



ΠΑΝΕΠΙΣΤΗΜΙΟ ΔΥΤΙΚΗΣ ΑΤΤΙΚΗΣ
ΣΧΟΛΗ ΜΗΧΑΝΙΚΩΝ
ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΥΠΟΛΟΓΙΣΤΩΝ

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

Ανάπτυξη ρομποτικού οχήματος κινούμενου με ανίχνευση γραμμής

Θεοδώρα Δ. Νινίγια

Εισηγητής: Δρ Ιωάννης Έλληνας, Καθηγητής

ΑΘΗΝΑ
ΣΕΠΤΕΜΒΡΙΟΣ 2018

Ανάπτυξη ρομποτικού οχήματος κινούμενου με ανίχνευση γραμμής

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

Σχεδίαση ολοκληρωμένου συστήματος συγγραφής πτυχιακής εργασίας

**Θεοδώρα Δ. Νινίγια
Α.Μ. 43343**

Εισηγητής:

Εισηγητής: Δρ Ιωάννης Έλληνας, Καθηγητής

Εξεταστική Επιτροπή:

Ημερομηνία εξέτασης

Ανάπτυξη ρομποτικού οχήματος κινούμενου με ανίχνευση γραμμής

ΔΗΛΩΣΗ ΣΥΓΓΡΑΦΕΑ ΠΤΥΧΙΑΚΗΣ ΕΡΓΑΣΙΑΣ

Η κάτωθι υπογεγραμμένη Νινίγια Θεοδώρα του Δημητράκ, με αριθμό μητρώου 43343 φοιτήτρια του Τμήματος Μηχανικών Η/Υ Συστημάτων Τ.Ε. του Α.Ε.Ι. Πειραιά Τ.Τ. πριν αναλάβω την εκπόνηση της Πτυχιακής Εργασίας μου, δηλώνω ότι ενημερώθηκα για τα παρακάτω:

«Η Πτυχιακή Εργασία (Π.Ε.) αποτελεί προϊόν πνευματικής ιδιοκτησίας τόσο του συγγραφέα, όσο και του Ιδρύματος και θα πρέπει να έχει μοναδικό χαρακτήρα και πρωτότυπο περιεχόμενο.

Απαγορεύεται αυστηρά οποιοδήποτε κομμάτι κειμένου της να εμφανίζεται αυτούσιο ή μεταφρασμένο από κάποια άλλη δημοσιευμένη πηγή. Κάθε τέτοια πράξη αποτελεί προϊόν λογοκλοπής και εγείρει θέμα Ηθικής Τάξης για τα πνευματικά δικαιώματα του άλλου συγγραφέα. Αποκλειστικός υπεύθυνος είναι ο συγγραφέας της Π.Ε., ο οποίος φέρει και την ευθύνη των συνεπειών, ποινικών και άλλων, αυτής της πράξης.

Πέραν των όποιων ποινικών ευθυνών του συγγραφέα σε περίπτωση που το Ίδρυμα του έχει απονείμει Πτυχίο, αυτό ανακαλείται με απόφαση της Συνέλευσης του Τμήματος. Η Συνέλευση του Τμήματος με νέα απόφασής της, μετά από αίτηση του ενδιαφερόμενου, του αναθέτει εκ νέου την εκπόνηση της Π.Ε. με άλλο θέμα και διαφορετικό επιβλέποντα καθηγητή. Η εκπόνηση της εν λόγω Π.Ε. πρέπει να ολοκληρωθεί εντός τουλάχιστον ενός ημερολογιακού δμήνου από την ημερομηνία ανάθεσης της. Κατά τα λοιπά εφαρμόζονται τα προβλεπόμενα στο άρθρο 18, παρ. 5 του ισχύοντος Εσωτερικού Κανονισμού.»

Ανάπτυξη ρομποτικού οχήματος κινούμενου με ανίχνευση γραμμής

ΕΥΧΑΡΙΣΤΙΕΣ

Η παρούσα πτυχιακή εργασία ολοκληρώθηκε μετά από επίμονες προσπάθειες, σε ένα ενδιαφέρον γνωστικό αντικείμενο της δημιουργίας ενός ρομποτικού οχήματος με χρήση Arduino καθώς και την εύρεση βέλτιστης απόδοσης του με την βοήθεια των ειδικά προσαρμοσμένων αποκωδικοποιητών στους κινητήρες του. Την προσπάθειά μου αυτή υποστήριξε ο επιβλέπων καθηγητής μου, Ιωάννης Έλληνας τον οποίο θα ήθελα να ευχαριστήσω.

Ανάπτυξη ρομποτικού οχήματος κινούμενου με ανίχνευση γραμμής

ΠΕΡΙΛΗΨΗ

Η παρούσα πτυχιακή εργασία ασχολείται με την ανάπτυξη ενός δίτροχου ρομποτικού οχήματος με βοηθητική ρόδα (ball caster) το οποίο αποσκοπεί στην κίνηση του στον χώρο ακολουθώντας μια μαύρη γραμμή βάσει αισθητήρων ανίχνευσης QTR-8RC (Reflectance Sensor Array) που βρίσκονται στο μπροστινό τμήμα της κατασκευής μας. Η κατασκευή βασίστηκε πάνω σε Arduino μικροελεγκτή. Αρχικά αναφέρονται τα βασικά χαρακτηριστικά μιας πλακέτας Arduino, διάφορες εφαρμογές του καθώς και η παρουσίαση της κατασκευής μου. Εν συνεχεία, γίνεται μια εισαγωγή στους κωδικοποιητές κινητήρων (motor encoders), τα είδη που υπάρχουν καθώς και την θεωρητική εφαρμογή τους στους DC κινητήρες της κατασκευής μου. Στο τρίτο μέρος, παρουσιάζεται η υλοποίηση του Line Follower οχήματος. Κλείνοντας, παραθέτω τα συμπεράσματά μου καθώς και τις προοπτικές εξέλιξης της κατασκευής μου.

ABSTRACT

This paper examines the development of a two wheel robot vehicle with an extra supplementary ball caster in front of my vehicle which aims for the balanced motion of my construction along a black line which can be detected from an Reflectance Sensor Array QTR-8RC. This construction is based on an Arduino platform. To begin with there is a brief presentation of the main features of Arduino microcontroller with some basic implementations of it and also the presentation of our Arduino project. Furthermore an introduction to motor encoders is being implemented and a variety of it's models with their specified use is presented. Additionally I analyse the procedure of my Line Follower Arduino project. Finally I give the conclusion and prospects of our development.

ΕΠΙΣΤΗΜΟΝΙΚΗ ΠΕΡΙΟΧΗ: Μικροελεγκτές και Encoders

ΛΕΞΕΙΣ ΚΛΕΙΔΙΑ: μικροελεγκτής, line follower, κωδικοποιητές κινητήρων

Πίνακας περιεχομένων

ΚΕΦΑΛΑΙΟ 1	12
ΕΙΣΑΓΩΓΗ.....	12
1.1 Περιγραφή του αντικειμένου της πτυχιακής εργασίας.....	12
1.2 Ιστορική αναδρομή.....	16
1.2.1 Ιστορική αναδρομή Arduino.....	16
1.2.2 Ιστορική αναδρομή Encoder.....	17
1.3 Ανασκόπηση της πτυχιακής εργασίας.....	18
ΚΕΦΑΛΑΙΟ 2	19
ΜΙΚΡΟΕΛΕΓΚΤΗΣ ARDUINO.....	19
2.1 Εισαγωγή.....	19
2.1.1 Εισαγωγή στους μικροελεγκτές.....	19
2.1.2 Εισαγωγή στον Arduino.....	20
2.2 Εκδόσεις του Arduino.....	20
2.3 Arduino Shields.....	25
2.4 Χαρακτηριστικά του Arduino.....	27
2.5 Παρουσίαση της κατασκευής μου.....	30
ΚΕΦΑΛΑΙΟ 3	36
ΚΩΔΙΚΟΠΟΙΗΤΕΣ (ENCODERS).....	36
3.1 Εισαγωγή.....	36
3.1.1 Εισαγωγή στους κωδικοποιητές.....	36
3.2 Χαρακτηριστικά περιστροφικού κωδικοποιητή.....	38
3.3 Χαρακτηριστικά γραμμικού κωδικοποιητή.....	40
3.4 Σφάλμα κωδικοποίησης.....	44
ΚΕΦΑΛΑΙΟ 4	45
ΥΛΟΠΟΙΗΣΗ ΚΑΤΑΣΚΕΥΗΣ.....	45
4.1 Εξαρτήματα.....	45
4.1.1 Micro metal gearmotor bracket.....	45
4.1.2 Wheel kit 70x8mm Pololu.....	46
4.1.3 Plastic Ball caster.....	47
4.1.5 Reflectance sensor array QTR-8RC Pololu.....	47
4.1.6 Magnetic encoder pair kit for micro metal gearmotors 12CPR 2.7.....	48
4.1.7 L293D motor driver.....	49
4.1.1 Micro gear motor.....	50
4.2 Διαφορές βηματικού κινητήρα και σερβοκινητήρα.....	51
4.3 Συνδεσμολογία εξαρτημάτων.....	52
ΚΕΦΑΛΑΙΟ 5	53
ΣΥΜΠΕΡΑΣΜΑΤΑ ΚΑΙ ΠΡΟΟΠΤΙΚΕΣ.....	53
5.1 Σύνοψη της πτυχιακής εργασίας.....	53
5.2 Προοπτικές.....	54
ΒΙΒΛΙΟΓΡΑΦΙΑ	58

Ευρετήριο πινάκων

Πίνακας 1: Χαρακτηριστικά Arduino Uno.....	25
Πίνακας 2: Ακροδέκτες τροφοδοσίας.....	27

Ευρετήριο εικόνων

Εικόνα 1: Δίτροχο αυτοκινητάκι.....	13
Εικόνα 2: Προφίλ δίτροχου line follower.....	13
Εικόνα 3: Arduino Logo.....	14
Εικόνα 4: Arduino Uno.....	19
Εικόνα 5: Arduino Leonardo.....	20
Εικόνα 6: Arduino Esplora.....	20
Εικόνα 7: Arduino Mega 2560.....	21
Εικόνα 8: Arduino Yun.....	22
Εικόνα 9: Arduino LilyPad.....	22
Εικόνα 10: Arduino Due.....	23
Εικόνα 11: Arduino Motor Shield.....	25
Εικόνα 12: Ψηφιακοί ακροδέκτες του Arduino Uno.....	26
Εικόνα 13: Αναλογικά pins Arduino Uno.....	27
Εικόνα 14: Ομαδοποίηση κωδικοποιητών.....	33
Εικόνα 15: Rotary encoder.....	34
Εικόνα 16: Linear magnetic encoder.....	35
Εικόνα 17: Optical rotary encoder.....	36
Εικόνα 18: Pololu magnetic encoders.....	37
Εικόνα 19: Τοποθέτηση πηνίων.....	38
Εικόνα 20: Έξοδος τετραγώνων για έναν αυξητικό κωδικοποιητή. Τα κανάλια A και B είναι εκτός φάσης κατά 90 μοίρες. Το κανάλι Z έχει έναν παλμό, ο οποίος χρησιμοποιείται ως αναφορά για την προσπέλαση.....	40
Εικόνα 21: Οπτικός γραμμικός κωδικοποιητής.....	40
Εικόνα 22: Metal gearmotor bracket.....	42
Εικόνα 23: Pololu wheels.....	43
Εικόνα 24: Ball caster.....	44
Εικόνα 25: Reflectance sensor array QTR-8RC.....	45
Εικόνα 26: Magnetic encoder pair kit for micro metal gearmotors 12CPR 2.7.....	45
Εικόνα 27: L293D shield.....	46
Εικόνα 28: Micro gearmotor.....	47
Εικόνα 29: Συνδεσμολογία Arduino-L293D-Motors.....	49
Εικόνα 30: QTR-8RC Reflectance Sensor Array pins.....	49
Εικόνα 31: 9V Battery Connector to Barrel Jack Plug.....	50
Εικόνα 32: Έξοδος Magnetic Encoder.....	51
Εικόνα 33: Motor encoder wiring.....	52
Εικόνα 34: Motor encoder installation.....	53

ΣΥΝΤΟΜΟΓΡΑΦΙΕΣ

PID PROPORTIONAL INTEGRAL DERIVATE

ROM READ ONLY MEMORY

CPU CENTRAL PROCCESING UNIT

EEPROM ELECTRICALLY ERASABLE PROGRAMMABLE READ ONLY MEMORY

USB UNIVERSAL SERIAL BUS

RAM RANDOM ACCESS MEMORY

WIFI WIRELESS FIDELITY

MP MEGAPIXEL

PWM PULSE WIDTH MODULATION

ICSP IN CIRCUIT SERIAL PROGRAMMING

MHz MEGAHEARTZ

V VOLT

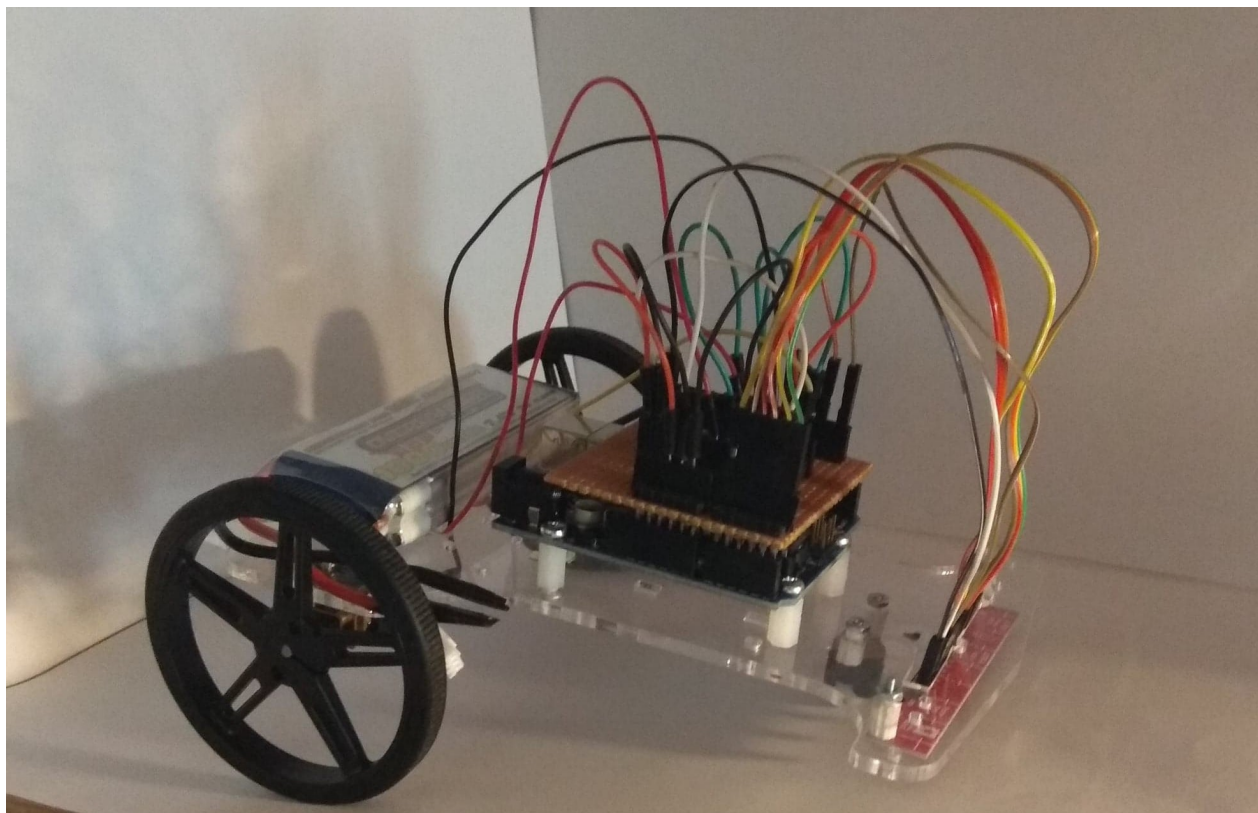
ΚΕΦΑΛΑΙΟ 1

ΕΙΣΑΓΩΓΗ

1.1 Περιγραφή του αντικειμένου της πτυχιακής εργασίας

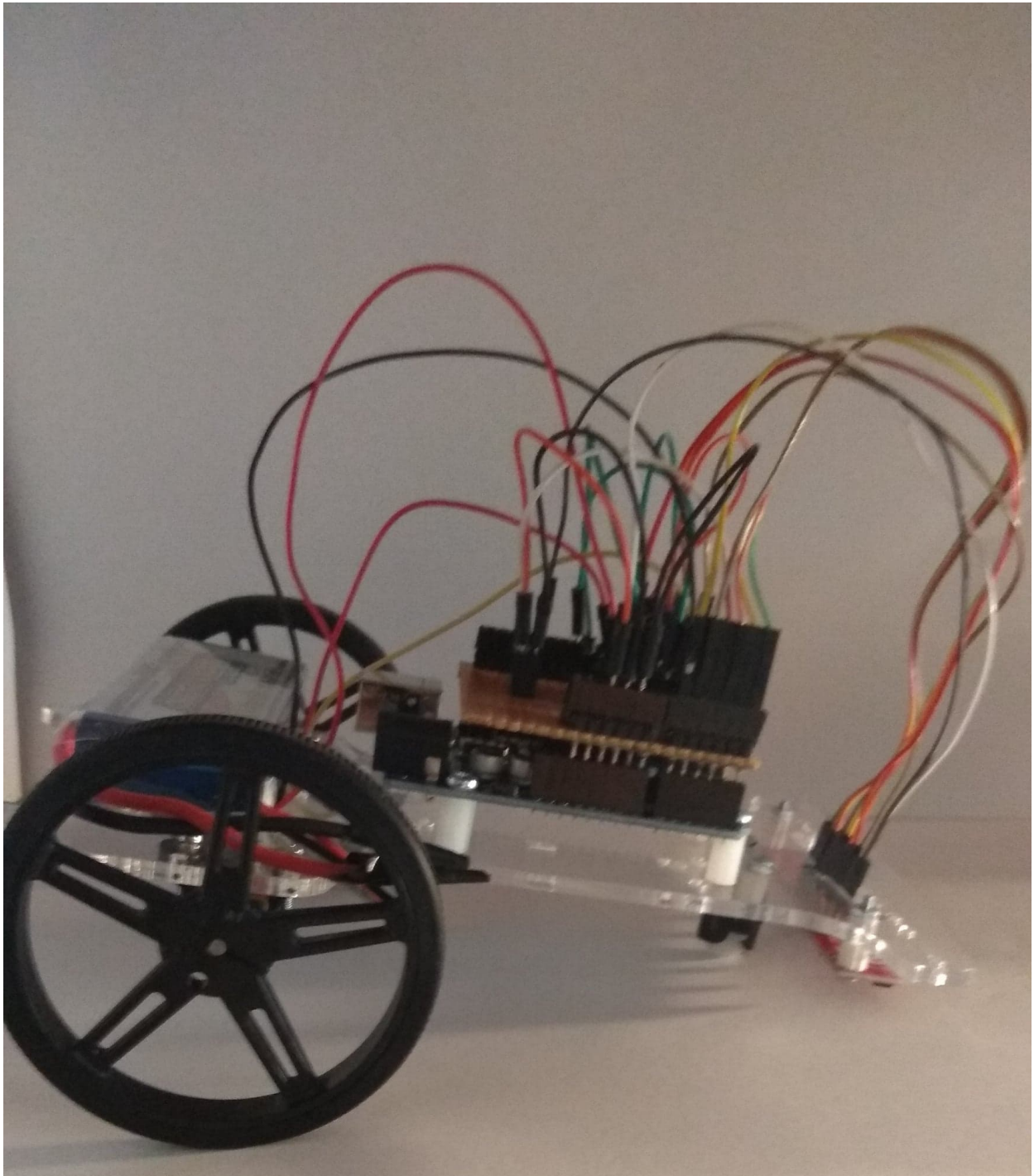
Σε αυτό το κεφάλαιο θα γίνει με απλά λόγια η επεξήγηση της πτυχιακής μου εργασίας. Η πτυχιακή μου έχει τίτλο «Ανάπτυξη ρομποτικού οχήματος κινούμενου με ανίχνευση γραμμής». Σε αυτό το σημείο θα πρέπει να τονίσω πως η χρήση των κωδικοποιητών(encoders) στους κινητήρες του αυτοκινήτου μου έγινε μόνο σε πειραματικό στάδιο και το εξηγώ μόνο σε θεωρητικό επίπεδο. Αρχικά, πρέπει να εξηγήσω τι εννοώ ακριβώς με τον όρο line follower. Πρόκειται για ένα ρομποτικό αυτοκινητάκι, δίτροχο, με μια βοηθητική ρόδα στο μπροστινό τμήμα του σασί, το οποίο μέσω led ανιχνευτών χρώματος μπορεί να ακολουθήσει μια μαύρη γραμμή στην περίπτωση μας (έχεις την δυνατότητα να του δώσεις οποιοδήποτε χρώμα να ανιχνεύσει αρκεί πρώτα να το αποθηκεύσεις σαν δεδομένο στην μνήμη του) και να κινηθεί κατά μήκος αυτής της γραμμής, ό,τι κι αν σημαίνει αυτό (στροφές, εμπόδια, διακοπές). Πώς γίνονται όμως όλα αυτά; Τα εξαρτήματα που χρειάστηκα είναι αρχικά ο Arduino που θα συνδεθούν όλα πάνω του. Χρειάστηκα και κάποια άλλα όμως συγκεκριμένα υλικά, όπως, οι κινητήρες μου που είναι από τα πιο σημαντικά καθώς και η πλακέτα που έπρεπε να χρησιμοποιήσω για την κίνηση τους, η L293D που έπρεπε να κουμπώσω πάνω στο arduino μου. Για να επικοινωνήσουν αυτές οι δύο πλακέτες έχω δημιουργήσει έναν κώδικα σε γλώσσα arduino με τις κατάλληλες παραμέτρους ώστε να αναγνωρίζει μέσω των ανιχνευτών χρώματος που βρίσκεται στο μπροστινό κομμάτι του αυτοκινήτου το μαύρο χρώμα και μέσω του PID (θα εξηγήσω στην συνέχεια πιο αναλυτικά) ελεγκτή να δίνει οδηγίες στον κινητήρα μου, τον δεξί ή τον αριστερό, είτε να χαμηλώσει ταχύτητα για να στρίψει δεξιά ή αριστερά αντίστοιχα, είτε να σταματήσει. Για να συνδεθούν όλα αυτά χρησιμοποίησα καλώδια διπλής όψης και σχεδίασα επίσης ένα σασί αυτοκινήτου από πλεξιγκλάς και τα εφόρμωσα όλα πάνω βιδωτά.

Ανάπτυξη ρομποτικού οχήματος κινούμενου με ανίχνευση γραμμής



Εικόνα 1: Δίτροχο αυτοκινητάκι

Ανάπτυξη ρομποτικού οχήματος κινούμενου με ανίχνευση γραμμής



Εικόνα 2: Line Follower προφίλ

Ανάπτυξη ρομποτικού οχήματος κινούμενου με ανίχνευση γραμμής

1.2 Ιστορική αναδρομή

1.2.1 Ιστορική αναδρομή Arduino

Η Arduino πλατφόρμα πρόκειται για μια πλακέτα παροχής λογισμικού. Η ανάπτυξή της ξεκίνησε το 2005 που θα αποσκοπούσε στον έλεγχο προγραμμάτων διαδραστικών σχεδίων από μαθητές. Ο Massimo Banzi και ο David Cueartielles ήθελαν να αναπτύξουν ένα φθινό και πρωτότυπο σύστημα για εκείνη την περίοδο. Έτσι ονόμασαν το σχέδιο από το Arduino της Ingea και ξεκίνησαν να παράγουν πλακέτες.

Το πρώτο Arduino που φτιάχτηκε ήταν το “Serial Arduino” και περιελάμβανε μια ATmega8 με σύνδεση RS-232 με τον μικροελεγκτή. Στην συνέχεια ακολούθησαν άλλες εκδόσεις που περιελάμβαναν USB μετατροπέα όπως το Arduino Extreme. Ο David Mellis ανέπτυξε το αρχικό λογισμικό Arduino που βασίζεται στην καλωδίωση, με πολλές συνεισφορές από τον Nicholas Zambetti. Το Arduino Extreme με USB υποδοχή για προγραμματισμό χρησιμοποιούσε τεχνολογία ATmega8. Ακολούθησε το Arduino Mini στο οποίο όλα τα εξαρτήματα ήταν πάνω στην πλακέτα, αυτό λέγεται τεχνολογία επιφανειακής συναρμολόγησης (SMT), και η τεχνολογία που χρησιμοποιούσε ήταν ATmega168. Όλες αυτές είναι βελτιώσεις που γίνοντουσαν στο Arduino κατά καιρούς. Η εξέλιξη της πλακέτας όμως δεν στάθηκε στο Arduino Mini αλλά συνεχίστηκε με μια ακόμα καλύτερη έκδοση του οποίου εκεί ο μικροελεγκτής αναβαθμίστηκε σε ATmega328 με το Arduino UNO. Την “επανάσταση” στους μικροελεγκτές Arduino ξεκίνησε η έκδοση του Arduino Leonardo, με ένα ATmega32U4 chip που εξαλείφει την ανάγκη για συνδεσιμότητα μέσω USB και μπορεί να χρησιμοποιηθεί ως ψηφιακό πληκτρολόγιο ή ποντίκι. Υπάρχουν βέβαια αρκετές εκδόσεις Arduino που χρησιμοποιούνται και για πιο εξειδικευμένες περιπτώσεις, όπως το Arduino Nano, Arduino Mega καθώς και το Arduino Esplora και το Arduino DUE.

Όσο αφορά το λογισμικό του μικροελεγκτή και το σχέδιο του, το ολοκληρωμένο περιβάλλον ανάπτυξης του Arduino είναι μια εφαρμογή γραμμένη σε Java που λειτουργεί σε πολλές πλατφόρμες και χρησιμοποιεί ως σχέδιο του μια διακλάδωση της πλατφόρμας Wiring. Η γλώσσα που χρησιμοποιείτε για προγραμματισμό του Arduino είναι βασισμένη στο Wiring και είναι παρόμοια της C++ ή C. Έχει φτιαχτεί για να εισάγει στον προγραμματισμό τους καλλιτέχνες ή νέους που δεν είναι εξοικειωμένοι με την ανάπτυξη λογισμικού.



Εικόνα 3: Arduino Logo

1.2.2 Ιστορική αναδρομή Encoder

Ο encoder ή στα ελληνικά όπως είναι γνωστός, κωδικοποιητής, είναι ένα κύκλωμα, μια συσκευή η οποία μετατρέπει πληροφορίες απο μια μορφή ή κώδικα σε άλλο για λόγους τυποποίησης ταχύτητας ή συμπίεσης. Δεν υπάρχουν σαφή στοιχεία για την ακριβή χρονολογική εμφάνιση τους για πρώτη φορά στον κόσμο του αυτοματισμού. Μπορούμε όμως να τα διακρίνουμε σε είδη, τα οποία μέσα στα έτη υπήρξε αρκετή βελτίωση και ανάπτυξή τους.

Ένας κωδικοποιητής είναι ένας αισθητήρας μηχανικής κίνησης που παράγει ψηφιακά σήματα σε απόκριση της κίνησης. Ως ηλεκτρομηχανική συσκευή, ένας κωδικοποιητής είναι σε θέση να παρέχει στους χρήστες του συστήματος κίνησης πληροφορίες σχετικά με τη θέση, την ταχύτητα και την κατεύθυνση. Υπάρχουν διαφορετικοί τύποι κωδικοποιητών: γραμμικοί, περιστροφικοί, θέσης και οπτικοί. Λόγω της ποικιλίας τους χρησιμοποιούνται σε πολλές βιομηχανίες. Γνωστή στην χρήση κωδικοποιητών είναι η αυτοκινητοβιομηχανία αλλά δεν λείπουν και από τον τομέα της ηλεκτρονικής, του εξοπλισμού γραφείου, από τον τομέα της σύγχρονης βιομηχανίας, από τον ιατρικό τομέα καθώς και τον στρατιωτικό. Σημαντικός ο ρόλος των κωδικοποιητών είναι και στην επιστήμη καθώς ο επιστημονικός εξοπλισμός εφαρμόζει κωδικοποιητές στην τοποθέτηση παρατηρητηρίου.

Με απλά λόγια, ο κωδικοποιητής είναι μια συσκευή ανίχνευσης που παρέχει ανατροφοδότηση. Οι κωδικοποιητές μετατρέπουν την κίνηση σε ένα ηλεκτρικό σήμα το οποίο μπορεί να διαβαστεί από κάποιο τύπο συσκευής ελέγχου κίνησης, όπως ένας μετρητής ή ένα PLC (Programmable Logic Controllers). Ο κωδικοποιητής στέλνει ένα σήμα ανάδρασης το οποίο μπορεί να χρησιμοποιηθεί για τον προσδιορισμό της θέσης του αριθμού της ταχύτητας ή της κατεύθυνσης. Μια συσκευή ελέγχου μπορεί να χρησιμοποιήσει αυτές τις πληροφορίες για να στείλει μια εντολή σε μια συγκεκριμένη λειτουργία. Σε παραδείγματα θα αναφερθούμε αναλυτικότερα πιο μετά. Όμως εκτός από τα είδη που προαναφέραμε οι κωδικοποιητές χωρίζονται ανάλογα με την ανάδρασή τους σε απόλυτους ή αυξομειωτικούς. Ένας γραμμικός κωδικοποιητής μετράει απόσταση μεταξύ δύο σημείων η σύνδεση γίνεται ενσύρματα, μέσω του καλωδίου γίνεται η μετατροπή του σήματος που φτάνει στην έξοδο. Το καλώδιο πρέπει να συνδέεται με το κινούμενο αντικείμενο και την συσκευή ελέγχου θέσης. Ο περιστροφικός επιστρέφει πληροφορίες συνήθως για την γωνιακή θέση ενός περιστρεφόμενου αντικειμένου. Οι κωδικοποιητές θέσης παράγουν ηλεκτρικό σήμα για τον προσδιορισμό της απόλυτης θέσης του αντικειμένου ενώ οι οπτικοί λειτουργούν με παλμούς φωτός. Από την ανάλυση αυτών των κυμάτων φωτός προκύπτουν πληροφορίες σχετικά με την θέση, την κατεύθυνση αλλά και το εύρος των κυμάτων.

1.3 Ανασκόπηση της πτυχιακής εργασίας

Για τη δημιουργία της πτυχιακής εργασίας έκανα μια έρευνα στο διαδίκτυο για να έχω περίπου μια ιδέα για το τι θα χρειαστώ. Ύστερα από μια έρευνα στις τιμές των προϊόντων που ήθελα, συνειδητοποίησα ότι μερικά χρειάζεται να τα παραγγείλω από το εξωτερικό. Αφότου είχα πλέον όλα τα υλικά το επόμενο βήμα ήταν να κοιτάξω πώς θα τα συνδέσω όλα μεταξύ τους ώστε να φτάσω πλέον στην δημιουργία ενός αυτόνομου 2-τροχού ρομποτικού αμαξιδίου. Χρειαζόμουν ένα σασί, πράγμα που έχουν όλα τα αυτοκίνητα, σχεδίασα ένα με κατάλληλες τρύπες επειδή κάποια εξαρτήματα χρειαζόταν να βιδωθούν πάνω, και το υλικό που χρησιμοποίησα ήταν το πλέξι γκλας ώστε να μην είναι αρκετά βαρύ και να έχει επιπτώσεις στην ταχύτητα αλλά και να μπορεί να “σηκώσει” το υπάρχον βάρος των εξαρτημάτων. Στη συνέχεια άρχισα να προσπαθώ να κάνω τις συνδεσμολογίες αφού είχα κάνει πρώτα μια έρευνα στο για το τι να προσέξω για τον μικροελεγκτή L293D ο οποίος θα ήταν υπεύθυνος για την κίνηση των κινητήρων μου. Συνέδεσα την LED σειρά στο μπροστινό κομμάτι του αμαξιού και ύστερα με το Arduino μου. Προγραμμάτισα τον βασικό κώδικα που είχε και στο ίδιο το website του προϊόντος ούτως ώστε να δω αν κάποιο από τα φωτάκια αυτά ήταν καμμένο και δεν αναγνώριζε το μαύρο-άσπρο. Αφότου σιγουρεύτηκα πώς οι ανιχνευτές της μαύρης γραμμής λειτουργούν έπρεπε να ελέγξω και τους κινητήρες μου. Κάθε κινητήρας για να λειτουργήσει πρέπει από κάπου να αντλεί ενέργεια. Αυτό το πέτυχα με μια μπαταρία λιθίου (LiPo Battery 7.4 V). Αφού σύνδεσα τον αρνητικό και τον θετικό πόλο αντίστοιχα και είδα πως συμπεριφέρεται ο κινητήρας μου έπρεπε να φτιάξω έναν κώδικα με PID έλεγχο κάθε φορά για να ελέγχω την ταχύτητα του στις στροφές και όταν έστω και λίγο βγαίνει εκτός της μαύρης γραμμής που πρέπει να ακολουθεί. Για να ρυθμίσεις τους PID μετρητές πρέπει να κάνεις αρκετές δοκιμές ως προς τις μετρήσεις που παίρνεις κάθε φορά και ως προς το μέγιστο της ταχύτητας που δημιουργείτε κάθε φορά όταν συναντάει την άσπρη γραμμή. Ο κώδικας Arduino φτιάχτηκε κομμάτι κομμάτι, πρώτα πρέπει να γίνεται έλεγχος με τα pins LED ώστε να ξέρω σε ποιά θέση βρίσκετε το αμάξι-δεξιά, κέντρο ή αριστερά-και με την σειρά του αυτό θα δίνει εντολή στον κινητήρα αν πρέπει να στρίψει ή να προχωρήσει ευθεία. Αφού καταφέραμε να εφαρμόσουμε έναν τύπο που θα μαζεύει όλα τα δεδομένα του αισθητήρα και μας έβγαζε μια τιμή που άλλαζε ανάλογα με την κίνηση του αισθητήρα. Στην συνέχεια ακολούθησε κώδικας που έπαιρνε την τιμή που έβγαζε ο τύπος και με τον PID αλγόριθμο υπολογιζόταν η ταχύτητα που έπρεπε να καταλήξει στους δύο τροχούς. Τέλος προσέθεσα και το τελευταίο κομμάτι που ήταν οι κωδικοποιητές στους δύο τροχούς για να έχουμε κάθε φορά μετρήσεις τις θέσης και της ταχύτητας οι οποίες θα υπολογιστούν μέσω Matlab και θα μας δώσουν μια τιμή που θα μας βοηθήσει στην ακριβής ταχύτητα που πρέπει να πάει στους τροχούς του αμαξιού. Το αμάξι μας είναι πλέον αυτόματο.

ΚΕΦΑΛΑΙΟ 2

ΜΙΚΡΟΕΛΕΓΚΤΗΣ ARDUINO

2.1 Εισαγωγή

2.1.1 Εισαγωγή στους μικροελεγκτές

Ο μικροελεγκτής είναι ένα ολοκληρωμένο, παρεμφερές του μικροεπεξεργαστή, ο οποίος έχει όλα τα εξαρτήματα που θα χρειαζόντουσαν υπό άλλες συνθήκες για να τελειοποιήσουμε μια κατασκευή, πάνω του ενσωματωμένα. Λόγω λοιπόν των πολλών ενσωματωμένων υποσυστημάτων που διαθέτει χρησιμοποιείται ευρύτατα σε όλα τα ενσωματωμένα συστήματα ελέγχου χαμηλού και μεσαίου κόστους. Όπως αυτά που χρησιμοποιούνται σε αυτοματισμούς, ηλεκτρονικά καταναλωτικά προϊόντα, ηλεκτρικές συσκευές καθώς και κάθε είδους αυτοκινούμενα τροχοφόρα οχήματα. Συνήθη υποσυστήματα είναι η προσωρινή μνήμη RAM, οι μνήμες προγράμματος ROM, FLASH, EPROM κλπ καθώς και CPU, πόρτες I/O, χρονιστές/απαριθμητές, μετατροπέα αναλογικού σε ψηφιακό και το αντίστροφο, timers και διάφορα άλλα. Θα μπορούσε να παρομοιαστεί με έναν μικροϋπολογιστή, όπως ακριβώς κι αυτός έχει επεξεργαστή, μνήμη και περιφερειακές συσκευές και εκτελεί προγράμματα έτσι και ένας μικροελεγκτής διαθέτει όλα τα παραπάνω χαρακτηριστικά που είπαμε και μάλιστα ολοκληρωμένα σε ένα μόνο chip. Οι κατασκευαστές μικροελεγκτών είναι μεγάλες εταιρείες αυτοματισμού και ηλεκτρονικών ειδών όπως η Microchip, η Texas Instruments, η Atmel, η Freescale (πρώην Motorola), Intel, Analog Devices. Γενικά οι μικροελεγκτές προγραμματίζονται σε γλώσσες χαμηλού επιπέδου αλλά τελευταία όλο και περισσότεροι προγραμματιστές χρησιμοποιούν γλώσσες υψηλού επιπέδου, ως γλώσσα υψηλού επιπέδου ονομάζεται η γλώσσα η οποία είναι αυστηρά δομημένη και υπάρχει συγκεκριμένος compiler ο οποίος μετατρέπει το πρόγραμμα σε γλώσσα μηχανής για το συγκεκριμένο μικροελεγκτή. Ο μικροελεγκτής οπότε αποτελεί τον “εγκέφαλο” των ηλεκτρονικών κυκλωμάτων.

Η πιο διαδεδομένη γλώσσα προγραμματισμού των μικροελεγκτών είναι η C, C++ και οι παραλλαγές τους. Σε προγράμματα όμως που απαιτείται μικρή χωρητικότητα μνήμης ή ταχύτητα η γλώσσα που προτιμάμε είναι η Assembly. Όμως οι μεγαλύτερες απαιτήσεις σε λειτουργικότητα και η ευκολία προγραμματισμού της C έναντι της assembly, σε

Ανάπτυξη ρομποτικού οχήματος κινούμενου με ανίχνευση γραμμής

συνδυασμό με την επάρκεια μνήμης των σύγχρονων μικροελεγκτών, έχουν γενικά εκτοπίσει την Assembly από τις περισσότερες εφαρμογές.

2.1.2 Εισαγωγή στον Arduino

Το Arduino είναι μια ηλεκτρονική πλατφόρμα ανοικτού κώδικα βασισμένη σε εύχρηστο υλικό και λογισμικό. Οι πίνακες Arduino είναι σε θέση να διαβάζουν εισόδους, φως σε έναν αισθητήρα, ένα δάκτυλο σε ένα κουμπί ή ένα μήνυμα Twitter και να το μετατρέπουν σε έξοδο, ενεργοποιώντας έναν κινητήρα, ενεργοποιώντας ένα LED, δημοσιεύοντας κάτι online. Μπορείτε να πείτε στην πλακέτα σας τι πρέπει να κάνει στέλνοντας ένα σύνολο οδηγιών στον μικροελεγκτή. Για να το κάνετε αυτό, χρησιμοποιείτε τη γλώσσα προγραμματισμού Arduino (βασισμένη στην καλωδίωση) και το λογισμικό Arduino (IDE), με βάση την επεξεργασία.

Το Arduino βασίζεται στον μικροελεγκτή της Atmel, Atmega καθώς και όλα τα σχέδια και το λογισμικό που χρειάζεται για την λειτουργία του. Διανέμεται δωρεάν ώστε να μπορεί να κατασκευαστεί από τον καθένα και εννοείται πως ο χρήστης μπορεί να συνδέσει πάνω πολλαπλές μονάδες εισόδου/εξόδου να προγραμματίσει τον μικροελεγκτή και να δέχεται δεδομένα από αυτές τις μονάδες εισόδου ώστε να τα επεξεργάζεται και να στέλνει και να στέλνει τις απαραίτητες εντολές στις θύρες εξόδου. Τα Arduino και τα Arduino συμβατά boards χρησιμοποιούν την τεχνολογία των shields, τυπωμένων boards επεκτάσεων κυκλωμάτων που συνδέονται στα κανονικά παρεχόμενα Arduino pin-headers. Τα shields μπορούν να παρέχουν έλεγχο στα motors, GPS, Ethernet, LCD εικόνας ή breadboarding (προτυποποίησης).

2.2 Εκδόσεις του Arduino

Δεκαέξι εκδοχές του Arduino Hardware έχουν χρησιμοποιηθεί εμπορικά μέχρι τώρα εγώ θα διαλέξω να συζητήσω για τα κύρια μοντέλα του Arduino που χρησιμοποιούνται στην αγορά.

➤ Arduino UNO

Αυτό το μοντέλο χρησιμοποιεί τον μικροελεγκτή ATmega328P και θεωρείται η πιο φιλική προς τον χρήστη Arduino πλακέτα από όλα τα μοντέλα. Λειτουργεί στα 5Volt ενώ οι τάσεις που μπορεί να δεχτεί είναι από 7 έως 20Volt. Έχει 14 ψηφιακές ακίδες εισόδου/εξόδου

Ανάπτυξη ρομποτικού οχήματος κινούμενου με ανίχνευση γραμμής

ενώ 6 από αυτές μπορούν να χρησιμοποιηθούν ως θύρες εξόδου PWM. Οι αναλογικές ακίδες εισόδου είναι 6. Έχει επίσης μια θύρα USB, ICSP και μια θύρα τροφοδοσίας. Το reset κουμπί υπάρχει πάνω στην πλακέτα πράγμα που το καθιστά εύκολο μιας και δεν θα χρειαστεί κώδικας.



Εικόνα 4: Arduino Uno

➤ Arduino Leonardo

Το Arduino Leonardo είναι παρόμοιο με το UNO, βασισμένο στην τεχνολογία ATmega32U4 όμως. Έχει 20 ψηφιακές ακίδες εισόδου / εξόδου (από τις οποίες 7 μπορούν να χρησιμοποιηθούν ως έξοδο PWM και 12 ως αναλογικές εισοδοί), έναν ταλαντωτή κρυστάλλου 16 MHz, μια σύνδεση micro USB, μια υποδοχή τροφοδοσίας, μια κεφαλίδα ICSP και ένα κουμπί επαναφοράς. Περιέχει όλα τα απαραίτητα για την υποστήριξη του μικροελεγκτή απλά συνδέστε τον με έναν υπολογιστή με καλώδιο USB ή ενεργοποιήστε τον με προσαρμογέα εναλλασσόμενου ρεύματος ή μπαταρία για να ξεκινήσετε.



Εικόνα 5: Arduino Leonardo

➤ Arduino Esplora

Το Arduino Esplora είναι ένας μικροελεγκτής που προέρχεται από το Arduino Leonardo, άρα έχει την ίδια λειτουργία. Το Esplora διαφέρει από όλους τους προηγούμενους Arduino, επειδή παρέχει μια σειρά ενσωματωμένων, έτοιμων προς χρήση, αισθητήρων για αλληλεπίδραση πάνω στην πλακέτα. Είναι σχεδιασμένο για ανθρώπους που θέλουν να ξεκινήσουν να τρέχουν με το Arduino χωρίς να χρειάζεται να μάθουν πρώτα για την ηλεκτρονική πλευρά. Το Esplora διαθέτει ενσωματωμένες ηχητικές και φωτεινές εξόδους και αρκετούς αισθητήρες εισόδου, συμπεριλαμβανομένου ενός joystick, ενός ρυθμιστή, ενός αισθητήρα θερμοκρασίας, ενός επιταχυνσιόμετρου, ενός μικροφώνου και ενός αισθητήρα φως. Έχει επίσης τη δυνατότητα να επεκτείνει τις δυνατότητές του με δύο υποδοχές εισόδου και εξόδου Tinkerkit και μία υποδοχή για έγχρωμη οθόνη TFT LCD.

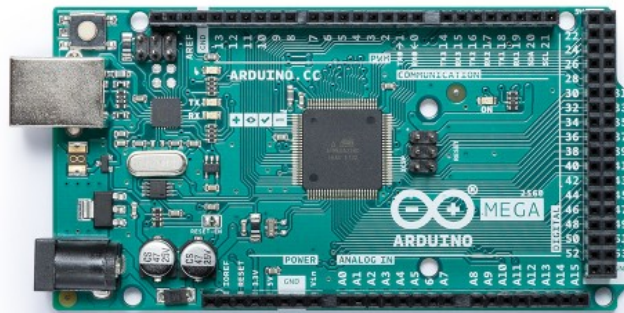


Εικόνα 6: Arduino Esplora

Ανάπτυξη ρομποτικού οχήματος κινούμενου με ανίχνευση γραμμής

➤ Arduino Mega 2560

Το Arduino Mega 2560 έχει σχεδιαστεί για πιο περίπλοκα έργα χρησιμοποιεί τεχνολογία ATmega2560. Με 54 ψηφιακούς ακροδέκτες εισόδου / εξόδου (από τις οποίες 15 μπορούν να χρησιμοποιηθούν ως έξοδο PWM), 16 αναλογικές εισόδους, 4 UART (σειριακές θύρες υλικού), ταλαντωτή κρυστάλλου 16 MHz, USB θύρα, υποδοχή τροφοδοσίας, και ένα κουμπί επαναφοράς. και μεγαλύτερο χώρο για τον κώδικά σας είναι η συνιστώμενη πλακέτα για έργα 3D εκτυπωτών και ρομποτικών εφαρμογών.



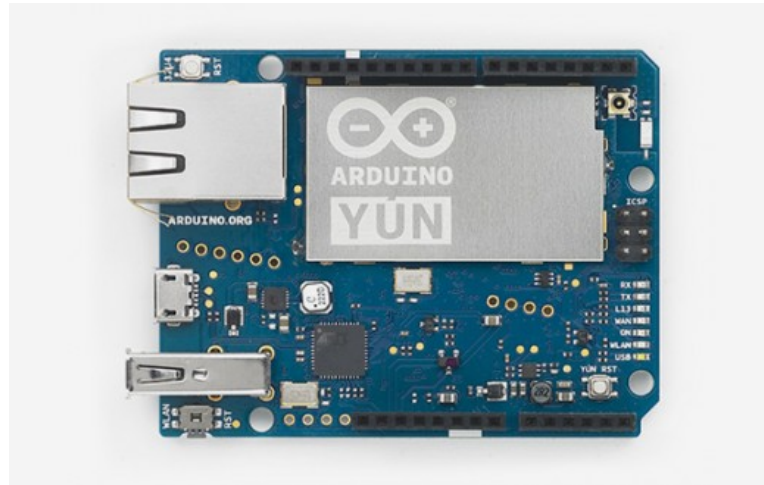
Εικόνα 7: Arduino Mega 2560

➤ Arduino Υύν

Το Arduino Yun είναι μια Arduino πλακέτα που βασίζεται σε τεχνολογία ATmega32u4 και το Atheros AR9331. Ο επεξεργαστής Atheros υποστηρίζει μια διανομή Linux που βασίζεται στο OpenWrt με το όνομα Linino OS. Ο πίνακας διαθέτει ενσωματωμένη υποστήριξη Ethernet και WiFi, θύρα USB-A, υποδοχή κάρτας micro-SD, 20 ακίδες

Ανάπτυξη ρομποτικού οχήματος κινούμενου με ανίχνευση γραμμής

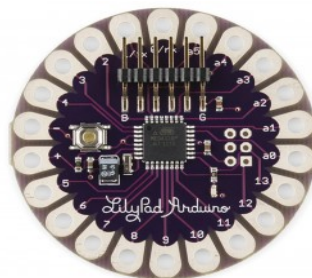
εισόδου / εξόδου (7 από αυτές μπορούν να χρησιμοποιηθούν ως έξοδο PWM και 12 ως αναλογικές εισόδους) έναν ταλαντωτή, μια θύρα USB, μια ICSP και 3 κουμπιά επαναφοράς



Εικόνα 8: Arduino Yun

➤ Arduino LilyPad

Το LilyPad Arduino βασίζεται στην ATmega168V (έκδοση χαμηλής ισχύος του ATmega168) ή στο ATmega328P. Το LilyPad Arduino σχεδιάστηκε και αναπτύχθηκε από τους Leah Buechley και SparkFun Electronics. Ενδύκνεται για έργα e-textile και wearables. Μπορεί να ραμμένο στο ύφασμα και παρόμοια τοποθετημένα τροφοδοτικά, αισθητήρες και ενεργοποιητές με αγώγιμο σπείρωμα.

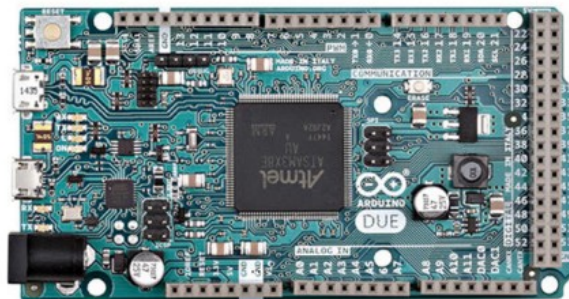


Εικόνα 9: Arduino LilyPad

Ανάπτυξη ρομποτικού οχήματος κινούμενου με ανίχνευση γραμμής

➤ Arduino Due

Το Arduino Due είναι ένας μικροελεγκτής που βασίζεται στον επεξεργαστή Atmel SAM3X8E ARM Cortex-M3. Είναι ο πρώτος Arduino βασισμένος σε πυρήνα ARM 32 bit. Διαθέτει 54 ψηφιακές ακίδες εισόδου / εξόδου (από τις οποίες 12 μπορούν να χρησιμοποιηθούν ως έξοδο PWM), 12 αναλογικές εισόδους, 4 UART (σειριακές θύρες υλικού), ρολόι 84 MHz, σύνδεση OTG USB, 2 DAC, 2 TWI, μια υποδοχή τροφοδοσίας, μια κεφαλίδα SPI, μια κεφαλίδα JTAG, ένα κουμπί επαναφοράς και ένα κουμπί διαγραφής. Προσοχή: Σε αντίθεση με τους περισσότερους πίνακες Arduino, ο πίνακας Arduino Due τρέχει στα 3.3V. Η μέγιστη τάση που μπορούν να ανεχθούν οι ακροδέκτες εισόδου / εξόδου είναι 3.3V. Η εφαρμογή τάσης υψηλότερης από 3.3V σε οποιονδήποτε ακροδέκτη I / O θα μπορούσε να προκαλέσει βλάβη στην πλακέτα.



Εικόνα 10: Arduino Due

2.3 Arduino Shields

Τα Arduino και γενικότερα όλα τα συμβατά boards Arduino χρησιμοποιούν την τεχνολογία των shields. Τα shields είναι επεκτατικά κυκλώματα τυπωμένων boards που συνδέονται στα κανονικά παρεχόμενα των Arduino pin-headers. Μας δίνουν την δυνατότητα περισσότερων κατασκευών και ως εκ τούτου με περισσότερες δυνατότητες. Είναι εύκολο να τοποθετηθούν και φτηνές για να παραχθούν. Φυσικά υπάρχει ένα όριο για τον αριθμό των shields που θα κουμπώσουν πάνω στην πλακέτα μας. Μερικά shields

Ανάπτυξη ρομποτικού οχήματος κινούμενου με ανίχνευση γραμμής

μπορεί να χρησιμοποιούν τα ίδια pins του Arduino για την λειτουργία τους οπότε μπορεί να μην είναι συμβατά μεταξύ τους.

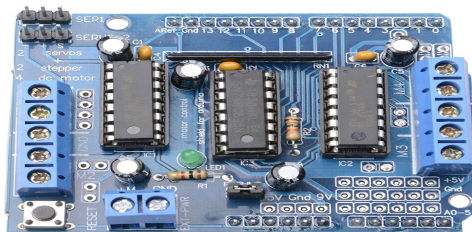
Τα πιο συνηθισμένα και δημοφιλή shields είναι και αυτά που θα αναλύσουμε παρακάτω:

- Ethernet Shield: Το Arduino Ethernet Shield επιτρέπει σε ένα board Arduino να συνδεθεί στο διαδίκτυο χρησιμοποιώντας τη βιβλιοθήκη Ethernet και να διαβάσει και να γράψει μια κάρτα SD χρησιμοποιώντας τη βιβλιοθήκη SD. Συνδέστε το shield στον υπολογιστή σας ή σε ένα διανομέα ή δρομολογητή δικτύου χρησιμοποιώντας ένα τυπικό καλώδιο Ethernet (CAT5 ή CAT6 με υποδοχές RJ45). Η σύνδεση με έναν υπολογιστή ενδέχεται να απαιτεί τη χρήση καλωδίου cross-over (αν και πολλοί υπολογιστές, συμπεριλαμβανομένων όλων των πρόσφατων Mac, μπορούν να κάνουν το cross-over εσωτερικά).
- Relay Shield: Το Arduino 4 Relays Shield είναι μια λύση για την οδήγηση φορτίων υψηλής ισχύος που δεν μπορούν να ελεγχθούν από τους ψηφιακούς IO του Arduino, λόγω των ορίων ρεύματος και τάσης του ελεγκτή. Το Shield διαθέτει τέσσερα ρελέ, κάθε ρελέ παρέχει 2 επαφές αλλαγής πόλων (NO και NC). για να αυξηθεί το όριο ρεύματος κάθε εξόδου, οι δύο επαφές μετάβασης έχουν τοποθετηθεί παράλληλα. Τέσσερις λυχνίες LED δείχνουν την κατάσταση ενεργοποίησης / απενεργοποίησης κάθε ρελέ.
- Motor Shield: Το Arduino Motor Shield βασίζεται στο L298, το οποίο είναι ένας διπλός οδηγός πλήρους γέφυρας, σχεδιασμένος για την οδήγηση επαγωγικών φορτίων όπως ηλεκτρονόμοι, ηλεκτρομαγνητικές βαλβίδες, DC και βηματικούς κινητήρες. Σας επιτρέπει να οδηγείτε δύο ηλεκτροκινητήρες συνεχούς ρεύματος με την πλακέτα Arduino, ελέγχοντας την ταχύτητα και την κατεύθυνση καθενός ανεξάρτητα. Μπορείτε επίσης να μετρήσετε την απορρόφηση ρεύματος κινητήρα από κάθε κινητήρα, μεταξύ άλλων χαρακτηριστικών. Εγώ για την κατασκευή μου χρησιμοποίησα την L293D motor shield για DC κινητήρες.
- XBEE Shield: Η ασπίδα Xbee επιτρέπει σε ένα board Arduino να επικοινωνεί ασύρματα χρησιμοποιώντας το Zigbee. Βασίζεται στη μονάδα Xbee από το MaxStream. Το δομοστοιχείο μπορεί να επικοινωνεί σε απόσταση μέχρι 30m σε κλειστό περιβάλλον και πάνω από 80m σε ανοιχτό περιβάλλον (χωρίς εμπόδια). Μπορεί να χρησιμοποιηθεί διαφήμιση ad Serial / USB converter που χρησιμοποιείται μέσω γραμμής εντολών και διαμορφώνεται για να λειτουργεί σε πλέγμα, μετάδοση.
- Proto Shield: Το ProtoShield σας διευκολύνει να σχεδιάσετε προσαρμοσμένα κυκλώματα. Μπορείτε εύκολα να κολλήσετε τα ενσωματωμένα τερματικά TH ή SMD στην πρωτότυπη περιοχή για να τα δοκιμάσετε με το Arduino. Η περιοχή SMD έχει σχεδιαστεί για μέγιστο 24 ακίδων ολοκληρωμένου κυκλώματος SOIC και η περιοχή

Ανάπτυξη ρομποτικού οχήματος κινούμενου με ανίχνευση γραμμής

ΤΗ περιέχει πολύ χώρο για τα απαραίτητα στοιχεία γύρω από το έργο. Μπορείτε ακόμη να κολλήσετε ένα μίνι πινάκιο (δεν περιλαμβάνεται) στην πρωτότυπη περιοχή για αδιάλυτη λειτουργία. Η ροτο περιοχή περιλαμβάνει επίσης δύο γραμμές ρεύματος (IOREF και GND), δύο λαμπτήρες LED και τα SPI σήματα ξεμπλοκαρίσματος για πίνακες με SPI μόνο στην κεφαλίδα ICSP όπως Zero.

- Bluetooth Shield: Δίνει την δυνατότητα στο Arduino να συνδεθεί με μια συσκευή μέσω bluetooth.
- GPS Shield: Η GPS πλακέτα έχει την δυνατότητα να επιστρέφει στοιχεία σχετικά με την θέση της κατασκευής μας.
- Grove OLED Display Shield: Το ορατό τμήμα του OLED μετρά 1,12 "διαγώνια και περιέχει 96x96 εικονοστοιχεία σε κλίμακα του γκρι. Επειδή η οθόνη χρησιμοποιεί OLED, δεν υπάρχει οπίσθιος φωτισμός και η αντίθεση είναι πολύ υψηλή. Αυτό το OLED χρησιμοποιεί τον driver SSD1327, ο οποίος διαχειρίζεται την οθόνη. Μπορείτε να επικοινωνήσετε με τον driver χρησιμοποιώντας 4-wire I2C (ρολόι, δεδομένα, ισχύ και GND)



Εικόνα 11: Arduino Motor Shield

2.4 Χαρακτηριστικά του Arduino

Τα περισσότερα arduino λειτουργούν με την τεχνολογία ATmega328P της Atmel. Ο μικροεπεξεργαστής ATmega328, έχει τρεις ομάδες μνήμης. Διαθέτει flash memory, στην οποία αποθηκεύονται τα Arduino sketch, SRAM (static random access memory), στην οποία δημιουργείται το sketch και χρησιμοποιεί τις μεταβλητές όταν τρέχει, και EEPROM, η οποία χρησιμοποιείται από τους προγραμματιστές για την αποθήκευση μακροχρόνιων πληροφοριών. Το Arduino, πρέπει να τροφοδοτηθεί με ρεύμα, είτε από τον υπολογιστή μέσω της σύνδεσης USB, είτε από εξωτερική τροφοδοσία που παρέχεται μέσω μιας υποδοχής φως των 2.1mm που βρίσκεται στην κάτω αριστερή γωνία. Για την αποφυγή

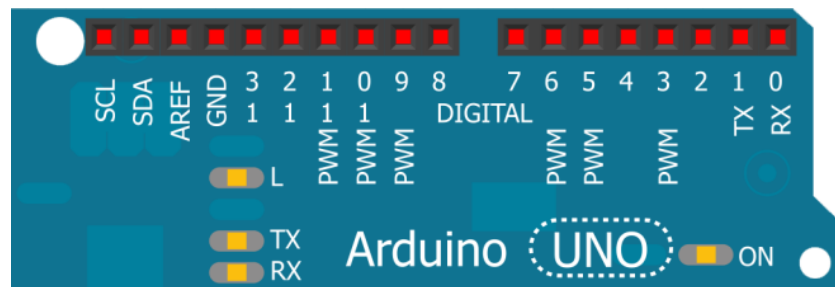
Ανάπτυξη ρομποτικού οχήματος κινούμενου με ανίχνευση γραμμής

προβλημάτων, η εξωτερική τροφοδοσία θα πρέπει να είναι από 7 ως 12V. Στον παρακάτω πίνακα φαίνονται ακριβώς όλα τα δεδομένα για το Arduino Uno που χρησιμοποίησα για την κατασκευή μου.

Μικροελεγκτής	ATMEGA328
Τάση λειτουργίας	5V
Τάση εισόδου	7-12V
Όρια τάσης εισόδου	6-20V
Ψηφιακοί ακροδέκτες I/O	14, (6 PWM έξοδοι)
Αναλογικοί ακροδέκτες εισόδου	6
Ισχύς συνεχόμενου ρεύματος ανά ακροδέκτη	40mA
Ισχύς συνεχόμενου ρεύματος για ακροδέκτη τάσης 3.3V	50mA
Μνήμη flash	32KB (ATMEGA328)
Μνήμη SRAM	2KB (ATMEGA328)
Μνήμη EEPROM	1KB (ATMEGA328)
Ταχύτητα ρολογιού	16MHz

Πίνακας 1: Χαρακτηριστικά Arduino Uno

Κάθε μικροελεγκτής Arduino διαθέτει εισόδους και εξόδους για την αλληλεπίδραση με το περιβάλλον του και τα εξαρτήματα. Κάθε ακροδέκτης (pin) είναι τόσο εισόδου όσο και εξόδου. Το Arduino Uno διαθέτει 20 ακροδέκτες, από τους οποίους 14 είναι ψηφιακοί και 6 είναι αναλογικοί. Αυτά θα τα δούμε και στις εικόνες που ακολουθούν.



Εικόνα 12: Ψηφιακοί ακροδέκτες του Arduino Uno

- Ακροδέκτες 0 και 1: Λειτουργούν ως RX και TX της σειριακής θύρας, όταν το πρόγραμμά ενεργοποιεί τη σειριακή θύρα. Έτσι, όταν το πρόγραμμά στέλνει δεδομένα στη σειριακή θύρα, αυτά προωθούνται και στη θύρα USB μέσω του ελεγκτή Serial-Over-USB, αλλά και στον ακροδέκτη 0 για να τα διαβάσει ενδεχομένως μια άλλη συσκευή. Αυτό φυσικά σημαίνει, ότι αν στο πρόγραμμά ενεργοποιήσει το σειριακό interface, χάνει 2 ψηφιακές εισόδους/εξόδους η πλατφόρμα.
- Ακροδέκτες 2 και 3: Λειτουργούν και ως εξωτερικά interrupts (interrupt 0 και 1 αντίστοιχα). Ρυθμίζονται μέσα από το πρόγραμμά, ώστε να λειτουργούν αποκλειστικά ως ψηφιακές εισοδοι, στις οποίες όταν συμβαίνουν συγκεκριμένες αλλαγές, η κανονική ροή του προγράμματος σταματάει άμεσα και εκτελείται μια συγκεκριμένη συνάρτηση.
- Ακροδέκτες 3, 5, 6, 9, 10 και 11: Μπορούν να λειτουργήσουν και ως ψευδο-αναλογικές έξοδοι με το σύστημα PWM (Pulse Width Modulation).



Εικόνα 13: Αναλογικά pins Arduino Uno

Στην άλλη πλευρά του Arduino, με τη σήμανση ANALOG IN όπως φαίνεται και στην εικόνα 6, υπάρχει μια ακόμη σειρά από 6 pin, αριθμημένα από το 0 ως το 5. Η τάση αναφοράς μπορεί να ρυθμιστεί με μια εντολή στο 1.1V (μεταξύ 2 και 5V) τροφοδοτώντας εξωτερικά με αυτή την τάση το pin με τη σήμανση AREF που βρίσκεται στην απέναντι

Ανάπτυξη ρομποτικού οχήματος κινούμενου με ανίχνευση γραμμής

πλευρά της πλακέτας. Έτσι, αν τροφοδοτηθεί ο ακροδέκτης AREF με 3.3V και στη συνέχεια διαβάσει κάποιον ακροδέκτη αναλογικής εισόδου στο οποίο εφαρμόζεται τάση 1.65V, το Arduino θα επιστρέψει την τιμή 512.

Ενώ οι ακροδέκτες που τροφοδοτούν την πλακέτα μας είναι οι ακόλουθοι:

Ακροδέκτες	Χαρακτηριστικά
Vin	Η τάση εισόδου της πλακέτας, όταν χρησιμοποιεί εξωτερική πηγή ενέργειας. Η τροφοδοσία τάσης γίνεται μέσω αυτού του ακροδέκτη.
5V	Η τάση που χρησιμοποιείται από τα διάφορα μέρη της πλακέτας και το μικροελεγκτή είναι 5V. Η τάση αυτή, την οποία δίνει αυτός ο ακροδέκτης, είναι είτε η τάση 5V που δίνει η σύνδεση με USB, είτε η ρυθμισμένη τάση που δίνεται μέσω του Vin.
3.3V	Η τάση αυτή παράγεται από το ολοκληρωμένο FTDI. Το όριο άντλησης ρεύματος είναι 50mA.
GND	Είσοδοι γείωσης.

Πίνακας 2: Ακροδέκτες τροφοδοσίας

2.5 Παρουσίαση της κατασκευής μου

Ανάπτυξη ρομποτικού οχήματος κινούμενου με ανίχνευση γραμμής

Τώρα θα σας παρουσιάσω πως υλοποιήθηκε η εφαρμογή Arduino για την κίνηση του δίτροχου οχήματος με την βοήθεια ανίχνευσης γραμμής. Αρχικά να πω ότι πρέπει να γίνει εγκατάσταση του Arduino IDE στον υπολογιστή μας. Τα βήματα που ακολούθησα ήταν:

- Άνοιγμα του προγράμματος Arduino
- Αρχείο->Δημιουργία
- Αποθηκεύουμε το Αρχείο με την ονομασία mycar.ino

Στην αρχή του κώδικα το πρώτο πράγμα που δηλώνουμε είναι οι βιβλιοθήκες που θα χρησιμοποιήσουμε. Σε κάθε arduino πρόγραμμα ακολουθείται η ίδια λογική. Εγώ χρειάστηκα δύο βιβλιοθήκες τις οποίες έπρεπε να τις εγκαταστήσω χειροκίνητα καθώς δεν συμπεριλαμβάνονταν στις ήδη υπάρχουσες βιβλιοθήκες. Οι κινήσεις μου ήταν:

- Σχέδιο->Συμπερίληψη Βιβλιοθήκης→Διαχείριση Βιβλιοθηκών
- Στο πεδίο της αναζήτησης αναγράφουμε τις βιβλιοθήκες που θέλουμε. Στην δικιά μου περίπτωση ήταν οι ακόλουθες

```
#include <EEPROMex.h>
#include <EEPROMVar.h>
#include <QTRSensors.h>
```

Περιγραφή βιβλιοθηκών:

- Αυτή είναι μια βιβλιοθήκη για μεταβλητές που πρέπει να παραμείνουν μέσω των κύκλων ισχύος. Αποθηκεύει αυτόματα σε EEPROM κάθε φορά που αλλάζει η τιμή.
- Η βιβλιοθήκη QTRSensors είναι για να μας επιστρέφει δεδομένα από τα ανιχνεύσιμα led μαύρης γραμμής στο μπροστινό τμήμα της κατασκευής μας.
- Η βιβλιοθήκη EEPROMex αποτελεί επέκταση της τυπικής βιβλιοθήκης Arduino EEPROM. Επεκτείνει τη λειτουργικότητα της αρχικής βιβλιοθήκης Arduino EEPROM με: Ανάγνωση, γραφή σε βασικούς τύπους. Αυτό περιλαμβάνει bytes, longs, ints, floats και doubles. Ανάγνωση, γραφή σε μεμονωμένα κομμάτια. Αυτό βοηθά στην αποτελεσματική χρήση της περιορισμένης μνήμης EEPROM. Και γενικότερα είναι μια βοηθητική βιβλιοθήκη.

Ανάπτυξη ρομποτικού οχήματος κινούμενου με ανίχνευση γραμμής

```
#define NUM_SENSORS 8 // number of sensors used
#define TIMEOUT 2500 // waits for 2500 microseconds for sensor outputs to go low
#define EMITTER_PIN 2 // emitter is controlled by digital pin 2
#define buz 11 // buzzer for calibration process
#define cal 10 // jumper-on → calibration, jumper-off → racing
#define motor_left_1 3
#define motor_left_2 5
#define motor_right_1 6
#define motor_right_2 9
#define leftSpeed 90
#define rightSpeed 90
#define max_leftSpeed 120
#define max_rightSpeed 120
#define KP 0.2
#define KD 5
int lastError = 0;
```

Με την δήλωση των `motor 1` & `motor 2` στην αρχή δημιουργούμε τον πρώτο και τον δεύτερο κινητήρα σαν παραμέτρους **`motor_left_1,2`** & **`motor_right_1,2`**. Στην συνέχεια οι `KP` και `KD` παράμετροι. Ένας ελεγκτής `PID` έχει τρεις παραμέτρους `Kp`, `Ki` και `Kd` που θα μπορούσαν να επηρεάσουν την απόδοση εξόδου. Ένα ρομπότ διαφορικής οδήγησης ελέγχεται από έναν ελεγκτή `PID`. Οι πληροφορίες κλάσης ανιχνεύονται από έναν αισθητήρα πυξίδας. Η ταχύτητα κίνησης προς τα εμπρός διατηρείται σταθερή. Ο ελεγκτής `PID` είναι σε θέση να ελέγξει τις πληροφορίες κλάσης για να ακολουθήσει μια δεδομένη κατεύθυνση.

Γνωρίζουμε ότι ο αναλογικός όρος, `Kp`, βασίζεται εξ ολοκλήρου στο σφάλμα και ότι, ας πούμε, το διπλασιασμό του σφάλματος θα σήμαινε διπλασιασμό του `Kp` (εφαρμογή αναλογικής δύναμης). Αυτό συνεπάγεται ότι η αύξηση του `Kp` είναι αποτέλεσμα της ροής του ρομπότ προς τη λάθος κατεύθυνση, έτσι `Kp` αυξάνεται για να διασφαλιστεί ότι το ρομπότ πηγαίνει προς τη σωστή κατεύθυνση ή τουλάχιστον προσπαθεί να μειώσει το σφάλμα καθώς ο χρόνος περνάει και έτσι μια αύξηση στο `Kp` θα επηρεάσει το ρομπότ με τέτοιο τρόπο ώστε να ρυθμίσετε την κεφαλή του ρομπότ έτσι ώστε να παραμένει στην σωστή διαδρομή.

Ο παράγωγος όρος, `Kd`, βασίζεται στον ρυθμό μεταβολής του σφάλματος, οπότε μια αύξηση στο `Kd` υποδηλώνει ότι ο ρυθμός αλλαγής του σφάλματος έχει αυξηθεί με την πάροδο του χρόνου, έτσι ώστε το διπλάσιο του σφάλματος να οδηγήσει σε διπλασιασμό της δύναμης. Μία αύξηση διπλάσια από την αλλαγή στην επικεφαλίδα του ρομπότ θα πραγματοποιηθεί εάν η επικεφαλίδα του ρομπότ διπλασιαστεί κατά λάθος από το προηγούμενο αποτέλεσμα ανάδρασης. Το `Kd` αναγκάζει το ρομπότ να αντιδράσει πιο γρήγορα καθώς το σφάλμα αυξάνεται. Το `Kp` βάσει κανόνα πρέπει να είναι μικρότερο του `Kd`. Στην συνέχεια ορίζονται οι ταχύτητες των τροχών, ο αριθμός των αισθητήρων, ποιός αισθητήρας πρέπει να είναι στο κέντρο της μαύρης γραμμής ώστε να την ακολουθήσει και επίσης ένας χρόνος αναμονής μέχρι οι αισθητήρες να αναγνωρίσουν την μαύρη γραμμή και η δήλωση του `EMITTER PIN` ανάβει τα `IR leds` που σημαίνει ότι ανάβουν όταν αναγνωρίζουν το μαύρο χρώμα της γραμμής.

Ανάπτυξη ρομποτικού οχήματος κινούμενου με ανίχνευση γραμμής

Αφού έχουμε δηλώσει όλα τα στοιχεία που χρειαζόντουσαν, επόμενο βήμα είναι να διαβάσουμε τους qtrrc αισθητήρες. Στην παράμετρο “sensorValues” ανατίθενται όλες οι τιμές που διαβάζουν ξεχωριστά και οι 8 αισθητήρες σε κάθε κίνηση του αυτοκινήτου. Οι τιμές κυμαίνονται απο 0 εως 10 αναλογικές έξοδοι.

```
// sensors 1 through 8 are connected to digital pins {4,7,14,15,16,17,18,19}
QTRSensorsRC qtrrc((unsigned char[]) {4,7,14,15,16,17,18,19},
  NUM_SENSORS, TIMEOUT, EMITTER_PIN);
unsigned int sensorValues[NUM_SENSORS];

void car_forward(int sp1, int sp2){
  analogWrite(motor_left_1, sp1);
  analogWrite(motor_right_2, sp2);
}

void setup() {
```

Ο κώδικας μας έχει μια void setup() η οποία είναι δεδομένο τμήμα κώδικα σε κάθε arduino αρχείο. Εκτελείται στην αρχή του προγράμματος για μία μόνο φορά. Στην δικιά μου κατασκευή θέλω αυτο το τμήμα να κάνει αυτόματο καλιμπράρισμα και να θέσει τους κινητήρες σε μηδενική ταχύτητα. Οι τρεις επόμενοι παράμετροι θα μας βοηθήσουν στον τύπο του PID control για την αποφυγή αλλαγής πορείας. Έχω συνδέσει και ένα buzzer για να σημάνει την έναρξη του οχήματος. Έπισης κράταω στην μνήμη του arduino τις μετρήσεις μου.

```
void setup(){
  delay(500);
  pinMode(buz, OUTPUT);
  pinMode(cal, INPUT_PULLUP);
  pinMode(motor_left_1, OUTPUT);
  pinMode(motor_left_2, OUTPUT);
  pinMode(motor_right_1, OUTPUT);
  pinMode(motor_right_2, OUTPUT);
  digitalWrite(motor_left_2, LOW);
  digitalWrite(motor_right_1, LOW);
  Serial.begin(9600);

  if (digitalRead(cal)==LOW){
    Serial.println("start calibration...");
    analogWrite(buz, 200); // buzzer-on for start of calibration
    for (int i = 0; i < 400; i++) { // make the calibration take about 10 seconds
      qtrrc.calibrate(); // reads all sensors 10 times at 2500 us per read (i.e. ~25 ms per call)
    }
    analogWrite(buz, 0); // buzzer-off for end of calibration

    // print the calibration minimum values measured when emitters were on
    for (int i = 0; i < NUM_SENSORS; i++) {
      unsigned int x=qtrrc.calibratedMinimumOn[i];
      EEPROM.writeInt(i*2, x);
      Serial.print(x);
      Serial.print(' ');
    }
  }
}
```

Ανάπτυξη ρομποτικού οχήματος κινούμενου με ανίχνευση γραμμής

Όταν το EMITTER pin είναι ενεργό πήρε κάποιες μετρήσεις από τον sensor απόστασης που έχουμε τοποθετήσει στην αρχή. Τώρα ζητάμε να τις εκτυπώσει και στην οθόνη μας (MIN, MAX). Όταν τελειώσει η αρχικοποίηση του των αισθητηρων μας αφαιρούμε το jumper και ξεκινάει. Και ύστερα κάνει την σύγκριση με τις τιμές που έχει από το καλιμπράρισμα που κάνει κάθε φορά. Και αναλόγως την τιμή που πέρνει κάθε φορά βοηθάει και το αυτοκίνητο μας να στρίβει κάθε φορά χωρίς μεγάλη απόκλιση.

```
Serial.println();
// print the calibration maximum values measured when emitters were on
for (int i = 0; i < NUM_SENSORS; i++) {
    unsigned int x=qtrrc.calibratedMaximumOn[i];
    EEPROM.writeInt(i*2+16, x);
    Serial.print(x);
    Serial.print(' ');
}
Serial.println();
Serial.println();
}
while (digitalRead(cal)==LOW){
    Serial.println("remove jumper...");
    delay(1000);
}
Serial.println("start racing...");
//recall MIN & MAX
qtrrc.calibrate();
for (int i = 0; i < NUM_SENSORS; i++) {
    qtrrc.calibratedMinimumOn[i]=EEPROM.readInt(i*2);
    Serial.print(qtrrc.calibratedMinimumOn[i]);
    Serial.print(' ');
}
Serial.println();
for (int i = 0; i < NUM_SENSORS; i++) {
    qtrrc.calibratedMaximumOn[i]=EEPROM.readInt(i*2+16);
    Serial.print(qtrrc.calibratedMaximumOn[i]);
    Serial.print(' ');
}
Serial.println();
Serial.println();
```

Ανάπτυξη ρομποτικού οχήματος κινούμενου με ανίχνευση γραμμής

```
}  
  
void loop(){  
  // read calibrated sensor values and obtain a measure of the line position from 0 to 5000  
  unsigned int position=qttrrc.readLine(sensorValues);  
  int error = position - 3500;  
  
  int motorSpeed = KP * error + KD * (error - lastError);  
  int left = leftSpeed - motorSpeed;  
  int right = rightSpeed + motorSpeed;  
  if (left>max_leftSpeed){  
    left=max_leftSpeed;  
  }  
  if (right>max_rightSpeed){  
    right=max_rightSpeed;  
  }  
  if (left<0){  
    left=0;  
  }  
  if (right<0){  
    right=0;  
  }  
  car_forward(left,right);  
  lastError = error;  
}
```

Αυτή η βιβλιοθήκη σας επιτρέπει να χρησιμοποιήσετε τη μέθοδο `calibrated()` για να βαθμονομήσετε εύκολα τους αισθητήρες σας για τις συγκεκριμένες συνθήκες που θα συναντήσετε. Η βαθμονόμηση των αισθητήρων μπορεί να οδηγήσει σε σημαντικά πιο αξιόπιστες μετρήσεις αισθητήρων, οι οποίες με τη σειρά τους μπορούν να σας βοηθήσουν να απλοποιήσετε τον κώδικα σας από τότε. Ως εκ τούτου, συνιστούμε να δημιουργήσετε μια φάση βαθμονόμησης στην ρουτίνα αρχικοποίησης της εφαρμογής σας. Αυτό μπορεί να είναι τόσο απλό όσο μια σταθερή διάρκεια κατά την οποία επαναλάβετε κλήση της μεθόδου `calibrated()`. Κατά τη διάρκεια αυτής της φάσης βαθμονόμησης, θα χρειαστεί να εκθέσετε τους αισθητήρες ανάκλασης στις πιο ελαφρές και σκοτεινότερες αναγνώσεις που θα συναντήσουν. Για παράδειγμα, εάν έχετε κάνει έναν ακολουθία γραμμής, θα θέλετε να το σύρετε κατά μήκος της γραμμής κατά τη διάρκεια της φάσης βαθμονόμησης, ώστε ο κάθε αισθητήρας να μπορεί να πάρει μια ανάγνωση για το πόσο σκοτεινή είναι η γραμμή και πόσο φωτεινό είναι το έδαφος.

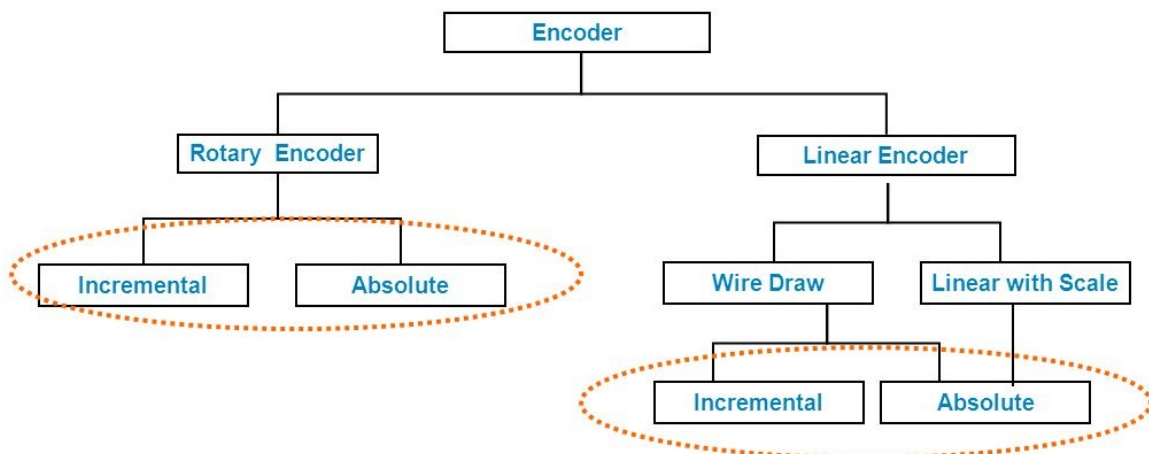
ΚΕΦΑΛΑΙΟ 3

ΚΩΔΙΚΟΠΟΙΗΤΕΣ (ENCODERS)

3.1 Εισαγωγή

3.1.1 Εισαγωγή στους κωδικοποιητές

Όπως αναφέρθηκε και πιο πάνω που έκανα μια γενική εισαγωγή στους κωδικοποιητές, πρόκειται για εξαρτήματα διαφορετικού μεγέθους και διαφορετικών λειτουργιών που κάνουν ευκολότερη την διαχείριση ενός αυτοματοποιημένου συστήματος. Ανάλογα με την χρήση τους και ανάλογα με το τι θέλουμε να μας επιστρέψουν ως ανατροφοδότηση χωρίζονται σε ειδικές κατηγορίες. Είναι δύο τα είδη των κωδικοποιητών γραμμικοί και περιστροφικοί. Ενώ ανάλογα με την έξοδο που παίρνουμε κατηγοριοποιείτε σε δύο είδη, τους βαθμιδωτούς και τους απόλυτους. Εγώ χρησιμοποίησα τους μαγνητικούς κωδικοποιητές της Pololu που χρησιμοποιεί αισθητήρα μαγνητικού δίσκου και αισθητήρα θέσης αλλά πιο αναλυτικά θα μιλήσουμε παρακάτω για αυτούς. Το παρακάτω διάγραμμα μας δίνει μια καθαρή εικόνα για το πως ξεχωρίζουμε τους κωδικοποιητές μεταξύ τους.



Εικόνα 14: Ομαδοποίηση κωδικοποιητών

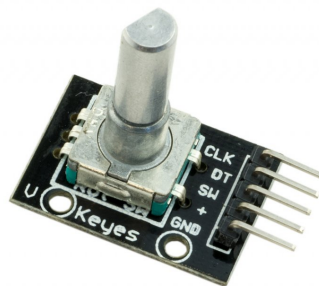
Ανάπτυξη ρομποτικού οχήματος κινούμενου με ανίχνευση γραμμής

➤ Rotary encoder(Περιστροφικός κωδικοποιητής)

Οι περιστροφικοί κωδικοποιητές είναι ηλεκτρομηχανικές συσκευές που χρησιμοποιούνται για την ανίχνευση σε μυριάδες εφαρμογές-στους κινητήρες που συνδέονται με drivers καθώς και τα αυτοματοποιημένα μηχανήματα από ηλεκτρονικά είδη ευρείας κατανάλωσης, ανελκυστήρες και παρακολούθηση της ταχύτητας του μεταφορέα μέχρι τον έλεγχο θέσης σε αυτοματοποιημένες βιομηχανικές μηχανές και ρομποτική. Παρακολουθούν την περιστροφή των αξόνων κινητήρα για τη δημιουργία ψηφιακών πληροφοριών θέσης και κίνησης. Είτε βαθμιαίοι είτε απόλυτοι, μαγνητικοί ή οπτικοί περιστροφικοί κωδικοποιητές στρέφουν την περιστροφή του άξονα κινητήρα για να παράγουν ψηφιακές πληροφορίες θέσης και κίνησης. Η χρήση τους συνιστάται σε βιομηχανικά και εμπορικά σχέδια. Οι περιστροφικοί κωδικοποιητές καθορίζονται από τον συντελεστή μορφής, το επίπεδο ανθεκτικότητας και την ανάλυση.

➤ .Linear encoder(γραμμικός κωδικοποιητής)

Ένας γραμμικός κωδικοποιητής είναι ένας αισθητήρας, μετατροπέας ή αναγνώστης που συνδυάζεται με μία κλίμακα που κωδικοποιεί τη θέση. Ο αισθητήρας διαβάζει την κλίμακα για να μετατρέψει την κωδικοποιημένη θέση σε ένα αναλογικό ή ψηφιακό σήμα, το οποίο στη συνέχεια μπορεί να αποκωδικοποιηθεί στη θέση του από ψηφιακό αναγνώστη (DRO) ή ελεγκτή κίνησης. Η κίνηση μπορεί να προσδιοριστεί με αλλαγή της θέσης με την πάροδο του χρόνου. Οι τεχνολογίες γραμμικού κωδικοποιητή περιλαμβάνουν οπτικό, μαγνητικό, επαγωγικό, χωρητικό και eddy current. Τα ρεύματα του Eddy (ονομαζόμενα επίσης ρεύματα Foucault) είναι βρόχοι ηλεκτρικού ρεύματος που προκαλούνται μέσα σε αγωγούς από ένα μεταβαλλόμενο μαγνητικό πεδίο στον αγωγό λόγω του νόμου επαγωγής του Faraday.



Εικόνα 15: Rotary encoder



Εικόνα 16: Linear magnetic encoder

3.2 Χαρακτηριστικά περιστροφικού κωδικοποιητή

Ας αναπτύξω αναλυτικότερα για το τι εστί περιστροφικός κωδικοποιητής και πως λειτουργεί επακριβώς. Οι περιστροφικοί κωδικοποιητές παρακολουθούν την κίνηση του άξονα κινήρα για μυριάδες κομμάτια βιομηχανικού εξοπλισμού και εμπορικές συσκευές. Για βιομηχανικές εφαρμογές, χρησιμοποιούνται συνήθως βαθμιδωτοί κωδικοποιητές (χρησιμοποιούνται όταν απαιτείται μόνο σχετική θέση ή κοστολογούν ένα πρόβλημα) με κινήρες επαγωγής εναλλασσόμενου ρεύματος. Αντίθετα, οι απόλυτοι κωδικοποιητές (οι οποίοι δίνουν διαφορετική δυαδική έξοδο σε κάθε θέση, έτσι ώστε η θέση του άξονα να καθορίζεται απολύτως) συνδυάζονται συχνά με κινήρες χωρίς ψήκτες μόνιμου μαγνήτη σε σερβο-εφαρμογές. Συχνά, η ανατροφοδότηση του κωδικοποιητή χρησιμοποιείται για να διασφαλιστεί ο συγχρονισμός των θέσεων του στατικού κινήρα και των περιστροφικών θέσεων του τροφοδοτούμενου ρεύματος, επομένως το ρεύμα εφαρμόζεται στις περιελίξεις όταν οι περιστροφικοί μαγνήτες βρίσκονται μέσα σε ένα κατάλληλο εύρος θέσης (για μεγιστοποίηση της ροπής). Οι περιστροφικοί κωδικοποιητές καθορίζονται από τον συντελεστή μορφής, το επίπεδο ανθεκτικότητας και την ανάλυση. Για τους βαθμιδωτούς κωδικοποιητές, η ανάλυση ορίζεται ως μετρήσεις ανά στροφή. Για απόλυτους κωδικοποιητές μονής στροφής, είναι θέσεις ανά στροφή, εκφρασμένες ως λέξη πολλαπλών δυαδικών ψηφίων. Οι κωδικοποιητές πολλών ταχυτήτων (εκείνοι που παρακολουθούν πολλαπλές περιστροφές 360 °) καθορίζονται από τις θέσεις ανά στροφή του άξονα εισόδου και τον αριθμό εσωτερικής σχέσης μετάδοσης των στροφών. Η απαιτούμενη ανάλυση εξαρτάται από τον αριθμό των θέσεων που απαιτούν μέτρηση. Αν μια μηχανή πρέπει να μετρήσει τη διαδρομή 25 ίντσών, οδηγεί σε αυξήσεις 0.001 ίντσες, για παράδειγμα, τότε απαιτεί έναν απόλυτο κωδικοποιητή με ανάλυση 25.000 μονάδων. Δηλαδή, η ανάλυση εδώ ορίζεται συνήθως με όρους bit: Ένας κωδικοποιητής 12 bit, για

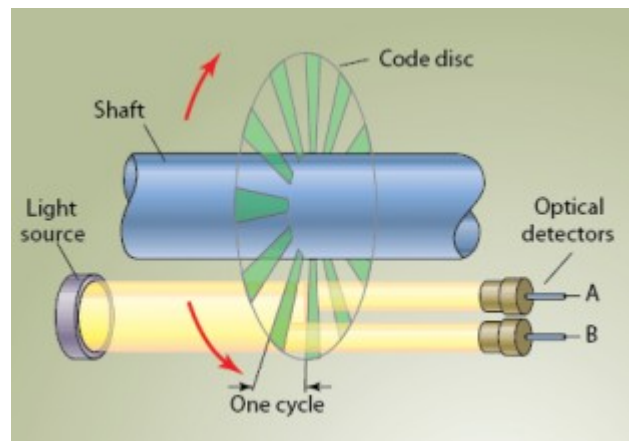
Ανάπτυξη ρομποτικού οχήματος κινούμενου με ανίχνευση γραμμής

παράδειγμα, αναφέρεται στον δυαδικό αριθμό 212 που είναι 4.096 στο δεκαδικό σύστημα. Έτσι, ένας κωδικοποιητής 12 bit έχει ανάλυση 4.096 βαθμών.

Η ακρίβεια σε αντίθεση με την ανάλυση εξαρτάται από τις αλληλεπιδράσεις ολόκληρου του συστήματος με την εφαρμογή και είναι ανιχνεύσιμη στο δίσκο κωδικοποίησης, μια υπομονάδα που θα αναλύσω περαιτέρω πιο κάτω, και την απόκλιση μεταξύ πραγματικής και θεωρητικής θέσης. Ένας καλός 12 ή 13-bit κωδικοποιητής, για παράδειγμα, είναι ακριβής στην μισή τιμή του λιγότερου σημαντικού bit.

Επαναληψιμότητα, είναι η ικανότητα του κωδικοποιητή να διαβάζει το ίδιο σημείο κάθε φορά που ο άξονας βρίσκεται σε μια συγκεκριμένη θέση, ορίζεται ως η απόκλιση της πραγματικής θέσης του κωδικοποιητή μεταξύ μεταγενέστερων πανομοιότυπων αναγνώσεων κώδικα. Αν και δεν έχει καμία σχέση με την ακρίβεια, είναι συνήθως τέσσερις έως δέκα φορές καλύτερα για τις μετρήσεις μας τις τελικές.

Ο πιο διαδεδομένος περιστροφικός κωδικοποιητής με κριτήριο τον σχεδιασμό του, είναι ο οπτικός περιστροφικός κωδικοποιητής. Αποτελούνται από μια πηγή φωτός LED, ανιχνευτή φωτός, δίσκο κώδικα και επεξεργαστή σήματος.



Εικόνα 17: Optical rotary encoder

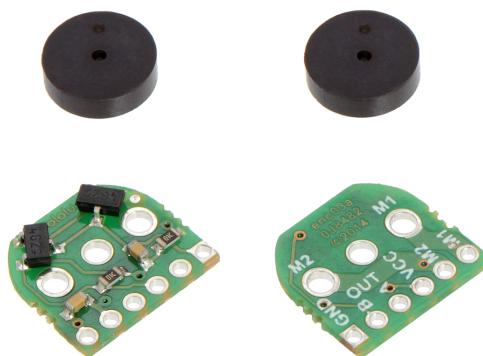
Μπορεί οι οπτικοί κωδικοποιητές ως προς τον σχεδιασμό τους να είναι καλύτεροι αλλά στις κατασκευές μας προτιμάμε τους μαγνητικούς κωδικοποιητές. Οι μαγνητικοί κωδικοποιητές είναι εγγενώς ανθεκτικότεροι και λειτουργούν αξιόπιστα κάτω από κραδασμούς, απότομες μεταβολές και υψηλή θερμοκρασία. Η είσοδος των μαγνητικών θραυσμάτων μπορεί να υποβαθμίσει την απόδοση ενός περιστροφικού μαγνητικού κωδικοποιητή, αλλά δεν υπάρχουν άλλοι παράγοντες. Επομένως, συχνά χρησιμοποιούνται περιστροφικοί μαγνητικοί κωδικοποιητές αντί για οπτικούς κωδικοποιητές. Η παθητική ποικίλλη απροθυμία ή οι μαγνητικές λωρίδες σε έναν περιστρεφόμενο κωδικοποιημένο ρότορα, τροχό ή ζώνη ανιχνεύονται είτε από αισθητήρα

Ανάπτυξη ρομποτικού οχήματος κινούμενου με ανίχνευση γραμμής

Hall είτε από μαγνητικό αισθητήρα. Η ταχύτητα και η ακρίβεια της θέσης του κινητήρα υπαγορεύουν ποιο από τα δύο είναι καλύτερα προσαρμοσμένο για μια εφαρμογή.

Οι περιστροφικοί μαγνητικοί κωδικοποιητές μπορούν να έχουν τη μορφή μικρών, φθηνών συσκευών που χρησιμοποιούνται σε εφαρμογές μεγάλου όγκου, όπως συστήματα πέδησης αντιμπλοκαρίσματος οχημάτων ή εξελιγμένες μονάδες που απαιτούν εργασίες ελέγχου κίνησης, όπως βιομηχανικά συστήματα αυτοματισμού και ιατρικό εξοπλισμό. Οι διαθέσιμες παραλλαγές είναι, βαθμιδωτά και απόλυτα μοντέλα, εκδόσεις χωρίς επαφή και ρουλεμαν και μονάδες στις οποίες η περιστρεφόμενη μονάδα και το σώμα του κωδικοποιητή είναι ουσιαστικά ξεχωριστές υποσυνιστώσες.

Εν ολίγοις, οι περισσότεροι μαγνητικοί κωδικοποιητές με Hall-effect χρησιμοποιούν έναν τροχό που συνδέεται με τον άξονα του κινητήρα για να ανιχνευθεί και ο τροχός αυτός μαγνητίζεται με βόρειους και νότιους πόλους γύρω από την περίμετρό του. Συνήθως κατασκευάζεται από ένα χυτευμένο με έγχυση φερριτή ενσωματωμένο με τη συστοιχία πόλων. Για να απεικονίσει, ένας κωδικοποιητής 15 mm περιέχει μια ρόδα διαμέτρου 7,6 mm, μαγνητισμένο με 32 πόλους (16 βόρεια και 16 νότια) χρησιμοποιώντας ένα στατικό αναπόσπαστο εξάρτημα. Το μέγεθος του εξαρτήματος συνήθως περιορίζει τον αριθμό των πόλων που μπορούν να ενσωματωθούν. Ο μέγιστος αριθμός των πόλων για έναν 32-πολικό κινητήρα είναι περίπου 0,75 mm, το μικρότερο πρακτικό μέγεθος που μπορεί να χειριστεί το εξάρτημα αυτό. Σκεφτείτε μια συνήθη ρύθμιση χρησιμοποιώντας ένα δίσκομαγνήτη και τρεις ψηφιακούς αισθητήρες Hall-effect (120 ° ηλεκτρικός χώρος) σε ένα κύκλωμα που λαμβάνει σήματα επικοινωνίας από τον τροχό. Οι αισθητήρες Hall επηρεάζουν τον διακόπτη όταν περνούν οι βόρειοι και νότιοι πόλοι.

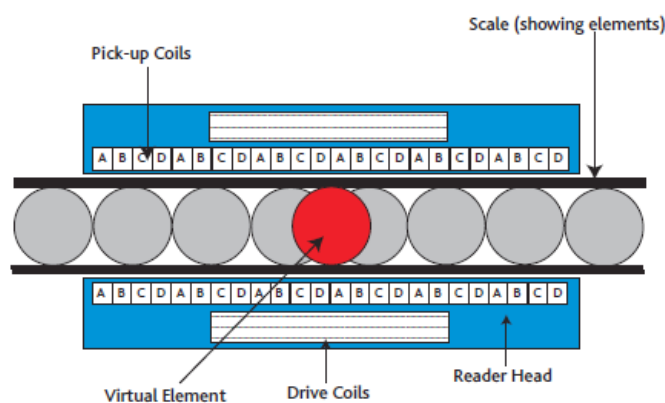


Εικόνα 18: Pololu magnetic encoders

3.3 Χαρακτηριστικά γραμμικού κωδικοποιητή

Ανάπτυξη ρομποτικού οχήματος κινούμενου με ανίχνευση γραμμής

Οι γραμμικοί κωδικοποιητές κινούνται σε μια ευρεία κλίμακα και μια κεφαλή ανάγνωσης που περιέχει ένα συγκρότημα πηνίων που υποστηρίζονται ηλεκτρονικά, τα οποία παρέχουν τετράγωνο σήμα ανάδρασης ή ημιτονοειδές-συνημιτονοειδές σήμα που επιτρέπουν άμεση ενσωμάτωση στις σερβομηχανισμένες εφαρμογές. Αυτοί οι κωδικοποιητές λειτουργούν με βάση την αρχή της ηλεκτρομαγνητικής επαγωγής. Αυτό το ηλεκτρομαγνητικό πεδίο δημιουργείται προκαλώντας ένα 10 kHz ημιτονοειδές ρεύμα μέσω ενός μόνο πηνίου μέσα στην κεφαλή ανάγνωσης. Αυτό το πεδίο αλληλεπιδρά με τα στοιχεία χρωμίου νικελίου που περιέχονται στην κλίμακα. Ένα σύνολο τεσσάρων πηνίων εντοπίζει τις ποικίλλες μορφές σημάτων που στη συνέχεια συνδυάζονται και επεξεργάζονται από το ηλεκτρονικό κύκλωμα για να παράξει ένα σήμα που ποικίλλει όσο η κεφαλή ανάγνωσης μετακινείται κατά μήκος της κλίμακας. Εξαρτόμενοι από τη θέση του κεφαλιού ανάγνωσης καθώς περνάει πάνω από το κάθε στοιχείο, η μετατόπιση φάσης αυτού του σήματος λήψης σε σχέση με το σήμα οδήγησης θα κυμαίνεται μεταξύ 0 και 360 μοίρες. Η Ψηφιακή Επεξεργασία Σήματος Υψηλής Ταχύτητας (DSP) μετατρέπει το αναλογικό σήμα σε ένα πρότυπο σήμα βιομηχανίας, το οποίο παράγει επίσης τον περιοδικό παλμό δείκτη αναφοράς. Οι γραμμικοί κωδικοποιητές μας επιστρέφουν ως ανατροφοδότηση μια απόλυτη θέση δεν χρησιμοποιεί μπαταρίες ούτε στατική μνήμη για να κρατήσει προηγούμενες μετρήσεις. Η κλίμακα περιβάλλεται από έναν ασάλινο σωλήνα που φιλοξενεί στοιχεία ακρίβειας χρωμίου νικελίου. Για απόλυτες και συγκεκριμένης φάσης μετρήσεις, ενσωματωμένα τμήματα κώδικα κλίμακας είναι τοποθετημένα ανάμεσα από αυτά τα στοιχεία με τέτοιο τρόπο ώστε να μην επηρεάζεται η γεωμετρία του συστήματος.



Εικόνα 19: Τοποθέτηση πηνίων

Στην παραπάνω εικόνα φαίνεται η τοποθέτηση των πηνίων στην κεφαλή ανάγνωσης. Υπάρχουν έξι συγκροτήματα πηνίων που λαμβάνουν πληροφορίες, κάθε συγκρότημα περιέχει τέσσερις πανομοιότυπες περιελίξεις που βρίσκονται ανα ένα διάστημα. Ως απόρροια αυτής της τοποθέτησης κάθε πηνίο ανα συγκρότημα είναι τοποθετημένο σε συστοιχία. Όσο αφορά την ανατροφοδότηση που παίρνουμε από τους γραμμικούς κωδικοποιητές αφορά αποκλειστικά και μόνο την αγορά που θέλουμε να πραγματοποιήσουμε. Κατά την επιλογή ενός γραμμικού κωδικοποιητή, το πρώτο πράγμα

Ανάπτυξη ρομποτικού οχήματος κινούμενου με ανίχνευση γραμμής

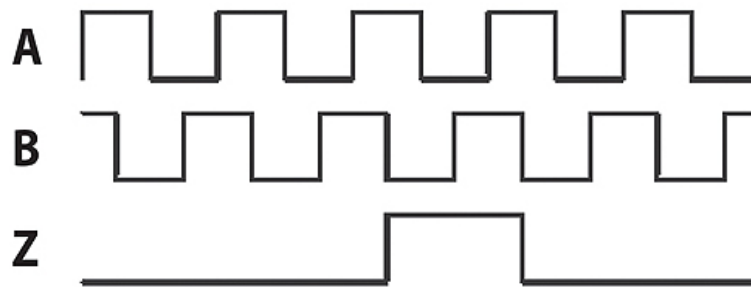
που πρέπει να λάβετε υπόψη είναι ποιος τύπος ανατροφοδότησης είναι απαραίτητος για την εφαρμογή - απόλυτη ή βαθμιαία.

Οι απόλυτοι κωδικοποιητές αποδίδουν μια μοναδική ψηφιακή τιμή σε κάθε θέση, η οποία τους επιτρέπει να διατηρούν ακριβείς πληροφορίες θέσης, ακόμη και όταν χάνονται οι τροφοδοσίες.

Οι βαθμιδωτοί κωδικοποιητές λειτουργούν με τη δημιουργία ενός συγκεκριμένου αριθμού παλμών ανά μονάδα κίνησης και την καταμέτρηση αυτών των παλμών καθώς κινείται το φορτίο. Επειδή απλώς καταμετρούν παλμούς, οι βαθμιδωτοί κωδικοποιητές θα χάσουν τη θέση τους εάν διακοπεί η παροχή ρεύματος. Προκειμένου να προσδιοριστεί η πραγματική θέση του φορτίου κατά την εκκίνηση ή την εκ νέου εκκίνηση, απαιτείται μια ακολουθία αναζήτησης. Αυτό σημαίνει ότι ο αισθητήρας (και το φορτίο) πρέπει να μετακινηθεί σε μια θέση αναφοράς και από εκεί μπορεί να αρχίσει να καθορίζει τη θέση του φορτίου. Λάβετε υπόψη ότι ακόμα και αν η πραγματική θέση του φορτίου κατά την εκκίνηση ή εκ νέου εκκίνηση δεν είναι συγκεκριμένη, η εκτέλεση μιας ακολουθίας ομαδοποίησης μπορεί να είναι ανεπιθύμητη από πλευράς χρόνου και παραγωγικότητας. Αυτό είναι ιδιαίτερα σημαντικό σε εφαρμογές με πολλές μεταβολές στις καταστάσεις τους και χαμηλές ταχύτητες, όπως εργαλεία, όπου η επιστροφή μπορεί να είναι μια χρονοβόρα διαδικασία.

Η έξοδος για τους απόλυτους και βαθμιδωτούς κωδικοποιητές διαφέρει και είναι επίσης ένα θέμα για την ενσωμάτωση τους στο σύστημα ελέγχου του συστήματος. Οι απόλυτοι γραμμικοί κωδικοποιητές παράγουν ψηφιακή έξοδο, ή "λέξη", που δηλώνει την πραγματική θέση της μονάδας. Η ανάλυση για έναν απόλυτο κωδικοποιητή καθορίζεται από τον αριθμό των δυαδικών ψηφίων στη λέξη. Οι βαθμιδωτοί κωδικοποιητές παράγουν έξοδο τετραγωνισμού, με δύο κανάλια που είναι 90 μοίρες εκτός φάσης. (Η έξοδος δύο καναλιών επιτρέπει την παρακολούθηση τόσο της θέσης όσο και της κατεύθυνσης. Εάν χρειάζεται μόνο μία θέση, τότε χρησιμοποιείται μόνο ένα κανάλι.) Μερικοί βαθμιδωτοί κωδικοποιητές παράγουν ένα τρίτο κανάλι με έναν μόνο παλμό, που χρησιμοποιείται ως δείκτης ή θέση αναφοράς για επιστρέφων. Ο αριθμός των παλμών ανά απόσταση (ίντσα ή χιλιοστόμετρο) καθορίζει την ανάλυση ενός αυξητικού κωδικοποιητή. Ωστόσο, η ανάλυση μπορεί να διπλασιαστεί μετρώντας τόσο την εμπρόσθια όσο και την τελική ακμή του παλμού από ένα κανάλι, ή μπορεί να τετραπλασιαστεί μετρώντας τα εμπρόσθια και τα τελικά άκρα των παλμών από τα δύο κανάλια.

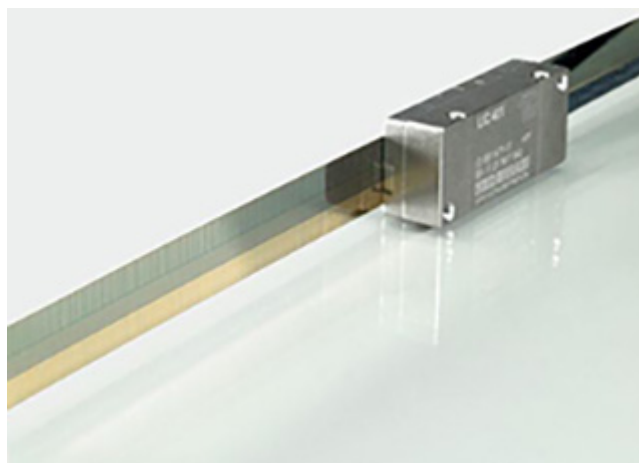
Ανάπτυξη ρομποτικού οχήματος κινούμενου με ανίχνευση γραμμής



Εικόνα 20: Έξοδος τετραγώνων για έναν αυξητικό κωδικοποιητή. Τα κανάλια A και B είναι εκτός φάσης κατά 90 μοίρες. Το κανάλι Z έχει έναν παλμό, ο οποίος χρησιμοποιείται ως αναφορά για την προσπέλαση.

Μόλις ληφθεί η απόφαση σχετικά με την βαθμιδωτή ή την απόλυτη ανατροφοδότηση, η επόμενη εξέταση είναι αν η τεχνολογία ανίχνευσης πρέπει να είναι οπτική ή μαγνητική. Ενώ οι οπτικοί κωδικοποιητές έχουν ιστορικά τη μοναδική επιλογή για αναλύσεις μικρότερες από 5 μm , οι βελτιώσεις στην τεχνολογία μαγνητικής κλίμακας τους επιτρέπουν πλέον να επιτύχουν αναλύσεις κάτω του 1 μm .

Οι οπτικοί κωδικοποιητές χρησιμοποιούν μια πηγή φωτός και ένα φωτοανιχνευτή για τον προσδιορισμό της θέσης, αλλά η χρήση του φωτός τις καθιστά ευαίσθητες στη βρωμιά και τα συντρίμμια, γεγονός που μπορεί να διαταράξει το σήμα. Η απόδοση των οπτικών κωδικοποιητών επηρεάζεται σε μεγάλο βαθμό από το κενό μεταξύ του αισθητήρα και της κλίμακας, το οποίο πρέπει να ρυθμιστεί και να διατηρηθεί σωστά για να διασφαλιστεί ότι η ακεραιότητα του σήματος δεν τίθεται σε κίνδυνο. Αυτό σημαίνει ότι η τοποθέτηση πρέπει να γίνει με προσοχή και να αποφευχθούν οι κραδασμοί και οι απότομες κινήσεις



Εικόνα 21: Οπτικός γραμμικός κωδικοποιητής

Ανάπτυξη ρομποτικού οχήματος κινούμενου με ανίχνευση γραμμής

Οι μαγνητικοί κωδικοποιητές χρησιμοποιούν μαγνητική κεφαλή ανάγνωσης και μαγνητική κλίμακα για τον προσδιορισμό της θέσης. Σε αντίθεση με τους οπτικούς κωδικοποιητές, οι μαγνητικοί κωδικοποιητές δεν επηρεάζονται από την βρωμιά, τα σκουπίδια ή απο υγρά. Οι κραδασμοί και οι δονήσεις είναι επίσης λιγότερο πιθανό να επηρεάσουν τους μαγνητικούς κωδικοποιητές. Ωστόσο, είναι ευαίσθητα σε μαγνητικά τσιπ, όπως είναι ο χάλυβας ή ο σίδηρος, καθώς μπορεί να παρεμβαίνουν στο μαγνητικό πεδίο.

Ενώ οι γραμμικοί κωδικοποιητές είναι συχνά ένα πρόσθετο στοιχείο ενός συστήματος, σε πολλές περιπτώσεις τα οφέλη τους υπερτερούν της πρόσθετης εργασίας και του κόστους. Για παράδειγμα, σε εφαρμογές με σφαιρικό κοχλία, μπορεί να επιλεγεί ένας κοχλίας χαμηλότερης ακρίβειας εάν χρησιμοποιείται ένας γραμμικός κωδικοποιητής, καθώς ο ανατροφοδοτούμενος κωδικοποιητής επιτρέπει στον ελεγκτή να αντισταθμίσει τα σφάλματα θέσης που εισάγονται από τη βίδα.

3.4 Σφάλμα κωδικοποίησης

Το σφάλμα του βαθμιδωτού κωδικοποιητή αποτελείται από τρεις τύπους: 1) σφάλμα ποσοτικοποίησης, 2) σφάλμα οργάνου, 3) σφάλμα παρεμβολής κύκλου (εάν ο κωδικοποιητής είναι έτσι εξοπλισμένος). Υπάρχει σφάλμα ποσοτικοποίησης επειδή ο κωδικοποιητής δεν μπορεί να υποδεικνύει κίνηση που εμφανίζεται σε ένα κβαντικό ψήφισμα στα σημεία μετάβασης. Αυτή είναι η συνιστώσα σφάλματος υψηλότερης συχνότητας και επαναλαμβάνει κάθε κβαντική κίνηση εισόδου. Σε έναν τέλειο κωδικοποιητή χωρίς μηχανική, οπτική ή ηλεκτρονική απόκλιση από την ιδανική, η σωστή γωνιακή θέση του άξονα εισόδου για μια δεδομένη ένδειξη ορίζεται ως η γωνιακή θέση στο μέσο μεταξύ της μετάβασης από την επόμενη χαμηλότερη ένδειξη στην μετάβαση για την επόμενη υψηλότερη ανάγνωση. Το σφάλμα ποσοτικοποίησης είναι η απόκλιση του άξονα εισόδου από τη μεσαία θέση για μια δεδομένη ένδειξη, με το μέγιστο σφάλμα $\pm 1/2$ της γωνιακής περιστροφής μεταξύ δύο διαδοχικών δυφίων. Για παράδειγμα, ένας περιστροφικός βαθμιδωτός κωδικοποιητής που παράγει 360 παλμούς ανά περιστροφή έχει σφάλμα ποσοτικοποίησης $\pm 1/2$ γωνιακών βαθμών.

Το σφάλμα οπτικού γραμμικού κωδικοποιητή προκαλεί μεγάλη ανησυχία για τις ευρείες εφαρμογές του. Η θερμοκρασία παίζει σημαντικό ρόλο στους πραγματικούς παράγοντες σφάλματος περιβάλλοντος και επηρεάζει την ακρίβεια του κωδικοποιητή. Σε αυτή τη μελέτη, ένα μοντέλο σφάλματος αυξητικού οπτικού γραμμικού κωδικοποιητή κατασκευάζεται και εκτιμάται από μια προηγούμενη ανάλυση, την επεξεργασία σφαλμάτων της προσαρμογής ελαχίστων τετραγώνων, την αποσύνθεση του εμπειρικού τρόπου λειτουργίας και τις μεθόδους διασταυρούμενης συσχέτισης. Τα σφάλματα των διαφορετικών θερμοκρασιών μπορούν να χρησιμοποιηθούν για τον διαχωρισμό σφαλμάτων κατά τον υπολογισμό της διασταυρούμενης συσχέτισης.

ΚΕΦΑΛΑΙΟ 4

ΥΛΟΠΟΙΗΣΗ ΚΑΤΑΣΚΕΥΗΣ

4.1 Εξαρτήματα

Στο κεφάλαιο αυτό αναφερθώ αποκλειστικά στην υλοποίηση της κατασκευής όσο αφορά το πρακτικό κομμάτι. Θα περιγράψω τα εξαρτήματα που χρησιμοποίησα καθώς και τις διαφορές μεταξύ άλλων εξαρτημάτων που δεν προτίμησα.

- 2X Micro metal gearmotor bracket
- 2X Pololu wheel 70x8mm black
- 1X Ball caster plastic-3/8"
- 1X Reflectance sensor array QTR-8RC Pololu
- 1X Magnetic encoder pair kit for micro metal gearmotors 12CPR 2.7
- 1X L293D motor driver
- 2X Micro gear motor 2600 RPM
- 1X Arduino UNO
- 1X Plexiglas base
- 1X LiPo Battery 7.4V

4.1.1 Micro metal gearmotor bracket

Το συγκεκριμένο εξάρτημα πρόκειται για αυτό που κράταγε τους κινητήρες πάνω στο σασί αυτοκινήτου σταθερούς. Αυτός ο συμπαγής βραχίονας επιτρέπει την εύκολη συναρμολόγηση δημοφιλών μεταλλικών ηλεκτροκινητήρων τύπου Sanyo. Ο πλαστικός βραχίονας περικλείει τα άλλα εκτεθειμένα γρανάζια και οι γλωττίδες συναρμογής συλλαμβάνουν τα παξιμάδια για εύκολη εγκατάσταση.



Εικόνα 22: Metal gearmotor bracket

4.1.2 Wheel kit 70x8mm Pololu

Αυτοί οι μαύροι πλαστικοί τροχοί συμπιπτουν με τους άξονες D των 3mm σε πολλούς κινητήρες, συμπεριλαμβανομένων των micro gearmotors που χρησιμοποιήσα για το αυτοκινητάκι και κάνουν μια ωραία λύση για μικρά ρομπότ. Οι τροχοί έχουν ελαστικά σιλικόνης και έχουν διάμετρο 70 mm (2,75 "). Δύο πρόσθετες οπές στερέωσης για 4-40 βίδες καθιστούν δυνατή τη χρήση του τροχού με αρκετούς από τους κεντρικούς διανομέα. Το γεγονός ότι έχουν μεγάλη διάμετρο βοηθάει στην όσο το δυνατόν ακρίβεια της κίνησης.



Εικόνα 23: Pololu wheels

4.1.3 Plastic Ball caster

Αυτό το kit μπάλας περιλαμβάνει ένα μαύρο περίβλημα ABS, μια πλαστική σφαίρα διαμέτρου 3/8 ", δύο αποστάτες (πάχους 1/16" και 1/8 ") και δύο σετ κοχλιών # 2. Το συνολικό ύψος του σφαιρικού σφαιρικού κυλίνδρου 0.4 ", μπορεί να αυξηθεί σε περίπου 0,6" με τη χρήση των διαχωριστικών που περιλαμβάνονται. Οι δύο βίδες μπορούν να χρησιμοποιηθούν για τη συναρμολόγηση του σφαιριδίου στο σασί (η απόσταση μεταξύ των οπών βιδών είναι 0,53 "). Σημειώστε ότι οι τροχοί είναι σχεδιασμένοι για μικρά ρομπότ. Δεν προορίζονται να φέρουν περισσότερο από λίγα κιλά. Με μεγαλύτερο βάρος από αυτό, η τριβή μεταξύ της σφαίρας και του περιβλήματος πιθανότατα θα την κρατήσει από το να κυλήσει καλά. Βοηθάει στην είναι ο πιο εύκολος τρόπος να ισορροπήσεις ένα δίτροχο αμαξίδιο.



Εικόνα 24: Ball caster

4.1.5 Reflectance sensor array QTR-8RC Pololu

Η συστοιχία αισθητήρων ανάκλασης QTR-8RC προορίζεται ως αισθητήρας απόστασης, αλλά μπορεί να χρησιμοποιηθεί ως αισθητήρας εγγύτητας γενικής χρήσης ή ανάκλασης. Η τεχνολογία του αποτελείται από οκτώ ζεύγη πομπού IR δέκτη (φωτοτρανζίστορ) ομοιόμορφα τοποθετημένα σε διαστήματα των 0,375 "(9,525 mm). Για να χρησιμοποιήσετε έναν αισθητήρα, πρέπει πρώτα να φορτίσετε τον κόμβο εξόδου εφαρμόζοντας τάση στον ακροδέκτη OUT μπορεί να διαβάσει την ανακλαστικότητα με την απόσυρση της εξωτερικά παρεχόμενης τάσης και του χρονισμού για πόσο χρονικό διάστημα χρειάζεται η τάση εξόδου να αποσυντεθεί λόγω του ενσωματωμένου φωτομετατροπέα. Αυτή η ικανότητα της μονάδας QTR-8RC να απενεργοποιήσει την ισχύ LED έχει πολλά πλεονεκτήματα:

- Δεν απαιτείται μετατροπέας αναλογικού σε ψηφιακό (ADC)

Ανάπτυξη ρομποτικού οχήματος κινούμενου με ανίχνευση γραμμής

- Βελτιωμένη ευαισθησία σε αναλογική έξοδο τάσης-διαιρέτη
- Παράλληλη ανάγνωση πολλών αισθητήρων είναι δυνατή με τους περισσότερους μικροελεγκτές
- Η παράλληλη ανάγνωση επιτρέπει τη βελτιστοποιημένη χρήση της δυνατότητας ενεργοποίησης ισχύος LED

Οι έξοδοι είναι όλες ανεξάρτητες, αλλά τα LED είναι διατεταγμένα σε ζεύγη για να μειώσουν κατά το ήμισυ την κατανάλωση ρεύματος. Οι λυχνίες LED ελέγχονται από ένα MOSFET με μια πύλη που τραβιέται κανονικά ψηλά, επιτρέποντας την απενεργοποίηση των LED με ρύθμιση της πύλης MOSFET σε χαμηλή τάση. Η απενεργοποίηση των ενδεικτικών λυχνιών ενδέχεται να είναι πλεονεκτική για τον περιορισμό της κατανάλωσης ενέργειας όταν οι αισθητήρες δεν χρησιμοποιούνται ή για τη μεταβολή της αποτελεσματικής φωτεινότητας των LED μέσω του ελέγχου PWM. Αυτός ο αισθητήρας σχεδιάστηκε για να χρησιμοποιείται με την πλακέτα παράλληλα με την επιφάνεια που ανιχνεύει.

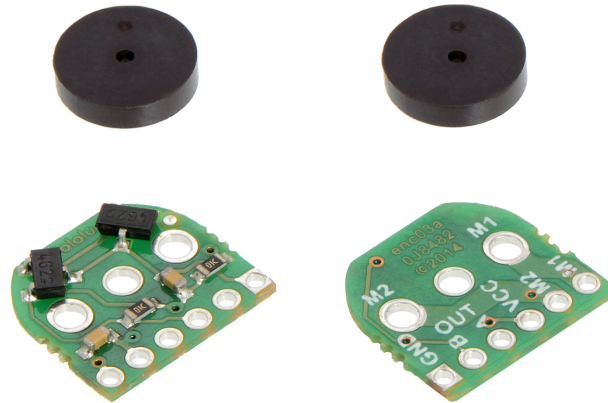


Εικόνα 25: Reflectance sensor array QTR-8RC

4.1.6 Magnetic encoder pair kit for micro metal gearmotors 12CPR 2.7

Αυτό το kit περιλαμβάνει δύο εξαρτήματα αισθητήρων Hall διπλού καναλιού και δύο μαγνητικούς δίσκους 6 πόλων που μπορούν να χρησιμοποιηθούν για την προσθήκη τετραγωνικής κωδικοποίησης σε δυο ηλεκτρομειωτήρες με μεταλλικούς μοτέρ με εκτεταμένους οπίσθιους άξονες (οι κινητήρες δεν περιλαμβάνονται σε αυτό το kit). Η κάρτα κωδικοποιητή ανιχνεύει την περιστροφή του μαγνητικού δίσκου και παρέχει μια ανάλυση 12 μετρήσεων ανά περιστροφή του άξονα του κινητήρα κατά την καταμέτρηση και των δύο άκρων και των δύο καναλιών. Για να υπολογίσετε τις μετρήσεις ανά περιστροφή του άξονα εξόδου του κιβωτίου ταχυτήτων, πολλαπλασιάστε τη σχέση μετάδοσης κατά 12.

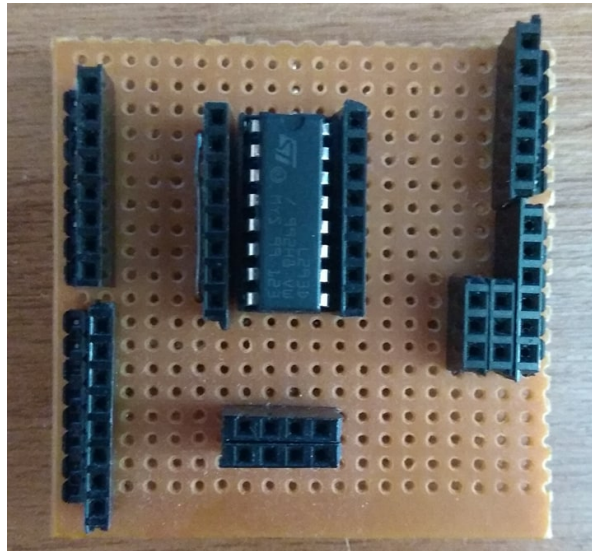
Ανάπτυξη ρομποτικού οχήματος κινούμενου με ανίχνευση γραμμής



Εικόνα 26: Magnetic encoder pair kit for micro metal gearmotors 12CPR 2.7

4.1.7 L293D motor driver

Οι συσκευές L293D είναι half-H drivers υψηλής ισχύος. Το L293D έχει σχεδιαστεί για να παρέχει αμφίδρομα ρεύματα κίνησης μέχρι 600 mA σε τάσεις από 4,5 V έως 36 V. Είναι σχεδιασμένο για την οδήγηση επαγωγικών φορτίων όπως ηλεκτρονόμοι, ηλεκτρομαγνητικές βαλβίδες, DC και διπολικούς βηματικούς κινητήρες, καθώς και άλλα φορτία υψηλού ρεύματος / υψηλής τάσης σε εφαρμογές θετικής παροχής. Κάθε έξοδος είναι ένα πλήρες κύκλωμα κίνησης totem pole, με έναν τρανζίστορ Darlington και μια πηγή ψευδο-Darlington. Οι οδηγοί ενεργοποιούνται ανά ζεύγη, με τους οδηγούς 1 και 2 ενεργοποιημένους από 1,2EN και τους οδηγούς 3 και 4 ενεργοποιημένους από 3,4EN. Το L293D χαρακτηρίζεται για λειτουργία από 0 ° C έως 70 ° C.



Εικόνα 27: L293D shield

4.1.1 Micro gear motor

Πρόκειται για σερβοκινητήρες μεγάλης ακρίβειας. Έχουν λόγο μετάδοσης 10:1 και λειτουργούν μέχρι 12V και έχουν ροπή στρέψης 4 oz-in. και μέγιστη ταχύτητα 2600 RPM. Κάθε micro gearmotor στρέφει έναν D-άξονα 3mm. Στην συνέχεια θέλω να επισημάνω τις διαφορές ενός βηματικού κινητήρα και ενός σερβοκινητήρα. Σε γενικές γραμμές, αυτοί οι τύποι κινητήρων μπορούν να λειτουργούν με τάσεις πάνω και κάτω από τις ονομαστικές τάσεις τους. οι χαμηλότερες τάσεις ενδέχεται να μην είναι πρακτικές και οι υψηλότερες τάσεις θα μπορούσαν να αρχίσουν να επηρεάζουν αρνητικά τη διάρκεια ζωής του κινητήρα.



Εικόνα 28: Micro gearmotor

4.2 Διαφορές βηματικού κινητήρα και σερβοκινητήρα

Για την συγκεκριμένη κατασκευή πρέπει να περιγράψω γιατί διάλεξα έναν σερβοκινητήρα έναντι του βηματικού. Στον χώρο της μηχανολογίας επικρατεί η αλήθεια ότι δεν υπάρχει τέλεια λύση υπάρχει η καλύτερη λύση για το πρόβλημα που αντιμετωπίζετε. Αυτό ισχύει ιδιαίτερα για τους σερβοκινητήρες και τους βηματικούς κινητήρες. Και οι δύο χρησιμοποιούνται ευρέως στη βιομηχανία. Όταν εφαρμόζονται σωστά, ωστόσο, τόσο οι βηματικοί όσο και οι σερβοκινητήρες μπορούν να παρέχουν αποτελεσματική και αξιόπιστη ισχύ για ένα εξαιρετικά επιτυχημένο σύστημα. Για την επιλογή μεταξύ των δύο υπάρχουν πολλά κριτήρια που πρέπει να λαμβάνουμε υπόψη, αλλά τα πιο σημαντικά είναι η ταχύτητα, η επιτάχυνση και το πιο μας συμφέρει οικονομικά.

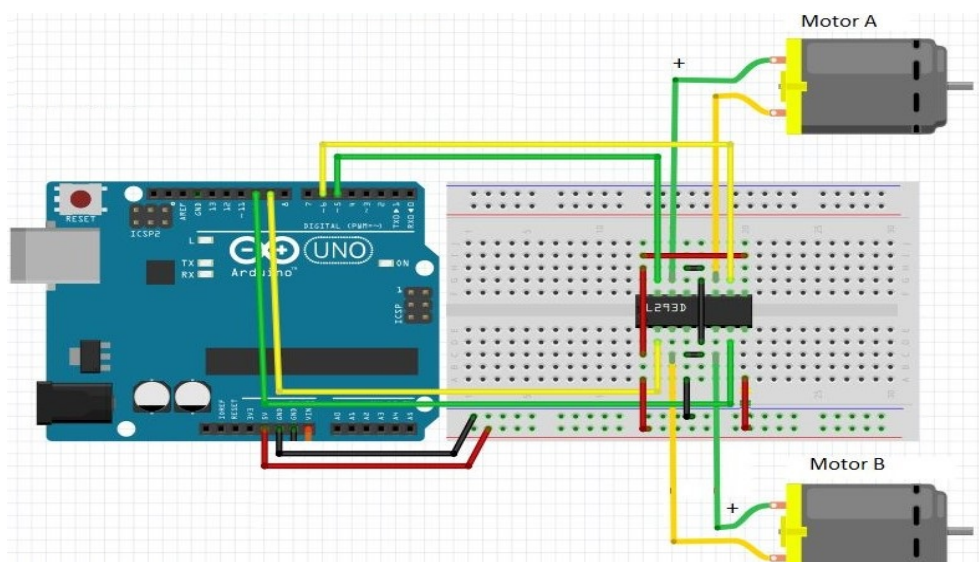
Οι βηματικοί κινητήρες αποτελούνται από έναν δρομέα με μόνιμους μαγνήτες και έναν στάτορα που φέρει τις περιελίξεις. Όταν το ρεύμα διέρχεται από τις περιελίξεις του στάτορα, παράγει μια κατανομή μαγνητικής ροής που αλληλεπιδρά με την κατανομή μαγνητικού πεδίου του δρομέα για να εφαρμόσει μια δύναμη περιστροφής. Το σημαντικό χαρακτηριστικό του είναι πως ακόμα και σε μηδενική ταχύτητα παράγουν πολύ υψηλή ροπή αυτό οφείλεται στις υψηλές μετρήσεις των μαγνητικών του πόλων. Επίσης είναι συμπαγή και πολύ οικονομικά. Το μειονέκτημα τους είναι ότι, οι βηματικοί κινητήρες έχουν περιορισμούς ταχύτητας. Γενικά λειτουργούν καλύτερα στις 1.200 RPM ή χαμηλότερες. Αν και παράγουν μεγάλη ροπή σε μηδενική ταχύτητα, η ροπή πέφτει καθώς αυξάνεται η ταχύτητα.

Όπως και οι βηματικοί κινητήρες, οι σερβοκινητήρες έχουν πολλές υλοποιήσεις. Ας εξετάσουμε τον πιο κοινό σχεδιασμό, ο οποίος ενσωματώνει έναν ρότορα με μόνιμους μαγνήτες και έναν στατικό στάτη με τις περιελίξεις. Και εδώ, το ρεύμα δημιουργεί μια κατανομή μαγνητικού πεδίου που ενεργεί στον ρότορα για να αναπτύξει ροπή. Οι σερβοκινητήρες έχουν σημαντικά χαμηλότερες μετρήσεις πόλων από τους βηματικούς κινητήρες, ωστόσο. Οι σερβοκινητήρες είναι πιο εξελιγμένοι από τους βηματικούς και τρέχουν ταχύτερα, γι αυτό τον λόγο μπορούν να έχουν και κιβώτιο ταχυτήτων. Παρέχουν επίσης πιο σταθερή ροπή στρέψης σε όλο το εύρος στροφών του κινητήρα. Η λειτουργία κλειστού βρόχου επιτρέπει στον ελεγκτή / μονάδα να αναζητήσει ότι το φορτίο παραμένει σε συγκεκριμένη θέση και ο κινητήρας θα κάνει συνεχείς ρυθμίσεις για να τον κρατήσει εκεί. Έτσι, οι σερβοκινητήρες μπορούν να παράσχουν de facto ροπή συγκράτησης.

Οι σερβοκινητήρες προσφέρουν ένα αναμφισβήτητο πλεονέκτημα απόδοσης.

4.3 Συνδεσμολογία εξαρτημάτων

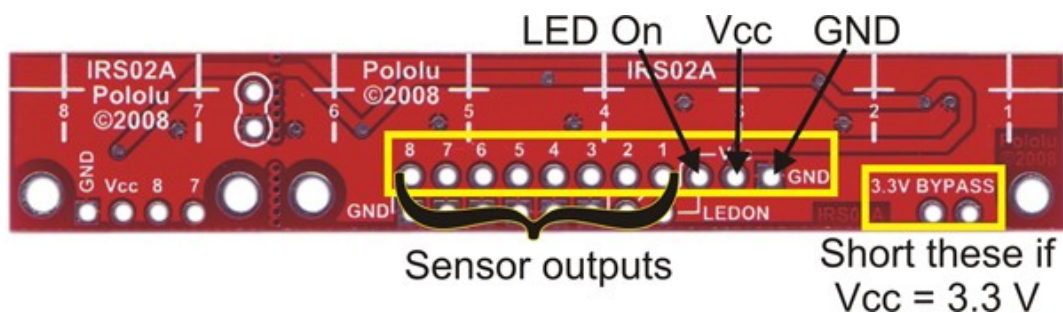
Ο μικροελεγκτής Arduino είναι ο πυρήνας της κατασκευής μας. Το L293D shield το τοποθετούμε πάνω στο arduino μας ως προέκταση. Όλα τα εξαρτήματα μας θα συνδέονται στον arduino ενώ το L293D είναι κατάλληλο για να κινήσει τους δύο κινητήρες μας. Για το πρώτο κινητήρα μας λοιπόν τον έναν ακροδέκτη τον συνδέουμε με το pin «2» του ολοκληρωμένου μας και τον δεύτερο ακροδέκτη στο pin «7» ενώ για τον δεύτερο κινητήρα ισχύει πως ο ένας ακροδέκτης θα συνδεθεί με το pin «10» του ολοκληρωμένου μου ενώ ο δεύτερος ακροδέκτης με το pin «15». Στην συνέχεια, συνδέουμε τον ακροδέκτη «3» του ολοκληρωμένου L293D με τον ακροδέκτη «10» του Arduino μας και τον ακροδέκτη «6» του ολοκληρωμένου L293D με τον ακροδέκτη «9» του Arduino καθώς και τον ακροδέκτη «11» του ολοκληρωμένου L293D με τον ακροδέκτη «5» του Arduino και τέλος συνδέουμε τον ακροδέκτη «14» του ολοκληρωμένου L293D με τον ακροδέκτη «6» του Arduino. Αυτή η συνδεσμολογία αποσκοπεί στην κίνηση των δύο κινητήρων μας, δηλαδή με απλά λόγια ο κώδικας του Arduino μεταφέρει το πρόγραμμα κίνησης στο ολοκληρωμένο L293D και αυτό με την σειρά του μεταφέρει το input στο output των κινητήρων. Όμως όπως γνωρίζουμε ένα ολοκληρωμένο πρέπει να έχει τροφοδοσία και γείωση για να λειτουργήσει για αυτόν τον λόγο συνδέουμε τον ακροδέκτη «8» του L293D με τον ακροδέκτη «5V» του Arduino ενώ για την γείωσή του συνδέουμε τον ακροδέκτη «4» του ολοκληρωμένου με τον ακροδέκτη «GND» του arduino. Για την τροφοδοσία των κινητήρων χρησιμοποίησα μπαταρία Lipo 7.4V και την σύνδεσα στο pin «8» του L293D, γενικά μπορεί με το ολοκληρωμένο μας να συνδεθεί πηγή τάσης έως 36V όμως οι κινητήρες μας αντέχουν από 6V-12V. Αναλυτικά φαίνεται και στην παρακάτω εικόνα.



Εικόνα 29: Συνδεσμολογία Arduino-L293D-Motors

Ανάπτυξη ρομποτικού οχήματος κινούμενου με ανίχνευση γραμμής

Για την συνδεσμολογία του QTR-8RC Reflectance Sensor Array χρησιμοποίησα 11 καλώδια. Ο αισθητήρας μας αποτελείται από 8 led τα οποία ανιχνεύουν χρώμα αυτά τα led συνδέονται με τα pin «4», «7», «14», «15», «16», «17», «18», «19» του Arduino μας. Τα υπόλοιπα 3 καλώδια είναι για την LED ένδειξη του αισθητήρα μας, την γείωση και την τροφοδοσία του στα 5V του Arduino.



Εικόνα 30: QTR-8RC Reflectance Sensor Array pins

ΚΕΦΑΛΑΙΟ 5

ΣΥΜΠΕΡΑΣΜΑΤΑ ΚΑΙ ΠΡΟΟΠΤΙΚΕΣ

5.1 Σύνοψη της πτυχιακής εργασίας

Από την πτυχιακή μου άσκηση συμπεράνα ότι μια κατασκευή μπορεί να υλοποιηθεί με πολύ απλά βήματα και γρήγορα. Διότι η κατασκευή μου πραγματοποιήθηκε ουσιαστικά από ένα κομμάτι plexiglas που βιδώθηκαν επάνω ο μικροελεγκτής Arduino, το L293D shield, ένας αισθητήρας ανίχνευσης χρώματος και 2 DC σερβοκινητήρες. Όλα τα παραπάνω συνδέθηκαν μεταξύ τους και δημιούργησαν το δίτροχο μου. Αργότερα υλοποιήσαμε τον κώδικα του arduino το οποίο ήταν ένα δύσκολο κομμάτι καθώς χρησιμοποιήσαμε ένα PID αλγόριθμο ώστε η κίνηση του δίτροχού μας όταν ανιχνεύει την μαύρη γραμμή να μην αποκλείει αρκετά πάνω στις στροφές. Διότι με 2600 RPM οι κινητήρες μας εκδηλώνουν μεγάλη στροφορμή πράγμα που καθιστά αισθητή κάθε μικρή αλλαγή στην κατεύθυνση του οχήματος. Κάθε δίτροχο έχει διαφορετικό PID ελεγκτή. Είδαμε επίσης ότι η σύνδεση του Arduino κώδικα έγινε απλά με ένα USB καλώδιο. Για να ξεκινήσει το δίτροχο μας μόλις του “φορτώσαμε” τον κώδικα έπρεπε να συνδέσουμε την Lipo battery 7.4V με το Arduino μας μέσω ενός 9V Battery Connector to Barrel Jack Plug.



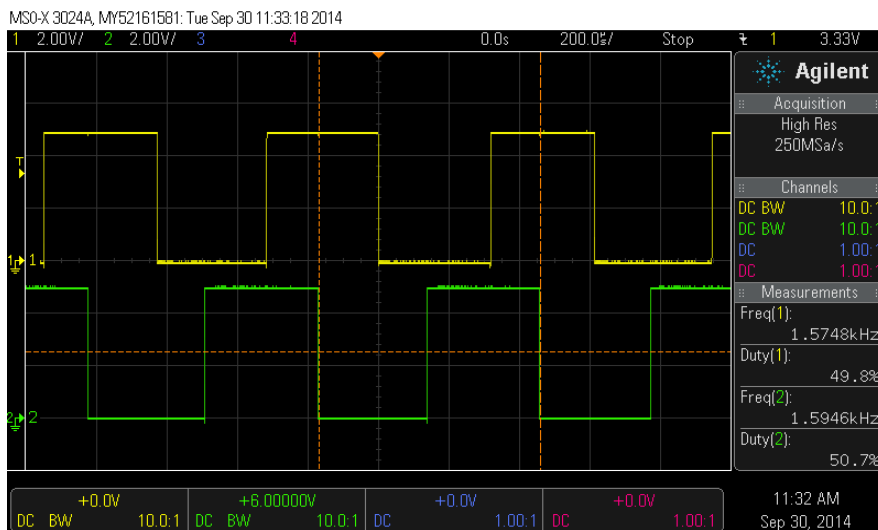
Εικόνα 31: 9V Battery Connector to Barrel Jack Plug

5.2 Προοπτικές

Η βασική ιδέα της κατασκευής μου είναι η αυτοματοποίηση. Το δίτροχό μου κινείται ακολουθώντας μια μαύρη γραμμή και οποιοδήποτε διαδρομή ακολουθεί αυτή η γραμμή. Κινείται στον χώρο βάσει του Arduino μικροελεγκτή και του αντίστοιχου κώδικα αναπτυγμένος σε arduino γλώσσα. Αυτή είναι η βάση της πτυχιακής όμως πάνω σε αυτήν μπορούν να προστεθούν διαφορετικά χαρακτηριστικά. Για παράδειγμα η γραμμή θα μπορούσε να εντοπίζεται μέσω κάμερας που θα αναγνωρίζει το μαύρο αυτοκίνητο και θα επιστρέφει στον Arduino μας την ανάλογη επιλογή. Επίσης θα μπορούσε η κατασκευή μου να επικοινωνεί μέσω bluetooth με κάποια συσκευή android και αυτή μέσω της κάμερας του κινητού θα μπορούσε να αναγνωρίσει την μαύρη γραμμή και να επιστρέφει το output στον Arduino μας μέσω android app. Χρήσιμη θα ήταν και η προσθήκη μαγνητικών κωδικοποιητών στους κινητήρες του δίτροχού μας. Είναι πολύ αποτελεσματικοί με το Arduino γιατί μας δίνουν ένα καθαρό και εύκολο να διαβάσεις ψηφιακό τετραγωνικό σήμα. Καθώς ο κινητήρας γυρίζει, ο δίσκος περιστρέφεται πάνω από τους αισθητήρες. Κάθε φορά που ένας μαγνητικός πόλος διέρχεται από έναν αισθητήρα, ο κωδικοποιητής εξάγει έναν ψηφιακό παλμό, που ονομάζεται επίσης "tick". Η ρύθμιση κωδικοποιητή φαίνεται παρακάτω. Ο κωδικοποιητής έχει δύο εξόδους, μία για κάθε αισθητήρα Hall. Οι αισθητήρες χωρίζονται κατά 90 μοίρες. Αυτό σημαίνει ότι οι τετραγωνικές εξόδους των αισθητήρων είναι 90 μοίρες εκτός φάσης. Αυτό ονομάζεται

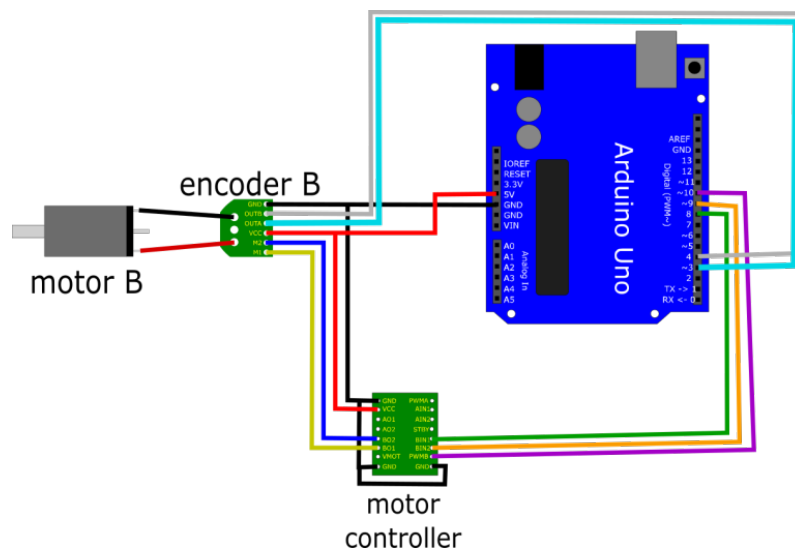
Ανάπτυξη ρομποτικού οχήματος κινούμενου με ανίχνευση γραμμής

έξοδος τετραγώνων. Η παρακάτω εικόνα (που λαμβάνεται από την ιστοσελίδα του Pololu) δείχνει την τυπική έξοδο ενός κωδικοποιητή.



Εικόνα 32: Έξοδος Magnetic Encoder

Γιατί είναι σημαντικό οι παλμοί εξόδου να είναι 90 μοίρες εκτός φάσης; Αυτό μας επιτρέπει να καθορίσουμε τόσο το μέγεθος όσο και την κατεύθυνση της περιστροφής του κινητήρα. Αν η έξοδος A είναι μπροστά από την έξοδο B, τότε ο κινητήρας στρέφεται προς τα εμπρός. Εάν η έξοδος A βρίσκεται πίσω από το B, ο κινητήρας στρέφεται προς τα πίσω. Η σύνδεση των κωδικοποιητών είναι αρκετά απλή. Δείτε το παρακάτω διάγραμμα για λεπτομέρειες σχετικά με τη σύνδεση του κωδικοποιητή για τον κινητήρα B και επαναλάβετε για τον κινητήρα A.



Εικόνα 33: Motor encoder wiring

Ανάπτυξη ρομποτικού οχήματος κινούμενου με ανίχνευση γραμμής

Λάβετε υπόψη ότι ο ακροδέκτης του encoder ΟΥΤΑ πρέπει να συνδεθεί με έναν ακροδέκτη διακοπής υλικού(interrupt pin) (ψηφιακός ακροδέκτης 2 ή 3 σε ένα Arduino Duemilanove ή Uno). Για αυτό το παράδειγμα δεν είναι πραγματικά απαραίτητο να χρησιμοποιείτε interrupt pins, αλλά αν δεν το κάνετε το Arduino σας δεν θα είναι σε θέση να κάνει τίποτα παρά να παρακολουθήσει τον κωδικοποιητή. Τα interrupt pins επιτρέπουν στο Arduino να παρακολουθεί τους κωδικοποιητές ενώ εκτελεί άλλο πρόγραμμα. Είναι επίσης σημαντικό όταν εγκαταστήσετε τους κινητήρες σας να μην τους τοποθετήσετε πολύ κοντά ο ένας στον άλλο. Αν οι κινητήρες σας είναι τοποθετημένοι πίσω-πίσω, οι τροχοί του μαγνητικού κωδικοποιητή θα αλληλεπιδρούν μεταξύ τους. Κρατώντας τους τροχούς του κωδικοποιητή σε απόσταση τουλάχιστον 20mm φαίνεται να αποφεύγονται τυχόν προβλήματα παρεμβολών.



Εικόνα 34: Motor encoder installation

Το παρακάτω σκίτσο απλά εκτυπώνει τις τιμές των δύο μεταβλητών μέτρησης. Όταν μετακινούνται οι κινητήρες, η έξοδος του κωδικοποιητή ενεργοποιεί την κατάλληλη μέθοδο συμβάντων κωδικοποιητή. Οι μέθοδοι συμβάντων κωδικοποιητή αυξάνουν ή μειώνουν τις μεταβλητές μέτρησης, παρακολουθώντας τον αριθμό των ticks από κάθε κωδικοποιητή. Αν ο κινητήρας γυρίζει προς τα εμπρός, η μέτρηση αυξάνεται και μειώνεται αν ο κινητήρας γυρίσει προς τα πίσω.

Ανάπτυξη ρομποτικού οχήματος κινούμενου με ανίχνευση γραμμής

```
#define RH_ENCODER_A 3
#define RH_ENCODER_B 5
#define LH_ENCODER_A 2
#define LH_ENCODER_B 4

// variables to store the number of encoder pulses
// for each motor
volatile unsigned long leftCount = 0;
volatile unsigned long rightCount = 0;

void setup() {
  pinMode(LH_ENCODER_A, INPUT);
  pinMode(LH_ENCODER_B, INPUT);
  pinMode(RH_ENCODER_A, INPUT);
  pinMode(RH_ENCODER_B, INPUT);

  // initialize hardware interrupts
  attachInterrupt(0, leftEncoderEvent, CHANGE);
  attachInterrupt(1, rightEncoderEvent, CHANGE);

  Serial.begin(9600);
}

void loop() {
  Serial.print("Right Count: ");
  Serial.println(rightCount);
  Serial.print("Left Count: ");
  Serial.println(leftCount);
  Serial.println();
  delay(500);
}

// encoder event for the interrupt call
void leftEncoderEvent() {
  if (digitalRead(LH_ENCODER_A) == HIGH) {
    if (digitalRead(LH_ENCODER_B) == LOW) {
      leftCount++;
    } else {
      leftCount--;
    }
  } else {
    if (digitalRead(LH_ENCODER_B) == LOW) {
      leftCount--;
    } else {
      leftCount++;
    }
  }
}
```

```
// encoder event for the interrupt call
void rightEncoderEvent() {
  if (digitalRead(RH_ENCODER_A) == HIGH) {
    if (digitalRead(RH_ENCODER_B) == LOW) {
      rightCount++;
    } else {
      rightCount--;
    }
  } else {
    if (digitalRead(RH_ENCODER_B) == LOW) {
      rightCount--;
    } else {
      rightCount++;
    }
  }
}
```

Βάση λοιπόν αυτής της τεχνοτροπίας θα μπορούσαν οι κωδικοποιητές να μου δίνουν επιπλέον στοιχεία για το δίδροχο μου. Ακόμα επίσης θα μπορούσαμε να προσθέσουμε εντολές μέσω του bluetooth ολοκληρωμένου μας για να γίνει το αυτοκινητο μας τηλεκατευθυνόμενο απο android εφαρμογη. Αξίζει να αναφέρουμε πως ο προγραμματισμός δεν έχει όρια και οι προοπτικές είναι πάρα πολλές για την συγκεκριμένη κατασκευή.

ΒΙΒΛΙΟΓΡΑΦΙΑ

[1] <https://el.wikipedia.org/wiki/Arduino>

[2] <https://www.arduino.cc/en/Main/Software>

[3] <https://www.arduino.cc/en/Main/arduinoShields>

[4] Juang, Hau-Shiue, and Kai-Yew Lurr. "Design and control of a two-wheel self-balancing robot using the arduino microcontroller board." *Control and Automation (ICCA), 2013 10th IEEE International Conference on*. IEEE, 2013.

[5] Badamasi, Yusuf Abdullahi. "The working principle of an Arduino." *Electronics, computer and computation (icecco), 2014 11th international conference on*. IEEE, 2014.

[6] Nayyar, Anand, and Vikram Puri. "A review of Arduino board's, Lilypad's & Arduino shields." *Computing for Sustainable Global Development (INDIACom), 2016 3rd International Conference on*. IEEE, 2016.

[7] <https://en.wikipedia.org/wiki/Encoder>

[8] Arduino, S. A. "Arduino." *Arduino LLC* (2015).

[9] Monk, Simon. "Programming Arduino: Getting Started with Sketches, (Tab)." *McGraw-Hill Education TAB* (2016).

[10] Buechley, Leah, and Michael Eisenberg. "The LilyPad Arduino: Toward wearable engineering for everyone." *IEEE Pervasive Computing 7.2* (2008).

[11] <https://en.wikipedia.org/wiki/ATmega328>

[12] <http://www.anaheimautomation.com/manuals/forms/encoder-guide.php>

[13] https://www.dynapar.com/technology/encoder_basics/magnetic_encoder/

[14] Miyashita, Kunio, Tadashi Takahashi, and M. U. N. E. S. A. D. A. Yamanaka. "Features of a magnetic rotary encoder." *IEEE Transactions on Magnetics 23.5* (1987): 2182-2184.

[15] <https://www.machinedesign.com/motion-control/what-s-difference-between-absolute-and-incremental-encoders>

[16] <http://andrewjkramer.net/motor-encoders-arduino/>

[17] <https://www.machinedesign.com/sensors/basics-rotary-encoders-overview-and-new-technologies-0>

[18] Åström, Karl Johan, and Tore Hägglund. "The future of PID control." *Control engineering practice 9.11* (2001): 1163-1175.

[19] Huang, Guoshing, and Shuocheng Lee. "PC-based PID speed control in DC motor." *Audio, Language and Image Processing, 2008. ICALIP 2008. International Conference on*. IEEE, 2008.