουργήσουν οι τέσσερις κινητήρες και όλα τα υπόλοιπα κυκλώματα για ικανοποιητικό χρονικό διάστημα χωρίς την χρήση σταθερής πηγής ενέργειας.

Έπειτα από έρευνα που πραγματοποιήθηκε καταλήξαμε σε μπαταρίες τεχνολογίας Lipo (ιόντων λιθίου και πολυμερών). Οι μπαταρίες αυτές είναι κατάλληλες για αυτόνομες κατασκευές συνεχούς ρεύματος όπως ελικόπτερα, αεροπλάνα, σκάφη και αυτοκίνητα αλλά και σε άλλες εφαρμογές όπως PDAs και Laptops, διότι μπορούν να αποθηκεύσουν μεγαλύτερες ποσότητες ενέργειας σε σύγκριση με τις κλασικές μπαταρίες ενώ ταυτόχρονα ζυγίζουν 20% λιγότερο. Τα τελευταία χρόνια πραγματοποιούνται και

έρευνες για υβριδικά ηλεκτροκίνητα αυτοκίνητα που θα χρησιμοποιούν μπαταρίες τέτοιου τύπου.

Οι μπαταρίες τεχνολογίας Lipo που συναντάμε στο εμπόριο έχουν μεγάλη ποικιλία και χωρίζονται σε πολλές κατηγορίες έτσι είναι πιο εύκολο να επιλεγεί την κατάλληλη σύμφωνα με τις ανάγκες της κατασκευής. Για τις ανάγκες της παρούσας κατασκευής βασικότερο κριτήριο είναι οι κινητήρες διότι καταναλώνουν την περισσότερη ενέργεια, έτσι σύμφωνα με τα εργοστασιακά χαρακτηριστικά των κινητήρων συμπεραίνουμε ότι η μπαταρία που θα πρέπει να επιλεγεί είναι της κατηγορίας των 11.1V και θα πρέπει να έχει δύναμη 1000mah ανά κινητήρα. στον πίνακα που ακολουθεί μερικές από τις μπαταρίες της κατηγορίας των 11.1V.

Σύμφωνα με τις ανάγκες των κινητήρων καταλήξαμε σε μπαταρία 4.400mah έτσι ώστε να έχει τη δυνατότητα να τροφοδοτήσει τον με 1000mah τον κάθε κινητήρα, που προτείνονται από τον κατασκευαστή, και τα υπόλοιπα 400mah να χρησιμοποιηθούν για τις ανάγκες των ηλεκτρονικών στοιχείων της κατασκευής. Όπως φαίνετε και από τον πίνακα (3.7) μπορεί να χρησιμοποιηθεί μπαταρία μεγαλύτερης ενεργειακής χωρητικότητας που θα δώσει στο σύστημα μεγαλύτερο χρόνο αυτονομίας ωστόσο θα πρέπει να ληφθεί υπόψη ότι μεγαλύτερη όσο μεγαλύτερη μπαταρία χρησιμοποιηθεί τόσο περισσότερο βάρος θα προστεθεί στην κατασκευή, δεδομένου ότι το συνολικό βάρος της κατασκευής δεν ξεπερνά το ένα κιλό και το βάρος της μπαταρίας είναι 317gr.

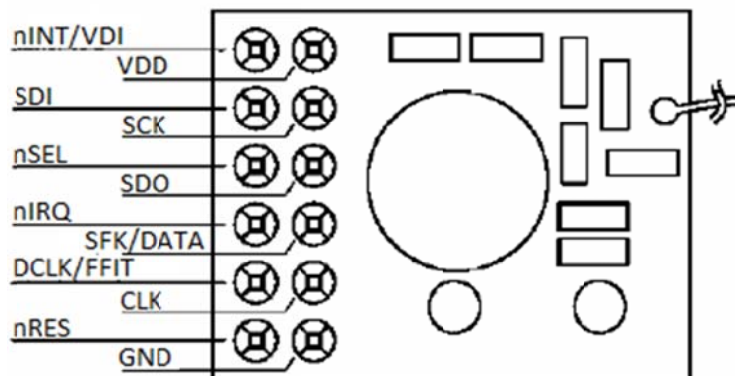
Κατηγορία κελιών	3S1P	3S1P	3S1P	3S1P	3S1P	3S1P
Χωρητικότητα(mah)	360	1550	2450	4000	4400	5800
Αποφόρτιση(c)	20	20	30	20	15	30
Βάρος(g)	32	129	219	304	317	443
Μέγιστος ρυθμός φόρτισης(C)	2	2	2	2	2	2
Μήκος(mm)	42	91	140	145	147	148
Ύψος(mm)	20	34	44	50	50	50
Πλάτος(mm)	23	20	15	19	19	26

Πίνακας 3.7 Κατηγορίες μπαταριών και χαρακτηριστικά

### 3.6 ΑΣΥΡΜΑΤΗ ΕΠΙΚΟΙΝΩΝΙΑ (RF)

Για την ασύρματη επικοινωνία επιλέχθηκε κατάλληλο όργανο ασύρματης επικοινωνίας το οποίο δίνει τη δυνατότητα στο χρήστη να στέλνει κατάλληλες πληροφορίες από τον ηλεκτρονικό υπολογιστή στην κατασκευή καθώς και να λαμβάνει πληροφορίες από την κατασκευή και να τις αποστέλλει στον ηλεκτρονικό υπολογιστή .

Χρησιμοποιήθηκε ο RFM12 διότι είναι συμβατός με τον μικροεπεξεργαστή που χρησιμοποιούμε, έχει χαμηλή κατανάλωση ενέργειας και πολύ γρήγορη και ασφαλή μεταφορά πληροφοριών σε αποστάσεις μέχρι και 200μ. Το πρωτόκολλο επικοινωνίας που χρησιμοποιεί είναι SPI και οι συχνότητες λειτουργίας του είναι 315/433/868/915 MHz. Στο σχήμα που ακολουθεί (3.19) παρουσιάζονται οι ακροδέκτες και η συνδεσμολογία που πρέπει να χρησιμοποιηθεί για την επικοινωνία του RFM12 με μικροεπεξεργαστή ATmega.



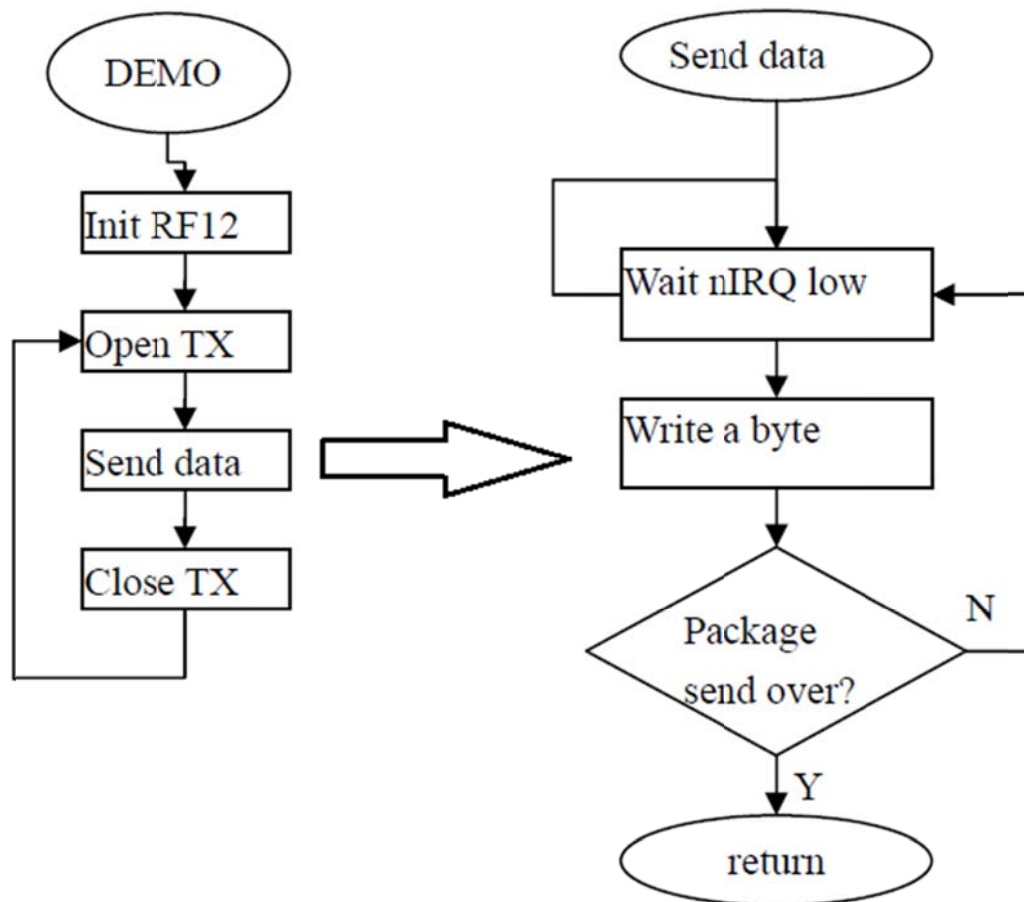
Σχήμα 3.19: Ακροδέκτες RFM12

ATmega	RFM12
SCK	SCK
MISO	SDO
MOSI	SDI
SS	nSEL
INT0	nIRQ

Πίνακας 3.8: Συνδεσμολογία RFM12 με μικροεπεξεργαστή ATmega

Το RF12 έχει τη δυνατότητα να λειτουργεί και σαν αποστολέας δεδομένων αλλά και σαν λήπτης. Για τις ανάγκες της εργασίας χρησιμοποιήθηκαν δύο RFM12. Ο ένας αποστέλλει πληροφορίες για το πώς θα κινηθεί η κατασκευή και λαμβάνει πληροφορίες από τα αισθητήρια της, και είναι συνδεδεμένος με τον υπολογιστή και ένας πάνω στην κατασκευή που λαμβάνει τις πληροφορίες κίνησης και αποστέλλει πληροφορίες από τα αισθητήρια.

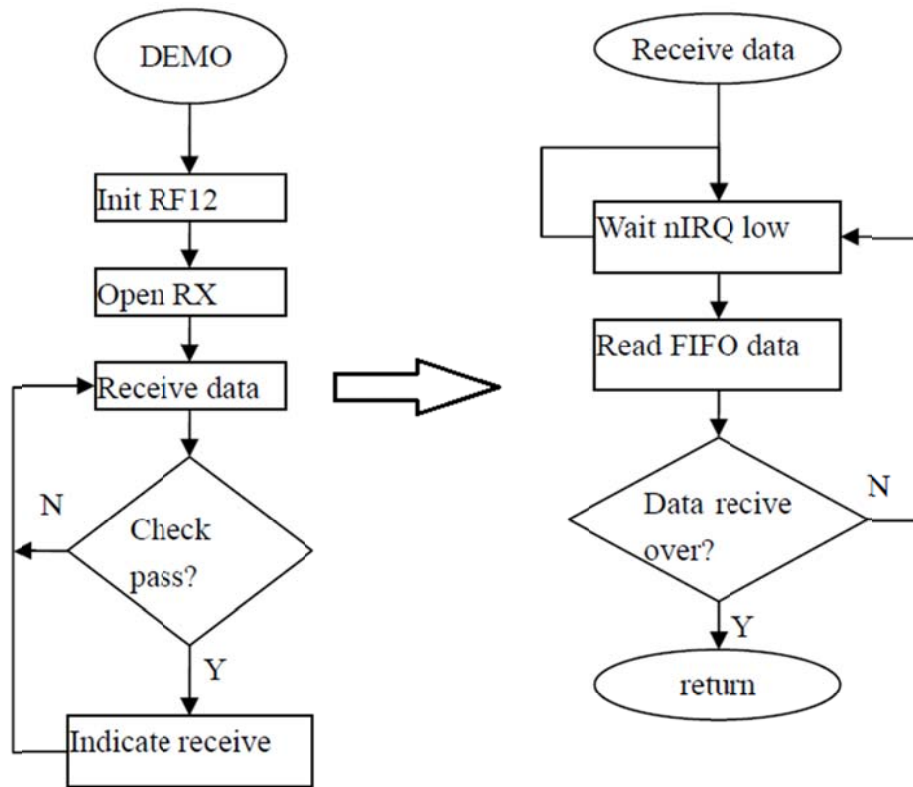
Στα παρακάτω σχήματα (3.20) και (3.21) παρουσιάζονται οι λειτουργίες που πραγματοποιούνται από το RFM12 κατά τη διαδικασία αποστολής και λήψης πληροφορίας με μορφή διαγράμματος ροής.



Σχήμα 3.20: Διάγραμμα ροής πομπού

Όπως φαίνεται από το διάγραμμα ροής πομπού του RFM12, στο πρώτο στάδιο γίνεται αρχικοποίηση και στη συνέχεια ο πομπός ανοίγει. Κατά το δεύτερο στάδιο το RFM12 θα αποστείλει ένα byte πληροφορίας και θα θέσει το nIRQ σε λογικό μηδέν μέχρι να τελειώσει η αποστολή. Τότε μπορεί να δοθεί καινούργιο πακέτο πληροφορίας για αποστολή.





Σχήμα 3.21:

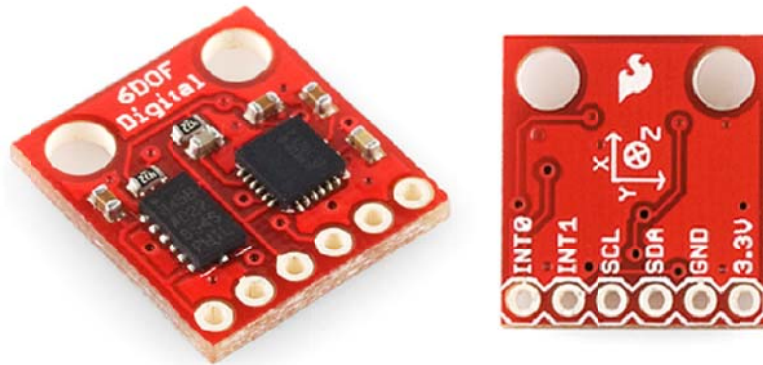
Διάγραμμα ροής δέκτη

Για την διαδικασία λήψης πληροφορίας κατά το πρώτο στάδιο γίνεται αρχικοποίηση του RFM12 στη συνέχεια ενεργοποιείτε η λειτουργία FIFO (first in first out) του δέκτη, η οποία παραμένει ενεργή μέχρι ο ακροδέκτης nIRQ να μεταβεί σε κατάσταση λογικού μηδέν. Όταν το nIRQ βρεθεί σε κατάσταση λογικού μηδέν μόνο τότε μπορεί να διαβαστεί η πληροφορία που έχει λάβει και αποθηκεύσει στη μνήμη τηρώντας το πρωτόκολλο FIFO. Τέλος αφού έχουμε λάβει και διαβάσει την πληροφορία θα πρέπει να γίνει εκκαθάριση της μνήμης έτσι ώστε να μπορέσουμε να λάβουμε και να αποθηκεύσουμε την καινούργια πληροφορία χωρίς λάθη.

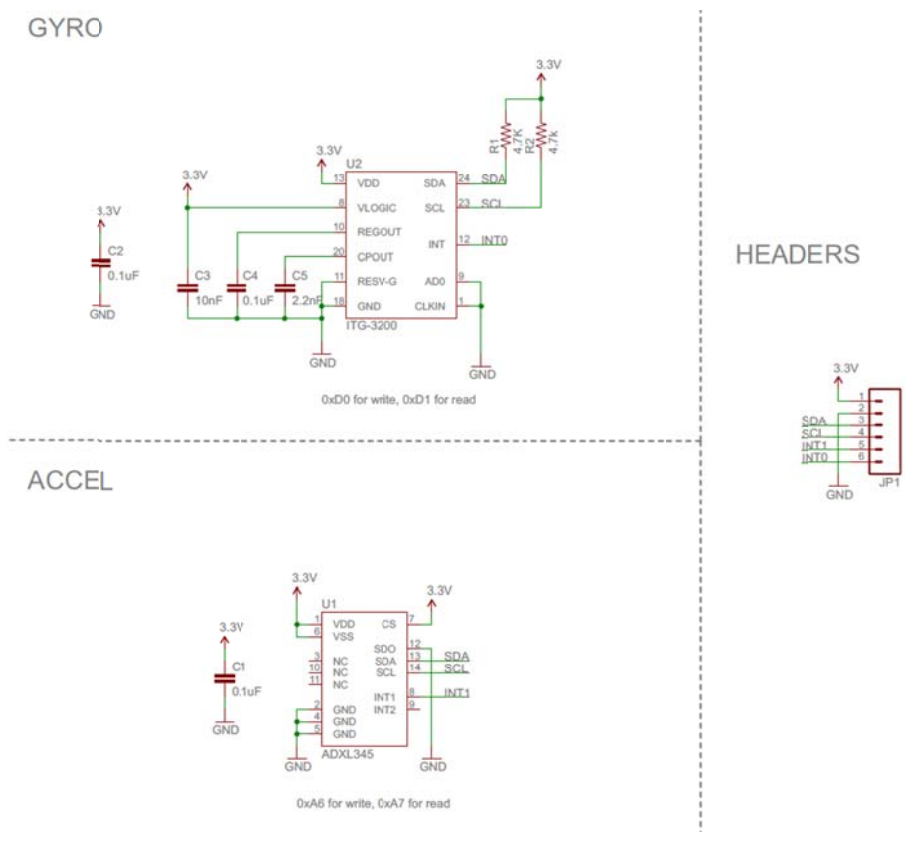
### 3.7 ΑΙΣΘΗΤΗΡΙΑ

Για να επιτύχουμε τον αυτόματο έλεγχο της πορείας της κατασκευής ήταν απαραίτητη η χρήση κατάλληλων αισθητηρίων. Τα αισθητήρια αυτά θα πρέπει να δίνουν δυνατότητα υπολογισμού της κλήσης, έτσι ώστε να δίδονται οι κατάλληλες εντολές στους κινητήρες και να διορθώνεται η πορεία σε περίπτωση που η κατασκευή τείνει να λάβει μεγάλη κλήση και να συντριβεί. Το αισθητήριο που τελικά χρησιμοποιήθηκε ήταν ένας ψηφιακός αισθητήρας έξι βαθμών ελευθερίας. Η πλακέτα αυτή χρησιμοποιεί δύο

αισθητήρες που ο καθένας έχει τρεις βαθμούς ελευθερίας, ένα επιταχυνσιόμετρο τριών αξόνων (ADXL345) και ένα γυροσκόπιο τριών αξόνων (ITG-3200), έτσι με τον συνδυασμό των δύο αυτών αισθητήρων έχουμε τη δυνατότητα να ελέγχουμε την κλίση ανά δευτερόλεπτο καθώς και την επιτάχυνση της κατασκευής στους τρεις άξονες X,Y και Z. Τα αισθητήρια λαμβάνουν τάση εισόδου 3,3V και επικοινωνούν με τον μικροεπεξεργαστή με τη βοήθεια πρωτοκόλλου I<sup>2</sup>c διαμέσου των ακροδεκτών SCL και SDA οι ακροδέκτες INT0 και INT1 δεν χρησιμοποιούνται.



Σχήμα 3.22 : Αισθητήρας έξι βαθμών ελευθερίας



Σχήμα 3.23 : Σχηματικό διάγραμμα αισθητηρίων

## ΚΕΦΑΛΑΙΟ 4

### ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΣ

#### 4.1 ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΣ ΤΟΥ ΜΙΚΡΟΕΠΕΞΕΡΓΑΣΤΗ

Για τον προγραμματισμό του μικροεπεξεργαστή επιλέχτηκε και χρησιμοποιήθηκε λογισμικό Arduino . Το λογισμικό αυτό είναι ελεύθερης πρόσβασης και σχεδιάστηκε στα πλαίσια της έρευνας Arduin of Ivrea που ξεκίνησε το 2005 στην πόλη Ivrea της Ιταλίας με σκοπό το την κατασκευής συσκευής ελέγχου για τις εργασίες πρωτότυπων κατασκευών που σχεδιάζονται από φοιτητές με μειωμένο κόστος. Το λογισμικό αυτό είναι συμβατό με windows και Linux και ο προγραμματισμός γίνεται σε γλώσσα προγραμματισμού C++.

Μια ακόμα ευκολία που παρέχει η χρήση Arduino ως γλώσσα προγραμματισμού είναι η πληθώρα βιβλιοθηκών που είναι στη διάθεση του χρήστη. Οι βιβλιοθήκες αυτές επιλύουν προβλήματα και βοηθούν το χρήστη στον προγραμματισμό καθώς ενημερώνονται συνεχώς και προστίθενται καινούργιες όταν αυτό θεωρηθεί απαραίτητο.

#### 4.2 ΠΡΟΓΡΑΜΜΑΤΙΖΟΝΤΑΣ ΤΟΝ ΜΙΚΡΟΕΠΕΞΕΡΓΑΣΤΗ

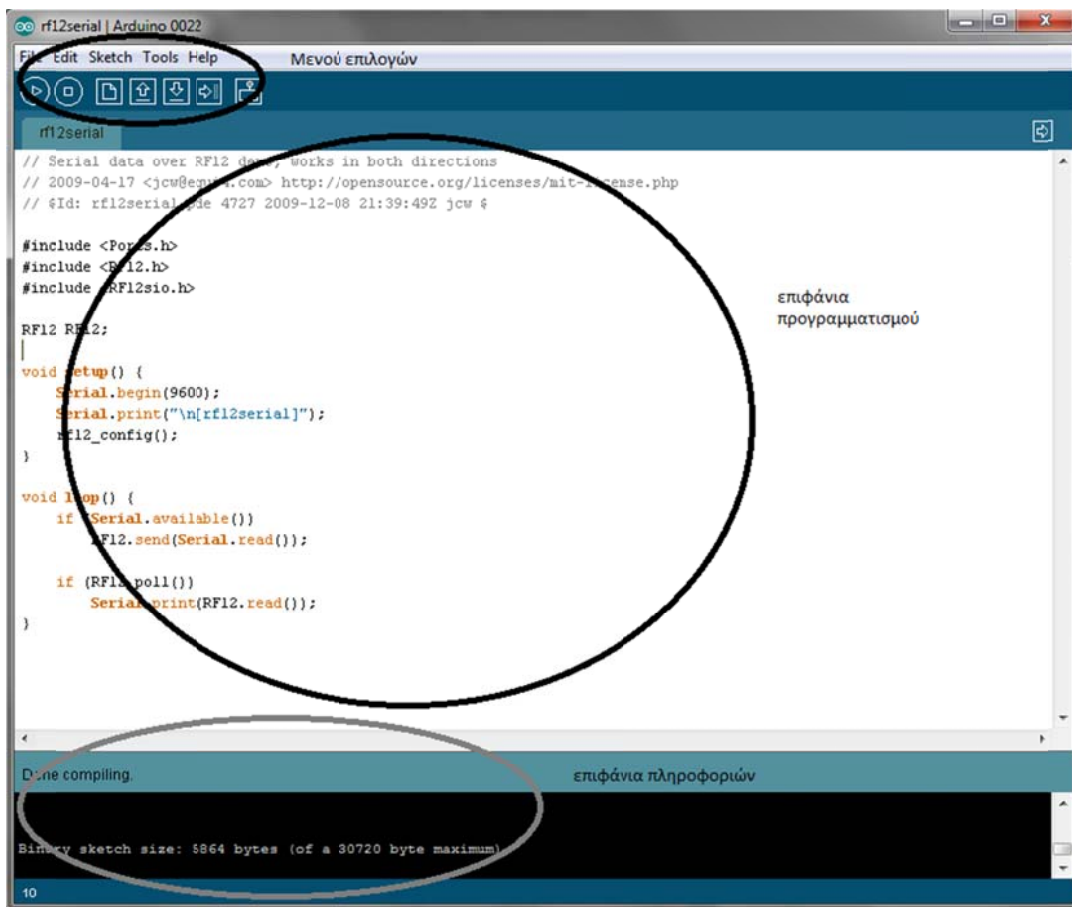
Το περιβάλλον προγραμματισμού είναι φιλικό στον χρήστη και περιλαμβάνει όλες τις βασικές εντολές και κανόνες προγραμματισμού που συναντάμε και στον προγραμματισμό επιπέδου C++.

Κάθε πρόγραμμα που υλοποιείτε με το λογισμικό Arduino αποτελείτε από τέσσερα βασικά μέρη που αναλύονται παρακάτω.

1. Στο πρώτο κομμάτι οι εντολές `#define` και `#include` που ορίζουν τις σταθερές και τις βιβλιοθήκες που θα χρησιμοποιηθούν στο κυρίως πρόγραμμα. Με την εντολή `#define` καταχωρείτε τιμή σε μία σταθερά που θα χρησιμοποιηθεί στο κυρίως πρόγραμμα ενώ η εντολή `#include` καταχωρεί τις βιβλιοθήκες που θα χρησιμοποιηθούν, και οι δύο εντολές πρέπει οριστούν στην αρχή του προγράμματος.
2. Το δεύτερο μέρος αποτελείτε από την ρουτίνα `setup` στην οποία ο χρήστης δηλώνει τις αρχικές τιμές στις εισόδους και τις εξόδους καθώς και μερικές υπορουτίνες που είναι απαραίτητες για την σωστή λειτουργία κάποιων βιβλιοθηκών.

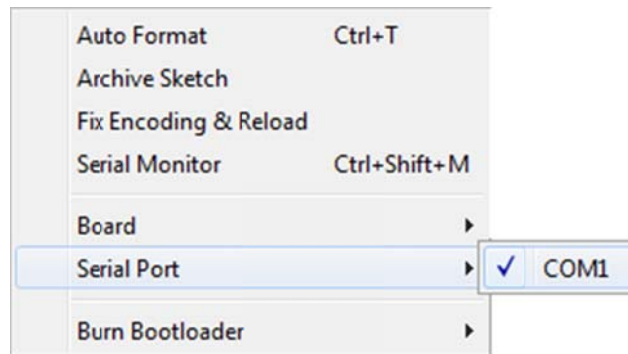
3. Το τρίτο και κυριότερο μέρος αποτελεί η ρουτίνα loop η οποία επαναλαμβάνετε συνεχώς, μέσα στην οποία βρίσκεται το κυρίως πρόγραμμα. Εκεί καλούνται και όλες οι υπορουτίνες που εκτελούν διάφορες λειτουργίες.
4. Το τελευταίο μέρος αποτελούν οι υπορουτίνες οι οποίες καλούνται μέσα από την κυρίως ρουτίνα και επιστρέφουν δεδομένα σε αυτή ή σε άλλες υπορουτίνες.

Το περιβάλλον προγραμματισμού αποτελείται από τρία βασικά μέρη: το μενού επιλογών, την επιφάνια προγραμματισμού, και την επιφάνια πληροφοριών. Οι περιοχές αυτές παρουσιάζονται στο ακόλουθο σχήμα.



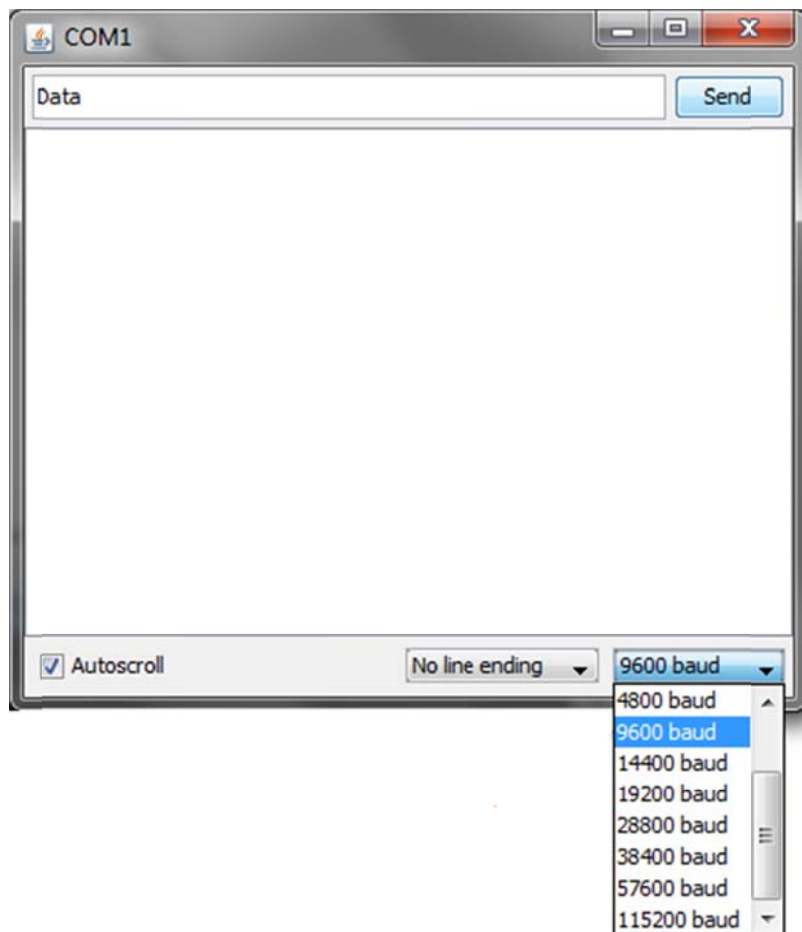
Σχήμα 3.24: Περιβάλλον προγραμματισμού Arduino

Από το μενού επιλογών ο χρήστης έχει τη δυνατότητα χρήσης βασικών επιλογών όπως να ανακτήσει και να αποθηκεύσει τον κώδικα που έχει δημιουργηθεί με την επιλογή "File". Με την επιλογή "Tools / Serial Port" ο χρήστης έχει την δυνατότητα να ενεργοποιήσει την θύρα επικοινωνίας που θα χρησιμοποιηθεί για τη σύνδεση του ηλεκτρονικού υπολογιστή με τον μικροεπεξεργαστή όπως φαίνεται παρακάτω.



Σχήμα 3.25 Επιλογή σειριακής θύρας

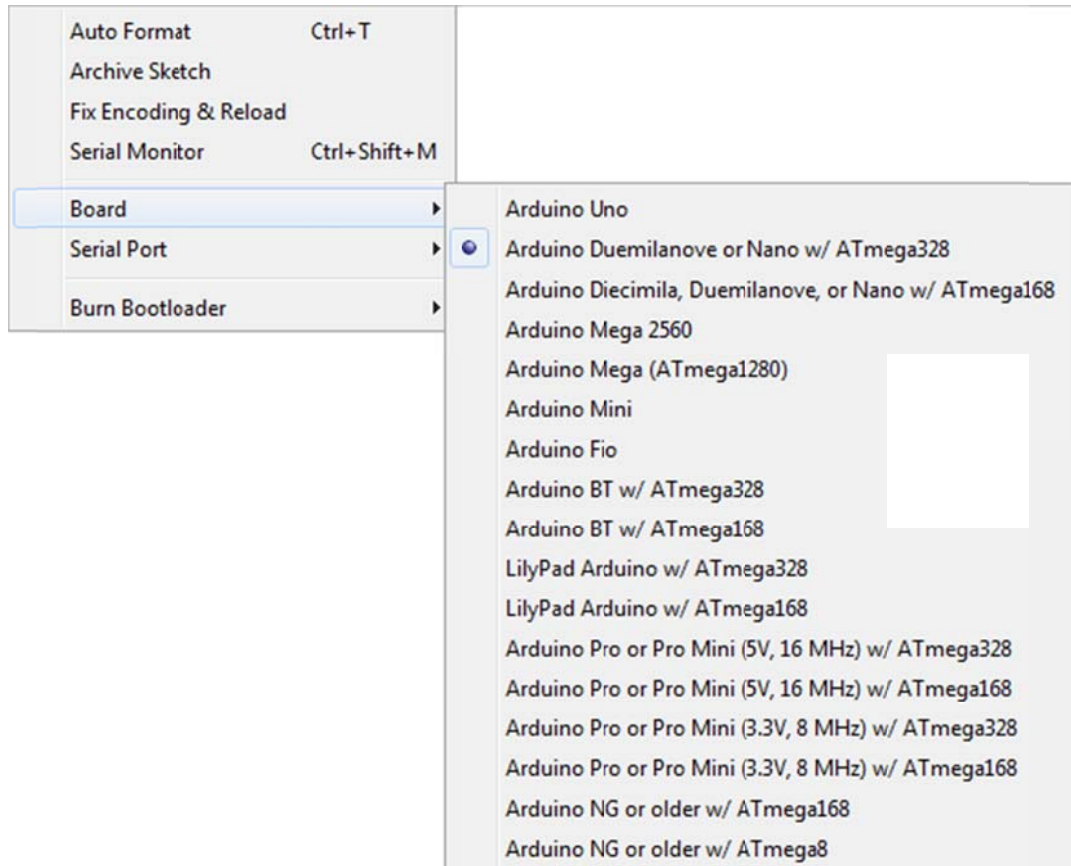
Από το ίδιο μενού μπορεί να ενεργοποιηθεί και το περιβάλλον παρακολούθησης σειριακής θύρας με την επιλογή “Tools / Serial Monitor” που παρουσιάζετε στην εικόνα που ακολουθεί.



Σχήμα 3.26 : Περιβάλλον παρακολούθησης σειριακής θύρας και επιλογής baud rate

Το περιβάλλον παρακολούθησης σειριακής θύρας εμφανίζει τις πληροφορίες που αποστέλλονται από τον μικροεπεξεργαστή στον ηλεκτρονικό υπολογιστή και δίνει την

δυνατότητα στον χρήστη να στείλει πληροφορίες από τον ηλεκτρονικό υπολογιστή στον μικροεπεξεργαστή μέσω της επιλογής “Send” και να επιλέγει την ταχύτητα επικοινωνίας της συριακής θύρας (Baud Rate). Τέλος θα πρέπει να επιλεγεί ο τύπος πλακέτας Arduino που θα χρησιμοποιηθεί από την επιλογή “Tools / Board” όπως φαίνεται στην εικόνα που ακολουθεί.



Σχήμα 3.27 : Επιλογή τύπου πλακέτας Arduino

Μετά την δημιουργία κώδικα και την επιλογή των κατάλληλων παραμέτρων θα πρέπει να πραγματοποιηθεί έλεγχος για συντακτικά λάθη και στη συνέχεια να μεταφερθεί ο κώδικας από τον ηλεκτρονικό υπολογιστή στον μικροεπεξεργαστή. Οι ενέργειες αυτές πραγματοποιούνται από το μενού “Sketch / Verify / Compile” για τον συντακτικό έλεγχο και από το “File / Upload to I/O Board” για την μεταφορά του κώδικα από τον ηλεκτρονικό υπολογιστή στον μικροεπεξεργαστή, αν σε κάποια από τις δύο διαδικασίες παρουσιαστεί σφάλμα εμφανίζεται σχετικό μήνυμα στην επιφάνεια πληροφοριών, το μήνυμα αυτό παρουσιάζει την πιθανή αιτία που προκάλεσε το σφάλμα.

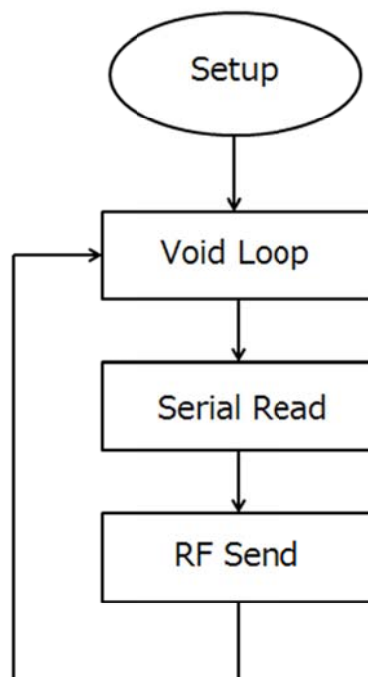
Η δεύτερη μπάρα στο μενού επιλογών αποτελείται από επτά εικονίδια γρήγορων επιλογών που εκτελούν τις βασικές λειτουργίες που αναλύθηκαν παραπάνω.

### 4.3 ΑΝΑΛΥΣΗ ΛΟΓΙΣΜΙΚΟΥ

Για τις ανάγκες πειραματισμού στα πλαίσια της εργασίας αναπτύχθηκε λογισμικό που δίνει τη δυνατότητα στον χρήστη να αλλάζει την ταχύτητα των κινητήρων ασύρματα με σκοπό τον έλεγχο της κίνησης που θα εκτελέσει η κατασκευή.

Για να γίνει η εφικτός ο έλεγχος σχεδιάστηκαν δύο προγράμματα. Το πρώτο πρόγραμμα αποστέλλει δεδομένα από τον ηλεκτρονικό υπολογιστή στην κεντρική πλακέτα ελέγχου μέσω της πλακέτας προγραμματισμού. Το δεύτερο πρόγραμμα λαμβάνει τις πληροφορίες από τον ηλεκτρονικό υπολογιστή και ελέγχει το πλάτος του παλμού που θα δεχτεί κάθε κινητήρας αλλάζοντας την ταχύτητα περιστροφής.

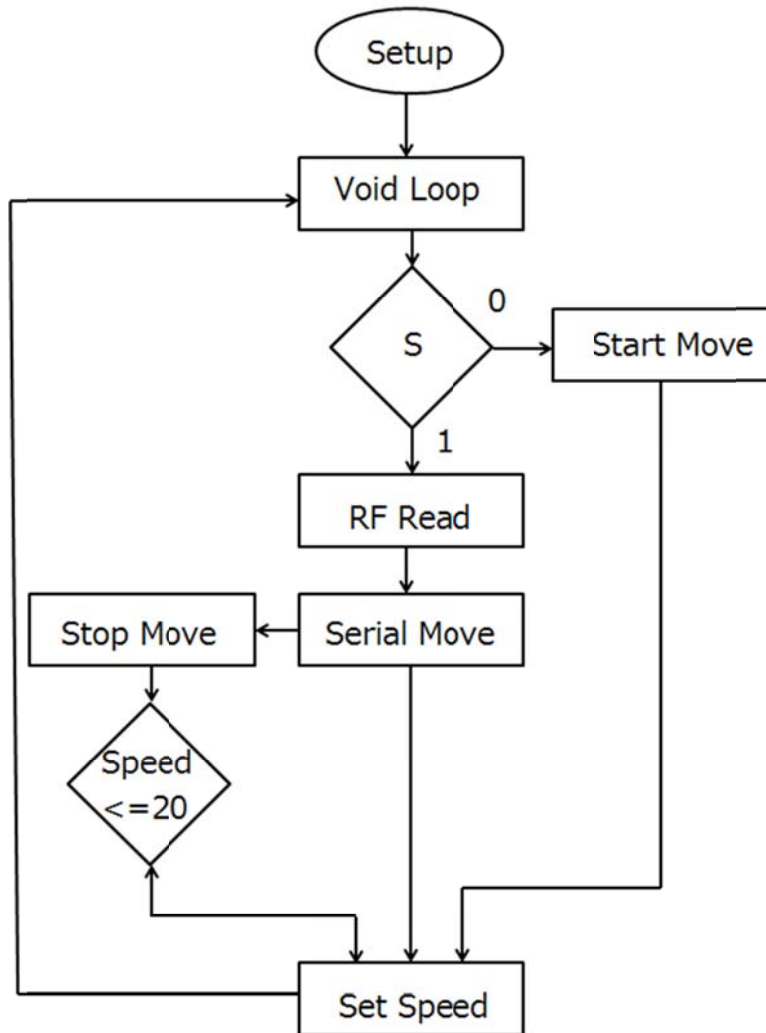
Το πρώτο πρόγραμμα που αποστέλλει πληροφορίες από τον ηλεκτρονικό υπολογιστή στην κεντρική πλακέτα ελέγχου παρουσιάζετε παρακάτω.



Σχήμα 4.28 : Διάγραμμα ροής αποστολής δεδομένων από τον ηλεκτρονικό υπολογιστή

Το δεύτερο πρόγραμμα σχεδιάστηκε για να λαμβάνει πληροφορίες η κεντρική πλακέτα ελέγχου και να πραγματοποιεί ανάλογες μεταβολές στην ταχύτητα κίνησης των τεσσάρων κινητήρων. Με την βοήθεια της μεταβλητής αυξομειώνουμε την ταχύτητα όλων των κινητήρων ταυτόχρονα, ενώ οι μεταβλητές M1,M2,M3 και M4 επηρεάζουν την ταχύτητα του κάθε κινητήρα ξεχωριστά. Το τελικό σήμα που θα λάβει ο κινητήρας αποτελείτε από το άθροισμα της μεταβλητής speed και M1 για τον κινητήρα ένα (1),

speed και M2 για τον κινητήρα δύο(2) ), speed και M3 για τον κινητήρα τρία (3) και speed και M4 για τον κινητήρα τέσσερα (4). Στην περίπτωση που δεν επιθυμάτε αλλαγή της ταχύτητας οι μεταβλητές M1,M2,M3 και M4 μηδενίζονται.



Σχήμα 4.29 : Διάγραμμα ροής κεντρικής πλακέτας ελέγχου



## ΚΕΦΑΛΑΙΟ 5

### ΤΟ ΠΡΟΤΕΙΝΟΜΕΝΟ ΕΝΑΕΡΙΟ ΟΧΗΜΑ

#### 5.1 Η ΤΕΛΙΚΗ ΜΟΡΦΗ ΤΟΥ ΠΡΟΤΕΙΝΟΜΕΝΟΥ ΟΧΗΜΑΤΟΣ

Στα πλαίσια της εργασίας κατασκευάστηκαν όλα τα δομικά τμήματα που παρουσιάστηκαν στα προηγούμενα κεφάλαια. Η κατασκευή τους πραγματοποιήθηκε στο Τμήμα Ηλεκτρονικής του ΤΕΙ Κρήτης.

Στην παρακάτω εικόνα παρουσιάζετε η μία από τις δύο αλουμινένιες ράβδους που αποτελούν το αλουμινένιο πλαίσιο στήριξης. Στην αριστερή πλευρά της εικόνας παρατηρούμε την οπή για την κεντρική βίδα στήριξης και μια περιοχή που έχει αφαιρεθεί με σκοπό την σταθερή και συμμετρική συναρμογή του πλαισίου στήριξης, στην δεξιά πλευρά παρατηρούμε την οπές στήριξης του κινητήρα καθώς και μερικές οπές κατάλληλα τοποθετημένες για την σωστή ψύξη του κινητήρα.



Εικόνα 5.30 : Ράβδος στήριξης

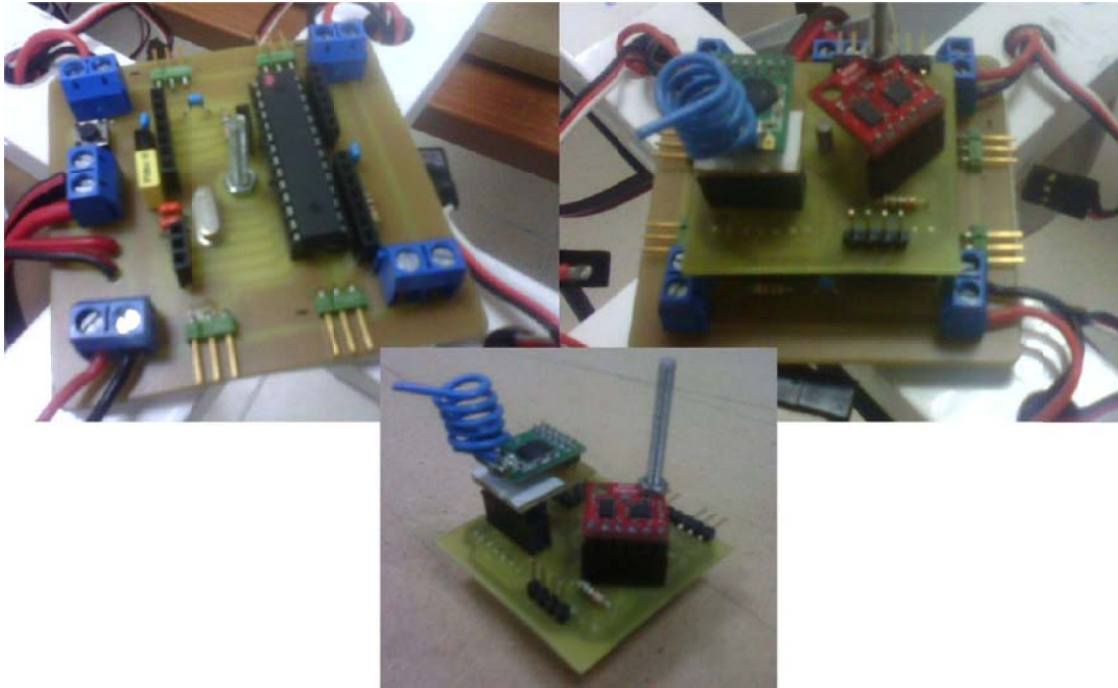
Αφού κατασκευάστηκε το πλαίσιο της κατασκευής τοποθετήθηκαν οι τέσσερις κινητήρες, συνοδευόμενοι από τους αντίστοιχους ελεγκτές ταχύτητας. Τέλος τοποθετήθηκε και η κεντρική πλακέτα ελέγχου. Η τελική μορφή της κατασκευής παρουσιάζεται στις εικόνα 5.31.



*Εικόνα 5.31 : Πλαίσιο στήριξης*

Όλα τα καλώδια οδηγούνται στο εσωτερικό των αλουμινένιων ράβδων και διαμέσου κατάλληλων οπών οδηγούνται στην κεντρική πλακέτα ελέγχου, ενώ οι ηλεκτρονικοί ελεγκτές ταχύτητας καλύπτονται με εύκαμπτο υλικό ώστε να προστατεύονται.

Στις εικόνες που ακολουθούν παρουσιάζονται η κεντρική πλακέτα ελέγχου (βλέπε σχήμα 5.32 αριστερή εικόνα) και η πλακέτα αισθητηρίων (βλέπε σχήμα 5.32 κεντρική εικόνα) οι πλακέτες αυτές εφαρμόζουν κατάλληλα όπως φαίνεται στην δεξιά εικόνα του σχήματος 5.32.



*Σχήμα 5.32 : Κεντρική πλακέτα ελέγχου και πλακέτα αισθητηρίων*

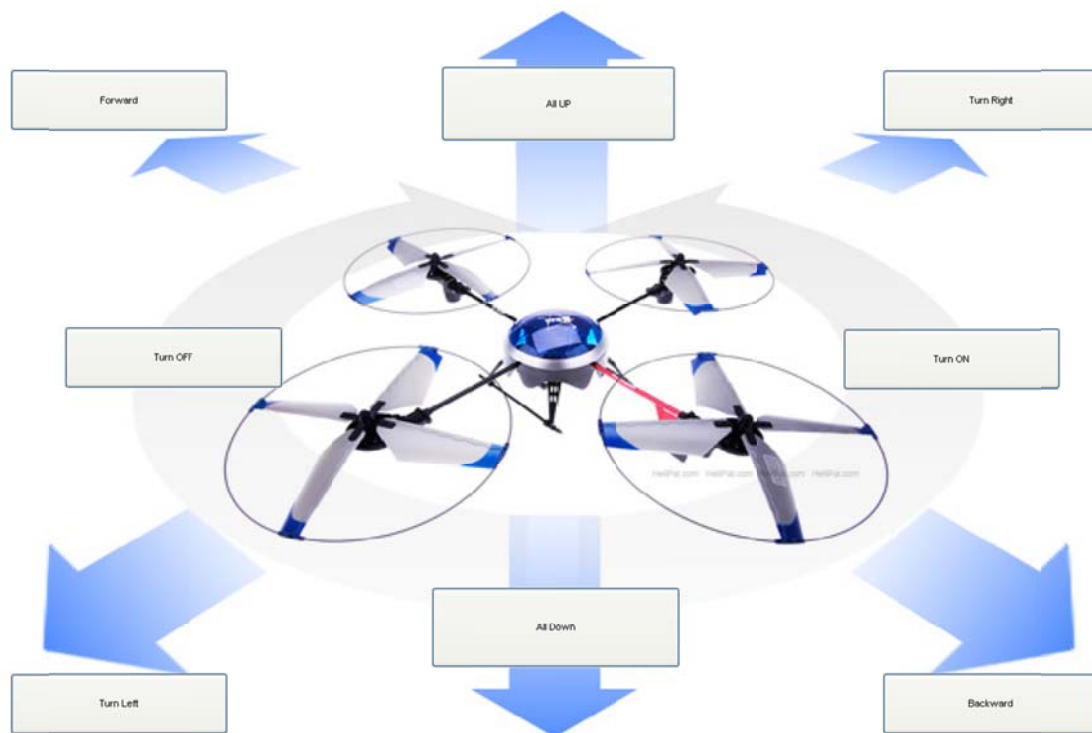
Για την προστασία των δομικών μερών και συγκεκριμένα της κεντρικής πλακέτας ελέγχου και των αισθητηρίων κατασκευάστηκε προστατευτικό καπάκι. Η τελική μορφή της κατασκευής παρουσιάζεται στην εικόνα 5.33.



*Σχήμα 5.33: Τελική μορφή κατασκευής*

## 5.2 ΓΡΑΦΙΚΟ ΠΕΡΙΒΑΛΛΟΝ ΔΕΙΠΑΦΗΣ

Για τον έλεγχο της κατασκευής δημιουργήθηκε κατάλληλο γραφικό περιβάλλον με τη βοήθεια του λογισμικού Matlab που επιτρέπει στον χρήστη να επικοινωνεί ασύρματα με την κεντρική πλακέτα ελέγχου μέσω της σειριακής θύρας. Το γραφικό περιβάλλον επιτρέπει στο χρήστη να εκτελεί όλες τις λειτουργίες επιλέγοντας κάθε φορά το κατάλληλο εικονίδιο χρησιμοποιώντας το ποντίκι του ηλεκτρονικού υπολογιστή. Στην εικόνα που ακολουθεί παρουσιάζετε το γραφικό περιβάλλον που δημιουργήθηκε καθώς και όλες οι επιτρεπόμενες επιλογές που είναι διαθέσιμες στον χρήστη.



Σχήμα 5.34 : Γραφικό περιβάλλον Matlab

Το γραφικό περιβάλλον που δημιουργήθηκε ενεργοποιεί την σειριακή επικοινωνία, με ανάλογο τρόπο με αυτόν που περιγράφηκε στην παράγραφο 4.2, και δίνει τη δυνατότητα στο χρήστη να αποστέλλει την κατάλληλη πληροφορία μέσω της σειριακής θύρας στην πλακέτα προγραμματισμού, χρησιμοποιώντας τον ασύρματο πομποδέκτη RFM12.

## 5.3 ΕΝΔΕΙΚΤΙΚΟ ΣΧΗΜΑ ΕΛΕΓΧΟΥ

Για τον έλεγχο της κίνησης της προτεινόμενης κατασκευής είναι απαραίτητη η χρήση κατάλληλων αισθητήρων. Οι βασικότεροι αισθητήρες είναι το γυροσκόπιο και ο αισθητήρας επιτάχυνσης. Το γυροσκόπιο έχει τη δυνατότητα να μετρά την κλίση ως προς

κάποιο άξονα, ενώ ο αισθητήρας επιτάχυνσης έχει τη δυνατότητα μέτρησης την επιτάχυνσης ως προς έναν άξονα. Με την χρήση τριών γυροσκοπίων και ενός επιταχυνσιογράφου έχουμε την δυνατότητα να βρίσκουμε την κλίση του ελικοπτέρου και να διορθώνουμε τα σφάλματα πορείας αλλάζοντας κατάλληλα την ροπή που παράγουν οι κινητήρες.

Η κεντρική πλακέτα ελέγχου λαμβάνει τα δεδομένα από την πλακέτα αισθητηρίων και στη συνέχεια στέλνει τα δεδομένα αυτά στο κύκλωμα ασύρματης επικοινωνίας, τα δεδομένα στέλνονται στον δέκτη ο οποίος είναι συνδεδεμένος με σειριακή θύρα ηλεκτρονικό υπολογιστή και με τη χρήση κατάλληλου λογισμικού υπολογίζονται οι παράμετροι ελέγχου των κινητήρων. Οι παράμετροι των κινητήρων αποστέλλονται ξανά από τον ηλεκτρονικό υπολογιστή στον μικροεπεξεργαστή και κατ' επέκταση στους ηλεκτρονικούς ελεγκτές ταχύτητας του κάθε κινητήρα, με αποτέλεσμα τον αυτόματο έλεγχο της πορείας του ελικοπτέρου τεσσάρων κινητήρων.

## ΚΕΦΑΛΑΙΟ 6

# ΣΥΜΠΕΡΑΣΜΑΤΑ – ΜΕΛΛΟΝΤΙΚΕΣ ΕΠΕΚΤΑΣΕΙΣ

### 6.1 ΣΥΜΠΕΡΑΣΜΑΤΑ

Στην παρούσα εργασία, αναπτύχθηκε μια πρωτότυπη εναέρια ρομποτική πλατφόρμα, ικανή να χρησιμοποιηθεί τόσο για την κατανόηση βασικών θεμάτων που σχετίζονται με την ρομποτική, τον έλεγχο, τις τηλεπικοινωνίες και την μηχανική, όσο και για να αποτελέσει μια πλατφόρμα πειραματισμού για την διεξαγωγή βασικών ερευνητικών πειραμάτων. Στα πλαίσια της εργασίας πραγματοποιήθηκε ο σχεδιασμός της τόσο σε επίπεδο υλικού (hardware), όσο και σε επίπεδο λογισμικού (software). Στη συνέχεια πραγματοποιήθηκε η ολοκλήρωση της προτεινόμενης κατασκευής και δοκιμάστηκε η λειτουργικότητα της.

Η συγκεκριμένη υλοποίηση έγινε λαμβάνοντας πάντα υπόψη τη σχέση κόστους – απόδοσης όλων των τεχνολογικών λύσεων που υιοθετήθηκαν. Ο συγκεκριμένος σχεδιασμός βασίζομενος σε ανοικτό λογισμικό δίνει αρκετές δυνατότητες για περαιτέρω ανάπτυξη. Εκτός όμως από τις επιλογές που έγιναν στο επίπεδο του λογισμικού, επιλέχθηκαν και εύκολα επεκτάσιμες λύσεις και όσον αφορά το υλικό. Ο σχεδιασμός των ηλεκτρονικών μερών έγινε έτσι ώστε να διασφαλίζεται η επεκτασιμότητα και η εύκολη διασυνδεσιμότητα και με άλλα αισθητήρια και συσκευές.

Η κατασκευή δοκιμάστηκε προσαρμοσμένη σε κατάλληλη βάση, και αποδείχθηκε –μετά από εκτενή πειραματισμό – ότι έχει τη δυνατότητα να πραγματοποιήσει με κατάλληλες τροποποιήσεις και ενδεχομένως και με την συμπληρωματική τοποθέτηση αισθητήρων, αυτόνομη ή ημιαυτόνομη πτήση.

### 6.2 ΜΕΛΛΟΝΤΙΚΕΣ ΕΠΕΚΤΑΣΕΙΣ

Η παρούσα εργασία αποτελεί μια βάση πάνω στην οποία υπάρχουν πολλές δυνατότητες επέκτασης τόσο σε θέματα υλικού και λογισμικού, όσο και σε θέματα που σχετίζονται με την ανάπτυξη πρωτότυπων μεθοδολογιών ελέγχου, προκειμένου να μπορέσει να λειτουργήσει αυτόνομα.

---

Συγκεκριμένα με βάση τη συγκεκριμένη κατασκευή αναμένεται στο μέλλον να βελτιστοποιηθεί ο σχεδιασμός, και να δοκιμαστούν και άλλες λύσεις που θα αυξήσουν την υπολογιστική ισχύ της κεντρικής μονάδας. Εκτός αυτού αναμένεται να δοκιμαστούν και εναλλακτικοί τύποι αισθητήρων που θα επιτρέψουν στο όχημα να έχει αυξημένη αντίληψη του περιβάλλοντος χώρου και να χρησιμοποιήσει τη συγκεκριμένη πληροφορία σαν είσοδο σε ένα εξελιγμένο σύστημα ελέγχου, συγκεκριμένα αναμένεται να τοποθετηθεί οπτικός αισθητήρας. Προκειμένου να αυξηθούν και οι δυνατότητες που σχετίζονται με την ανίχνευση θέσης, αναμένεται να τοποθετηθεί και ένας αισθητήρας GPS, που θα δώσει στο συγκεκριμένο όχημα, αυξημένες δυνατότητες αντίληψης του περιβάλλοντος χώρου.

Όσον αφορά την ανάπτυξη συστημάτων ελέγχου αναμένεται να μελετηθεί η ανάπτυξη ελεγκτών βασισμένων σε τεχνικές υπολογιστικής νοημοσύνης (νευρωνικά δίκτυα και ασαφή λογική), καθώς και σε συμβατικές τεχνικές ελέγχου.

**ΒΙΒΛΙΟΓΡΑΦΙΑ**

- [1] Ν. Ι. Βιτζιλαίος, “Σχεδίαση, μελέτη και κατασκευή συστήματος αυτόνομης πλοήγησης μη επανδρωμένου εναέριου οχήματος κάθετης απογείωσης-προσγείωσης,” Τμήμα Μηχανικών Παραγωγής και Διοίκησης, Μεταπτυχιακό Δίπλωμα Ειδίκευσης, Πολυτεχνείο Κρήτης, 2005.
- [2] Π. Σπανουδάκης, “Σχεδίαση νέου μη – επανδρωμένου οχήματος καθέτου απογείωσης – προσγείωσης,” Τμήμα Μηχανικών Παραγωγής και Διοίκησης, Μεταπτυχιακό Δίπλωμα Ειδίκευσης, Πολυτεχνείο Κρήτης, 2003.
- [3] P. Spanoudakis, L. Doitsidis, N. C. Tsourveloudis, and K. P. Valavanis, "The Market for VTOL UAVs," *Unmanned Systems Magazine*, vol. 21, pp. 14-18, 2003.
- [4] Ν. Ι. Βιτζιλαίος, “Ευστάθεια πτήσης μη επανδρωμένων ελικοπτέρων: τεχνικές ελέγχου & επιλογή βέλτιστων χαρακτηριστικών,” Διδακτορικό Δίπλωμα, Πολυτεχνείο Κρήτης, 2010.
- [5] [http://www.barnardmicrosystems.com/L4E\\_quadrotor.htm](http://www.barnardmicrosystems.com/L4E_quadrotor.htm)
- [6] [www.sparkfun.com](http://www.sparkfun.com)
- [7] <http://www.giantcod.co.uk/>
- [8] <http://www.hobbyking.com/hobbyking/store/index.rc>
- [9] [www.wikipedia.org](http://www.wikipedia.org)



**ΠΑΡΑΡΤΗΜΑ****Πρόγραμμα κεντρικής πλακέτας ελέγχου :**

```
#include <Servo.h>
#include <Ports.h>
#include <RF12.h>
#include <RF12sio.h>

RF12 RF12;
int speed =10;
int M1 =0;
int M2 =0;
int M3 =0;
int M4 =0;
int angle;
char move;
Servo myservo1;
Servo myservo2;
Servo myservo3;
Servo myservo4;
////////////////////////////////////
void setup()
{
  myservo1.attach(3);
  myservo2.attach(4);
  myservo3.attach(8);
  myservo4.attach(9);
  speed=10;
  arm();
  Serial.begin(9600);
```

```
rf12_config();
speed,M1,M2,M3,M4=start(move);
}
////////////////////////////////////
void loop()
{

  if (RF12.poll())
  {
    move = RF12.read();
    Serial.print("\n move \n");
    speed,M1,M2,M3,M4 =serial(move);
    if (speed <30) speed = 30;
    if (speed >50) speed = 50;
    setSpeed(speed,M1,M2,M3,M4);
  }

}
////////////////////////////////////
void arm()
{
  setSpeed(speed,M1,M2,M3,M4);
  delay(3000);
}
////////////////////////////////////
int start(char move)
{
  Serial.print("\n\n ==START=- \n\n");
  Serial.print(" _____\n");
  Serial.print("|press q to start |\n");
  Serial.print("|press a for up |\n");
```

```
Serial.print("|press z for down  |\n");
Serial.print("|press w to end  |\n");
Serial.print("|_____|\n");
Serial.print("P.S. total fail menu\n\n\n");
while (move!='q')
  if (RF12.poll())
    move = RF12.read();
Serial.print(" -=CAUTION TURNING MOTORS ON=- \n\n");
while (speed<=25){
  speed+=2;
  setSpeed(speed,M1,M2,M3,M4);
  delay(1000);
}
return(speed,M1,M2,M3,M4);
}
////////////////////////////////////
void setSpeed(int speed,int M1,int M2,int M3,int M4)
{
  int angle = map(speed, 0, 100, 0, 180);
  int angle1 = map((angle+M1), 0, 100, 0, 180);
  int angle2 = map((angle+M2), 0, 100, 0, 180);
  int angle3 = map((angle+M3), 0, 100, 0, 180);
  int angle4 = map((angle+M4), 0, 100, 0, 180);
  Serial.print(speed,DEC );
  Serial.print('\n');
  myservo1.write(angle1);
  myservo2.write(angle2);
  myservo3.write(angle3);
  myservo4.write(angle4);
  return;
}
```

```
////////////////////////////////////
```

```
int serial(char move)
{
switch (move)
{
case'a':
    {
    Serial.print("up\n");
    speed+=2;
    M1=M2=M3=M4=0;
    break;
    }
case'z':
{
    Serial.print("down\n");
    speed-=2;
    M1=M2=M3=M4=0;
break;}
case'w':
{
    Serial.print("\n\n ==CAUTION TURNING MOTORS OFF=- \n\n");
    while (speed>=20)
    {
    speed-=2;
    setSpeed(speed,M1=0,M2=0,M3=0,M4=0);
    delay(300);
    }
    speed,M1,M2,M3,M4=start(move);
break;
}
case's':
```

```
{
  Serial.print("Left \n");
  M1=5; M2=-5; M3=M4=0;
  delay(500);
  break;
}
case'd':
{
  Serial.print("Backword \n");
  M3=5; M4=-5; M1=M2=0;
  delay(500);
  break;
}
case'x':
{
  Serial.print("Right \n");
  M1=-5; M2=5; M3=M4=0;
  delay(500);
  break;
}
case'c':
{
  Serial.print("Forword \n");
  M3=-5; M4=5; M1=M2=0;
  delay(500);
  break;
}
default:
  M1=M2=M3=M4=0;
}
return(speed,M1,M2,M3,M4);
```

```
}
```

**Πρόγραμμα πλακέτας προγραμματισμού για ασύρματη επικοινωνία :**

```
// Serial data over RF12 demo, works in both directions  
// 2009-04-17 <jcw@equi4.com> http://opensource.org/licenses/mit-license.php  
// $Id: rf12serial.pde 4727 2009-12-08 21:39:49Z jcw $
```

```
#include <Ports.h>  
#include <RF12.h>  
#include <RF12sio.h>  
RF12 RF12;  
void setup()  
{  
  Serial.begin(9600);  
  Serial.print("\n[rf12serial]");  
  rf12_config();  
}  
void loop()  
{  
  if (Serial.available())  
    RF12.send(Serial.read());  
  
  if (RF12.poll())  
    Serial.print(RF12.read());  
}
```

**Πρόγραμμα συλλογής δεδομένων από επιταχυνσιόμετρο :**

```
#include <Wire.h>
#define DEVICE (0x53) //ADXL345 device address
#define TO_READ (6) //num of bytes we are going to read each time (two bytes for
each axis)

byte buff[TO_READ] ; //6 bytes buffer for saving data read from the device
char str[512]; //string buffer to transform data before sending it to the serial
port

void setup()
{
Wire.begin(); // join i2c bus (address optional for master)
Serial.begin(9600); // start serial for output

//Turning on the ADXL345
writeTo(DEVICE, 0x2D, 0);
writeTo(DEVICE, 0x2D, 16);
writeTo(DEVICE, 0x2D, 8);
}

void loop()
{
int regAddress = 0x32; //first axis-acceleration-data register on the ADXL345
int x, y, z;

readFrom(DEVICE, regAddress, TO_READ, buff); //read the acceleration data from the
ADXL345
```

```
//each axis reading comes in 10 bit resolution, ie 2 bytes. Least Significant Byte first!!
//thus we are converting both bytes in to one int
x = (((int)buff[1]) << 8) | buff[0];
y = (((int)buff[3])<< 8) | buff[2];
z = (((int)buff[5]) << 8) | buff[4];

x-=41;
y+=4;
z-=239;

//we send the x y z values as a string to the serial port
sprintf(str, "%d %d %d", x, y, z);
Serial.print(str);
Serial.print(10, BYTE);

//It appears that delay is needed in order not to clog the port
delay(1000);
}

//----- Functions
//Writes val to address register on device
void writeTo(int device, byte address, byte val)
{
  Wire.beginTransmission(device); //start transmission to device
  Wire.send(address); // send register address
  Wire.send(val); // send value to write
  Wire.endTransmission(); //end transmission
}

//reads num bytes starting from address register on device in to buff array
```



```

void readFrom(int device, byte address, int num, byte buff[]) {
  Wire.beginTransmission(device); //start transmission to device
  Wire.send(address); //sends address to read from
  Wire.endTransmission(); //end transmission

  Wire.beginTransmission(device); //start transmission to device
  Wire.requestFrom(device, num); // request 6 bytes from device

  int i = 0;
  while(Wire.available()) //device may send less than requested (abnormal)
  {
    buff[i] = Wire.receive(); // receive a byte
    i++;
  }
  Wire.endTransmission(); //end transmission
}

```

**Πρόγραμμα συλλογής δεδομένων από γυροσκόπιο :**

```

#include <Wire.h> // I2C library, gyroscope
#define GYRO_ADDR 0x68 // ego
// #define GYRO_ADDR 0x69 // gyro address, binary = 11101001 when AD0 is
// connected to Vcc (see schematics of your breakout board)
#define SMPLRT_DIV 0x15
#define DLPF_FS 0x16
#define INT_CFG 0x17
#define PWR_MGM 0x3E
#define TO_READ 8 // 2 bytes for each axis x, y, z
// offsets are chip specific.
/*
int offx = 120;

```

```
int offy = 20;
int offz = 93;
*/
int offx = -23;
int offy = -28;
int offz = -5;

//initializes the gyroscope
void initGyro()
{
  /******
  * ITG 3200
  * power management set to:
  * clock select = internal oscillator
  * no reset, no sleep mode
  * no standby mode
  * sample rate to = 125Hz
  * parameter to +/- 2000 degrees/sec
  * low pass filter = 5Hz
  * no interrupt
  *****/

  writeTo(GYRO_ADDR, PWR_MGM, 0x00);
  writeTo(GYRO_ADDR, SMPLRT_DIV, 0x07); // EB, 50, 80, 7F, DE, 23, 20, FF
  writeTo(GYRO_ADDR, DLPF_FS, 0x1E); // +/- 2000 dgrs/sec, 1KHz, 1E, 19
  writeTo(GYRO_ADDR, INT_CFG, 0x00);

}
void getGyroscopeData()
{
```

```

/*****
Gyro ITG-3200 I2C
registers:
temp MSB = 1B, temp LSB = 1C
x axis MSB = 1D, x axis LSB = 1E
y axis MSB = 1F, y axis LSB = 20
z axis MSB = 21, z axis LSB = 22
*****/

int regAddress = 0x1B;
int temp, x, y, z;
byte buff[TO_READ];
char str[50]; // 50 should be enough to store all the data from the gyro
readFrom(GYRO_ADDR, regAddress, TO_READ, buff); //read the gyro data from the
ITG3200

temp = (buff[0] << 8) | buff[1];
x = ((buff[2] << 8) | buff[3]) + offx;
y = ((buff[4] << 8) | buff[5]) + offy;
z = ((buff[6] << 8) | buff[7]) + offz;
//we send the x y z values as a string to the serial port
sprintf(str, "%d,%d,%d,%d", temp, x, y, z);
Serial.print(str);
Serial.print(10, BYTE);

}
void setup()
{
  Serial.begin(9600);
  Wire.begin();
  initGyro();
}
void loop()

```

```
{
  getGyroscopeData();
  delay(1000);
}
//----- Functions
//Writes val to address register on device
void writeTo(int device, byte address, byte val)
{
  Wire.beginTransmission(device); //start transmission to device
  Wire.send(address);    // send register address
  Wire.send(val);    // send value to write
  Wire.endTransmission(); //end transmission
}

//reads num bytes starting from address register on device in to buff array
void readFrom(int device, byte address, int num, byte buff[]) {
  Wire.beginTransmission(device); //start transmission to device
  Wire.send(address);    //sends address to read from
  Wire.endTransmission(); //end transmission

  Wire.beginTransmission(device); //start transmission to device
  Wire.requestFrom(device, num); // request 6 bytes from device

  int i = 0;
  while(Wire.available() //device may send less than requested (abnormal)
  {
    buff[i] = Wire.receive(); // receive a byte
    i++;
  }
  Wire.endTransmission(); //end transmission
}
```