

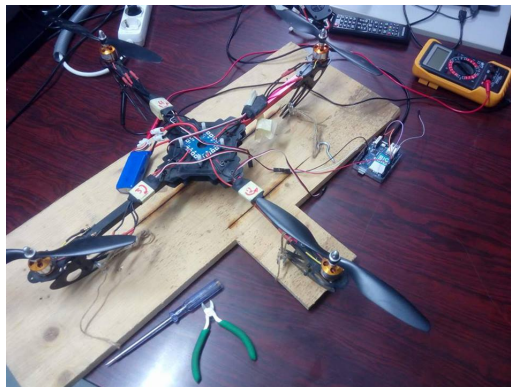


ΧΑΡΟΚΟΠΕΙΟ ΠΑΝΕΠΙΣΤΗΜΙΟ

ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΤΗΛΕΜΑΤΙΚΗΣ

Ανάπτυξη Συστήματος Τηλεμετρίας Τετρακοπτέρου με Χρήση Arduino

Πτυχιακή εργασία
Κοκκίνης-Ντρένης Νικόλαος



Αθήνα, 2018

ΧΑΡΟΚΟΠΕΙΟ ΠΑΝΕΠΙΣΤΗΜΙΟ

ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΤΗΛΕΜΑΤΙΚΗΣ

Τριμελής Εξεταστική Επιτροπή

Επιβλέπων

Βαρλάμης Ηρακλής

Επίκουρος Καθηγητής, Τμήμα Πληροφορικής και Τηλεματικής, Χαροκόπειο Πανεπιστήμιο

Μέλη

Δημητρακόπουλος Γεώργιος

Επίκουρος Καθηγητής, Τμήμα Πληροφορικής και Τηλεματικής, Χαροκόπειο Πανεπιστήμιο

Τσερπές Κωνσταντίνος

Επίκουρος Καθηγητής, Τμήμα Πληροφορικής και Τηλεματικής, Χαροκόπειο Πανεπιστήμιο

Ο Κοκκίνης-Ντρένης Νικόλαος δηλώνω υπεύθυνα ότι:

1. Είμαι ο κάτοχος των πνευματικών δικαιωμάτων της πρωτότυπης αυτής εργασίας και από όσο γνωρίζω η εργασία μου δε συκοφαντεί πρόσωπα, ούτε προσβάλει τα πνευματικά δικαιώματα τρίτων.
2. Αποδέχομαι ότι η ΒΚΠ μπορεί, χωρίς να αλλάξει το περιεχόμενο της εργασίας μου, να τη διαθέσει σε ηλεκτρονική μορφή μέσα από τη ψηφιακή Βιβλιοθήκη της, να την αντιγράψει σε οποιοδήποτε μέσο ή/και σε οποιοδήποτε μορφότυπο καθώς και να κρατά περισσότερα από ένα αντίγραφα για λόγους συντήρησης και ασφάλειας.

Ευχαριστίες

Πριν την παρουσίαση των αποτελεσμάτων της παρούσας πτυχιακής εργασίας, αισθάνομαι την υποχρέωση να ευχαριστήσω ορισμένους από τους ανθρώπους που γνώρισα, συνεργάστηκα μαζί τους και έπαιξαν πολύ σημαντικό ρόλο στην πραγματοποίησή της. Πρώτο από όλους θέλω να ευχαριστήσω τον επιβλέποντα καθηγητή της διπλωματικής εργασίας, Καθηγητή Ηρακλή Βαρλάμη για την πολύτιμη καθοδήγηση του και την εμπιστοσύνη και εκτίμηση που μου έδειξε. Στη συνέχεια θα ήθελα να ευχαριστήσω τον Καθηγητή Ανδρεάτο Αντώνιο και τον Λιανό Δημήτριο οι οποίοι με τα πλούσια πνευματικά προσόντα και το ήθος τους συνέβαλαν ουσιαστικά στην ολοκλήρωση αυτής της εργασίας.

Περιεχόμενα

1	Εισαγωγή	9
1.1	Τι είναι το Arduino	9
1.2	Μια σύντομη ιστορία του Arduino	10
1.3	Τι είναι Physical Computing	10
1.4	Τι είναι ο μικροελεγκτής	11
1.5	Η πλατφόρμα Arduino	11
1.5.1	Η πλακέτα Arduino	12
1.5.2	Ακροδέκτες εισόδου/εξόδου	12
1.6	Το Λογισμικό (IDE)	13
1.7	Γλώσσα Arduino	14
1.8	Αισθητήρες	14
2	Μελέτη προηγούμενων πρότζεκ	16
2.1	Ελεύθερη πτήση με 3 βαθμούς ελευθερίας	16
2.2	Ελεύθερη πτήση με 6 βαθμούς ελευθερίας	16
2.3	Οπτική Ανίχνευση με την χρήση κάμερας	17
2.4	Ελεγκτής Fuzzy	17
2.5	Δυναμικά Χαρακτηριστικά & Έλεγχος PID	17
2.6	Σύνοψη και σύγκριση των προαναφερόμενων πρότζεκτ	19
3	Σχεδιασμός	20
3.1	Περιγραφή σχεδιασμού	20
4	Μελέτη κίνησης τετρακοπτέρου	22
4.1	Εισαγωγή	22
4.2	Υλικά	24
4.3	Συνδεσμολογία	26

4.4	Κώδικας	27
5	Μελέτη επικοινωνίας μέσω Wi-Fi	30
5.1	Εισαγωγή	30
5.2	Κατάσταση παραστατικής μεταφοράς - REpresentational State Transfer (REST) . .	30
5.3	Λειτουργία Arduino Yun	31
5.4	Σύνδεση του Arduino Yun με το Wi-Fi	32
6	Μελέτη γραφικού περιβάλλοντος πλοήγησης του τετρακοπτέρου	33
6.1	Εισαγωγή	33
7	Μελέτη Web-Camera	34
7.1	Εισαγωγή	34
7.2	Υλικά	34
8	Μελέτη αισθητήρων για την συλλογή δεδομένων	36
8.1	Εισαγωγή	36
8.2	Συνδεσμολογία	36
8.3	Κώδικας	37
9	Συμπεράσματα και μελλοντικές επεκτάσεις	38

Περίληψη

Αναντίρρητα, στη σημερινή εποχή, με την τεχνολογική πρόοδο και εξέλιξη που υπάρχει έχει καταστεί εφικτή η αυτοματοποίηση και απλοποίηση πολλών πτυχών της καθημερινότητάς μας, λόγο ότι υπάρχει πληθώρα αισθητήρων που μπορούν να χρησιμοποιηθούν για να γίνουν αρκετές και διαφορετικές μετρήσεις, με απώτερο στόχο τη βελτίωση του τρόπου ζωής μας. Παρ'όλα αυτά η συλλογή δεδομένων, σε ένα εξωτερικό περιβάλλον παραμένει ένα δύσκολο επίτευγμα, μιας και πρέπει να υπάρξουν πολλά ηλεκτρικά κυκλώματα και πολλοί αισθητήρες για τις μετρήσεις. Την τελευταία τετραετία έχει αναπτυχθεί αρκετά ο κλάδος των μη επανδρωμένων ιπτάμενων οχημάτων (drones) που κατά κύριο λόγο χρησιμοποιούνται για στρατιωτικούς σκοπούς, ειδικότερα για επιθέσεις από αέρος, την εξερεύνηση μιας περιοχής για την ανεύρεση κάποιας κρυφής βάσης κ.ά. Σε αντίθεση με τα προηγούμενα τα drones μπορούν να χρησιμοποιηθούν ταυτόχρονα και για εμπορικούς αλλά και για ερευνητικούς σκοπούς, όπως οι μετρήσεις δεδομένων που γίνονται από μετεωρολογικούς σταθμούς, μετρήσεις της ατμοσφαιρικής ρύπανσης σε περιοχές που έχουν δύσκολη πρόσβαση, όπως έγινε στην Ιαπωνία μετά την έκρηξη του πυρηνικού αντιδραστήρα και έστελναν drone σε διάφορες περιοχές να μετρήσουν τα επίπεδα ραδιενέργειας για να δουν αν μπορούν να ξαναγίνουν κατοικήσιμες.

Τις τελευταίες δεκαετίες, η χρήση των τετρακοπτέρων έχει διευρυνθεί και εντατικοποιηθεί για πολλές εφαρμογές, καθώς η ανάγκη για αεροσκάφη με μεγαλύτερη ευελιξία και ικανότητα πλεύσης, έχει οδηγήσει σε αύξηση αυτής της έρευνας. Για αυτό το λόγο ο σχεδιασμός των τεσσάρων μοτέρ επιτρέπει στα τετρακόπτερα να είναι σχετικά απλά στο σχεδιασμό τους, αλλά εξαιρετικά αξιόπιστα και ευέλικτα. Η συνεχιζόμενη έρευνα λειτουργεί εξελικτικά και αυξητικά ως προς τις ικανότητες των τετρακοπτέρων, έχοντας ήδη κάνει πρόοδο στην επικοινωνία πολλαπλών τεχνικών, στην εξερεύνηση του περιβάλλοντος και στην ευελιξία. Στην περίπτωση που καταστεί εφικτό ο συνδυασμός των αναπτυσσόμενων ιδιοτήτων, τα τετρακόπτερα θα καταστούν ικανά για προηγμένες και αυτόνομες αποστολές που δεν είναι επί του παρόντος εφικτές με οποιαδήποτε άλλα οχήματα.

Παράλληλα, αξίζει να αναφερθεί ότι η χρήση των τετρακοπτέρων δεν περιορίζεται μόνο στις στρατιωτικές αποστολές, αλλά μπορούν να χρησιμοποιηθούν και ως ευέλικτες πλατφόρμες δοκιμών από χομπίστες. Αρχικά, έχουν οικονομικό κόστος παραγωγής, υπάρχει ποικιλία ως προς τα μεγέθη και η ευκολία της συναρμολόγησης του απλού μηχανικού σχεδιασμού τους, τα καθιστά απλά στην κατασκευή και στην συντήρηση τους ακόμα και από ερασιτέχνες. Ωστόσο, λόγω του διεπιστημονικού χαρακτήρα της λειτουργίας ενός τετρακοπτέρου, οι ακαδημαϊκοί από έναν αριθμό επιστημονικών πεδίων πρέπει να συνεργαστούν για να επιφέρουν σημαντικές βελτιώσεις στον τρόπο λειτουργίας των τετρακοπτέρων.

Στόχος αυτής της εργασίας είναι να φτιαχτεί ένα τετρακόπτερο που να είναι διαχειρίσιμο μέσω ενός συνδέσμου (URL) και που μπορεί να πλοηγηθεί μέσω του γραφικού περιβάλλοντος ενός περιηγητή (browser), ώστε να μην είναι αναγκαία η χρήση τηλεχειριστηρίου από πιο έμπειρους και εξειδικευμένους χρήστες. Επίσης, στο browser θα λαμβάνουμε τα αποτελέσματα, πρωτίστως εικόνα από την κάμερα που είναι ενσωματωμένη πάνω στο τετρακόπτερο, για να

μπορεί να υπάρχει καλύτερος έλεγχος του και δευτερευόντως δεδομένα από τους 5 αισθητήρες και που θα προβάλλονται τα αποτελέσματα τους στον χρήστη.

Το υλικό είναι αρκετά απλοποιημένο και για τον λόγο αυτό δεν θα αναλυθεί ο κώδικα της εφαρμογής. Για όποιον επιθυμεί περισσότερα πράγματα διατίθεται στο τελευταίο μέρος της εργασίας ένας συνδεδεμένος για το github που είναι αναλυτικά όλος ο κώδικας για όλες τις διεργασίες και είναι ελεύθερος ο καθένας να πειραματιστεί με την φαντασία του και να δημιουργήσει ελεύθερα το ακατόρθωτο.

Λέξεις κλειδιά:]Arduino,Electronic stability control,τετρακόπτερο,αισθητήρας θερμοκρασίας, υγρασίας, θορύβου, φωτίας

1 Εισαγωγή

Οι αισθητήρες χρησιμοποιούνται σε αναρίθμητα αντικείμενα καθημερινής χρήσης, όπως τα κουμπιά του ανελκυστήρα με αισθητήρα αφής ή το θερμόμετρο που βάζουμε όταν αρρωσταίνουμε και άλλες πολυάριθμες εφαρμογές που οι περισσότεροι άνθρωποι δεν αντιλαμβανόμαστε την χρήση τους τώρα. Με την πρόοδο της πλατφόρμας των μικροεπεξεργαστών και των εύχρηστων μικρο ελεγκτών, οι χρήσεις των αισθητήρων έχουν επεκταθεί πέρα από τα παραδοσιακά πεδία μέτρησης της θερμοκρασίας, της πίεσης ή της ροής. Επιπλέον, πριν μια 5ετία ήταν αρκετά δύσκολο να φτιάξει κανείς τα δικά του κυκλώματα καθώς έπρεπε να είχε πολλές βαθιές τεχνικές γνώσεις και αυτός ο παράγων δεν βοηθούσε τόσο στην πρόοδο της τεχνολογίας και της επιστήμης για τον λόγο ότι όσο πιο λίγα άτομα ασχολούνται και πειραματίζονται με κάτι, τόσο πιο δύσκολα επιλύονται προβλήματα πάνω σε αυτό.

Με τη ίδρυση και κατασκευή του Arduino πολλά από αυτά τα προβλήματα έγιναν ευκολότερα και πιο αντιληπτά ακόμα για άτομα τα οποία δεν έχουν κάποια τεχνική κατάρτιση στον τομέα αυτό, ιδιαίτερα σήμερα που η τεχνολογία και η δημιουργία είναι κρίσιμο και ζωτικό κομμάτι του πολιτισμού μας, μιας και έχουμε μάθει στην συνεχόμενη πληροφόρηση και αυτοματοποίηση των πραγμάτων. Με την παροχή των αμέτρητων αισθητήρων και με την έμπνευση του Arduino όπως και του physical computing, πλέον μπορούμε να χρησιμοποιήσουμε ο καθένας την φαντασία του και να φτιάξουμε κατασκευές είτε για την διευκόλυνση μας είτε για την κατανόηση του κόσμου μας.

1.1 Τι είναι το Arduino

Το Arduino είναι ένας ανοιχτού κώδικα physical computer που βασίζεται στην απλή λογική της εισόδου/εξόδου (I/O) και έχει ένα περιβάλλον ανάπτυξης που χρησιμοποιεί την Processing η οποία είναι μια γλώσσα ανοιχτού λογισμικού για μικρο επεξεργαστές. Επίσης, ο κύριος λόγος που χρησιμοποιείται είναι για την ανάπτυξη αυτόνομων και διαδραστικών αντικειμένων ή για συλλογή δεδομένων μέσω των συνδεδεμένων ανιχνευτών. Το Arduino είναι ένα εργαλείο για την πραγματοποίηση διαδραστικών εφαρμογών, σχεδιασμένο απλοϊκά ώστε να γίνεται αντιληπτό από αρχάριους, αλλά εξακολουθεί να είναι αρκετά ευέλικτο ακόμα και για τους εμπειρογνώμονες για να τους βοηθήσει στην ανάπτυξη πιο σύνθετων εφαρμογών. Από την ίδρυση του το 2005, έχουν πωληθεί περισσότερες από 200,000 πλακέτες Arduino και υπάρχει μια ραγδαία αύξηση του αριθμού των πρότζεκτ που το χρησιμοποιούν σαν υπολογιστικό πυρήνα. Η κοινότητα του Arduino είναι τεράστια οπότε, υπάρχει ένας μεγάλος αριθμός σεμιναρίων, βιβλίων και φόρουμ διαθέσιμα για όλο το φάσμα των χρηστών.

Το Arduino διαφέρει από τις άλλες πλατφόρμες για τους παρακάτω λόγους:

- Το προγραμματιστικό περιβάλλον του μπορεί να τρέξει σε διαφορετικά προγραμματιστικά περιβάλλοντα (Windows, Linux, Macintosh).

- Βασίζεται στην IDE Processing, ένα εύκολο στη χρήση περιβάλλον ανάπτυξης που χρησιμοποιείται από καλλιτέχνες και σχεδιαστές.
- Μπορεί να προγραμματιστεί μέσω ενός καλωδίου USB, είτε μέσω Wi-Fi άμα έχει συνδεθεί στο ρούτερ του οικιακού δικτύου.
- Έχει ανοιχτό hardware και software. Οπότε, όποιος επιθυμεί μπορεί να κατεβάσει το διάγραμμα του κυκλώματος, να αγοράσει τα εξαρτήματα που χρειάζεται και να φτιάξει το δικό του.
- Το hardware είναι αρκετά φθηνό. Ένα USB board κοστίζει περίπου €20 και η αντικατάσταση ενός κατεστραμμένου κυκλώματος δεν κοστίζει παραπάνω από €5.
- Υπάρχει μια αρκετά μεγάλη και ενεργή κοινότητα χρηστών, που είναι πρόθυμοι να βοηθήσουν και να απαντήσουν σε διάφορα προβλήματα που μπορεί να αντιμετωπίσουν οι χρήστες.

1.2 Μια σύντομη ιστορία του Arduino

Το Arduino γεννήθηκε το 2005 στο Interaction Design Institute Ivrea, στην Ιταλία, ως μια ανοιχτού κώδικα πλατφόρμα. Οι ιδρυτές του έργου ήταν ο Massimo Banzi και ο David Cuartielles. Ο Hernando Barragan, ένας φοιτητής στο ίδιο ινστιτούτο της Ιταλίας, μαζί με τον Diego Gonzalez Joven που είχε αναπτύξει την Wiring ως θέμα της μεταπτυχιακής του διατριβής, η οποία ήταν υπό την επίβλεψη του Massimo Banzi και του Casey Reas (ένας από τους εμπνευστές της Processing). Η ιδέα πίσω από την Wiring ήταν να επιτρέπει την εύκολη χρήση του προγραμματισμού και της ηλεκτρονικής από καλλιτέχνες και σχεδιαστές σε μια παρόμοια νοοτροπία με την οποία ο Casey Reas και ο Ben Fry, είχαν αναπτύξει την Processing μερικά χρόνια νωρίτερα. Το Arduino χτίστηκε γύρω από την Wiring, αλλά σχεδιάστηκε και αναπτύχθηκε ανεξάρτητα μετά το 2005.

1.3 Τι είναι Physical Computing

Ο όρος Physical Computing περιγράφει τη ζεύξη του φυσικού χώρου με το χώρο των υπολογιστών. Μέσα από τη χρήση μικροεπεξεργαστών ανοικτού κώδικα και αισθητήρων, σε συνδυασμό με τη χρήση ειδικού λογισμικού δημιουργούνται και D.I.Y. διαδραστικές κατασκευές, εφαρμογές και εγκαταστάσεις οι οποίες προτείνουν πρωτότυπες εμπειρίες.

Περιλαμβάνει το σχεδιασμό διαδραστικών εφαρμογών που μπορούν να επικοινωνούν με ανθρώπους χρησιμοποιώντας αισθητήρες και ενεργοποιητές που ελέγχονται από ένα λογισμικό που τρέχει μέσα σε ένα μικροελεγκτή (ένας μικρός υπολογιστής με ένα μόνο ολοκληρωμένο κύκλωμα).

Στο παρελθόν, η χρήση της ηλεκτρονικής απαιτούσε να ασχοληθεί κανείς με τη μηχανική και απαιτούσε αρκετούς φυσικούς πόρους όπως και χρόνο για την κατασκευή ενός κυκλώματος. Αυτά τα προβλήματα αποθάρρυναν δημιουργικούς ανθρώπους να ασχοληθούν με αυτά. Επίσης,

τα περισσότερα εργαλεία που προορίζονται για τους μηχανικούς απαιτούσαν εκτεταμένη γνώση και είχαν μεγάλο κόστος για να αποκτηθούν.

Όμως τα τελευταία χρόνια, οι μικροελεγκτές έχουν γίνει φθηνότεροι και πιο εύχρηστοι επιτρέποντας έτσι τη δημιουργία καλύτερων και πιο διαδραστικών εργαλείων.

Το Arduino, λοιπόν, έχει φέρει αυτά τα εργαλεία πιο κοντά σε έναν αρχάριο χρήστη, επιτρέποντάς του να μπορέσει να αρχίσει την δημιουργία μικρών projects μετά από 2 ή 3 ημέρες. Με το Arduino, ένας σχεδιαστής ή καλλιτέχνης μπορεί να μάθει βασικές γνώσεις ηλεκτρονικής και να γνωρίσει κάθε είδους αισθητήρα πολύ γρήγορα και να υλοποιήσει ένα πρότζεκ χωρίς μεγάλο κόστος ή εξειδικευμένες γνώσεις.

1.4 Τι είναι ο μικροελεγκτής

Όπως ανέφερα και πιο πάνω είναι ένας πολύ απλός και μικρός υπολογιστής που είναι πολύ καλός σε τρία συγκεκριμένα πράγματα: λαμβάνει πληροφορίες από αισθητήρες, ελέγχει απλούς κινητήρες και άλλες συσκευές που προκαλούν φυσική αλλαγή και αποστέλλει πληροφορίες σε άλλους υπολογιστές ή συσκευές.

Γενικά ενεργεί ως πύλη μεταξύ του φυσικού κόσμου και του κόσμου των υπολογιστών. Οι μικροελεγκτές βρίσκονται παντού, από τα πλυντήρια μέχρι και τους διακόπτες για τα φώτα. Για αυτό το λόγο η τιμή τους έχει μειωθεί και η χρήση τους έχει γίνει ευρέως διαδεδομένη και ευκολότερη. Έτσι μπορεί κάποιος να εξερευνήσει ενσωματωμένα υπολογιστικά συστήματα στα πιο απίθανα μέρη ή να αναπτύξει απλές συσκευές με πιο πολυσύνθετες λειτουργίες.

1.5 Η πλατφόρμα Arduino

Το Arduino αποτελείται από δύο κύρια μέρη:

- Την πλακέτα Arduino, το οποίο είναι το κομμάτι του hardware για την κατασκευή ενός πρότζεκ.
- Το Arduino IDE, το κομμάτι του software που τρέχει στον υπολογιστή.

Το IDE χρησιμοποιείται για την δημιουργία ενός σκίτσου (ένα μικρό πρόγραμμα), το οποίο στη συνέχεια “ανεβαίνει” στην πλακέτα Arduino. Το σκίτσο λέει στην πλακέτα Arduino τι πρέπει να κάνει.

Πριν λίγα χρόνια, το να δουλεύει κανείς πάνω στο hardware σήμαινε ότι έπρεπε να φτιάξει ένα κύκλωμα από το μηδέν. Όπως επίσης, να χρησιμοποιούσε εκατοντάδες διαφορετικά εξαρτήματα όπως αντιστάσεις, πυκνωτές, πηνία, τρανζίστορ, κ.ο.κ.. Κάθε κύκλωμα ήταν δύσκολο να αλλάξει, καθώς θα έπρεπε ο χρήστης να κόψει καλώδια, να κάνει συγκολλήσεις κ.ο.κ.

Με την εμφάνιση των ψηφιακών τεχνολογιών και των μικροελεγκτών αυτές οι λειτουργίες αντικαταστάθηκαν με προγράμματα λογισμικού. Το λογισμικό επιδέχεται ευκολότερα αλλαγές από το hardware. Με λίγες εντολές μπορεί να αλλάξει ολοκληρωτικά η λογική της συσκευής.

1.5.1 Η πλακέτα Arduino

Η πλακέτα Arduino είναι μια μικρή πλακέτα μικροελεγκτή, η οποία είναι ένα μικρό κύκλωμα που περιέχει ένα ολόκληρο υπολογιστή σε ένα μικρό ολοκληρωμένο κύκλωμα.

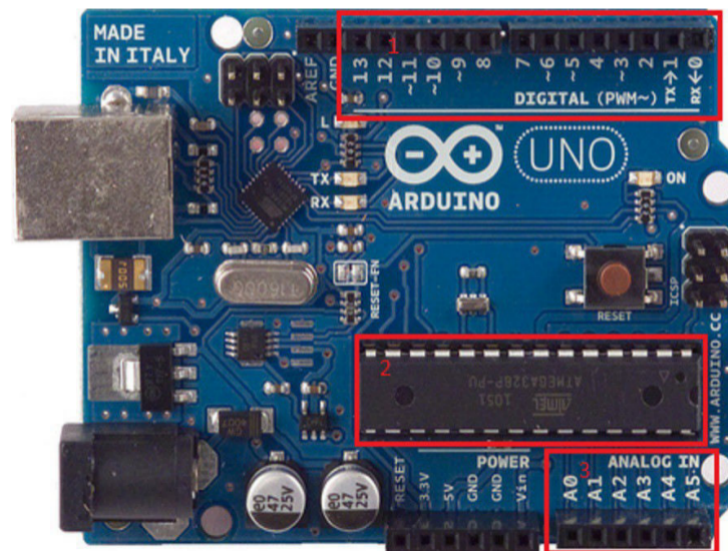
Μπορούμε να σκεφτούμε την πλακέτα Arduino ως ένα μικρό εγκέφαλο που μας επιτρέπει να συνδέσουμε μαζί ένα ευρύ φάσμα αισθητήρων και ενεργειών. Η πλακέτα Arduino είναι χτισμένη γύρω από ένα 8-bit μικροελεγκτή ATmel AVR.

Ανάλογα με την πλακέτα, βρίσκουμε διάφορες μάρκες, συμπεριλαμβανομένων των ATmega8, ATmega168, ATmega328, ATmega1280, και ATmega2560. Οι είσοδοι/έξοδοι μιας πλακέτας Arduino μπορούν να χρησιμοποιηθούν και σαν είσοδοι/έξοδοι για άλλα κυκλώματα χτισμένα γύρω από το Arduino. Το Arduino Uno είναι η πιο απλή και διαδεδομένη πλακέτα που μπορούμε να χρησιμοποιήσουμε.

1.5.2 Ακροδέκτες εισόδου/εξόδου

Το Arduino Uno έχει 14 ακροδέκτες ψηφιακής εισόδου/εξόδου και 6 ακροδέκτες αναλογικής εισόδου, όπως φαίνεται στο παρακάτω σχήμα (Εικόνα 1). Αυτοί οι ακροδέκτες αντιστοιχούν στους ακροδέκτες εισόδου/εξόδου του ATmega μικροελεγκτή. Η αρχιτεκτονική της πλακέτας διαθέτει αυτούς τους ακροδέκτες για να είναι ευκολότερο να συνδεθεί η πλακέτα με εξωτερικά κυκλώματα. Οι ακροδέκτες μπορούν να οριστούν είτε ως ακροδέκτες εισόδου είτε ως ακροδέκτες εξόδου μέσα στο σκίτσου (πρόγραμμα).

1. Ψηφιακοί ακροδέκτες (digital pins): Το Arduino Uno έχει 14 ψηφιακούς ακροδέκτες, οι οποίοι είναι αριθμημένοι από το 0 έως το 13 και μπορούν να οριστούν και ως ακροδέκτες εισόδου αλλά και ως εξόδου. Και στις 2 αυτές λειτουργίες οι ακροδέκτες μπορούν μόνο να στέλνουν ή να λαμβάνουν ψηφιακά σήματα τα οποία αποτελούνται από 2 καταστάσεις: ON (HIGH ή 5V) και OFF (LOW ή 0V). Οι ακροδέκτες που έχουν οριστεί ως έξοδοι μπορούν να περέχουν ρεύμα σε εξωτερικές συσκευές ή κυκλώματα. Αντίθετα, οι ακροδέκτες που έχουν οριστεί ως είσοδοι διαβάζουν ρεύματα από συσκευές που είναι συνδεδεμένες σε αυτούς. Τέλος, 6 από αυτούς του ακροδέκτες (PWM pins) μπορούν να χρησιμοποιηθούν και ως ακροδέκτες διαμόρφωσης εύρους παλμού (PWM). Με την χρήση αυτών των ακροδεκτών μπορούμε να εξομοιώσουμε αναλογικές τιμές.
2. Αναλογικοί ακροδέκτες εισόδου (analog input pins): Οι μικροελεγκτές ATmega, που χρησιμοποιούν οι πλακέτες Arduino, περιέχουν 6 κανάλια analog-to-digital μετατροπέα (ADC). Η λειτουργία αυτής της συσκευής είναι να μετατρέψει μία αναλογική τάση εισόδου



Σχήμα 1: 1. Ψηφιακοί Ακροδέκτες, 2. ATmega328 μικροελεγκτής, 3. Αναλογικοί Ακροδέκτες

σε ένα ψηφιακό αριθμό ανάλογο με το μέγεθος της τάσης εισόδου σε σχέση με την τάση αναφοράς (5V). Ο μετατροπέας ATmega για τις πλακέτες Arduino έχει ανάλυση 10-bit, το οποίο σημαίνει ότι θα επιστρέφει ακραίους από 0 έως 1023. Για παράδειγμα, μια είσοδος 0V θα επιστρέφει 0, μια είσοδος 5V θα επιστρέφει 1023, ενώ μια είσοδος 2.5V θα επιστρέφει 512. Αυτοί οι ακροδέκτες μπορούν να οριστούν και ως ακροδέκτες εισόδου ή εξόδου ακριβώς όπως και οι ψηφιακοί ακροδέκτες. Αν σε κάποιο project χρειαστούν παραπάνω ψηφιακοί ακροδέκτες, μπορούμε να χρησιμοποιήσουμε και τους αναλογικούς ακροδέκτες σαν να είναι ψηφιακοί.

1.6 Το Λογισμικό (IDE)

Το Arduino IDE είναι μια εφαρμογή cross-platform γραμμένη σε Java και προέρχεται από τον IDE για την Processing και την Wiring. Ένα IDE είναι ένα πρόγραμμα που έχει σχεδιαστεί για να διευκολύνει την ανάπτυξη λογισμικού. Το Arduino IDE περιλαμβάνει ένα πρόγραμμα επεξεργασίας πηγαίου κώδικα και έναν compiler που κάνει τον κώδικα κατανοητό στον μικροελεγκτή. Όταν, λοιπόν, “ανέβει” το σκίτσο (πρόγραμμα) στην πλακέτα: ο κώδικας μεταφράζεται στη γλώσσα C και περνά στο avr-gcc compiler, ένα σημαντικό κομμάτι του λογισμικού ανοιχτού κώδικα που κάνει την τελική μετάφραση και έτσι ο κώδικας γίνεται κατανοητός από το μικροελεγκτή.

Ο κύκλος προγραμματισμού για Arduino είναι ο εξής:

1. Σύνδεση της πλακέτας σε μια θύρα USB ενός υπολογιστή.
2. Γραφή σκίτσου.
3. "Ανέβασμα" του σκίτσου στην πλακέτα μέσω της USB σύνδεσης και αναμονή για επανεκκίνηση της πλακέτας.

4. Η πλακέτα εκτελεί το σκίτσο που μόλις “ανέβηκε”.

1.7 Γλώσσα Arduino

Η γλώσσα Arduino υλοποιείται σε C/C++ και βασίζεται στην Wiring. Όταν γράφεται ένα σκίτσο γίνεται χρήση της βιβλιοθήκης Wiring, η οποία περιλαμβάνεται στο Arduino IDE. Αυτό μας επιτρέπει να φτιάχνουμε εκτελέσιμα προγράμματα με δύο μόνο λειτουργίες:

- setup()
- loop()

Όπου στην setup αρχικοποιούνται οι μεταβλητές, καθορίζουμε ποιες κεφαλές θα είναι εισόδου και ποιες εξόδου, είτε καλούμε κάποιες συναρτήσεις να ξεκινήσουν.

Ενώ η loop είναι η κύρια λειτουργία του προγράμματος που θα επαναλαμβάνεται συνεχόμενα όσο το Arduino έχει παροχή ενέργειας, είτε απο κάποια μπαταρία, είτε απο το λάπτοπ.

Όπως αναφέρθηκε, η Wiring είναι εμπνευσμένη από την Processing και για αυτό παρατηρείται πως και η δομή της γλώσσας Arduino κληρονομείται από την Processing.

1.8 Αισθητήρες

Στο Arduino μπορούν να χρησιμοποιηθούν μεγάλο εύρος απο αισθητήρες. Η λογική λειτουργίας του κάθε αισθητήρα είναι ίδια σχεδόν σε όλους. Έχουν τρεις με τέσσερις κεφαλές (pins). Υπάρχουν 3 κεφαλές στους αναλογικούς και 4 στους ψηφιακούς αισθητήρες.

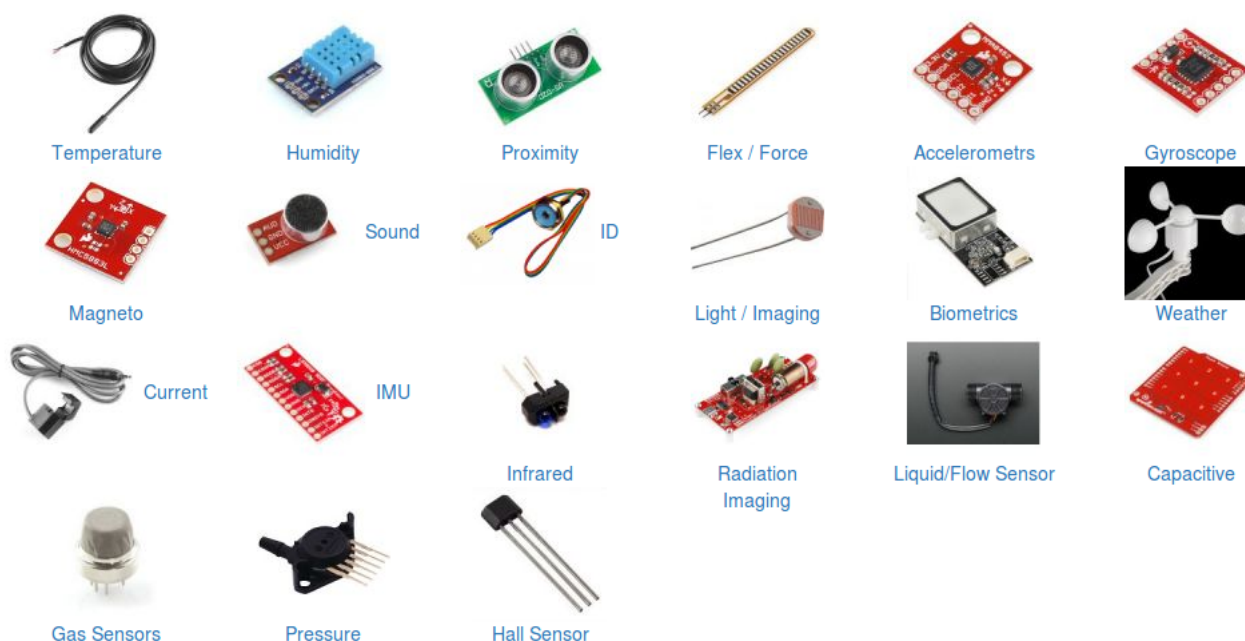
- Τάση ρεύματος
- Γείωση
- Αναλογική κεφαλή
- Ψηφιακή κεφαλή

Ενώ, υπάρχουν και άλλοι που δεν έχουν μια απο τις δύο κεφαλές, αναλογική ή ψηφιακή. Οπότε και η συνδεσμολογία τους στο κύκλωμα είναι αρκετά απλοποιημένη. Η τάση ρεύματος συνδέεται στην έξοδο ρεύματος του arduino, η γείωση με την γείωση και η αναλογική ή ψηφιακή κεφαλή με την αναλογική ή ψηφιακή αντίστοιχα του Arduino.

Η διαφορά στα δεδομένα και στις μετρήσεις γίνεται μέσω σκίτσου (προγράμματος)

Μια λίστα με τους πιο χρησιμοποιημένους είναι:

1. Επιταχυνσιόμετρο
2. Βιομετρίας
3. Χωρητικότητας
4. Αισθητήρας υγρασίας
5. Απεικόνισης
6. Υπέρυθρες
7. Υγρό / ροή
8. Αισθητήρας πίεσης
9. Ακτινοβολίας
10. Ήχου
11. Ταυτότητας
12. Αισθητήρας θερμοκρασίας
13. Αισθητήρας αερίου
14. Αισθητήρας Flex / Δύναμης
15. Αισθητήρας φωτός
16. Αισθητήρας κίνησης
17. Αισθητήρας κλίσης
18. Έγχρωμος αισθητήρας
19. Γυροσκόπιο
20. Μαγνητόμετρο
21. Αισθητήρας καιρού
22. Πυξίδα
23. Αισθητήρας ρεύματος
24. GPS



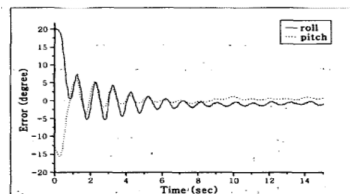
Σχήμα 2: Αισθητήρες

2 Μελέτη προηγούμενων πρότζεκ

Η έρευνα που αφορά τα τετρακόπτερα καλύπτει τους τομείς του σχεδιασμού, του ελέγχου, της σταθερότητας, της επικοινωνίας, την αποφυγή συγκρούσεων και την πλοήγηση. Αυτοί οι τομείς έχουν προσελκύσει τους ερευνητές και την ακαδημαϊκή κοινότητα για την έρευνα και την ανάπτυξη τους.

2.1 Ελεύθερη πτήση με 3 βαθμούς ελευθερίας

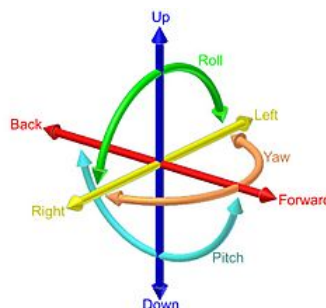
Ο [Park et al., 2001] μελέτησε το όχημα ελεύθερης πτήσης με 3 βαθμούς ελευθερίας (3-DOF). Οι βαθμοί ελευθερίας είναι ο αριθμός των παραμέτρων που καθορίζουν την μελέτη ενός φυσικού συστήματος. Δηλαδή, μελέτησε την συμπεριφορά ενός τετρακοπτέρου στους 3 άξονες προσπαθώντας να δημιουργήσει μια καλή συμπεριφορά πτήσης ώστε να στέκεται σταθερό.



Σχήμα 3: Τα αποτελεσματα απο το ελεγχο συμπεριφοράς της 3-DOF

2.2 Ελεύθερη πτήση με 6 βαθμούς ελευθερίας

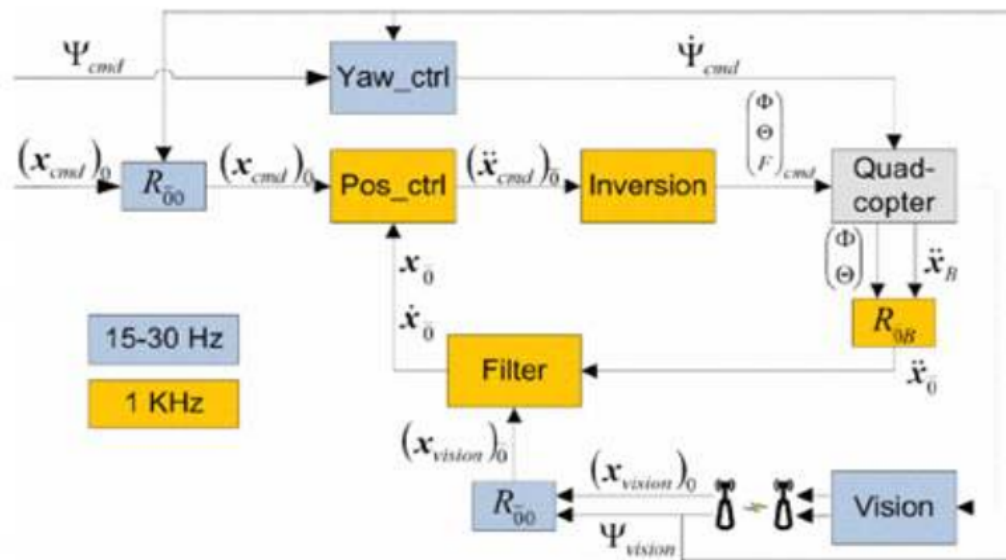
Οι [Mian and Daobo, 2008] ανέπτυξαν ένα μη γραμμικό μοντέλο και μια μη γραμμική στρατηγική ελέγχου για ένα 6-DOF. Πιο περιληπτικά αναφέρθηκε στην ελευθερία κίνησης ενός άκαμπτου σώματος σε τρισδιάστατο χώρο. Συγκεκριμένα, το σώμα είναι ελεύθερο να αλλάζει θέση ως μετατόπιση προς τα εμπρός / προς τα πίσω, προς τα επάνω / προς τα κάτω, προς τα αριστερά / δεξιά (ταλαντεύεται) σε τρεις κάθετους άξονες, σε συνδυασμό με αλλαγές στον προσανατολισμό μέσω περιστροφής γύρω από τρεις κάθετους άξονες, yaw (κανονικός άξονας), pitch (εγκάρσιος άξονας) και roll (διαμήκης άξονας).



Σχήμα 4: Άξονες υπολογισμού θέσης στην 6-DOF

2.3 Οπτική Ανίχνευση με την χρήση κάμερας

Ο [Achtelik et al., 2009] έκανε έρευνα πάνω στον χειρισμό και στην αυτοματοποίηση ενός τετρακοπτέρου. Η έρευνα του βασίστηκε με την οπτική ανίχνευσης της θέσης του τετρακοπτέρου με την χρήση μιας κάμερας που ήταν ενσωματωμένη πάνω στο ίδιο το τετρακόπτερο. Η κίνηση του τετρακοπτέρου ήταν βασισμένη πάνω στην οπτική ανατροφοδότηση από την κάμερα και ενός αισθητήρα αδράνειας.



Σχήμα 5: Μοντέλο σχεδιασμού για την πλοήγηση του τετρακοπτέρου από τον Achtelik

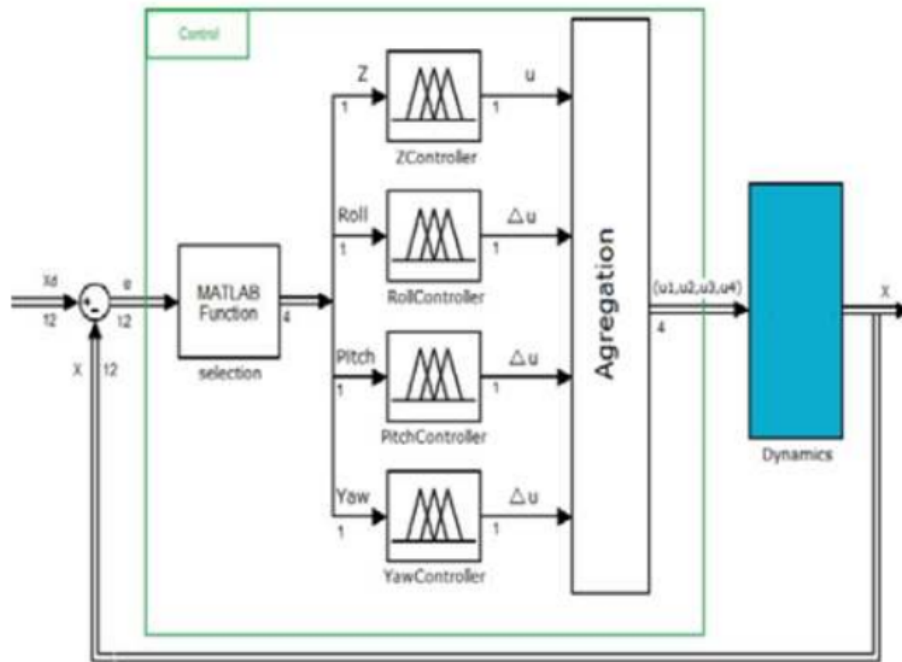
2.4 Ελεγκτής Fuzzy

Ο [Santos et al., 2010] δούλεψε πάνω στον έξυπνο fuzzy ελεγκτή για τα τετρακόπτερα. Ο ελεγκτής αυτός έχει σχεδιαστεί και υλοποιηθεί για να ελέγχει μια προσομοίωση ενός τετρακοπτέρου. Σαν είσοδο δέχεται τις τιμές που έχουν οι μεταβλητές του ύψους, του yaw (κανονικός άξονας), του pitch (εγκάρσιος άξονας) και του roll (διαμήκης άξονας). Η έξοδος αυτού του συστήματος είναι η ισχύς που πρέπει να έχει το κάθε μοτέρ ξεχωριστά για να φτάσει τις προσδοκίες των στόχων που έχουν καθοριστεί στην αρχή του συστήματος.

Τα αποτελέσματα των προσομοιώσεων αναδεικνύουν ότι ο ελεγκτής fuzzy βγάζει πάρα πολύ έγκυρα αποτελέσματα σαν έξοδο για την πραγματοποίηση των καθορισμένων στόχων.

2.5 Δυναμικά Χαρακτηριστικά & Έλεγχος PID

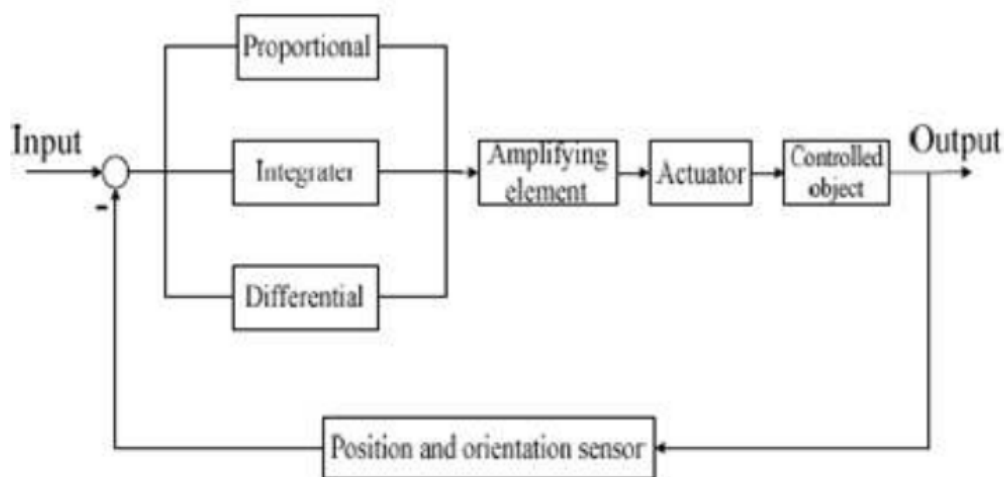
Ο [Li and Li, 2011] έκανε έρευνα πάνω στην ανάλυση των δυναμικών χαρακτηριστικών και τον έλεγχο του PID ενός τετρακοπτέρου. Στην δημοσίευση του περιγράφει την αρχιτεκτονική ενός τετρακοπτέρου και αναλύει το μοντέλο των δυνάμεων που υπάρχουν σε αυτό. Επίσης, σχεδίασε



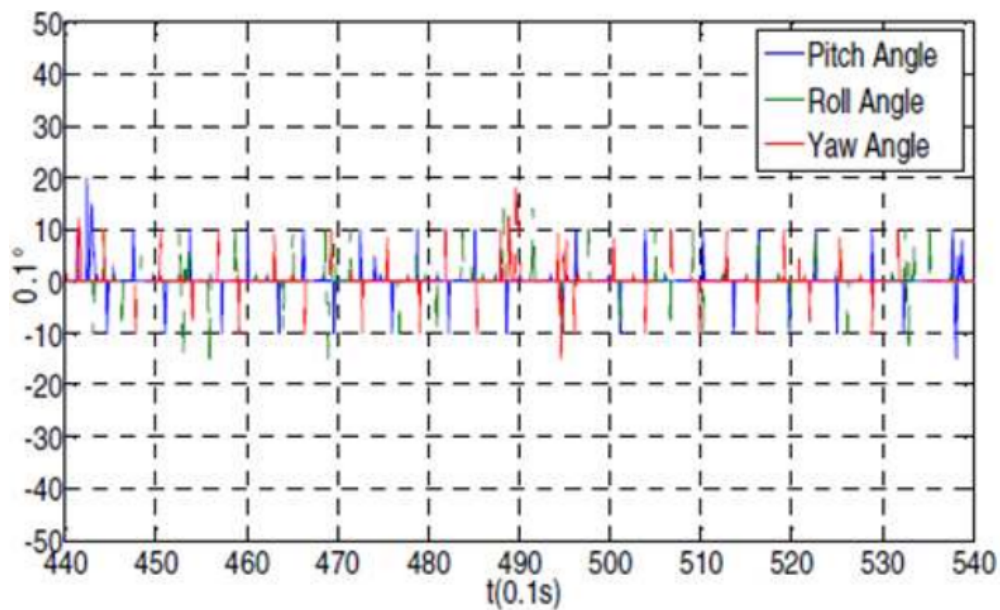
Σχήμα 6: Διάγραμμα ελέγχου χρησιμοποιώντας το Fuzzy

έναν ελεγκτή που έχει σαν στόχο να ρυθμίσει τη στάση (θέση και προσανατολισμό) ενός 6-DOF μοντέλου.

Ακόμα δημιούργησε ένα μοντέλο Simulink (αναπτύχθηκε από το MathWorks, είναι ένα γραφικό περιβάλλον προγραμματισμού για τη μοντελοποίηση) για τον PID (η δυνατότητα χρήσης των τριών όρων ελέγχου της αναλογικής, ολοκληρωτικής και παράγωγης επίδρασης στον ελεγκτή).



Σχήμα 7: Μοντέλο Simulink για τον PID.



Σχήμα 8: Αποτελέσματα της ερευνάς του

2.6 Σύνοψη και σύγκριση των προαναφερόμενων πρότζεκτ

Αρ.	Τίτλος Έρευνας	Πλεονεκτήματα	Μειονεκτήματα
1.	Ελεύθερη πτήση με 3 βαθμούς ελευθερίας	Απλό και βασικό σχέδιο ελέγχου	Περιορισμένος βαθμός ελευθερίας.
2.	Ελεύθερη πτήση με 6 βαθμούς ελευθερίας	Ελαττώνει το αρχικό σφάλμα με την πρόσθεση των γωνιών κύλισης,βήματος και στροφής	Μόνος σχεδιασμός για εξισορρόπηση της θέσης του τετρακοπτέρου.
3.	Οπτική Ανίχνευση με την χρήση κάμερας	Το σύστημα εντοπισμού είναι ιδιαίτερα παραμετροποιήσιμο	Ευαίσθητο στο φως και ακατάλληλο για χρήση σε χώρους υψηλής φωτεινότητας .
4.	Ελεγκτής Fuzzy	Γρήγορη δυναμική απόκριση και μικρό ποσοστό λάθους	Ο σχεδιασμός του ελεγκτή είναι πολύ περίπλοκος.
5.	Δυναμικά Χαρακτηριστικά & Έλεγχος PID	Ισχυρή προσαρμοστική ικανότητα	Το σύστημα είναι ασταθές χωρίς τις τιμές των K_p , K_i και K_d .

3 Σχεδιασμός

3.1 Περιγραφή σχεδιασμού

Η κύρια ιδέα της παρούσας πτυχιακής εργασίας ήταν να υλοποιηθεί ένα τετρακόπτερο το οποίο θα συνδέεται μέσω Wi-Fi με έναν υπολογιστή που ο χρήστης θα μπορεί να το ελέγξει. Η επικοινωνία θα γίνεται μεταξύ ενός Server (τετρακόπτερο) και ενός Client (ιστοσελίδα). Επίσης, στο τετρακόπτερο υπάρχει μία web-camera, έτσι ώστε ο χρήστης να μπορεί να το κινεί ακόμα και αν βρίσκεται σε διαφορετικό χώρο από αυτό. Σε αυτό το σημείο πρέπει να αναφέρω ότι η επικοινωνία επιτυγχάνεται μόνο όταν ο Server (τετρακόπτερο) και ο Client (ιστοσελίδα) βρίσκονται στο ίδιο δίκτυο γιατί διαφορετικά δεν εμφανίζεται η ιστοσελίδα πλοήγησης του τετρακοπτέρου. Τέλος, έγινε προσθήκη των κάτωθι αισθητήρων:

- Αισθητήρα θερμοκρασίας
- Αισθητήρα υγρασίας
- Αισθητήρα φωτιάς
- Αισθητήρα απόστασης
- Αισθητήρα ήχου

Για να γίνει συλλογή των δεδομένων του περιβάλλοντος που θα πετάξει το τετρακόπτερο ώστε μελλοντικά να μπορέσει να γίνει κάποια ανάλυση τους για την μεγαλύτερη αυτοματοποίηση του. Τις τιμές αυτών των δεδομένων μπορεί να τις δει ο χρήστης μέσα από το γραφικό περιβάλλον που παρέχει η χρήση της ιστοσελίδας στον χρήστη-πλοηγητή.

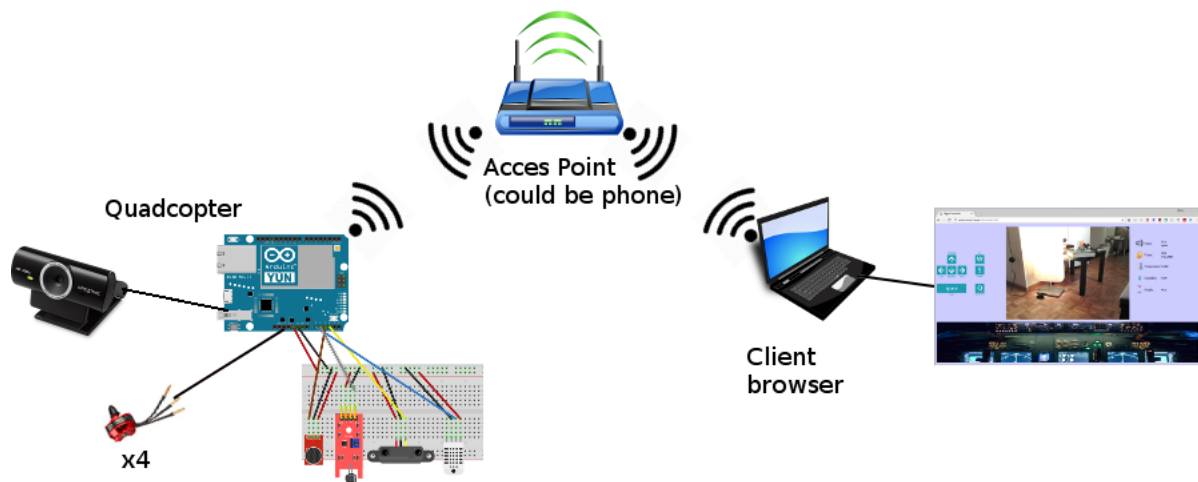
Στην συνέχεια της εργασίας θα δούμε πιο αναλυτικά τα παρακάτω στάδια που έχει μέχρι στιγμής το πρότζεκτ:

1. Μελέτη κίνησης τετρακοπτέρου
2. Μελέτη επικοινωνίας μέσω Wi-Fi
3. Μελέτη γραφικού περιβάλλοντος για την πλοήγηση του τετρακοπτέρου
4. Μελέτη Web-Camera
5. Μελέτη αισθητήρων για την συλλογή δεδομένων
6. Συμπεράσματα και μελλοντικές επεκτάσεις

Για κάθε ένα από αυτά τα στάδια που αναφέρονται πιο πάνω θα τα αναλύσουμε το καθένα ξεχωριστά σαν ένα αυτόνομο κεφάλαιο, έτσι ώστε ο κάθε αναγνώστης να μπορεί να το καταλάβει

πιο απλοϊκά, σε περίπτωση που θελήσει κανείς να συνεχίσει το πρότζεκτ ή να πάρει κάποια ιδέα υλοποίησης. Κάθε ένα από τα κεφάλαια θα έχει μια εισαγωγή που θα εξηγεί το τι θα αναλύσουμε σε αυτό, υλικά που απαιτούνται για κάποια συνδεσμολογία κυκλώματος που μπορεί να γίνει ή μια γενική λογική από τον κώδικα που χρειάστηκε να γραφτεί για να υλοποιηθούν οι ενέργειες που πραγματοποιεί το τετρακόπτερο.

Ο κώδικας για όλη την εργασία θα υπάρχει στο github και είναι ελεύθερος για τον κάθενα να πειραματιστεί μαζί του και να τον τροποποιήσει ανάλογα με τις ανάγκες του.



Σχήμα 9: Διάγραμμα υλοποίησης και επικοινωνίας

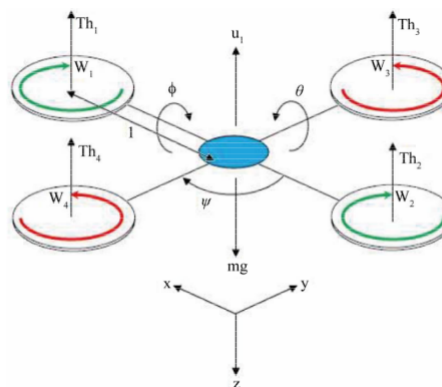
4 Μελέτη κίνησης τετρακοπτέρου

4.1 Εισαγωγή

Σε αυτό το κεφάλαιο θα εξετάσουμε την κίνηση του τετρακοπτέρου. Στο τέλος αυτού ο σκοπός μας είναι να κατασκευαστεί ένα τετρακόπτερο το οποίο θα κάνει προκαθορισμένες κινήσεις που θα του δίνει ένας χρήστης μέσω μιας ιστοσελίδας και να διαβάζουμε τα δεδομένα από διάφορες μετρήσεις που θα κάνουν οι συνδεδεμένοι αισθητήρες με το Arduino Yun που είναι πάνω στο τετρακόπτερο.

Γενικά να γνωρίζουμε ότι τα τετρακόπτερα χρησιμοποιούν δύο ζεύγη πανομοιότυπων σταθερών ελικοειδών ελίκων: δύο δεξιόστροφα (CW) και δύο αριστερόστροφα (CCW). Αυτά χρησιμοποιούν ανεξάρτητη μεταβολή της ταχύτητας σε κάθε μοτέρ του τετρακοπτέρου. Με την αλλαγή της ταχύτητας περιστροφής των μοτέρ είναι δυνατόν να δημιουργηθεί ειδικά μία επιθυμητή ολική ώθηση, θα μπορούσαμε να εντοπίσουμε το κέντρο της ώθησης τόσο στον πλευρικό όσο και στον κάθετο άξονα.

Με τα δύο μοτέρ να περιστρέφονται κατά τη φορά των δεικτών του ρολογιού και τα άλλα δύο κατά την αντίθετη φορά, το συνολικό άθροισμα της ροπής αντίδρασης είναι μηδέν και αυτός είναι ο λόγος που επιτρέπει στο τετρακόπτερο να παραμένει σταθερό στον αέρα.



Σχήμα 10: Μαθηματικό μοντέλο των δυνάμεων που ασκούνται σε ένα οποιοδήποτε τετρακόπτερο

Όπου:

1. Το U_1 είναι το άθροισμα της ώθησης κάθε μοτέρ.
2. Τα Th_1 , Th_2 , Th_3 και Th_4 είναι ώθηση που παράγεται από το μπροστινό, πίσω, αριστερά και δεξιά μοτέρ αντίστοιχα.
3. Το m είναι η μάζα.
4. Το g είναι η επιτάχυνση βαρύτητας.
5. Το l είναι το μισό μήκος του τετρακοπτέρου.

6. Τα χ , y και z είναι το φίλτρο τριών αξόνων.

7. Τα θ , ϕ και ψ είναι τρεις γωνίες Euler που αντιπροσωπεύουν το pitch, roll και yaw.

Η δυναμική σύνθεση του τετρακοπτέρου να κινείται από τη θέση προσγείωσης σε ένα σταθερό σημείο του χώρου παρουσιάζεται στον παρακάτω τύπο:

$$R_{xyz} = \begin{bmatrix} C_\theta C_\phi & C_\theta S_\phi S_\psi - S_\theta C_\psi & C_\theta S_\phi C_\psi + S_\theta S_\psi \\ C_\phi S_\theta & S_\phi S_\theta S_\psi + C_\phi C_\psi & S_\phi S_\theta C_\psi - C_\phi S_\psi \\ -S_\theta & C_\theta S_\psi & C_\theta C_\psi \end{bmatrix}$$

Που:

$$S_\theta = \sin(\theta), S_\phi = \sin(\phi), S_\psi = \sin(\psi),$$

$$C_\theta = \cos(\theta), C_\phi = \cos(\phi), C_\psi = \cos(\psi)$$

Τα τετρακόπτερα διαφέρουν από τα συμβατικά ελικόπτερα, τα οποία χρησιμοποιούν δρομείς που μπορούν να μεταβάλλουν δυναμικά το βήμα των λεπίδων τους καθώς κινούνται γύρω από το πλήθος των μοτέρ. Στις πρώτες μέρες πτήσης, τα τετρακόπτερα θεωρήθηκαν πιθανές λύσεις σε ορισμένα από τα επίμονα προβλήματα της κάθετης πτήσης. Τα ζητήματα ελέγχου που προκαλούνται από τη ροπή μπορούν να εξαλειφθούν με αντίθετη περιστροφή και οι σχετικά βραχείες λεπίδες τους είναι πολύ ευκολότερες στην κατασκευή. Ορισμένα σχέδια για επανδρωμένα οχήματα εμφανίστηκαν στη δεκαετία του 1920 και του 1930. Τα οχήματα αυτά ήταν μεταξύ των πρώτων επιτυχημένων οχημάτων κατακόρυφης απογείωσης και προσγείωσης (VTOL). Ωστόσο, τα πρωτότυπα υπέφεραν από κακή απόδοση και τα τελευταία πρωτότυπα απαιτούσαν υπερβολικά έμπειρους χειριστές για να μπορεί να δώσει μια ικανοποιητική πτήση.

Στα τέλη της δεκαετίας του 2000, η πρόοδος στην ηλεκτρονική επέτρεψε την παραγωγή φτηνών και ελαφρών ελεγκτών πτήσης, επιταχυνσιόμετρα (IMU), παγκόσμιου συστήματος εντοπισμού θέσης και κάμερες. Αυτό είχε σαν αποτέλεσμα η τετραπλή διάταξη να γίνει δημοφιλής για τα μικρά μη επανδρωμένα οχήματα. Με το μικρό τους μέγεθος και την ευελιξία τους, αυτά τα τετρακόπτερα μπορούν να πετούν σε εσωτερικούς και εξωτερικούς χώρους.

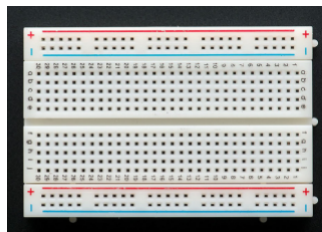
Πλέον τα τετρακόπτερα είναι φθηνότερα και πιο ανθεκτικά από τα συμβατικά ελικόπτερα λόγω της μηχανικής τους απλότητας. Οι μικρότερες λεπίδες τους επίσης δίνουν ένα μεγάλο πλεονέκτημα επειδή διαθέτουν λιγότερη κινητική ενέργεια, μειώνοντας την ικανότητά τους να προκαλέσουν βλάβη. Ωστόσο, καθώς αυξάνεται το μέγεθος, τα τετρακόπτερα αναπτύσσουν μειονεκτήματα έναντι των συμβατικών ελικοπτερόν. Η αύξηση του μεγέθους των λεπίδων αυξάνει την ορμή τους. Αυτό σημαίνει ότι οι αλλαγές στην ταχύτητα των λεπίδων διαρκούν περισσότερο, γεγονός που επηρεάζει αρνητικά τον έλεγχο τους. Τα ελικόπτερα δεν αντιμετωπίζουν αυτό το πρόβλημα καθώς η αύξηση του μεγέθους του δίσκου του μοτέρ δεν επηρεάζει σημαντικά την ικανότητα ελέγχου της κλίσης του πτερυγίου.

Παρακάτω θα αναφέρουμε πιο αναλυτικά τι εξαρτήματα χρειαστήκαμε για την δημιουργία ενός τετρακοπτέρου. Την συνδεσμολογία και τις ρυθμίσεις που κάναμε μεταξύ τους για να μπορέσει να είναι σε θέση να πλοηγηθεί ευπρεπώς όπως και κάποια ιδιαίτερα σημεία στον

κώδικα για την πλοήγηση του.

4.2 Υλικά

Μια breadboard.



Σχήμα 11: Breadboard

Το Arduino Yun



Σχήμα 12: Το Arduino Yun

4 ESC(electronic speed controller) Είναι ένα ηλεκτρονικό κύκλωμα με σκοπό να μεταβάλλει την ταχύτητα του ηλεκτροκινητήρα, την κατεύθυνσή του και ενδεχομένως να δρα και ως δυναμική πέδηση. Τα ESC χρησιμοποιούνται συχνά σε ηλεκτροκίνητα ραδιοελεγχόμενα μοντέλα, με μεγαλύτερη ποικιλία από αυτά, να χρησιμοποιείται συχνότερα σε κινητήρες χωρίς ψήκτρες, παρέχοντας κατ'ουσίαν μία ηλεκτρονικά παραγόμενη τριφασική πηγή ηλεκτρικής ενέργειας χαμηλής τάσης για τον κινητήρα.



Σχήμα 13: ESC

4 Brushless DC electric motor Ο ηλεκτροκινητήρας συνεχούς ρεύματος (κινητήρες BLDC, μοτέρ BL) που είναι επίσης γνωστοί ως ηλεκτροκινητήρες (ECM, ηλεκτροκινητήρες EC)

ή σύγχρονοι κινητήρες συνεχούς ρεύματος, είναι σύγχρονοι κινητήρες που τροφοδοτούνται με ηλεκτρικό ρεύμα DC μέσω μετατροπέα ή τροφοδοτικού που οδηγεί την κάθε φάση του κινητήρα μέσω ενός ελεγκτή κλειστού βρόχου. Ο ελεγκτής παρέχει παλμούς ρεύματος στις περιελίξεις του μοτέρ που ελέγχουν την ταχύτητα και τη ροπή του κινητήρα. Ένα ESC μπορεί να είναι μια αυτόνομη μονάδα που συνδέεται στο κανάλι ελέγχου του, σαν ρυθμιστή γκαζιού του δέκτη ή ενσωματώνεται στον ίδιο τον δέκτη



Σχήμα 14: Brushless DC electric motor

Μερικά Breadboard jumper καλώδια



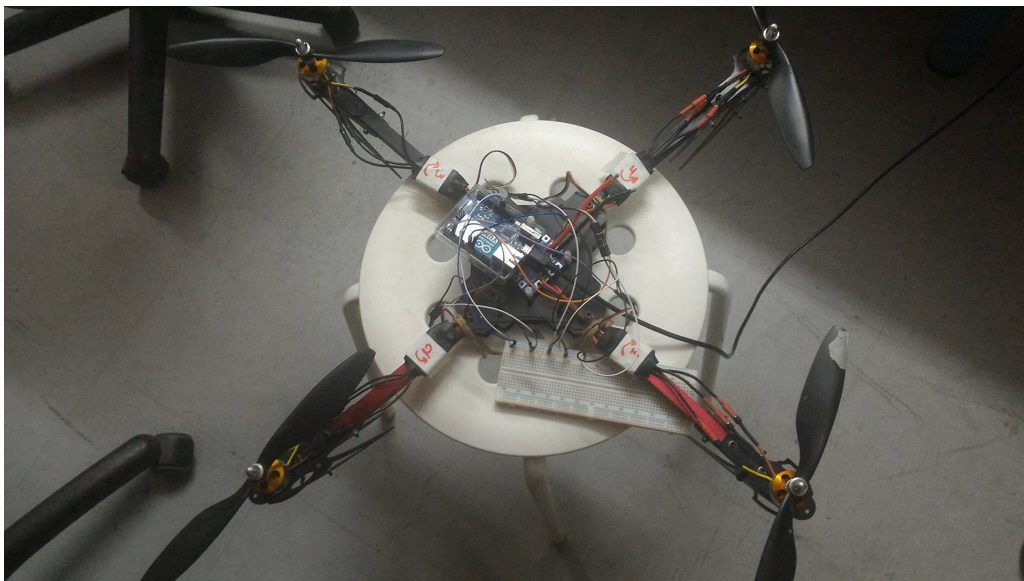
Σχήμα 15: Breadboard jumper καλώδια

Μια Lipo(lithium polymer battery) Μπαταρία Μια μπαταρία πολυμερούς λιθίου, ή πιο σωστά μπαταρία λιθίου-πολυμερούς ιόντων (συντομογραφία LiPo, LIP, Li-poly και άλλα), είναι μια επαναφορτιζόμενη μπαταρία τεχνολογίας ιόντων λιθίου χρησιμοποιώντας έναν πολυμερή ηλεκτρολύτη αντί για ένα υγρό. Υψηλής αγωγιμότητας πολυμερή (gel) που σχηματίζουν αυτόν τον ηλεκτρολύτη. Αυτές οι μπαταρίες παρέχουν μεγαλύτερη συγκεντρωμένη ενέργεια από τους άλλους τύπους μπαταριών λιθίου και χρησιμοποιούνται σε εφαρμογές όπου το βάρος είναι ένα κρίσιμο χαρακτηριστικό.



Σχήμα 16: Lipo Μπαταρία

Τον σκελετό του τετρακοπτέρου Ο συγκεκριμένο σκελετός του τετρακοπτέρου απο το hackerspace.gr, ήταν ένα πάλιο πρότζεκ που προσπάθησαν να αλλάξουν το τηλεχειριστήριο του συγκεκριμένου τετρακοπτέρου με ένα τηλεχειριστήριο xbox. Η τροποποίηση που έγινε στον αρχικό σκελετό ήταν να αντικατασταθούν όλα τα τσιπάκια και οι πλακέτες που είχε και παραμείνουν τα μοτέρ.



Σχήμα 17: Σκελετός του τετρακοπτέρου

4.3 Συνδεσμολογία

Πρέπει να τοποθετήσουμε πάνω στον σκελετό τους 4 Brushless DC κινητήρες και να συνδέσουμε τα 3 καλώδια που έχουν(τάση, σήμα, γείωση) με τα 3 αντίστοιχα των ESC.

Πάνω στον σκελετό υπήρχε μια πλακέτα για να συνδέσουμε την μπαταρία και με ένα "κολλητήρι" συνδέουμε τις παροχές ρεύματος και τις γειώσεις των 4 ESC και έχουμε 2 καλώδια ακροδέκτες που θα συνδεθούν με την Lipo μπαταρία.

Έπειτα συνδέουμε τους 4 ακροδέκτες του Arduino Yun (3, 6, 9, 10) για να δώσουμε

αναλογικό σήμα στα ESC, την τάση του Arduino YUN με την τάση των καλωδίων σήματος των ESC και την γείωση αντίστοιχα.

4.4 Κώδικας

Σε αυτό το κεφάλαιο εξετάζουμε τα πιο σημαντικά σημεία του κώδικα για την κίνηση του τετρακοπτέρου.

Αυτό που πρέπει να προσεχθεί είναι ότι το κάθε είδος ESC έχει δικό του dataset για το πως πρέπει να ρυθμιστεί πριν μπει σε λειτουργία. Και η ρύθμιση του γίνεται μέσω κώδικα.

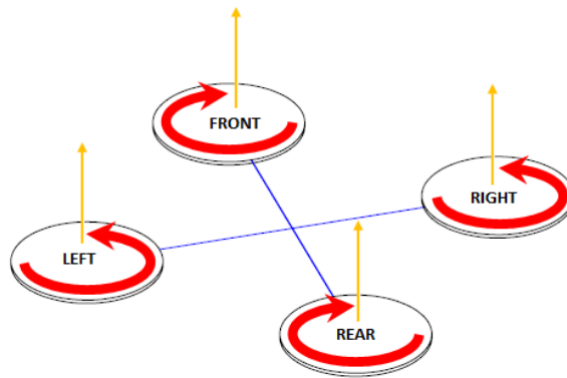
Για να λειτουργήσει το τετρακόπτερο, πρέπει πρώτα να λάβει το σήμα της μέγιστης ταχύτητας, μετά να περιμένουμε 2 δευτερόλεπτα και μετά να του δώσουμε το σήμα της ελάχιστης ταχύτητας.

Αλλιώς το ESC δεν μπορεί να οπλίσει και στείλει το σήμα στα μοτέρ. Αυτό το βλέπουμε στο παράδειγμα παρακάτω για το ένα ESC.

```
void setSpeed(int speed, int numberofservo) {
  int angle = map(speed, 0, 100, 0, 180);
  if(numberofservo == 1 ){
    myservo1.write(angle);
  }
}
void arm() {
  // arm the speed controller, modify as necessary for your ESC
  setSpeed(100,1);
  delay(2000); //delay 2 second, some speed controllers may need longer
  setSpeed(0,1);
  delay(2000); //delay 2 second, some speed controllers may need longer
}
```

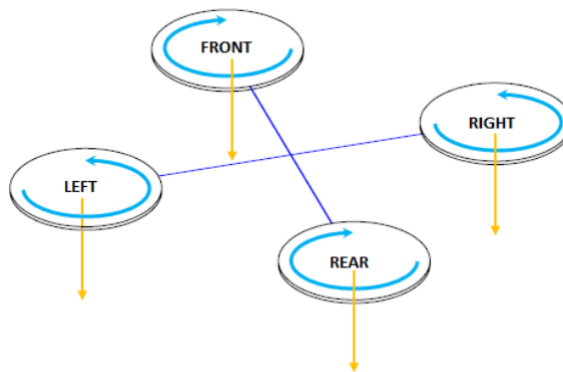
Επίσης η λογική βάση για την πτήση του τετρακοπτέρου είναι ότι για την εντολή "πάνω" όλα τα μοτέρ αυξάνουν την δύναμη τους κατά 5%, αντίστοιχα το ίδιο συμβαίνει και με την εντολή "κάτω".

Στην παρακάτω εικόνα παρατηρούμε το πως κινούνται οι έλικες κατά την απογείωση για να γίνει η μεταβολή της ταχύτητας που επισημίναμε πιο πάνω.



Σχήμα 18: Κίνηση Τετρακπτέρου κατα την απογείωση

Και στην παρακάτω εικόνα παρατηρούμε την προσγείωση.



Σχήμα 19: Κίνηση Τετρακπτέρου κατα την προσγείωση

Για την εντολή "μπροστά" να αυξήσουν την δύναμη των δυο πίσω μοτέρ κατά 5%. Αντίστοιχα το ίδιο ισχύει και για την εντολή "πίσω".

Για την εντολή "αριστερά" να αυξήσουν την δύναμη των δυο δεξιά μοτέρ κατά 5%. Αντίστοιχα το ίδιο ισχύει και για την εντολή "δεξιά".

Στο σημείο αυτό θα δούμε λίγο πιο συγκεκριμένα πως μπορεί να "σηκωθεί" στο Yun ένας web server και πως μπορούμε να καταλάβουμε τα δεδομένα ποιανού αισθητήρα ζητάει το γραφικό περιβάλλον.

Πιο αναλυτικά μέσω του κώδικα (που θα δούμε παρακάτω) κάθε ακροδέκτης μπορεί να έχει ένα συγκεκριμένο URL (πχ `arduinoyun.local/arduino/flame`) που να διαβάζεται η τιμή του αισθητήρα ή να "γράφεται" κάποιος συγκεκριμένος παλμός (πχ στο ESC).

Ένα γενικό κομμάτι κώδικα αυτού παρατηρούμε στο πιο κάτω παράδειγμα που μπορεί στην συνέχεια να προσαρμοστεί από κάθε χρήστη στις ανάγκες του.

```
#include <YunServer.h>
#include <YunClient.h>
YunServer server;
void loop()
```

```
{
  YunClient client = server.accept();
  if (client) {
    String command = client.readString();
    command.trim();      //kill whitespace
    if (command == "arm") {
      do something.....
    }
  }
}
```

5 Μελέτη επικοινωνίας μέσω Wi-Fi

5.1 Εισαγωγή

Σε αυτό το κεφάλαιο της πτυχιακής θα μελετήσουμε την επικοινωνία μέσω Wi-Fi ενός Arduino Yun με μία ιστοσελίδα σε localhost. Σκοπός του βήματος αυτού, είναι να δίνονται εντολές στο τετρακόπτερο μέσω μια ιστοσελίδας, η οποία αποτελεί τον Client και εκείνο (Server) να κινείται αναλόγως. Η επικοινωνία βασίζεται στο REST (REpresentational State Transfer, Κατάσταση παραστατικής μεταφοράς) API που υποστηρίζει το Arduino Yun.

5.2 Κατάσταση παραστατικής μεταφοράς - REpresentational State Transfer (REST)

Το Representational State Transfer (REST) [Paul and Harvey, 2009] αναφέρεται σε ένα αρχιτεκτονικό στυλ για την υλοποίηση Web υπηρεσιών. Τέτοιες Web υπηρεσίες ονομάζονται Web υπηρεσίες RESTful. Αν και το ίδιο το REST δεν είναι πρότυπο, οι Web υπηρεσίες RESTful υλοποιούνται χρησιμοποιώντας Web πρότυπα.

Η βασική έννοια στην αρχιτεκτονική REST είναι εκείνη του πόρου. Ένας πόρος είναι οποιοδήποτε κομμάτι πληροφορίας το οποίο μπορεί να προσδιοριστεί με μοναδικό τρόπο. Στην αρχιτεκτονική REST οι αιτούντες και οι υπηρεσίες ανταλλάσσουν μηνύματα τα οποία συνήθως περιέχουν τόσο δεδομένα όσο και μεταδεδομένα. Το τμήμα των δεδομένων ενός μηνύματος αντιστοιχεί σε μια συγκεκριμένη αναπαράσταση του πόρου, όπως περιγράφεται από τα συνοδευτικά μεταδεδομένα (τη μορφή), τα οποία μπορεί επίσης να περιέχουν πρόσθετες οδηγίες επεξεργασίας. Η ανταλλαγή ενός πόρου μπορεί να γίνει σε πολλές αναπαραστάσεις. Οι δύο επικοινωνούντες συνεργάτες έχουν τη δυνατότητα να συμφωνήσουν εκ των προτέρων σε μια συγκεκριμένη αναπαράσταση.

Οι υπηρεσίες στην αρχιτεκτονική REST δεν διατηρούν την κατάσταση μιας αλληλεπίδρασης με έναν αιτούντα. Δηλαδή αν μια αλληλεπίδραση απαιτεί μια κατάσταση, όλες οι καταστάσεις πρέπει να αποτελούν μέρος των μηνυμάτων που ανταλλάσσονται. Με το να είναι οι υπηρεσίες μη καταστασιακές αυξάνουν την αξιοπιστία τους απλοποιώντας την επεξεργασία της ανάκαμψης (μια υπηρεσία που παρουσίασε βλάβη μπορεί να επανακάμψει γρήγορα). Επιπλέον, τέτοιες υπηρεσίες έχουν βελτιωμένη επεκτασιμότητα, επειδή δεν χρειάζεται να διατηρούν την κατάστασή τους και έτσι δεν καταναλώνουν μνήμη, αντιπροσωπεύοντας ενεργές αλληλεπιδράσεις. Η αξιοπιστία και η επεκτασιμότητα αποτελούν απαραίτητες ιδιότητες του Ιστού. Η συμμόρφωση με το αρχιτεκτονικό ύφος REST επιτρέπει την κάλυψη αυτών των απαιτήσεων.

Η αρχιτεκτονική REST χρησιμοποιεί μια απλή διασύνδεση για το χειρισμό των πόρων κατά ένα γενικό τρόπο. Αυτή η διασύνδεση βασικά υποστηρίζει τη μέθοδο της δημιουργίας, ανάκτησης,

ενημέρωσης και διαγραφής (create, retrieve, update, and delete, CRUD). Τα μεταδεδομένα των αντίστοιχων μηνυμάτων περιλαμβάνουν το όνομα της μεθόδου και το αναγνωριστικό του πόρου στον οποίο στοχεύει η μέθοδος. Εκτός από την περίπτωση της μεθόδου ανάκτησης, το μήνυμα περιλαμβάνει μια αναπαράσταση του πόρου. Επομένως, τα μηνύματα είναι αυτοπεριγραφικά. Τα αναγνωριστικά που περιλαμβάνονται στα μηνύματα είναι θεμελιώδη για τη REST. Υποδηλώνουν δηλαδή περαιτέρω οφέλη αυτού του αρχιτεκτονικού ύφους.

Για παράδειγμα, με το ρητό ορισμό του αναγνωριστικού του πόρου, η REST δίνει τη δυνατότητα για στρατηγικές αποθήκευσης στην κρυφή μνήμη (caching) σε διάφορα επίπεδα και σε κατάλληλους ενδιάμεσους κατά μήκος της διαδρομής των μηνυμάτων. Ένας ενδιάμεσος θα μπορούσε να διαπιστώσει ότι διαθέτει ένα έγκυρο αντίγραφο του πόρου και είναι σε θέση να ικανοποιήσει μία αίτηση ανάκτησης χωρίς να μεταβιβάσει περαιτέρω την αίτηση. Αυτό συμβάλλει στην επεκτασιμότητα του γενικού περιβάλλοντος. Κάθε μέθοδος σε μια RESTful Web υπηρεσία προσδιορίζεται από ένα μοναδικό URL. Κατά συνέπεια, όταν ο διακομιστής λαμβάνει ένα αίτημα, ξέρει αμέσως ποια λειτουργία να εκτελέσει. Τέτοιες Web υπηρεσίες μπορούν να χρησιμοποιηθούν σε ένα πρόγραμμα ή άμεσα από έναν Web browser.

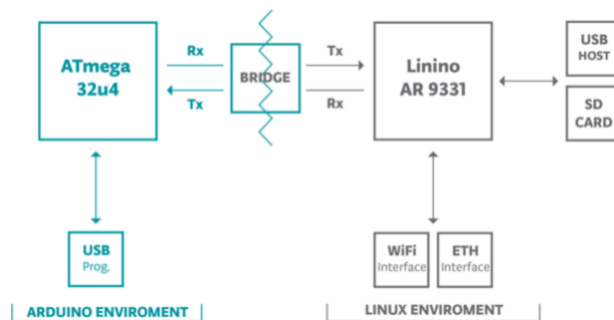
5.3 Λειτουργία Arduino Yun

Το Arduino Yun είναι το πρώτο μέλος μια πρωτοποριακής γραμμής προϊόντων με Wi-Fi που συνδυάζουν τη δύναμη του Linux με την ευκολία χρήσης του Arduino. Το πρώτο Arduino Yun βασίζεται σε έναν ATmega32u4 επεξεργαστή και τον Atheros AR9331. Ο επεξεργαστής Atheros υποστηρίζει μια διανομή Linux βασισμένη στο OpenWrt που ονομάζεται OpenWrt-Yun. Η πλακέτα έχει ενσωματωμένη υποστήριξη Ethernet και Wi-Fi, μια θύρα USB-A, υποδοχή κάρτας micro-SD, 20 ψηφιακούς ακροδέκτες εισόδου / εξόδου (εκ των οποίων 7 είναι δυνατόν να χρησιμοποιηθούν ως έξοδοι PWM και 12 ως αναλογικές εισόδους), έναν 16 MHz κρύσταλλο ταλαντωτή, μία micro USB σύνδεση, μια κεφαλή ICSP και 3 κουμπιά επαναφοράς.

Το Yun ξεχωρίζει από τις άλλες πλακέτες Arduino στο ότι μπορεί να επικοινωνήσει με την διανομή Linux, προσφέροντας ένα ισχυρό δικτυωμένο υπολογιστή με την ευκολία του Arduino. Εκτός από τις εντολές Linux, όπως η cURL, μπορεί να γραφτεί και κώδικας σε python. Το Yun είναι παρόμοιο με το Arduino Leonardo στον επεξεργαστή ATmega32u4 που έχει ενσωματωμένη σύνδεση με USB εξαλείφοντας έτσι την ανάγκη για ένα δεύτερο επεξεργαστή. Αυτό επιτρέπει στο Yun να εμφανιστεί σε έναν συνδεδεμένο υπολογιστή ως ποντίκι, πληκτρολόγιο ή ως virtual (CDC) serial / COM port.

Ιστορικά, η διασύνδεση του Arduino με πολύπλοκα web services ήταν μια πρόκληση λόγω της περιορισμένης διαθέσιμης μνήμης. Τα web services τείνουν να χρησιμοποιούν λεπτομερείς μορφές κειμένου, όπως XML, οι οποίες απαιτούν αρκετή RAM για να αναλυθούν. Στο Arduino Yun υπάρχει η βιβλιοθήκη Bridge, η οποία μεταβιβάζει όλες τις συνδέσεις δικτύου και επεξεργασίας των συναλλαγών HTTP στο Linux επεξεργαστή. Δηλαδή, η βιβλιοθήκη Bridge διευκολύνει την επικοινωνία μεταξύ των δύο επεξεργαστών δίνοντας έτσι τη δυνατότητα σε

προγράμματα του Arduino να τρέξουν shell scripts, να επικοινωνούν με διασυνδέσεις δικτύου και να λαμβάνουν πληροφορίες από τον επεξεργαστή AR9331.



Σχήμα 20: Επικοινωνία μεταξύ των δύο επεξεργαστών μέσω της βιβλιοθήκης Bridge

5.4 Σύνδεση του Arduino Yun με το Wi-Fi

Για να συνδέσουμε το Arduino Yun στο Wi-Fi πρέπει πρώτα να το συνδέσουμε στο υπολογιστή μας μέσω της θύρας USB. Όταν συνδεθεί θα πρέπει να περιμένουμε λίγο μέχρι ο linux επεξεργαστής ξεκινήσει και να το θέσει σαν ένα access point. Έπειτα επιλέγουμε από τον υπολογιστή μας το δίκτυο Wi-Fi που δημιουργείται και συνδεόμαστε σε αυτό.

Μέτα μέσα από ένα browser ανοίγουμε την ιστοσελίδα <http://arduinoyun.local/> είτε πατάμε το ip που συνήθως είναι 192.168.240.1.

Στην συνέχεια, μας εμφανίζεται ένα παράθυρο που από εκεί πραγματοποιούμε τις ρυθμίσεις που θέλουμε όπως αλλαγή ονόματος, σύνδεση στο οικιακό ρούτερ, προσθήκη/αφαίρεση κωδικού κ.α.

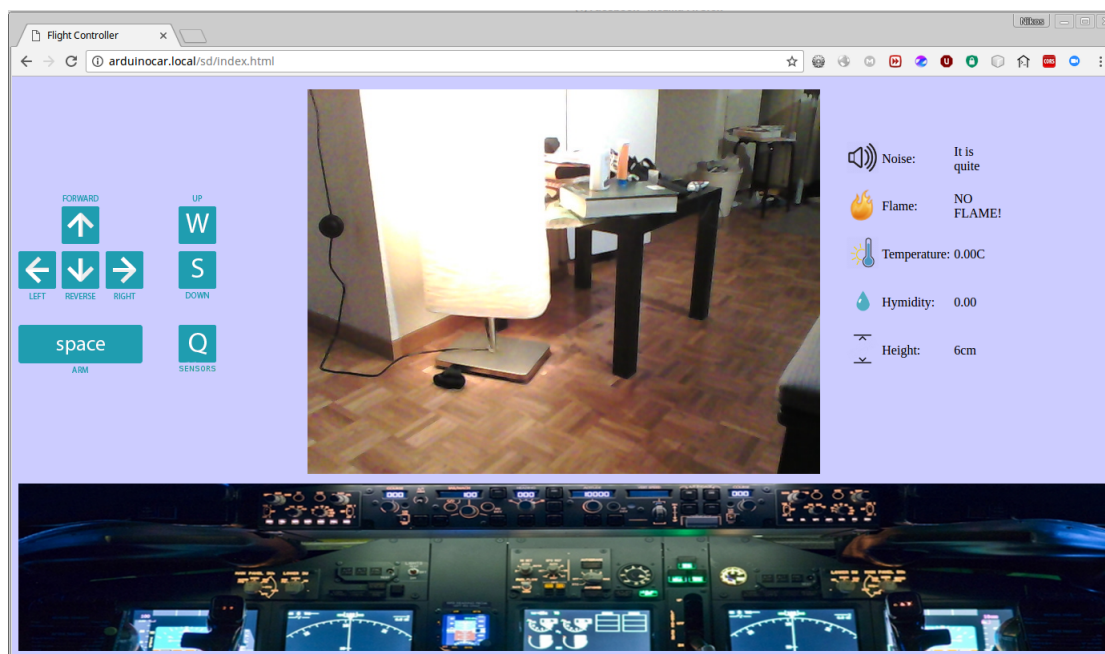
6 Μελέτη γραφικού περιβάλλοντος πλοήγησης του τετρακοπτέρου

6.1 Εισαγωγή

Σε αυτό το κεφάλαιο περιγράφεται το γραφικό περιβάλλον που υπάρχει για την πλοήγηση του τετρακοπτέρου. Το Arduino Yun μπορεί να πάρει στο κύκλωμα του μια sd card που μέσα στην sd υπάρχει το html αρχείο για την πλοήγηση του τετρακοπτέρου. Όπως αναφέραμε και πιο πάνω το Yun λειτουργεί σαν ένας web server όπου έχει μια συγκεκριμένη ip και μπορεί και “σερβίρει” το περιεχόμενο του html αρχείο στον χρήστη.

Το αρχείο καταλαβαίνει τα πλήκτρα που πατάει ο χρήστης απο τον browser και στέλνει την ανάλογη εντολή στο τετρακόπτερο. Ακόμα, το τετρακόπτερο στέλνει πίσω στο χρήστη την τιμή των δεδομένων που έχουν οι αισθητήρες μέσω του REST API που υπάρχει στο Yun.

Επειδή υπήρχε μεγάλη καθυστέρηση στην λήψη δεδομένων, όταν οι αισθητήρες έστελναν συνέχεια τα δεδομένα τους και δεν μπορούσε να λάβει το Arduino τις εντολές πλοήγησης, για το λόγο οτι δεν είναι πολύ-νηματικό και είναι σειριακό (δηλαδή κάθε νέα εντολή μπαίνει σε μια λίστα και εκτελεί την πρώτη στην σειρά), υπάρχει η επιλογή να πατάει ο χρήστης το κουμπί “Q” για όταν θέλει να πάρει δεδομένα. Οπότε αυτό μείωσε αισθητά την καθυστέρηση που υπήρχε στην πλοήγηση του.



Σχήμα 21: Γραφικό περιβάλλον του χρήστη για την πλοήγηση του τετρακοπτέρου

7 Μελέτη Web-Camera

7.1 Εισαγωγή

Το κεφάλαιο αυτό αφορά την προσθήκη της ψηφιακής κάμερας με την οποία γίνεται ζωντανή μετάδοση στην ιστοσελίδα (client).

Για να μπορέσει να λειτουργήσει η κάμερα πρέπει πρώτα να εγκατασταθούν κάποια πακέτα λογισμικού. Έπειτα μέσω μιας σύνδεσης ssh με την πλακέτα, στον linux επεξεργαστή, “τρέχουμε” την παρακάτω εντολή, έτσι ώστε να εγκατασταθούν κάποιοι οδηγοί UVC.

```
opkg install kmod-video-uvnc
```

Στη συνέχεια, η παρακάτω εντολή εγκαθιστά τη βιβλιοθήκη fswebcam με την οποία μπορεί εύκολα οποιοσδήποτε να τραβήξει φωτογραφία από το τερματικό σε περίπτωση που χρειαστεί στο μέλλον.

```
opkg install fswebcam
```

Κατόπιν, εγκαθιστούμε τη βιβλιοθήκη mjpg streaming για να μπορέσουμε να κάνουμε τη ζωντανή μετάδοση με την ακόλουθη εντολή:

```
opkg install mjpg-streamer
```

Μετά από αυτό, λόγω του πακέτου που εγκαταστάθηκε είναι εύκολο να γίνει streaming τοπικά, δίνοντας την παρακάτω εντολή στον επεξεργαστή linux:

```
mjpg_streamer -i "input_uvc.so -d /dev/video0 -r 640x480 -f 25" -o "output_http.so  
-p 8080 -w /www/webcam" &.
```

Το γνώρισμα -r αρχικοποιεί την ανάλυση του βίντεο (χρησιμοποιείται η χαμηλότερη ανάλυση γιατί σε υψηλότερη το βίντεο καθυστερούσε), ενώ το γνώρισμα -p αρχικοποιεί τη θύρα στην οποία θα είναι διαθέσιμο το stream.

7.2 Υλικά

Μια κάμερα Creative που την συνδέται μέσω του USB port του Arduino Yun.



Σχήμα 22: Κάμερα Creative

Το Arduino Yun υποστηρίζει σίγουρα κάθε Logitech Webcam και ένα πλήθος από τις Creative. Μέχρι στιγμής έχουν δοκιμάσει 4 διαφορετικά μοντέλα από κάμερες και όλες δούλεψαν σε ικανοποιητικό βαθμό, ωστόσο δεν έχει δοκιμαστεί άλλη κάμερα διαφορετικής εταιρίας από αυτές τις δύο που αναφέρονται, αλλά υπάρχει σαν μελλοντικός στόχος δοκιμής άλλης κάμερας, μήπως και χρειάζονται παραπάνω ενέργειες για την λειτουργία τους.

Στην αρχή χρησιμοποιήθηκε μια κάμερα GoPro, αλλά είναι πιο βαριά από μια webcam και πρέπει να συνδεθεί και αυτή μέσω του Wi-Fi, έτσι ώστε να έχουμε την εικόνα ξεχωριστά σε ένα άλλο τάμπλετ ή PC. Αυτός ήταν ο κύριος λόγος της επιλογής χρήσης της webcam για να είναι όλα ενσωματωμένα στην ιστοσελίδα και ο χρήστης να μην χρειάζεται να έχει πολλές ηλεκτρονικές συσκευές.

8 Μελέτη αισθητήρων για την συλλογή δεδομένων

8.1 Εισαγωγή

Ο έλεγχος και η εύκολη συνδεσμολογία των αισθητήρων είναι ο λόγος ύπαρξης του Arduino. Ο κύριος λόγος που των αισθητήρων στο τετρακόπτερο είναι να δούμε πως θα λειτουργούσε και να είναι πιο διαδραστικό.

Επίσης, όπως έχει αναφερθεί και στην αρχή της εργασίας, όλοι οι αισθητήρες έχουν σχετικά την ίδια συνδεσμολογία και λαμβάνουν τις μετρήσεις τους είτε σε ψηφιακή είτε σε αναλογική μορφή είτε και στα 2.

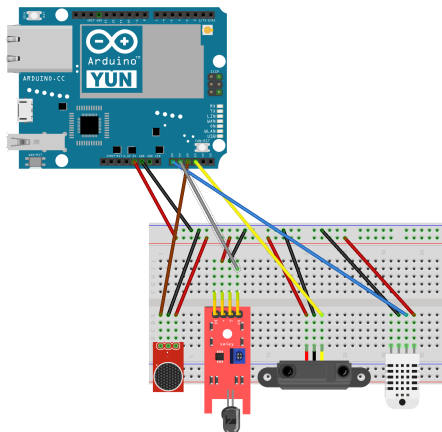
Παρακάτω θα μιλήσουμε λίγο πιο αναλυτικά για την συνδεσμολογία των αισθητήρων που χρησιμοποιήθηκαν όπως και για κάποια γενικά σημεία του κώδικα ώστε να γίνει αντιληπτό το πόσο εύκολο είναι πλέον να δημιουργούμε μικρά κυκλώματα.

8.2 Συνδεσμολογία

Στο παρακάτω σχήμα βλέπουμε τους αισθητήρες που έχουν συνδεθεί στο κύκλωμα μας. Το κόκκινο χρώμα των καλωδίων υποδηλώνει το καλώδιο της τροφοδοσίας(τάση) του κυκλώματος και με το μαύρο την γείωση. Τα πολύχρωμα καλώδια υποδηλώνουν το καλώδιο που μεταφέρετε το σήμα από τον αισθητήρα στο Arduino.

Οι αισθητήρες που παρατηρούμε από αριστερά προς δεξιά είναι : αισθητήρας ήχου, αισθητήρας φωτιάς, αισθητήρας απόστασης και αισθητήρας (DHT11) που είναι υγρασίας και θερμοκρασίας.

Οι τιμές των δεδομένων που λαμβάνει το Arduino από τους συνδεδεμένους ακροδέκτες του κυκλώματος μεταφέρονται στον linux επεξεργαστή και μέσω του Rest Api που έχει υλοποιηθεί σε αυτό μπορούν να εμφανιστούν στο γραφικό παράθυρο.



Σχήμα 23: Συνδεσμολογία του Arduino με τους αισθητήρες

8.3 Κώδικας

Μια γενική περιγραφή για να υλοποιήσεις μια λειτουργία ή να πάρεις δεδομένα από το τους αισθητήρες που είναι "καρφίτσωμένοι" στο Arduino περιγράφεται απο τον παρακάτω κώδικα.

```
#include <dht.h>
#define dht_apin A0 // Analog Pin sensor is connected to
dht DHT;

void setup(){
    Serial.begin(9600);
} //end "setup()"

void loop(){
    //Start of Program
    DHT.read11(dht_apin);
    Serial.print("temperature = ");
    Serial.print(DHT.temperature);
    Serial.println("C ");
} // end loop()
```

Δηλαδή η λογική είναι να γίνει include η βιβλιοθήκη του αισθητήρα, να δηλωθεί σε ποίον ακροδέκτη είναι συνδεδεμένο το καλώδιο σήματος. Στην setup() να δηλώσεις το κανάλι επικοινωνίας είτε να αρχικοποιηθούν κάποια πράγματα. Τέλος στην loop() να εκτελείται συνεχώς η ενέργεια που χρειαζόμαστε.

9 Συμπεράσματα και μελλοντικές επεκτάσεις

Τα συμπεράσματα που βγήκαν μετα απο αυτή την πολύμηνη εργασία είναι ότι με το Arduino είναι σχετικά εύκολο να υλοποιηθούν πολλά prototype πρότζεκ και με την βοήθεια της κοινότητας αλλά και του φόρουμ λύνονται αρκετά προβλήματα που κάθε αρχάριος χρήστης θα συναντήσει.

Όσο αφορά την πτήση του τετρακοπτέρου η καθυστέρηση 5-7 δευτερολέπτων δεν είναι βολική για μια άνετη πλοήγηση, θα έπρεπε πολλά πράγματα να χωριστούν και σε άλλους μικρο-ελεγκτές κάποια απο κομμάτια της διαχείρισης του τετρακοπτέρου, πχ να υπάρχει ένα arduino για την πλοήγηση και ένα για να μαζεύει τα δεδομένα από τους αισθητήρες, η κάμερα να είναι συνδεδεμένη σε ένα τάμπλετ ή κινητό τηλέφωνο για να μην τρώει πολύ bandwidth της επικοινωνίας του Arduino με τον browser και να είχαμε σαν αποτελέσματα καλύτερες ταχύτητες απόκρισης και στο τετρακόπτερο αλλά και στην εικόνα με τους αισθητήρες.

Επίσης, οι παρατηρήσεις μου είναι οτι το Arduino Yun είναι αρκετά βαρύ, για μικρού σκέλους τετρακόπτερο και θα υπάρχει μεγάλη δυσκολία στο να πετάξει ή οτι θα πετάει πολύ κοντά και θα καταναλώνει όλη την ισχύ της μπαταρίας προσπαθώντας να δώσει παραπάνω κίνηση στους έλικες. Στα τελικά πειράματα αποδείχθη οτι είναι προτιμότερο να φτιαχτεί κάτι με ένα ESP8266 που έχει το 1/3 το βάρος του Yun και κάνουν σχεδόν την ίδια δουλειά, απλώς απαιτεί περισσότερο χρόνο να αφιερώσει κανείς γιατί θέλει ειδικές ρυθμίσεις και βιβλιοθήκες να προστεθούν στο Arduino IDE και θα χρειαστεί τελείως διαφορετική διαδικασία στην ρύθμιση του με το Wi-Fi (πρέπει όλα να είναι γραμμένα στο κώδικα).

Επιπλέον, υπάρχει και το Ardupilot που είναι Arduino πλακέτα για να πιλοτάρεις τετρακόπτερα και αεροπλανάκια τηλεκατευθυνόμενα, επειδή όμως έχει πολλά έτοιμα modules και ήθελα να εντρυφήσω σε πιο πολλά πράγματα πάνω στην διαχείριση της πλοήγησης του τετρακοπτέρου, ώστε να είναι πιο απλοϊκό προς τα εμένα, δεν χρησιμοποιήθηκε σε αυτή την εργασία, αλλά συνιστάται ανεπιφύλακτα για βελτίωση του τετρακοπτέρου στο μέλλον.

Επιπλέον, η σκέψη μου είναι ότι με τα δεδομένα που θα συγκεντρώθουν και με την "εξόρυξη" τους, να σχεδιαστούν κάπως καλύτεροι αλγόριθμοι για να γίνονται πιο αυτοματοποιημένες ενέργειες του τετρακοπτέρου και να μπορέσει να χρησιμοποιηθεί απο διάφορες δράσεις, όπως ανίχνευση θυμάτων σε ένα σεισμό ή ανίχνευση μιας πυρκαγιάς σε ένα πυκνό δάσος κ.α. Μιας που θα έχει το προτέρημα να κινείται από αέρα έχοντας καλύτερη εικόνα του περιβάλλοντος, μπορεί να χρησιμοποιηθεί και να αναλυθεί ακόμα ηχητικό σήμα για άμα ζητάει κάποιος βοήθεια, ώστε να μπορεί αυτόματα να ενημερώνει τον χειριστή για την τοποθεσία του.

Αναφορές

- [Achtelik et al., 2009] Achtelik, M., Zhang, T., Kuhnlenz, K., and Buss, M. (2009). Visual tracking and control of a quadcopter using a stereo camera system and inertial sensors. In *Mechatronics and automation, 2009. icma 2009. international conference on*, pages 2863--2869. IEEE.
- [Anderson, 1997] Anderson, S. (1997). Historical overview of v/stol aircraft technology. *NASA Technical Memorandum*.
- [Azfar and Hazry, 2011] Azfar, A. Z. and Hazry, D. (2011). Simple gui design for monitoring of a remotely operated quadcopter unmanned aerial vehicle. In *2011 IEEE 7 th International Colloquium on Signal Processing and its Applications*.
- [Benett, 1979] Benett, S. (1979). A history of control engineering.
- [Chen et al., 2013] Chen, M. Y., Edwards, D. H., Boehmer, E. L., Eller, N. M., Slack, J. T., Speck, C. R., Brown, S. M., Williams, H. G., Wilson, S. H., Gillum, C. S., et al. (2013). Designing a spatially aware and autonomous quadcopter. In *Systems and Information Engineering Design Symposium (SIEDS), 2013 IEEE*, pages 213--218. IEEE.
- [Hanafi et al., 2013] Hanafi, D., Qetkeaw, M., Ghazali, R., Than, M. N. M., Utomo, W. M., and Omar, R. (2013). Simple gui wireless controller of quadcopter. *International Journal of Communications, Network and System Sciences*, 6(1):52.
- [Hoffmann,] Hoffmann, G. The stanford testbed of autonomous rotorcraft for multi agent control (starmac). In *the Proceedings of the 23rd Digital Avionics System Conference*.
- [Kushner, 2011] Kushner, D. (2011). The making of arduino. *IEEE Spectrum*.
- [Leishman, 2006] Leishman, G. J. (2006). *Principles of helicopter aerodynamics with CD extra*. Cambridge university press.
- [Leong et al., 2012] Leong, B. T. M., Low, S. M., and Ooi, M. P.-L. (2012). Low-cost microcontroller-based hover control design of a quadcopter. *Procedia Engineering*, 41:458--464.
- [Li and Li, 2011] Li, J. and Li, Y. (2011). Dynamic analysis and pid control for a quadrotor. In *Mechatronics and Automation (ICMA), 2011 International Conference on*, pages 573--578. IEEE.
- [Margolis, 2011] Margolis, M. (2011). *Arduino Cookbook: Recipes to Begin, Expand, and Enhance Your Projects*. " O'Reilly Media, Inc."
- [Mian and Daobo, 2008] Mian, A. A. and Daobo, W. (2008). Nonlinear flight control strategy for an underactuated quadrotor aerial robot. In *Networking, Sensing and Control, 2008. ICNSC 2008. IEEE International Conference on*, pages 938--942. IEEE.

- [Park et al., 2001] Park, D., Park, M.-S., and Hong, S.-K. (2001). A study on the 3-dof attitude control of free-flying vehicle. In *Industrial Electronics, 2001. Proceedings. ISIE 2001. IEEE International Symposium on*, volume 2, pages 1260--1265. IEEE.
- [Paul and Harvey, 2009] Paul, D. and Harvey, D. (2009). How to program eighth edition. *Prentice Hall 8 edition*.
- [Ryan and Hedrick, 2005] Ryan, A. and Hedrick, J. K. (2005). A mode-switching path planner for uav-assisted search and rescue. In *Decision and Control, 2005 and 2005 European Control Conference. CDC-ECC'05. 44th IEEE Conference on*, pages 1471--1476. IEEE.
- [Sajid et al., 2016] Sajid, M., Yang, Y. J., Kim, G. B., and Choi, K. H. (2016). Remote monitoring of environment using multi-sensor wireless node installed on quad-copter drone. In *Robotics and Intelligent Sensors (IRIS), 2016 IEEE International Symposium on*, pages 213--216. IEEE.
- [Salih et al., 2010] Salih, A. L., Moghavvemi, M., Mohamed, H. A., and Gaeid, K. S. (2010). Flight pid controller design for a uav quadrotor. *Scientific Research and Essays*, 5(23):3660--3667.
- [Santos et al., 2010] Santos, M., Lopez, V., and Morata, F. (2010). Intelligent fuzzy controller of a quadrotor. In *Intelligent Systems and Knowledge Engineering (ISKE), 2010 International Conference on*, pages 141--146. IEEE.
- [Stafford, 2015] Stafford, J. (2015). How a quadcopter works| clay allen. *University of Alaska, Fairbanks. Retrieved*, pages 01--20.
- [Weng, 2011] Weng, K. W. (2011). Quadcopter. *Robot Head to Toe Magazine*, 10:1--3.

Λίγκ για τον κώδικα: <https://github.com/NickKok/Quadcopter>