



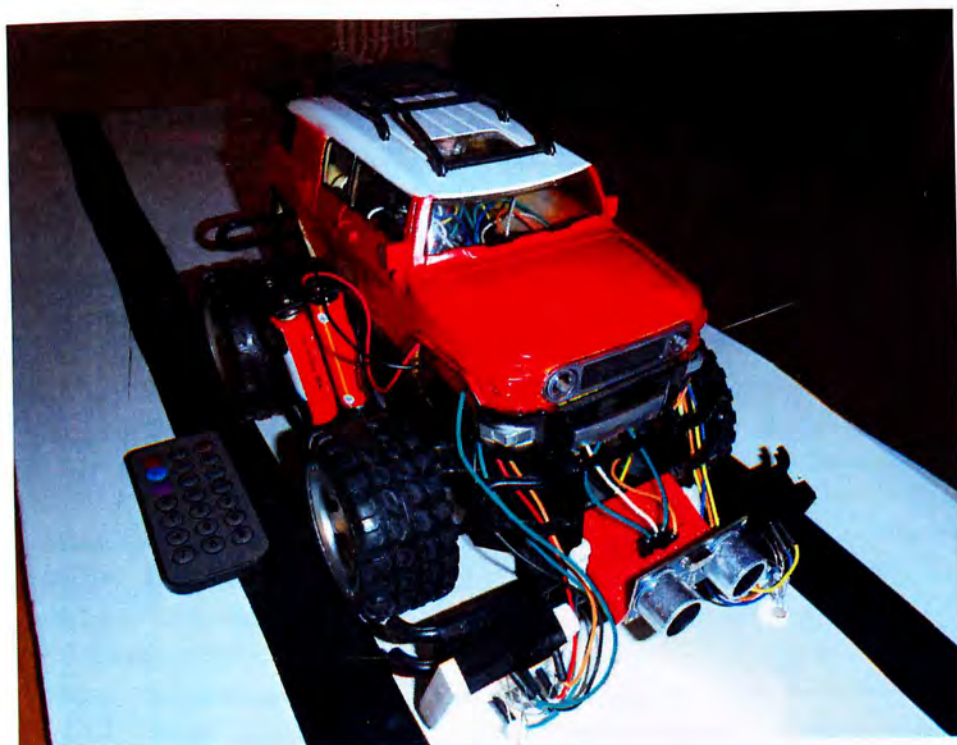
Σχολή Τεχνολογικών Εφαρμογών

Τμήμα Αυτοματισμού

Πτυχιακή εργασία

Θέμα:

«ΚΑΤΑΣΚΕΥΗ ΑΥΤΟΝΟΜΟΥ ΟΧΗΜΑΤΟΣ ΜΕ ΠΡΟΣΑΡΜΟΖΟΜΕΝΟ ΕΛΕΓΧΟ ΠΟΡΕΙΑΣ»



ΕΠΙΒΛΕΠΩΝ ΚΑΘΗΓΗΤΗΣ: ΜΙΧΑΛΗΣ ΠΑΠΟΥΤΣΙΔΑΚΗΣ

ΦΟΙΤΗΤΗΣ: ΜΑΤΘΑΙΟΣ ΠΑΝΤΕΛΗΣ Α.Μ. 34621

Πειραιάς, Σεπτέμβριος 2012

ΒΙΒΛΙΟΘΗΚΗ
ΤΕΙ ΠΕΙΡΑΙΑ

Περιεχόμενα:

Περίληψη.....	1
1. Κεφάλαιο 1° Cruise Control.....	2
1.1 Τι είναι το Cruise Control.....	2
1.2 Η ιστορία του Cruise Control.....	2
1.2.1 Adaptive Cruise Control.....	3
1.3 Πως λειτουργεί.....	3
1.4 Πλεονεκτήματα-Μειονεκτήματα.....	7
2. Κεφάλαιο 2° Adaptive Cruise Control.....	8
2.1 Τι είναι το Adaptive Cruise Control.....	8
2.2 Πως λειτουργεί.....	9
2.2.1 Τύποι.....	9
2.2.2 Χειρισμός.....	11
2.2.3 Απεικόνιση δεδομένων.....	12
2.3 Βελτιώσεις.....	13
2.3.1 Το CMBS.....	13
2.3.2 Audi Pre-Sense.....	15
2.3.3 Volvo ACC with queue assist.....	16
3. Κεφάλαιο 3° Υλοποίηση.....	17
3.1 Η κατασκευή.....	17
3.1.1 Arduino.....	18
3.1.2 DC κινητήρες.....	20
3.1.3 Ολοκληρωμένο κύκλωμα L293NE.....	21
3.1.4 Φωτοαντιστάσεις.....	23
3.1.5 Αισθητήρας υπερήχων.....	24
3.1.6 Δέκτης υπερύθρων.....	26
3.1.7 Το breadboard.....	28
3.1.8 Τροφοδοσία.....	28
3.2 Ο προγραμματισμός.....	29
3.2.1 Δηλώσεις μεταβλητών.....	29
3.2.2 Setup.....	30
3.2.3 Void loop() και δέκτης υπερύθρων.....	31
3.2.4 Φωτοαντιστάσεις και υπολογισμός απόστασης.....	34
3.2.5 Σειριακή θύρα.....	34
3.2.6 Στρίψιμο τροχών.....	35
3.2.7 Φρενάρισμα.....	36

3.2.8	Αυξομείωση ταχύτητας-απόστασης.....	38
3.2.9	Απενεργοποίηση.....	39
4.	Κεφάλαιο 4° Μελλοντικές βελτιώσεις.....	40
4.1	Cooperative Adaptive Cruise Control.....	40
4.1.1	Τι είναι	40
4.1.2	Υποσυστήματα.....	42
4.1.3	Πλεονεκτήματα-μειονεκτήματα.....	43
5.	Βιβλιογραφία	45

Περίληψη:

Σε αυτήν την πτυχιακή εργασία θα προσπαθήσω να εξομοιώσω υπό κλίμακα ένα από τα χρησιμότερα και πιο αναπτυσσόμενα αυτόματα συστήματα ελέγχου στα αυτοκίνητα, τον Προσαρμοζόμενο Έλεγχο Πορείας (Adaptive Cruise Control).

Ο Προσαρμοζόμενος Έλεγχος Πορείας εφαρμόζεται σε ολοένα και περισσότερα αυτοκίνητα και αναπτύσσεται από διάφορες εταιρείες. Είναι ένα ιδιαίτερα χρήσιμο εργαλείο για την άνεση και ασφάλεια του οδηγού και των επιβαινόντων σε ένα όχημα.

Επίσης, οι μεγάλες βελτιώσεις και τα νέα συνεργαζόμενα συστήματα που έχουν κάνει την εμφάνιση τους στον τομέα, θα αναλυθούν και θα παρουσιαστούν με κατανοητό τρόπο.

Εγώ λοιπόν θα προσπαθήσω να μετατρέψω ένα κοινό τηλεκατευθυνόμενο αυτοκινητάκι του εμπορίου σε ένα αυτόνομο όχημα με δικιά του «σκέψη» το οποίο θα κινείται ανάμεσα σε δύο λωρίδες και ανάλογα με την κατάσταση του δρόμου μπροστά του θα αποφασίζει μόνο του εάν μπορεί να επιταχύνει έως την ταχύτητα που του έχω θέσει η εάν χρειάζεται να ελαττώσει ταχύτητα η ακόμα και να φρενάρει και να έρθει σε πλήρη στάση για αποφυγή σύγκρουσης.

Κεφάλαιο 1^ο: Cruise Control (Έλεγχος πορείας)

1.1 Τι είναι το Cruise Control.

Το cruise control ή έλεγχος πορείας, είναι ένα σύστημα που συναντάται σε αυτοκίνητα, φορτηγά και λεωφορεία και το οποίο ελέγχει αυτόματα την ταχύτητα του οχήματος.

Ο οδηγός του οχήματος ορίζει την επιθυμητή ταχύτητα και το σύστημα αναλαμβάνει το γκάζι ώστε να διατηρήσει αυτήν την ορισμένη ταχύτητα.

1.2 Η ιστορία του Cruise Control.

Το Cruise Control έκανε τη εμφάνιση του για πρώτη φορά την δεκαετία του 1910. Ήταν τότε που η αμερικανική εταιρεία Peerless διαφήμισε ένα σύστημα το οποίο «θα διατηρούσε σταθερή ταχύτητα είτε σε ανηφόρα είτε σε κατηφόρα».

Αυτό γινόταν δυνατό με την βοήθεια ενός centrifugal governor ο οποίος ήλεγχε την θέση του γκαζιού και αυξομειώνει τη δύναμη του ανάλογα με την αντίσταση που έβρισκε ο κινητήρας του οχήματος.

Το 1945, ο μηχανικός Ralph Teetor χρησιμοποίησε την ταχύτητα του άξονα του αυτοκινήτου για να υπολογίσει την ταχύτητα του οχήματος και με τη βοήθεια ενός πηνίου ρύθμιζε την ποσότητα του γκαζιού. Το 1958, η Chrysler χρησιμοποίησε αυτή την τεχνολογία στο μοντέλο Imperial το οποίο ήταν και το πρώτο με το συγκεκριμένο σύστημα.

Μια δεκαετία αργότερα, το 1968, ο Daniel Aaron Wisner, μηχανικός της RCA εφηύρε τον ηλεκτρονικό έλεγχο πορείας. Η εφεύρεση αυτή αποτέλεσε την πρώτη εφαρμογή ηλεκτρονικών βοηθημάτων στη οδήγηση αυτοκινήτων και ήταν η αρχή μιας νέας εποχής, αυτής της χρησιμοποίησης υπολογιστών στα αυτοκίνητα.

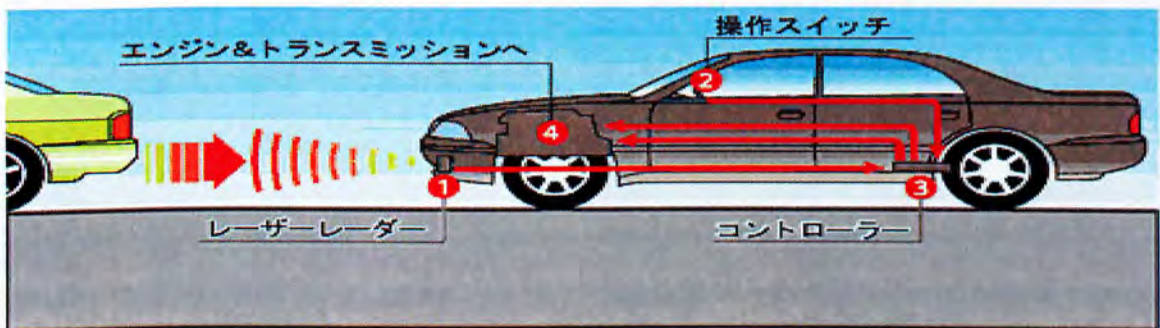
Χρειάστηκαν δύο δεκαετίες μέχρι η Motorola να αναπτύξει ένα ολοκληρωμένο κύκλωμα το οποίο να υλοποιεί την εφεύρεση του Wisner. Έτσι από εκείνη την εποχή ξεκίνησε η εφαρμογή του ηλεκτρονικού ελέγχου πορείας σε σχεδόν όλα τα οχήματα όλων των εταιρειών ως βασικός εξοπλισμός.

Το πλεονέκτημα του ηλεκτρονικού ελέγχου πορείας έναντι του μηχανικού, είναι η πολύ ευκολότερη αναβάθμιση του με συστήματα αποφυγής σύγκρουσης και περεταίρω ελέγχου του κινητήρα.

1.2.1 Adaptive Cruise Control (Προσαρμοζόμενος Έλεγχος Πορείας).

Πλέον, τα περισσότερα νέα μοντέλα χρησιμοποιούν μια ακόμα πιο αναβαθμισμένη έκδοση του ηλεκτρονικού ελέγχου πορείας, τον προσαρμοζόμενο έλεγχο πορείας. Ο προσαρμοζόμενος έλεγχος πορείας έκανε την εμφάνιση του για πρώτη φορά το 1995 από την Mitsubishi με το όνομα "Distance Control" ωστόσο η μειωμένη αποτελεσματικότητά του το καταδίκασε σε αποτυχία.

Η Mitsubishi με τη βοήθεια λέιζερ ανίχνευε το μπροστινό όχημα και προσαρμόζε την ταχύτητα. Όμως η χρήση λέιζερ αποτύγχανε σε συνθήκες δυνατής βροχής ή χιονιού ή σε περίπτωση που το μπροστινό όχημα ήταν βαριά σκονισμένο και δεν υπήρχε αρκετή αντανάκλαση.

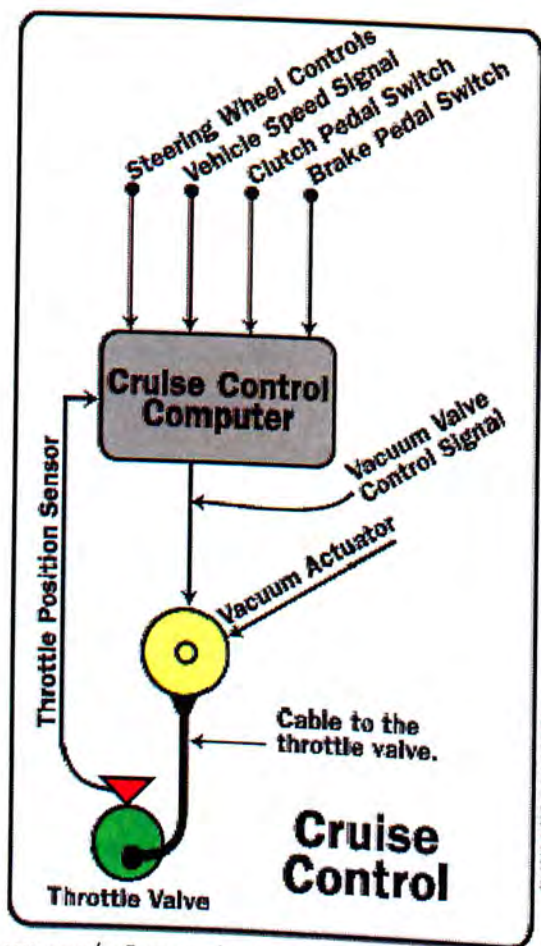


1 Το σύστημα Distance Control της Mitsubishi (βλέπε [5])

Ουσιαστικά, από το 2006 και μετά διάφορες εταιρείες έχουν σχεδόν τελειοποιήσει το σύστημα το οποίο θα το δούμε αναλυτικά στη συνέχεια.

1.3 Πως λειτουργεί.

Το Cruise Control στα μοντέρνα μοντέλα αυτοκίνητων είναι συνήθως απενεργοποιημένο και ενεργοποιείται με το πάτημα διακόπτων οι οποίοι βρίσκονται συνήθως στο τιμόνι του οχήματος. Τα περισσότερα μοντέλα διαθέτουν διακόπτες "on/off" ή και μόνο "on" σε κάποιες περιπτώσεις οι οποίοι πρέπει να πιεστούν όταν εκκινήσει ο κινητήρας του οχήματος. Επίσης συναντάμε διακόπτες "set", "resume", "accelerate" και "coast" οι οποίοι εξυπηρετούν διάφορες λειτουργίες του συστήματος. Μερικές φορές υπάρχει και διακόπτης "cancel".



Ο εγκέφαλος του συστήματος βρίσκεται κάτω από το καπό του αυτοκινήτου και αποτελείται συνήθως από έναν μικροϋπολογιστή. Σε αυτόν συνδέονται τα διάφορα αισθητήρια (πχ ταχύτητας) καθώς και ο σερβομηχανισμός του γκαζιού. Στο διάγραμμα βλέπουμε ένα τυπικό σύστημα ελέγχου πορείας.

Ένα καλό σύστημα επιταχύνει γρήγορα μέχρι τη επιθυμητή ταχύτητα χωρίς όμως να την υπερβεί. Αυτό επιτυγχάνεται με την εφαρμογή της θεωρίας των συστημάτων αυτομάτου ελέγχου. Με έναν κλειστό βρόχο ο ελεγκτής δέχεται τις πληροφορίες από τα αισθητήρια και αναλόγως αυξομειώνει την ταχύτητα του οχήματος.

Τα περισσότερα συστήματα ελέγχου πορείας χρησιμοποιούν ελεγκτές PID για τον αποτελεσματικότερο έλεγχο της ταχύτητας.

Ο PID ελεγκτής είναι ο πιο ενδεδειγμένος για τη συγκεκριμένη εφαρμογή. Εάν για παράδειγμα υπήρχε απλός ελεγκτής P σε περίπτωση ανηφόρας ο ελεγκτής θα έδινε περισσότερη ισχύ ωστόσο πιθανόν το όχημα να συνέχιζε να χάνει ταχύτητα. Εκεί υπερέχει ο PID καθώς μετά από μικρό χρονικό διάστημα ο I θα ανιχνεύσει τη διαφορά ανάμεσα στην απόσταση που έχει διανύσει ενώ ο D ανιχνεύει την αρνητική επιτάχυνση πολύ πριν πέσει η ταχύτητα και επεμβαίνει αμέσως.

Όλα τα συστήματα ελέγχου πορείας πρέπει να απενεργοποιούνται είτε με διακόπτη είτε με το πάτημα του πεντάλ του φρένου. Όσο είναι ενεργοποιημένος ο έλεγχος πορείας, το πεντάλ του γκαζιού μπορεί να χρησιμοποιηθεί για επιτάχυνση ωστόσο μόλις αυτό αφηθεί η ταχύτητα θα επιστρέψει στην ορισμένο από τον οδηγό.

Πολλά συστήματα διαθέτουν ένα όριο στην ταχύτητα την οποία μπορούν να αναπτύξουν μέσω του συστήματος ελέγχου πορείας. Παρόλα αυτά, σε περίπτωση κατηφορικού δρόμου ή πάτημα του πεντάλ του γκαζιού από τον οδηγό, το σύστημα δε θα εμποδίσει το όχημα να επιταχύνει.

1.4 Πλεονεκτήματα-Μειονεκτήματα.

Τα οχήματα με σύστημα ελέγχου πορείας έχουν αρκετά πλεονεκτήματα έναντι των κοινών οχημάτων. Για παράδειγμα:

- Η χρησιμότητα του σε μακρινά ταξίδια μέσω αυτοκινητοδρόμων είναι ιδιαίτερα μεγάλη καθώς εκτός από την ξεκούραση που προσφέρει στον οδηγό, διευκολύνει τον ίδιο να αλλάζει στάση σώματος διαρκώς για τη δική του άνεση.
- Το σύστημα δεν έχει μεγάλες εναλλαγές στον τρόπο που δίνει εντολές στον κινητήρα με αποτέλεσμα την εξοικονόμηση καυσίμου.
- Μερικοί οδηγοί τείνουν να ξεπερνούν τα όρια ταχύτητας των δρόμων το οποίο μπορεί να καταλήξει είτε σε ατύχημα είτε σε κάποιο πρόστιμο. Βέβαια σε περίπτωση κατηφορικού δρόμου ο οδηγός πρέπει να βρίσκεται σε εγρήγορση.

Ωστόσο δεν πρέπει να παραλείπονται κάποια αρνητικά στοιχεία:

- Το γεγονός ότι δε χρειάζεται κάποιος χειρισμός από την πλευρά του οδηγού εκτός της ελαφριάς στροφής του τιμονιού ανά διαστήματα μπορεί να οδηγήσει τον οδηγό στην ύπνωση. Επίσης σε πιθανή αδιαθεσία του οδηγού το όχημα είναι ουσιαστικά ανεξέλεγκτο.
- Οδηγώντας σε συνθήκες βροχής ή παγετού υπάρχει ο κίνδυνος πιθανής ολίσθησης του οχήματος η οποία με το πάτημα του πεντάλ του φρένου για απενεργοποίηση του συστήματος μπορεί να καταλήξει σε απώλεια ελέγχου του οχήματος.

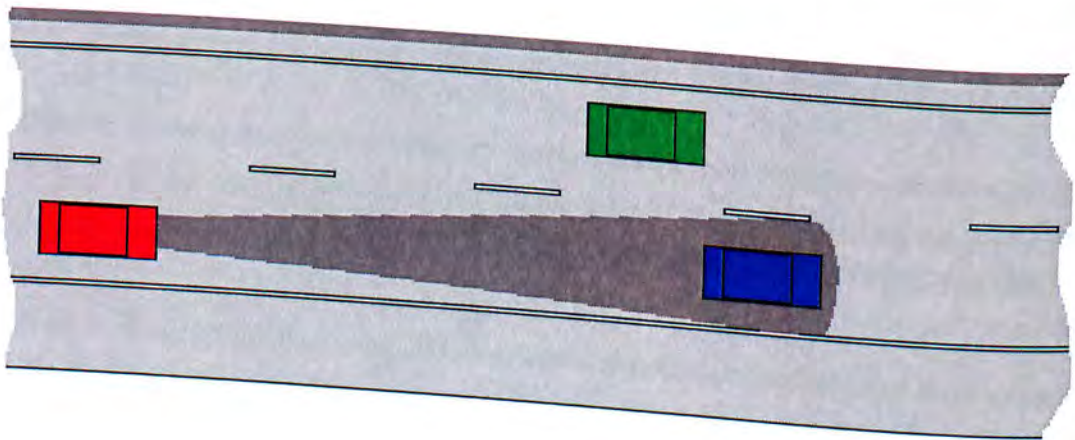
Μπορεί το σύστημα στις περισσότερες περιπτώσεις να εξοικονομεί καύσιμο, ωστόσο σε δρόμο με συνεχείς ανηφόρες και κατηφόρες υπάρχει η περίπτωση ένας έμπειρος οδηγός να έχει καλύτερη αντίδραση από τον υπολογιστή ο οποίος «διαβάζει» μόνο δεδομένα-αριθμούς χωρίς να έχει επαφή με το εξωτερικό περιβάλλον.

Κεφάλαιο 2^ο: Adaptive Cruise Control (Προσαρμοζόμενος έλεγχος πορείας)

2.1 Τι είναι το Adaptive Cruise Control.

Την τελευταία δεκαετία, ενώ το συμβατικό cruise control έχει γίνει βασικός εξοπλισμός σε μεγάλο αριθμό αυτοκινήτων, το σύστημα έχει καταστεί άχρηστο για συνεχή χρήση λόγω της μεγάλης αύξησης της κίνησης στους δρόμους. Για αυτό το λόγο πολλοί κατασκευαστές αυτοκινήτων προσπάθησαν μέσω βελτιώσεων να προσαρμόσουν αυτό το πολύ χρήσιμο σύστημα στα νέα δεδομένα των δρόμων. Το αποτέλεσμα αυτής της προσαρμογής και βελτίωσης του συμβατικού ελέγχου πορείας είναι ο Προσαρμοζόμενος Έλεγχος Πορείας.

Το adaptive cruise control είναι παρόμοιο με το απλό cruise control. Ωστόσο, διαφέρει σε μεγάλο βαθμό καθώς εκτός από έλεγχο της ταχύτητας του οχήματος, το νέο σύστημα αυτόματα αυξομειώνει την ταχύτητα ανάλογα με την κατάσταση του δρόμου μπροστά κρατώντας σταθερή απόσταση από το προπορευόμενο όχημα.



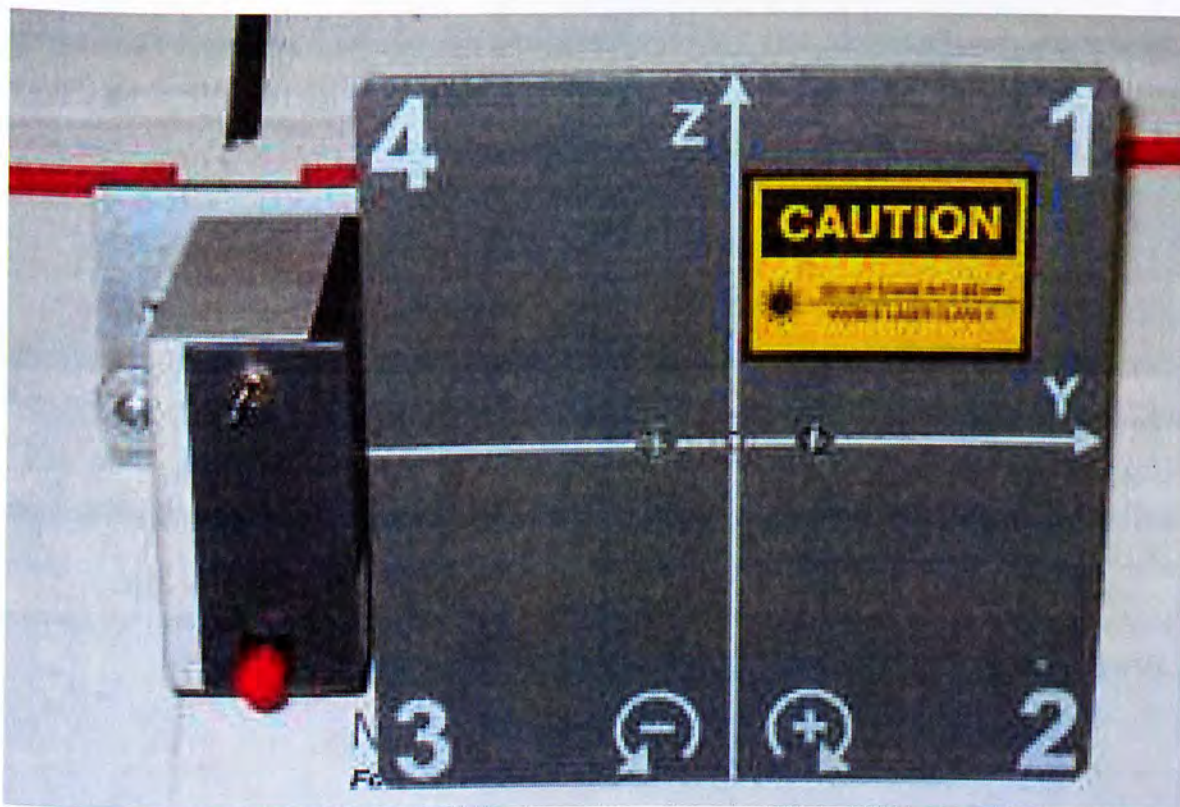
Το κόκκινο αυτοκίνητο ακολουθεί το μπλε με την ίδια ταχύτητα (βλέπε [12])

2.2 Πως λειτουργεί.

Η αυξομείωση της ταχύτητας επιτυγχάνεται με τη χρήση ενός αισθητήρα ραντάρ, ενός ψηφιακού επεξεργαστή σήματος και ενός ειδικού ελεγκτή. Το ραντάρ υπολογίζει την απόσταση και την ταχύτητα του προπορευόμενου οχήματος και στη συνέχεια στέλνει ένα σήμα στον κινητήρα ή στο σύστημα φρένων για επιτάχυνση ή επιβράδυνση του οχήματος ενώ η απόσταση από το προπορευόμενο όχημα διατηρείται σταθερή. Όταν ο δρόμος είναι και πάλι καθαρός, το σύστημα θα δώσει πάλι εντολή για επιτάχυνση και επιστροφή στην επιθυμητή ταχύτητα.

2.2.1 Τύποι.

Αρχικά, η απόσταση από το προπορευόμενο όχημα μετριόταν με τη βοήθεια λέιζερ. Ωστόσο το λέιζερ έχανε την αποτελεσματικότητά του σε άσχημες καιρικές συνθήκες όπως χιόνι ή δυνατή βροχή καθώς και στην περίπτωση που το προπορευόμενο όχημα είναι βαριά σκονισμένο και δεν υπάρχει αρκετή αντανάκλαση. Επίσης τοποθετούνταν μέσα σε ένα αρκετά μεγάλο μεταλλικό κουτί το οποίο δεν το καθιστούσε πρακτικό.



Σύστημα με λέιζερ της Volkswagen

Έτσι επικράτησε η λύση της εύρεσης απόστασης μέσω ραντάρ. Το ραντάρ το οποίο είναι διακριτικά τοποθετημένο στο μπροστινό μέρος του αυτοκινήτου, συνήθως σε κάποιο κομμάτι του προφυλακτήρα εκπέμπει ένα σήμα συχνότητας συνήθως γύρω στα 75GHz, έχει εμβέλεια από 150 έως 200 μέτρα και λειτουργεί σε ταχύτητες από 0 έως 180 χμ/ώρα.



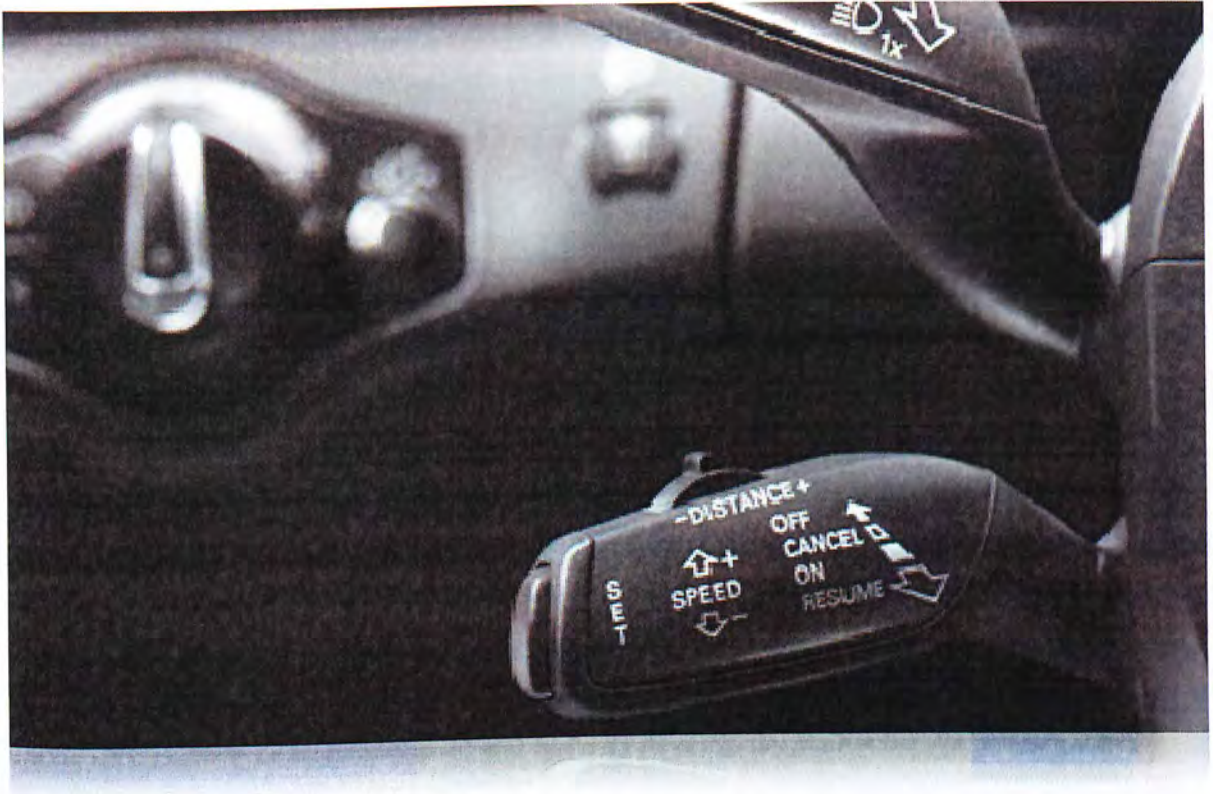
Το ραντάρ (δεξιά) τοποθετημένο στον προφυλακτήρα (βλέπε [4])

Η πλειοψηφία των εφαρμογών χρησιμοποιεί ένα ραντάρ ενώ υπάρχουν και εταιρείες που έχουν δοκιμάσει και πολλαπλά ραντάρ για καλύτερη σάρωση του οδοστρώματος, τοποθετώντας από ένα ραντάρ σε κάθε άκρη του προφυλακτήρα.

Η τεχνική αυτή δε χρησιμοποιεί δορυφόρο ή κάποιο άλλο βοήθημα (πχ κάποια κατασκευή του δρόμου) αλλά μόνο τα αισθητήρια και τον μικροϋπολογιστή που βρίσκεται πάνω στο αυτοκίνητο.

2.2.2 Χειρισμός.

Ο χειρισμός του συστήματος είναι παρόμοιος με τον απλό έλεγχο πορείας αλλά λόγω της πολυπλοκότερης λειτουργίας υπάρχουν και περισσότεροι διακόπτες για τον οδηγό. Παρακάτω βλέπουμε ένα χαρακτηριστικό παράδειγμα:



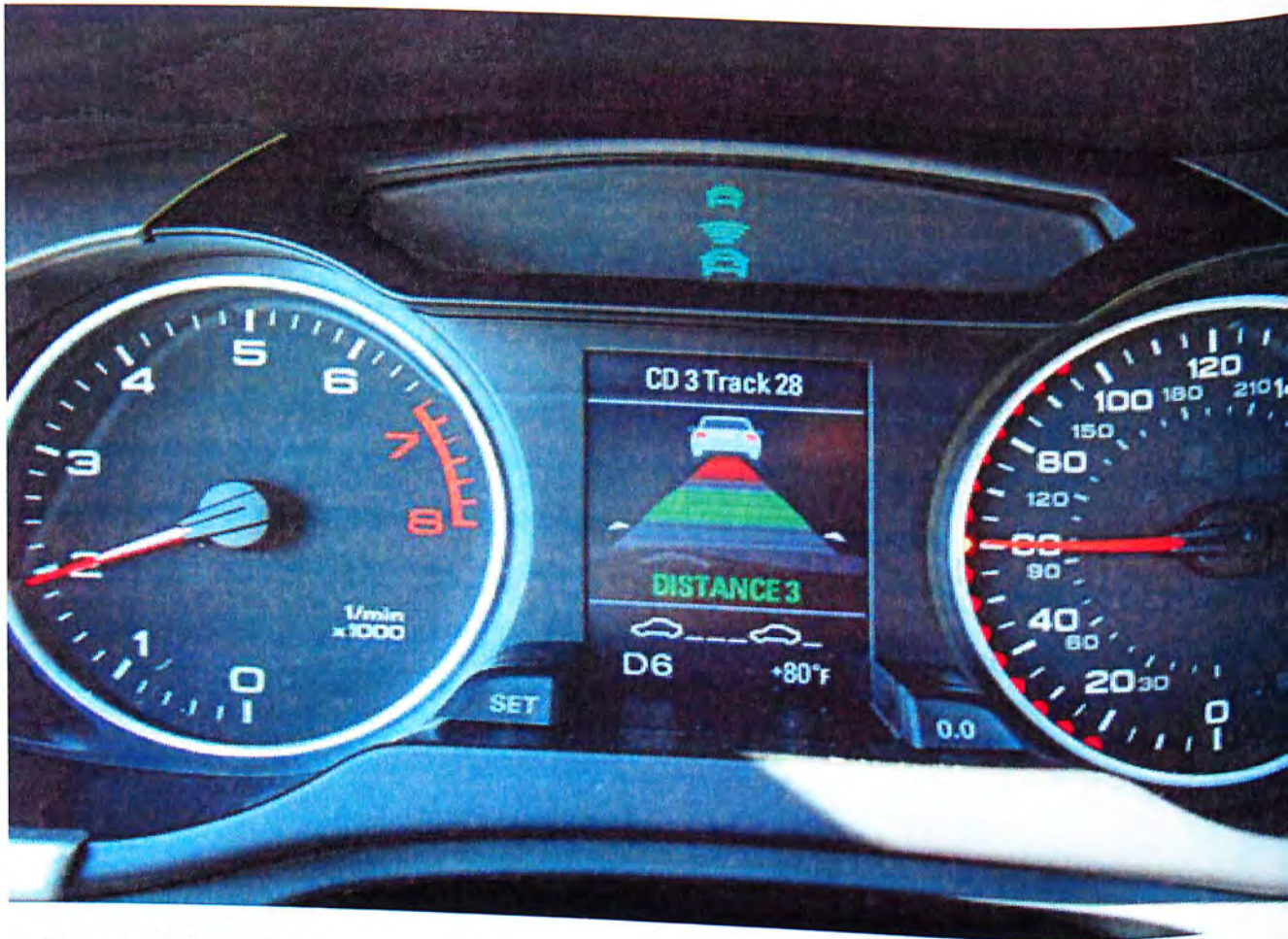
Τα πλήκτρα "on/off", "set", "resume", "cancel" είναι γνωστά από το συμβατικό σύστημα.

Η προσθήκη των πλήκτρων "Distance+/-", "Speed+/-" είναι αυτή που ξεχωρίζει το απλό σύστημα ελέγχου πορείας από το προσαρμοζόμενο.

- Ο διακόπτης "Speed+/-" αυξάνει ή μειώνει την επιθυμητή ταχύτητα.
- Ο διακόπτης "Distance+/-" αυξάνει ή μειώνει την απόσταση από το προπορευόμενο όχημα.

2.2.3 Απεικόνιση δεδομένων.

Όταν ο οδηγός ενεργοποιήσει την απόσταση και την ταχύτητα που επιθυμεί, το σύστημα αναλαμβάνει τα υπόλοιπα. Μέσω ενδείξεων σε μια οθόνη η οποία βρίσκεται είτε στο καντράν του αυτοκινήτου είτε στην κεντρική πλατφόρμα, ο οδηγός ενημερώνεται για την κατάσταση της πορείας.



Audi Adaptive Cruise Control

Στην παραπάνω εικόνα βλέπουμε το πλαίσιο πληροφοριών που εφαρμόζει η εταιρεία Audi στα αυτοκίνητα της. Στο πάνω μέρος το αυτοκίνητο ενημερώνει τον οδηγό ότι έχει εντοπίσει ένα κινούμενο όχημα μπροστά και το ακολουθεί. Στο κεντρικό κομμάτι φαίνεται η απόσταση την οποία κρατάει το όχημα από το προπορευόμενο. Αυτή τη στιγμή είναι στο "Distance3" το οποίο είναι μια από τις προεπιλεγμένες αποστάσεις που διαθέτει το συγκεκριμένο μοντέλο. Τέλος, στο κοντέρ βλέπουμε το κόκκινο λαμπάκι να είναι αναμμένο στα 100 χμ/ώρα που είναι και η ταχύτητα που έχει θέσει ο οδηγός και την οποία έχει αναπτύξει το αυτοκίνητο.

2.3 Βελτιώσεις.

Παρόλη την χρησιμότητα και την αποτελεσματικότητα των συστημάτων προσαρμοζόμενου ελέγχου πορείας που εφαρμόζεται από την πλειοψηφία των εταιριών, δε μπορούμε να παραβλέψουμε κάποιες ελλείψεις του συστήματος.

Για παράδειγμα, σε περίπτωση ακινητοποιημένου οχήματος (λόγω ατυχήματος, μπουλιαρίσματος κλπ) το σύστημα δε θα αντιδράσει προκαλώντας ατύχημα σε περίπτωση που ο οδηγός δεν προνοήσει.

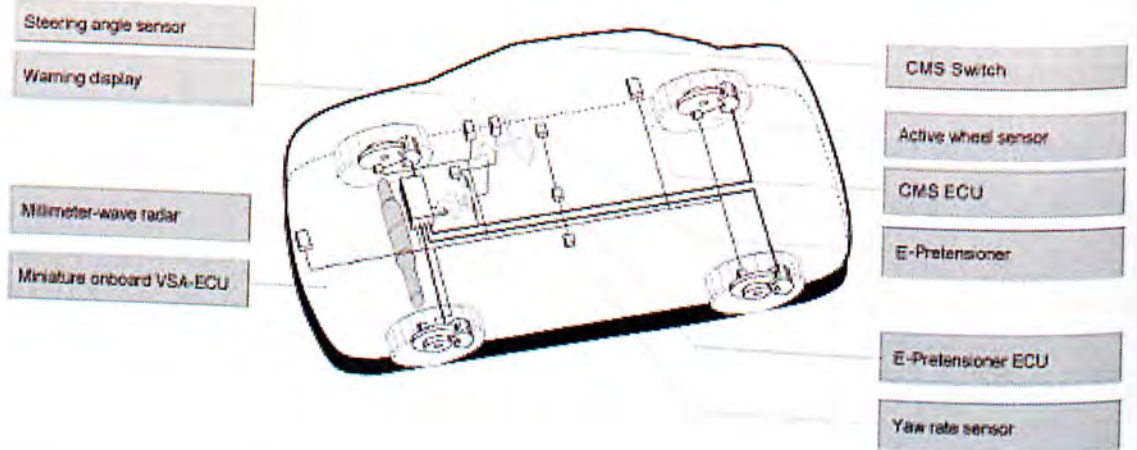
Για αυτό το λόγο αρκετές εταιρίες έχουν αναπτύξει συστήματα τα οποία λειτουργούν ταυτόχρονα με το σύστημα προσαρμοζόμενου ελέγχου πορείας και βοηθάνε είτε στην πρόληψη είτε στην αποτροπή σύγκρουσης με το προπορευόμενο όχημα.

2.3.1 Το CMBS.

Από τις πρώτες εταιρίες που ανέπτυξαν συστήματα αποφυγής σύγκρουσης ήταν η Honda το 2003 στο μοντέλο Inspire. Το σύστημα χρησιμοποιούσε το ραντάρ για να ελέγξει την κατάσταση του δρόμου και εφάρμοζε τα φρένα σε περίπτωση που ο οδηγός του οχήματος αγνοούσε τις φωτεινές και ηχητικές ενδείξεις. Επίσης με τη χρήση ενός μηχανισμού (E-Pretensioner) συνδεδεμένου στη ζώνη του οδηγού, έδινε σήματα στον οδηγό τραβώντας τη ζώνη σαν επιπλέον ένδειξη. Το σύστημα CMBS (Collision Mitigation Brake System) & E-Pretensioner όπως ονομάστηκε αποτελούνταν από:

- Ένα ραντάρ με εμβέλεια 100 μέτρων και οπτική γωνία 16 μοιρών.
- Αισθητήρια σε διάφορα σημεία του οχήματος (τιμόνι, τροχούς, φρένα)
- Μικροϋπολογιστή ελέγχου ο οποίος δέχεται τα σήματα από τα αισθητήρια και μέσω κατάλληλου προγραμματισμού υπολογίζει την απόσταση και τη σχετική ταχύτητα των οχημάτων καθώς και την πιθανότητα σύγκρουσης. Εάν διαπιστωθεί πιθανότητα σύγκρουσης ο M/Y προειδοποιεί τον οδηγό μέσω ενδείξεων στην οθόνη του οχήματος.
- Μικροϋπολογιστή ο οποίος ελέγχει μία υδραυλική μονάδα η οποία είναι συνδεδεμένη με το σύστημα φρένων του οχήματος και τα ενεργοποιεί.
- Μικροϋπολογιστή συνδεδεμένο σε ένα μηχανισμό για επαναφορά της ζώνης του οδηγού (E-Pretensioner).
- Το σύστημα απεικόνισης δεδομένων στο καντράν του οδηγού το οποίο δέχεται σήματα από τον M/Y ελέγχου.

CMS & E-Pretensioner System Overview



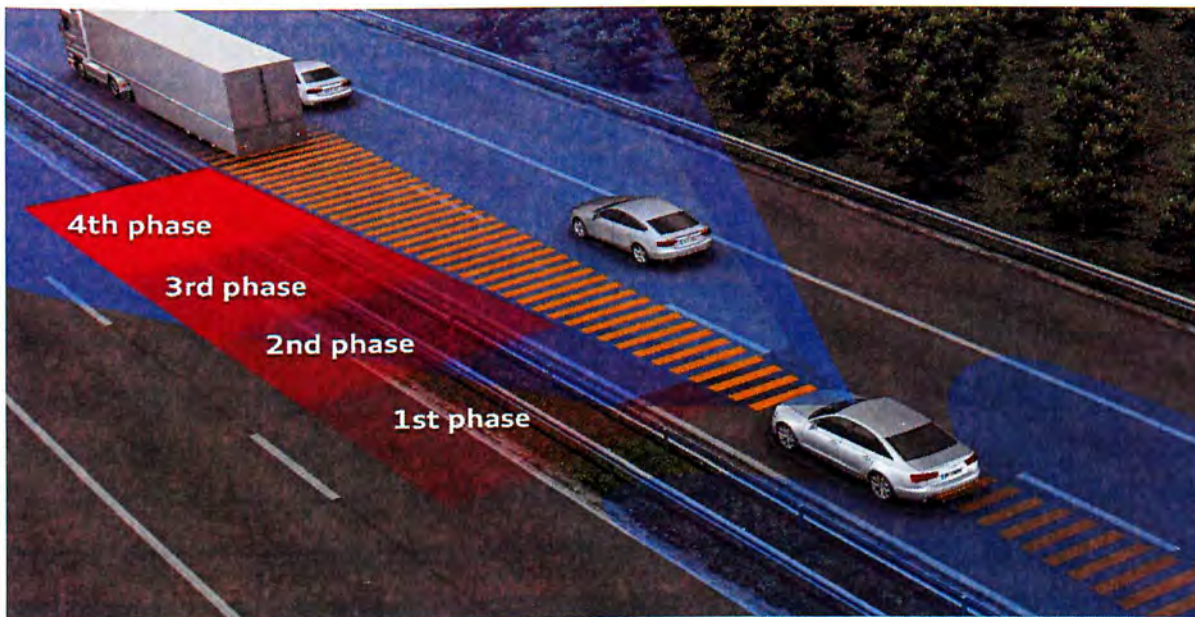
Το σύστημα CMBS της Honda. (Βλέπε [24])

Το σύστημα λειτουργεί σε τρία στάδια:

1. Πρώτη προειδοποίηση: Όταν ανιχνευτεί πιθανότητα σύγκρουσης με προπορευόμενο όχημα ή εάν έχει μειωθεί σημαντικά η απόσταση ανάμεσα στα οχήματα, ενεργοποιείται ένα ηχητικό σήμα ενώ εμφανίζεται και η ένδειξη "BRAKE" (ΦΡΕΝΑ) στην οθόνη του οδηγού.
2. Δεύτερη προειδοποίηση: Εάν η απόσταση των οχημάτων συνεχίσει να μειώνεται ο Μ/Υ ελέγχου δίνει σήμα για ελαφρύ φρενάρισμα και ο μηχανισμός E-Pretensioner τραβάει ελαφρά τη ζώνη του οδηγού 2-3 φορές σαν επιπλέον ένδειξη. Αν ο οδηγός πιέσει το φρένο σε αυτή τη φάση, θεωρείται φρενάρισμα ανάγκης και το σύστημα θα ενεργοποιήσει το βοηθούμενο σύστημα φρεναρίσματος.
3. Ελαχιστοποίηση ζημιάς από σύγκρουση: Αν ανιχνευτεί ότι η σύγκρουση είναι αναπόφευκτη, το σύστημα θα ενεργοποιήσει σε μέγιστο βαθμό τα φρένα ενώ το E-Pretensioner θα σφίξει τις ζώνες αρκετά ώστε να αναπληρώσει πιθανά κενά λόγω ρουχισμού ώστε να επιτυγχάνεται μέγιστη ασφάλεια.

2.3.2 AUDI Pre-Sense Plus.

Ακολούθησαν αρκετές εταιρίες οι οποίες συνεχώς βελτίωναν την εκάστοτε έκδοση του συστήματος. Ας δούμε την νεότερη έκδοσή του η οποία είναι το σύστημα της AUDI με το όνομα Pre-Sense Plus.



AUDI Pre-Sense Plus (βλέπε [25])

Το σύστημα χρησιμοποιεί δύο ραντάρ τοποθετημένα στο μπροστινό μέρος του οχήματος, συνήθως στον προφυλακτήρα, τα οποία σε συνεργασία με μια κάμερα τοποθετημένη στο παρμπρίζ του αυτοκινήτου στέλνουν δεδομένα στον μικροϋπολογιστή ελέγχου. Εάν ανιχνευτεί πιθανότητα σύγκρουσης το σύστημα μπαίνει σε λειτουργία σε 4 στάδια:

1. Το σύστημα ειδοποιεί ηχητικά και οπτικά τον οδηγό για την ανάγκη φρεναρίσματος και προετοιμάζει τα φρένα για πιθανή μανούβρα του οχήματος.
2. Εάν ο οδηγός δεν ανταποκριθεί και η σοβαρότητα της κατάστασης συνεχίζει να αυξάνει, ένα «νευρικό» φρενάρισμα προειδοποιεί περεταίρω για την ανάγκη μείωσης της ταχύτητας ενώ οι ζώνες «κλειδώνουν» στη σωστή θέση. Ταυτόχρονα τα φρένα ετοιμάζονται για το ενδεχόμενο εκτάκτου φρεναρίσματος. Εάν ο οδηγός φρενάρει σε αυτό το σημείο, το σύστημα θα εφαρμόσει ακριβώς όση δύναμη χρειάζεται για την αποφυγή της σύγκρουσης.
3. Εάν ακόμα δεν υπάρχει αντίδραση από τον οδηγό, το σύστημα θα φρενάρει ελαφρά.
4. Τέλος, εάν η σύγκρουση είναι αναπόφευκτη το σύστημα θα εφαρμόσει πλήρη ισχύ στα φρένα και αυτόματα θα ανάψει τα αλάρμ προειδοποιώντας διερχόμενους οδηγούς.

2.3.3 Volvo ACC with queue assist.

Η εταιρία Volvo προχωρώντας ένα βήμα παραπάνω πρόσθεσε ένα τελευταίο αλλά πολύ χρήσιμο χαρακτηριστικό στα συστήματα προσαρμοζόμενου ελέγχου πορείας που εφαρμόζει στα οχήματα της. Αυτό είναι η δυνατότητα ο υπολογιστής να αναλαμβάνει την οδήγηση του οχήματος σε κατάσταση μπουτιλιαρίσματος με συνεχείς ακινητοποιήσεις και εκκινήσεις του αυτοκινήτου.

Έτσι, χωρίς ο οδηγός να ακουμπάει τα πεντάλ, το όχημα επιταχύνει και επιβραδύνει έως και σε πλήρη στάση αυτόματα ακόμα και σε πολύ χαμηλές ταχύτητες.

Υπάρχουν βέβαια πολλά θετικά στοιχεία σε αυτό το χαρακτηριστικό καθώς η οδήγηση σε κατάσταση μπουτιλιαρίσματος μπορεί να είναι πολύ κουραστική και εκνευριστική για τον οδηγό.

Ωστόσο υπάρχουν και κάποια αρνητικά στοιχεία. Σε μια «τυφλή» στροφή το σύστημα χάνει το προπορευόμενο όχημα με αποτέλεσμα να επιταχύνει και να δημιουργεί κινδύνους. Επίσης όταν ένα άλλο αυτοκίνητο αλλάξει λωρίδα και εισέλθει στο κενό μεταξύ του οχήματος μας και του προπορευόμενου, το σύστημα δε θα το αναγνωρίσει παρά μόνο όταν αυτό εισέλθει κατά περίπου τα 2/3 στη λωρίδα με αποτέλεσμα να μη φρενάρι αρκετά έγκαιρα.

Το σύστημα queue assist αν και πολύ χρήσιμο στην τωρινή του κατάσταση επιδέχεται αρκετές βελτιώσεις τις οποίες ήδη δουλεύει η εταιρία.

Κεφάλαιο 3^ο: Υλοποίηση

3.1 Η κατασκευή.

Για την υλοποίηση της κατασκευής μου, προμηθεύτηκα ένα κοινό τηλεκατευθυνόμενο αυτοκινητάκι του εμπορίου. Αυτό αποτελείται από 2 DC κινητήρες, ένας τοποθετημένος στο πίσω μέρος για να κινεί το όχημα εμπρός και πίσω και ένας μπροστά ο οποίος λειτουργεί σαν τιμόνι στρίβοντας τις μπροστινές ρόδες.

Ένα ολοκληρωμένο κύκλωμα γέφυρας H-Bridge (LN 298N) οδηγεί τους κινητήρες επιτρέποντας την περιστροφή αριστερόστροφα και δεξιόστροφα.

Επίσης ήταν αναγκαίο να χρησιμοποιηθεί ένας μικροϋπολογιστής ο οποίος θα είναι το μυαλό της κατασκευής. Για αυτό το λόγο επέλεξα τον Arduino duemilano πάνω στον οποίο έχω συνδέσει όλα τα περιφερειακά αισθητήρια και μέσω κατάλληλου προγραμματισμού ελέγχω το όχημα.

Για τον έλεγχο της πορείας έχω τοποθετήσει στο μπροστινό μέρος της κατασκευής από ένα αισθητήριο φωτοαντίστασης σε κάθε μεριά του προφυλακτήρα, τα οποία με τη βοήθεια φωτισμού από λαμπάκια τύπου LED, ανιχνεύουν τις μαύρες ταινίες-λωρίδες μέσα στις οποίες πρέπει να κινείται το αυτοκίνητο μας.

Επιπλέον στο μπροστινό μέρος έχω τοποθετήσει έναν αισθητήρα υπερήχων ο οποίος βοηθάει στον υπολογισμό της απόστασης από το προπορευόμενο όχημα και την εύρεση της σχετικής του ταχύτητας.

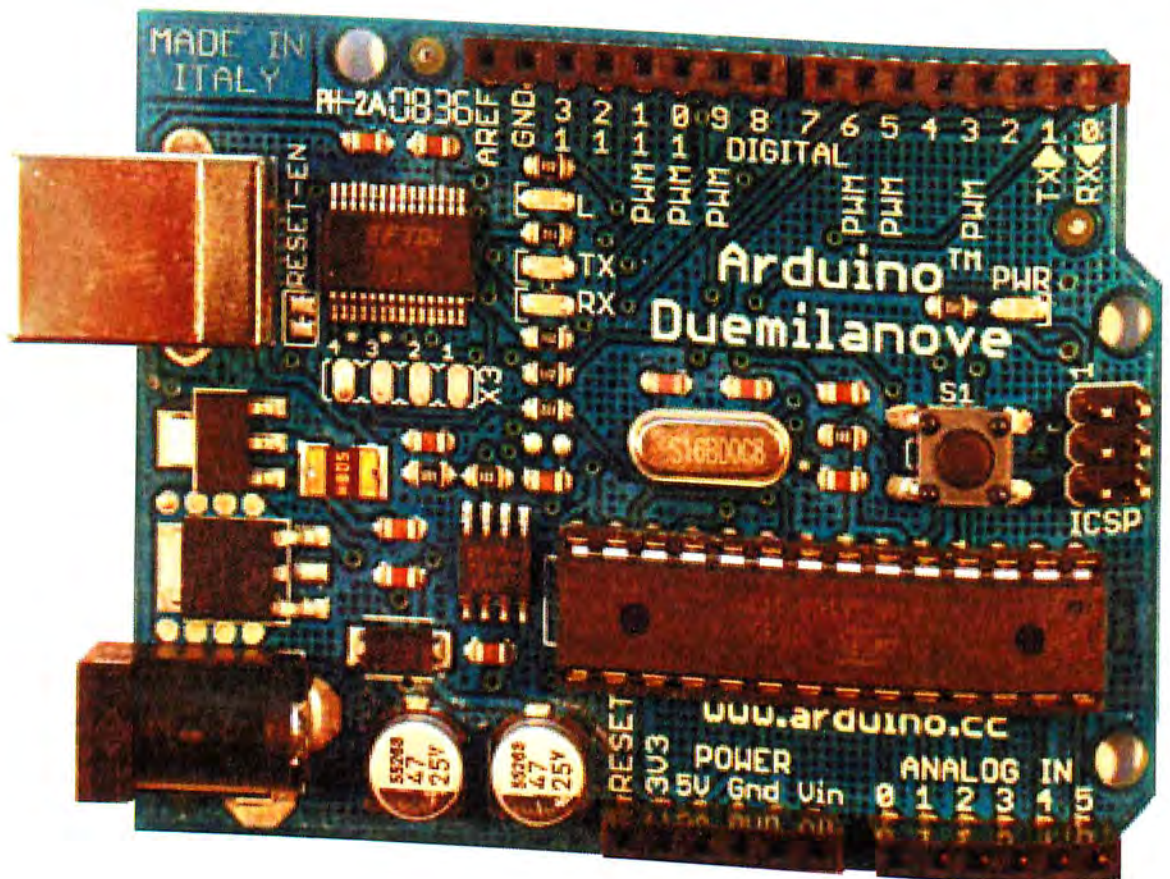
Τέλος για τον χειρισμό του οχήματος από απόσταση έχω χρησιμοποιήσει έναν δέκτη υπερύθρων ο οποίος δέχεται τα σήματα από το τηλεχειριστήριο.

Όλα αυτά συνδέονται μέσω ενός μπρεντμπορντ για ευκολότερη επεξεργασία, διόρθωση λαθών και επέκταση μελλοντικά.

Για την τροφοδοσία του οχήματος χρειάστηκαν 2 ειδών μπαταρίες. Ένα σετ από 8 AA συνολικής τάσης 12V για την τροφοδοσία των κινητήρων και μια ακόμα μπαταρία των 6V για την τροφοδοσία της πλακέτας του Arduino.

3.1.1 Ο μικροϋπολογιστής Arduino Duemilano.

Για τον έλεγχο τόσο των κινητήρων όσο και των σημάτων από τα αισθητήρια χρησιμοποιώ τον μικροϋπολογιστή Arduino Duemilano. Πρόκειται για μια υπολογιστική πλατφόρμα βασισμένη σε μια απλή μητρική πλακέτα με ενσωματωμένο μικροελεγκτή και εισόδους/εξόδους, και η οποία μπορεί να προγραμματιστεί με τη γλώσσα Wiring (ουσιαστικά πρόκειται για τη C++ με κάποιες μετατροπές).



Μία πλακέτα Arduino αποτελείται από ένα μικροελεγκτή Atmel AVR (ATmega328 και ATmega168 στις νεότερες εκδόσεις, ATmega8 στις παλαιότερες) και συμπληρωματικά εξαρτήματα για την διευκόλυνση του χρήστη στον προγραμματισμό και την ενσωμάτωση του σε άλλα κυκλώματα. Όλες οι πλακέτες περιλαμβάνουν ένα γραμμικό ρυθμιστή τάσης 5V και έναν κρυσταλλικό ταλαντωτή 16MHz (ή κεραμικό αντηχητή σε κάποιες παραλλαγές). Ο μικροελεγκτής είναι από κατασκευής προγραμματισμένος με ένα *bootloader*, έτσι ώστε να μην χρειάζεται εξωτερικός προγραμματιστής.

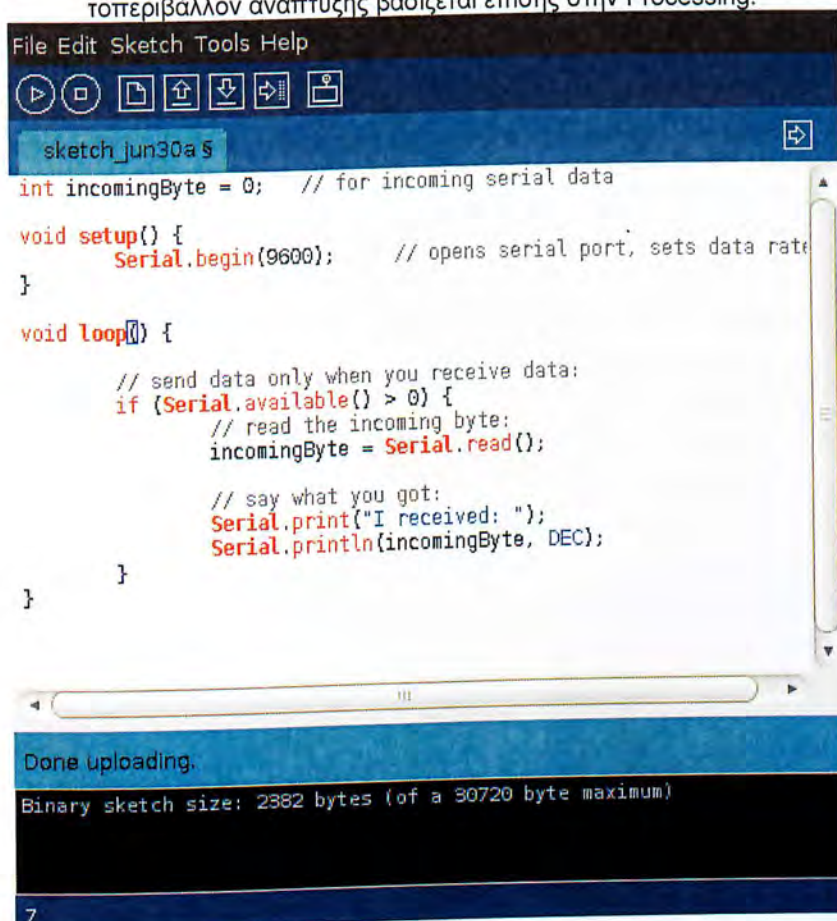
Γενικά όλες οι πλακέτες είναι προγραμματισμένες μέσω μιας σειριακής σύνδεσης RS-232, αλλά ο τρόπος με τον οποίο αυτό υλοποιείται ποικίλλει ανάλογα με την έκδοση. Οι σειριακές πλακέτες Arduino περιέχουν ένα απλό κύκλωμα αντιστροφής για την μετατροπή ανάμεσα στα σήματα των επιπέδων RS-232 και TTL. Οι πλακέτες Arduino που κυκλοφορούν σήμερα στην αγορά, συμπεριλαμβανόμενης και της Diecimila, προγραμματίζονται μέσω USB, εφαρμόζοντας ένα τσίπ προσαρμογέα USB-to-serial όπως το FTDI

FT232. Κάποιες παραλλαγές, όπως το Arduino mini και το ανεπίσημο Boarduino, χρησιμοποιούν προσαρμογέα USB-to-serial σε μορφή πλακέτας ή καλωδίου.

Η πλακέτα του Arduino έχει εκτεθειμένες τις περισσότερες επαφές εισόδου/εξόδου για χρήση με άλλα κυκλώματα. Το Diecimila, για παράδειγμα, παρέχει 14 ψηφιακές επαφές εισόδου/εξόδου, από τις οποίες οι 6 μπορούν να παράξουν σήματα PWM, και 6 αναλογικές εισόδους. Αυτές οι επαφές είναι διαθέσιμες στην κορυφή της πλακέτας μέσω θηλυκών συνδέσεων μεγέθους 0,1 ιντσών. Διάφορες plug-in πλακέτες εφαρμογών γνωστές σαν "shields" είναι, επίσης, διαθέσιμες στο εμπόριο..

Το IDE του Arduino είναι γραμμένο σε Java και μπορεί να τρέξει σε πολλαπλές πλατφόρμες. Περιλαμβάνει επεξεργαστή κώδικα (επεξεργαστή κειμένου με διάφορα εύχρηστα εργαλεία) και μεταγλωττιστής και έχει την ικανότητα να φορτώνει εύκολα το πρόγραμμα μέσω σειριακής θύρας από τον υπολογιστή στην πλακέτα.

Το περιβάλλον ανάπτυξης είναι βασισμένο στην Processing, ένα περιβάλλον ανάπτυξης σχεδιασμένο να εισαγάγει στον προγραμματισμό καλλιτέχνες μη εξοικειωμένους με την ανάπτυξη λογισμικού. Η συγκεκριμένη γλώσσα προγραμματισμού προέρχεται από την Wiring, μια γλώσσα που μοιάζει με την C η οποία παρέχει παρόμοια λειτουργικότητα για μια πιο περιορισμένης σχεδίασης πλακέτα, της οποίας το περιβάλλον ανάπτυξης βασίζεται επίσης στην Processing.



```
File Edit Sketch Tools Help
[Icons]
sketch_jun30a 5
int incomingByte = 0; // for incoming serial data
void setup() {
  Serial.begin(9600); // opens serial port, sets data rate
}
void loop() {
  // send data only when you receive data:
  if (Serial.available() > 0) {
    // read the incoming byte:
    incomingByte = Serial.read();

    // say what you got:
    Serial.print("I received: ");
    Serial.println(incomingByte, DEC);
  }
}
```

Done uploading.

Binary sketch size: 2382 bytes (of a 30720 byte maximum)

7

Το περιβάλλον του Arduino (βλέπε [19])

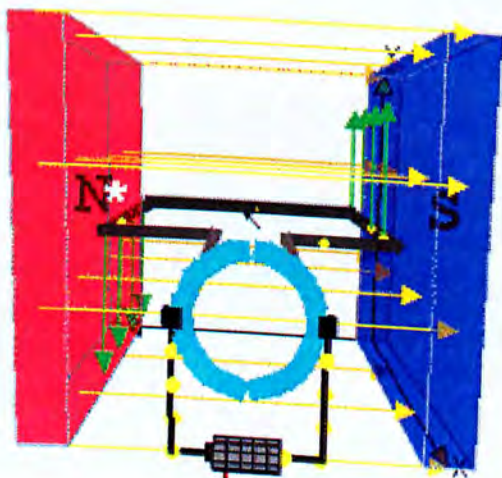
Με τη χρήση του Arduino σαν μικροελεγκτή εξομοιώνω την χρησιμότητα του πραγματικού μικροϋπολογιστή που βρίσκεται κάτω από το καπό των αυτοκινήτων και ο οποίος είναι «το μυαλό» του αυτοκινήτου. Αντίστοιχα, στην κατασκευή μου ενώ πολλά αισθητήρια ενεργούν αυτόνομα, όλα είναι συνδεδεμένα πάνω στον Arduino και μόνο αυτός έχει τη δυνατότητα να δώσει εντολή για κάποια ενέργεια.

3.1.2 DC κινητήρες.

Οι DC κινητήρες (κινητήρες συνεχούς ρεύματος, βλέπε [1]) χρησιμοποιούνται ευρέως σε συσκευές όπως ηλεκτροκίνητα οχήματα, μηχανές ξυρίσματος, υαλοκαθαριστήρες αυτοκινήτων και είναι η καλύτερη επιλογή για την κίνηση του οχήματος που έχω κατασκευάσει.

Ένας κινητήρας συνεχούς ρεύματος αποτελείται από τα παρακάτω στοιχεία:

1. Τον στάτη ο οποίος αποτελείται από:
 - a. Το ζύγωμα
 - b. Τους μαγνητικούς πόλους (N και S)
 - c. Την περιέληξη
 - d. Το κλεμοκυβώτιο
 - e. Το περίβλημα
2. Το ρότορα ο οποίος αποτελείται από:
 - a. Τον άξονα
 - b. Το επαγωγικό τύμπανο
 - c. Τον συλλέκτη
 - d. Την περιέληξη
 - e. Τον ανεμιστήρα

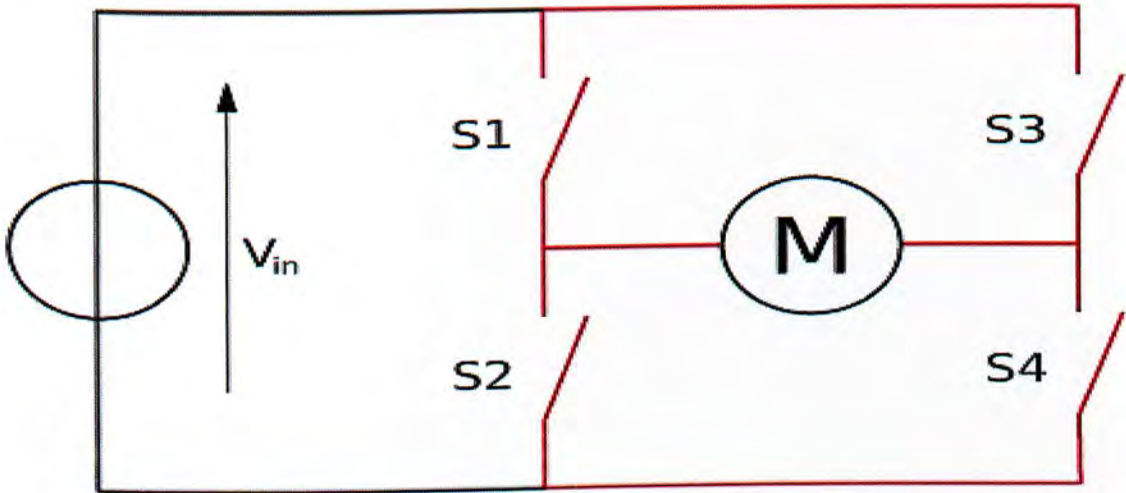


Εάν ένας ρευματοφόρος αγωγός (ή πηνίο) τοποθετηθεί εντός μαγνητικού πεδίου μαγνητικής επαγωγής B , έτσι ώστε να τέμνει τις μαγνητικές γραμμές, τότε επί του αγωγού αναπτύσσεται δύναμη (F) η οποία τείνει να τον κινήσει.

Η διπλανή διάταξη αποτελείται από μια πηγή συνεχούς τάσης, από μια αντίσταση και από δυο λείους κυλινδρικούς αγωγούς. Ο ένας αγωγός είναι συνδεδεμένος με τον θετικό και ο άλλος με τον αρνητικό πόλο της πηγής. Όταν αρχίσει να ρέει ρεύμα στο κύκλωμα, δημιουργούνται οι δυνάμεις που τείνουν να κινήσουν τον αγωγό.

3.1.3 Το ολοκληρωμένο κύκλωμα L293NE.

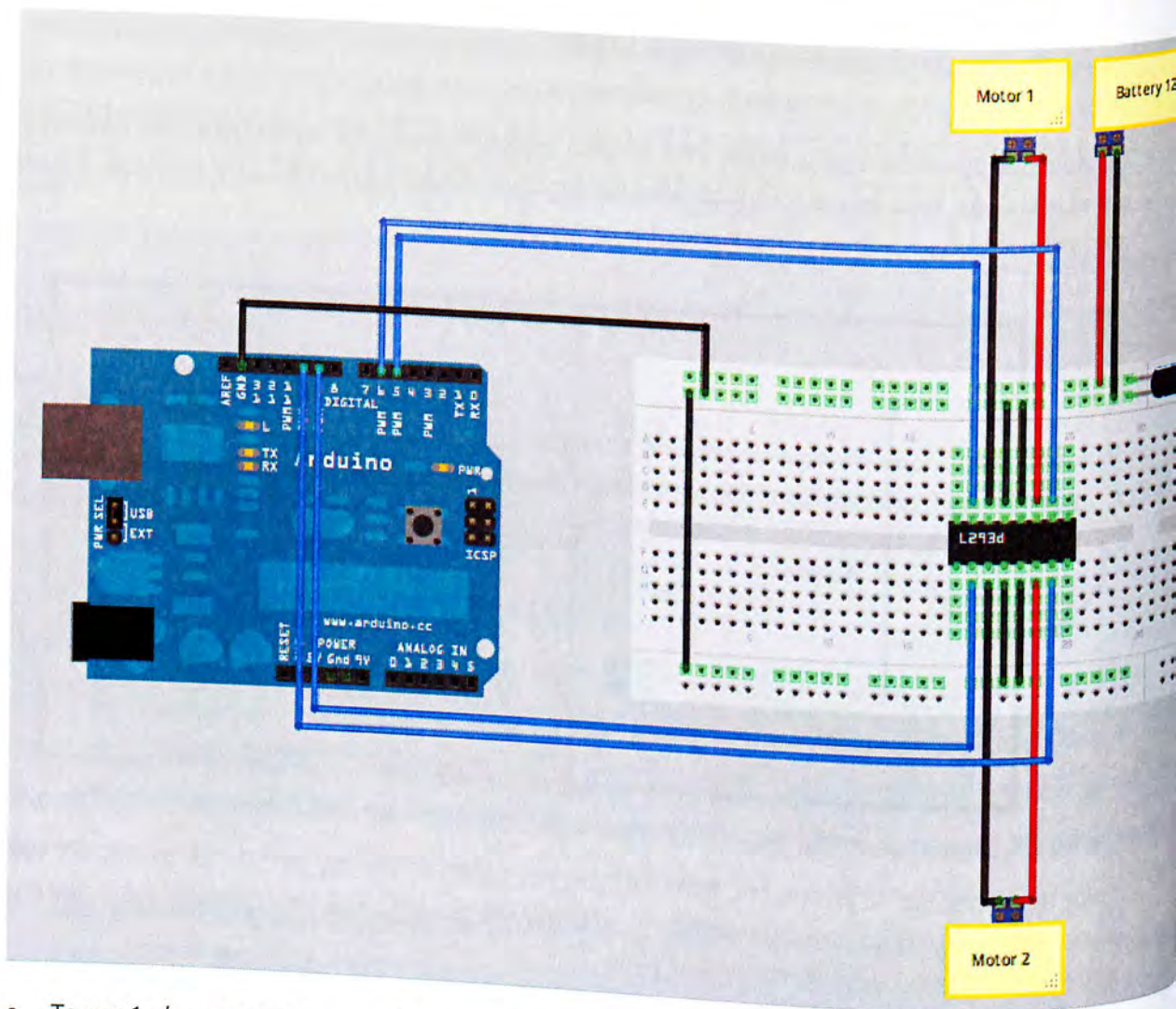
Το ολοκληρωμένο κύκλωμα γέφυρας «H» L293NE χρησιμοποιείται για την οδήγηση των κινητήρων και την περιστροφή τους αριστερόστροφα και δεξιόστροφα. Παρακάτω βλέπουμε ένα τυπικό διάγραμμα μιας γέφυρας H:



Είναι προφανές ότι το όνομα της προήλθε από τον τρόπο σύνδεσης των στοιχείων της. Οι γέφυρες αυτές μπορούν είτε να κατασκευαστούν με ηλεκτρονικά στοιχεία είτε να αγοραστούν έτοιμες σαν ένα ολοκληρωμένο κύκλωμα. Λειτουργεί ως εξής:

Όταν οι διακόπτες S1 και S4 κλείσουν και οι S3 και S2 είναι ανοιχτοί, ο κινητήρας αποκτά θετική πόλωση ενώ αντίστροφα εάν S1 και S4 ανοίξουν και S3 και S2 κλείσουν ο κινητήρας θα έχει αρνητική πόλωση και θα αλλάξει φορά περιστροφής.

Παρακάτω περιγράφεται αναλυτικά ο τρόπος σύνδεσης που έχω χρησιμοποιήσει για την κατασκευή μου καθώς δίνεται με πιο ακριβή τρόπο μια περιγραφή του τρόπου λειτουργίας του ολοκληρωμένου κυκλώματος.



- Το πιν 1 είναι το πιν ενεργοποίησης του τσιπ και συνδέεται με μια έξοδο του Arduino η οποία δίνει λογικό «1» για όση ώρα λειτουργεί το αυτοκινητάκι.
- Το πιν8 είναι η τροφοδοσία των κινητήρων και είναι συνδεδεμένο με την πηγή, στη συγκεκριμένη περίπτωση με τις μπαταρίες με συνολική τάση 12V.
- Το πιν16 είναι η τροφοδοσία του ολοκληρωμένου και συνδέεται με τα 5V που μπορεί να δώσει ο Arduino.
- Τα πιν 4,5,12,13 πάνε στη γείωση.
- Τα πιν 2,7,10,15 δέχονται σήμα από τις εξόδους PWM του Arduino.
- Τα πιν 3,6,11,14 δίνουν σήμα για τροφοδοσία των κινητήρων ανάλογα με το πόση τάση έχουν δεχτεί τα 2,7,10,15.

3.1.4 Οι φωτοαντιστάσεις (LDR).

Μια φωτοαντίσταση είναι μια αντίσταση της οποίας η τιμή μειώνεται με την αύξηση του φωτός που προσπίπτει στην επιφάνεια της. Μια φωτοαντίσταση φτιάχνεται από υψηλής αντίστασης ημιαγωγούς. Αν πέσει φως στην συσκευή, με αρκετά υψηλή συχνότητα, τότε φωτόνια απορροφούνται από τον ημιαγωγό και δεσμευμένα ηλεκτρόνια αποκτούν αρκετή ενέργεια, ώστε να αποσπαστούν από τα άτομα που τα δεσμεύουν. Τα ελεύθερα ηλεκτρόνια που δημιουργούνται άγουν ρεύμα και αυτό έχει σαν αποτέλεσμα την μείωση της αντίστασης.



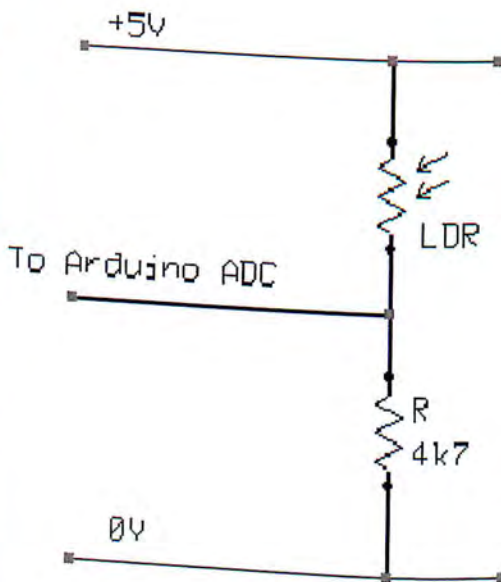
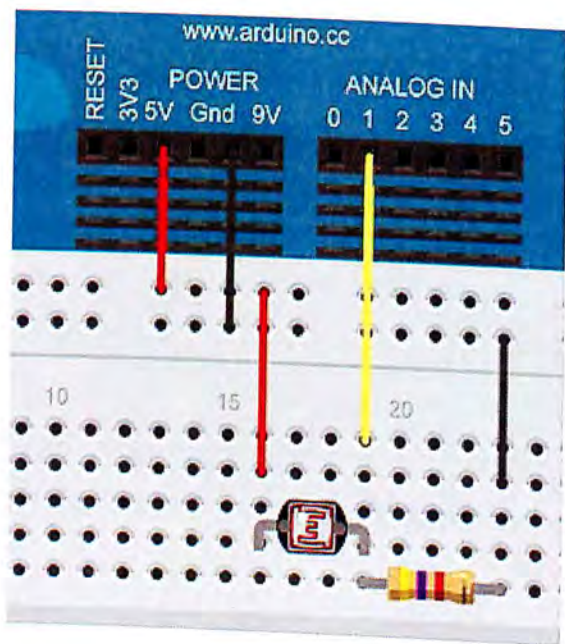
Τυπική φωτοαντίσταση

Υπάρχουν δυο τύποι φωτοαντιστάσεων ανάλογα με τους ημιαγωγούς που χρησιμοποιούνται. Ο πρώτος τύπος είναι οι ενδογενείς και ο δεύτερος είναι φωτοαντιστάσεις με ημιαγωγούς που έχουν προσμίξεις. Στον πρώτο τύπο φωτοαντίστασης τα ηλεκτρόνια προκειμένου να αποδεσμευτούν χρειάζονται αρκετή ενέργεια ενώ στον δεύτερο λιγότερη. Έτσι η δεύτερη κατηγορία φωτοαντιστάσεων μπορεί να επηρεάζεται και από φως χαμηλότερων συχνοτήτων.



Τοποθετημένα κατάλληλα και με ένα λαμπάκι LED να δίνει αρκετό φως, θα σαρώνουν την επιφάνεια και μόλις εμφανιστεί η μαύρη λωρίδα η οποία απορροφά το μεγαλύτερο μέρος του εκπεμπόμενου φωτός, ο μικροελεγκτής θα αντιλαμβάνεται αυτή τη διαφοροποίηση και θα δίνει εντολή για στρόφι των τροχών.

Το κύκλωμα το οποίο έχω χρησιμοποιήσει είναι το παρακάτω:



Αριστερά φαίνεται ο τρόπος σύνδεσης των καλωδίων πάνω στο μπρέντμπορντ ενώ δεξιά το διάγραμμα του κυκλώματος. Η σύνδεση της φωτοαντίστασης με την πλακέτα Arduino γίνεται μέσω μια αναλογικής εισόδου της πλακέτας καθώς η τιμή που επιστρέφει η φωτοαντίσταση είναι αυτής της μορφής.

Το αισθητήριο της φωτοαντίστασης προσαρμόζεται και λειτουργεί ικανοποιητικά σε κάθε επιφάνεια εάν φυσικά υπάρχει αρκετό φως. Στη συγκεκριμένη εφαρμογή αποτελεί την βέλτιστη λύση καθώς εκτός από αποτελεσματικό είναι και πολύ οικονομικό.

3.1.5 Αισθητήρας Υπερήχων.

Οι αισθητήρες υπερήχων παρέχουν μια οικονομικά αποδοτική μέθοδο ανίχνευσης με ιδιότητες που δεν υπάρχουν σε άλλες τεχνολογίες. Με τη χρήση μιας ευρείας ποικιλίας μετατροπέων υπερήχων και διάφορα φάσματα συχνοτήτων, ένας αισθητήρας υπερήχων μπορεί να σχεδιαστεί για να λύσει πολλά προβλήματα εφαρμογών που είναι απαγορευτικά στο κόστος ή απλά δεν μπορούν να λυθούν από άλλους αισθητήρες. Οι αισθητήρες υπερήχων εκπέμπουν συνεχώς ηχητικούς παλμούς υψηλής συχνότητας προς την επιφάνεια του στόχου και ανακλώνται πίσω στον αισθητήρα. Τα ηλεκτρονικά του αισθητήρα μετρούν το

χρόνο λήψης του σήματος και τον μετατρέπουν σε μονάδα μήκους.

Δεδομένου ότι η ταχύτητα του ήχου επηρεάζεται από τη θερμοκρασία αέρα, οι αισθητήρες μας υπερήχων περιλαμβάνουν έναν ενσωματωμένο αισθητήρα θερμοκρασίας. Οι μετρήσεις στάθμης/απόστασης αντισταθμίζονται αυτόματα σε όλη την κλίμακα λειτουργίας του αισθητήρα.

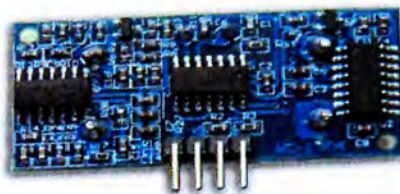
Κλασικές εφαρμογές περιλαμβάνουν την ανίχνευση προσέγγισης, την παρουσία ή την απουσία αντικειμένου, την ανίχνευση εμποδίων σε αυτοματοποιημένα οχήματα, την μέτρηση απόστασης, μέτρηση στάθμης, κ.λπ... κ.λπ...

Βασικά τεχνικά χαρακτηριστικά (βλέπε [26])

- Άμεση απεικόνιση μετρούμενης τιμής σε mm/cm ή % σε ψηφιακή (LED) ένδειξη
- Αντιστάθμιση θερμοκρασίας για μέτρηση ακριβείας σε εφαρμογές με διακύμανση θερμοκρασίας
- Δυνατότητα ρύθμισης του αισθητήρα μέσω της ψηφιακής οθόνης
- ακρίβεια $\pm 1\%$ (με εσωτερική αντιστάθμιση θερμοκρασίας)
- 1 ή 2 rpr ή rpr σήματα για όλους τους αισθητήρες
- Αναλογικά σήματα εξόδου 4-20 mA και 0-10 V (2 αναλογικά σήματα σε ένα αισθητήρα)
- Τάση λειτουργίας 9-30 V για ποικίλες εφαρμογές, με προστασία πολικότητας
- Λειτουργία teach-in για αναγνώριση των συμβατικών σημείων ανίχνευσης
- Αυτόματος συγχρονισμός για ταυτόχρονη λειτουργία μέχρι και 10 μονάδων σε περιορισμένους χώρους
- Αυτόματη μεταστροφή μεταξύ εξόδων ρεύματος & τάσης για απλή λειτουργία
- Υλικό περιβλήματος επινικελωμένος ορείχαλκος, , πλαστικά μέρη, PBT, TPU (προαιρετικά από ανοξείδωτο χάλυβα)
- Βαθμός προστασίας IP67
- Αντοχή σε δονήσεις & κραδασμούς σύμφωνα με το IEC 60068-2-27

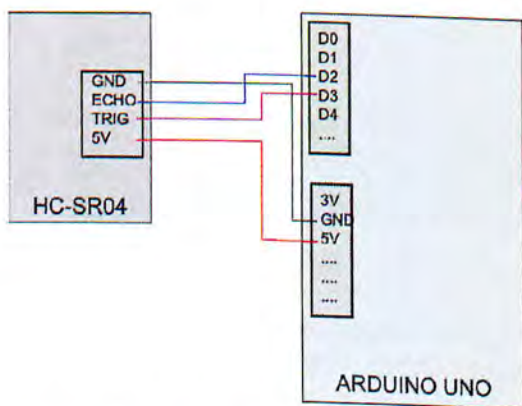
Οι διαφορετικοί τρόποι λειτουργίας και συνθέσεις των αισθητήρων & συστημάτων υπερήχων επιτρέπουν τις χρήσεις στις περισσότερες εφαρμογές αυτοματισμού

Συγκεκριμένα στη δική μας εφαρμογή έχω χρησιμοποιήσει τον HC-SR04:



Όπως βλέπουμε η συγκεκριμένη φόρμα αποτελείται στο μπροστινό μέρος από τον πομπό και τον δέκτη υπερήχων. Ο πομπός στέλνει συνεχώς κύματα προς την κατεύθυνση που βλέπει και όταν βρεθεί κάποιο εμπόδιο τα κύματα επιστρέφουν, τα λαμβάνει ο δέκτης και γίνεται η κατάλληλη επεξεργασία ώστε να προσδιοριστεί η μετρούμενη απόσταση.

Οι 4 ακροδέκτες της μονάδας (VCC, Trig, Echo, GND) αποτελούν το δίαυλο επικοινωνίας μεταξύ του αισθητηρίου και του μικροελεγκτή. Η συνδεσμολογία είναι η παρακάτω:



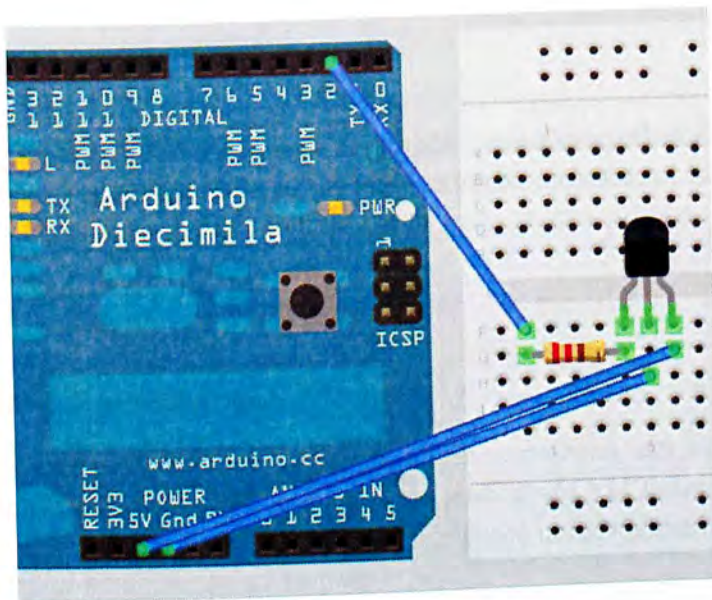
Όταν το Trig λάβει λογικό «1», ξεκινάει την αποστολή κυμάτων. Εάν βρεθεί εμπόδιο, ο ακροδέκτης Echo θα λάβει λογικό «1» και με βάση την διαφορά απόστασης θα μείνει σε λογικό «1» για συγκεκριμένο χρονικό διάστημα. Έπειτα μέσω λογισμικού ο μικροελεγκτής θα υπολογίσει την απόσταση.

Με αυτόν τον απλό τρόπο αναπαριστώ την λειτουργία του ραντάρ στο μπροστινό μέρος του αυτοκινητάκι. Καθώς το αυτοκινητάκι κινείται, το αισθητήριο υπερήχων συνεχώς μετράει την απόσταση από το προπορευόμενο όχημα, εάν αυτό φυσικά υπάρχει, και επιστρέφει τα δεδομένα στον μικροελεγκτή. Εκεί με τους κατάλληλους αλγορίθμους αποφασίζεται η επόμενη κίνηση (επιτάχυνση-επιβράδυνση-φρενάρισμα-σταθερή ταχύτητα).

3.1.6 Δέκτης υπερύθρων.

Για την ενεργοποίηση του οχήματος και την εκκίνηση λειτουργίας του, έχω τοποθετήσει έναν δέκτη υπερύθρων συνδεδεμένο με μια είσοδο του Arduino. Ο δέκτης αυτός λαμβάνει τα σήματα του τηλεκοντρόλ της κατασκευής. Με αυτό τον τρόπο αναπαριστώ τους διακόπτες που έχει στη διάθεση του ο οδηγός του οχήματος.

Παρακάτω φαίνεται ο τρόπος σύνδεσης του δέκτη υπερύθρων με τον Arduino καθώς και η μορφή του:



Βλέπουμε ότι ο δέκτης υπέρυθρων έχει 3 ακροδέκτες.

- Ο ακροδέκτης 1 αποτελεί την έξοδο του δέκτη και συνδέεται μέσω μια αντίστασης με το ψηφιακό πιν του Arduino ο οποίος με κατάλληλο προγραμματισμό αποκωδικοποιεί το σήμα που δέχεται.
- Ο ακροδέκτης 2 συνδέεται με τη γείωση.
- Ο ακροδέκτης 3 είναι η τροφοδοσία. 5V είναι αρκετά.

Για τον χειρισμό των εντολών του οδηγού έχω προσαρμόσει τα πλήκτρα ενός κοινού τηλεκοντρόλ στις απαιτήσεις της εφαρμογής.



Πιέζοντας το πλήκτρο play ενεργοποιούμε το σύστημα προσαρμοζόμενου ελέγχου πορείας.

Πιέζοντας το πλήκτρο off απενεργοποιούμε το σύστημα.

Τα πλήκτρα vol+ vol- αυξάνουν ή ελαττώνουν την επιθυμητή ταχύτητα του οχήματος.

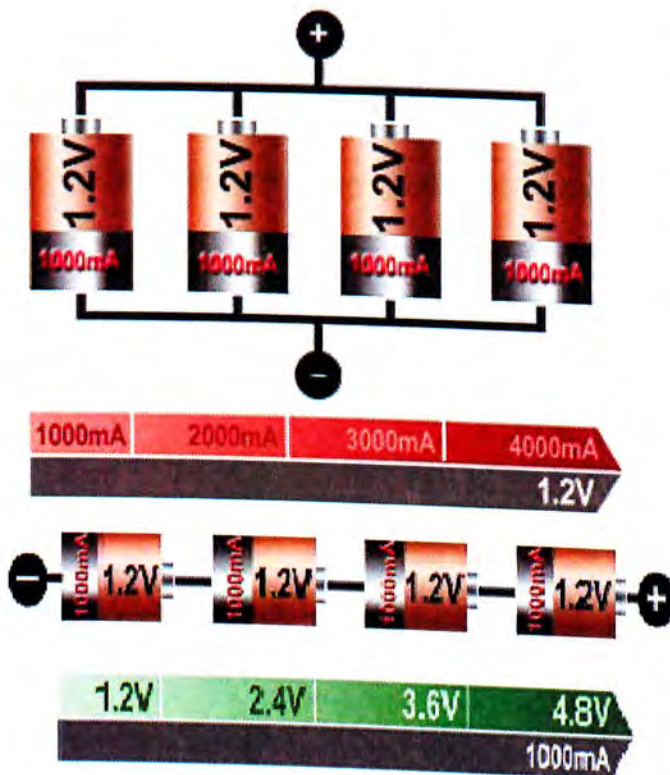
Τα πλήκτρα με τα βέλη αυξάνουν ή ελαττώνουν την επιθυμητή απόσταση από το προπορευόμενο όχημα.

3.1.7 Το breadboard .

Για τη σύνδεση όλων των εξαρτημάτων μεταξύ τους, προτίμησα τη χρήση ενός breadboard 62x14 για την ευκολότερη διόρθωση λαθών και πιθανή προσθήκη μελλοντικών εξαρτημάτων για επέκταση της κατασκευής.

3.1.8 Τροφοδοσία.

Για την τροφοδοσία του οχήματος έχω χρησιμοποιήσει μπαταρίες. Οι κινητήρες λόγω του αυξημένου βάρους της κατασκευής χρειάστηκαν 12V τάσης ώστε να λειτουργούν αποτελεσματικά. Επίσης μια μπαταρία 6V είναι αρκετή για τη λειτουργία του Arduino.



Για να επιτύχω την επιθυμητή τάση των 12V χρησιμοποίησα 2 «πακέτα» των 4 AA μπαταριών του 1,5V και συνδέοντας τα σε σειρά πετυχαίνω το αποτέλεσμα που θέλω. Σε περίπτωση που τις συνδέω παράλληλα θα πετύχαινα τάση 6V αλλά αυξημένη στο διπλάσιο ένταση ρεύματος, όπως βλέπουμε και στο διπλανό σχήμα.

3.2 Ο προγραμματισμός.

Όπως αναφέραμε και πριν, ο προγραμματισμός του Arduino γίνεται σε μια παραλλαγή της γλώσσας C++ (βλέπε [3]). Ας δούμε αναλυτικά, κομμάτι κομμάτι τον τρόπο με τον οποίο πέτυχα την εξομοίωση του προσαρμοζόμενου ελέγχου πορείας καθώς και την παραμονή ανάμεσα στις λωρίδες που έχω θέσει σαν πίστα.

3.2.1 Δηλώσεις μεταβλητών.

```
1. #include "Ultrasonic.h"
2. #include <IRremote.h>
3. Ultrasonic ultrasonic( 12, 13 );
4. int x,s1,s2,s3,s4,g1,g2,onoff,dist1,dist2,k,tt,x5144,x4539;
5. const int motor1Pin1 = 9; // H-bridge leg 1 (pin 2, 1A)
6. const int motor1Pin2 = 10; // H-bridge leg 2 (pin 7, 2A)
7. const int motor2Pin1 = 5;
8. const int motor2Pin2 = 6;
9. const int enablePin = 8; // H-bridge enable pin
10. int right_ldr = 1;//analog pin to which LDR is connected
11. int left_ldr = 2;
12. int right_value = 0; //variable to store LDR values
13. int left_value = 0;
14. unsigned long t1,t2,t3,t4,p1,p2,ir;
15. int RECV_PIN = 7;
16. IRrecv irrecv(RECV_PIN);
17. decode_results results;
```

Στις πρώτες 2 σειρές (1 και 2) έχουμε την δήλωση των βιβλιοθηκών των οποίων θα κάνει χρήση το πρόγραμμα στη συνέχεια. Οι βιβλιοθήκες προσφέρουν επιπλέον δυνατότητες προς χρήση από τα προγράμματα, για παράδειγμα εάν δουλεύει με εξωτερικές συσκευές η εάν χρειάζεται χειρισμός δεδομένων. Η εντολή `#include` υποδηλώνει ότι θα χρησιμοποιηθεί η βιβλιοθήκη το όνομα της οποίας ακολουθεί μέσα σε `" "` ή `<>`.

Στη σειρά 3 έχουμε την ρουτίνα εξυπηρέτησης του αισθητήρα υπερήχων. Αυτή επιστρέφει την απόσταση από εμπόδιο ή προπορευόμενο όχημα. Συντάσσεται με τον τρόπο τον οποίο βλέπουμε `Ultrasonic(int TP, int EP)` όπου TP ο αριθμός του ψηφιακού πιν του Arduino στο οποίο καταλήγει ο ακροδέκτης Trig του HC-SR04 και EP ο αριθμός ψηφιακού πιν στο οποίο καταλήγει ο ακροδέκτης Echo του HC-SR04.

Στη σειρά 4 βλέπουμε τη δήλωση των ακέραιων μεταβλητών που θα χρησιμοποιηθούν στη συνέχεια από το πρόγραμμα. Αναλυτικά:

- X: Η απόσταση που μετράει ο HC-SR04.
- S1,S2: Η επιθυμητή ταχύτητα που θέτει ο οδηγός καθώς και μια βοηθητική μεταβλητή.
- S3,S4: Η τάση που δίνουμε στον κινητήρα που ελέγχει τις εμπρός ρόδες του οχήματος.
- G1,G2,K: Περιοριστικές μεταβλητές.
- Onoff: Η μεταβλητή που καθορίζει λειτουργία ή μη.
- Dist1,dist2: Αποστάσεις ανάμεσα στις οποίες διατηρείται σταθερή η ταχύτητα.

Έπειτα στις σειρές 5 έως 13 έχουμε κάποιες ακέραιες μεταβλητές οι οποίες παραμένουν σταθερές για όλη τη διάρκεια του προγράμματος. Αυτές αφορούν τα πινς των κινητήρων καθώς και το πιν ενεργοποίησης του ολοκληρωμένου που τους ελέγχει. Επίσης δηλώνουν σε ποια πινς είναι συνδεδεμένα τα αισθητήρια των φωτοαντιστάσεων.

Στη σειρά 14 έχουμε δήλωση μεταβλητών οι οποίες είναι πιθανό να πάρουν μεγάλο μέγεθος και γι αυτό το λόγο πρέπει να δηλωθούν ως μεταβλητές αορίστου μεγέθους. Αυτές αφορούν κυρίως χρόνους καθώς και τα δεδομένα από τον υπέρυθρο δέκτη.

Στις σειρές 15-17 έχουμε την ρύθμιση των παραμέτρων του δέκτη υπέρυθρων. Η συνάρτηση `IRrecv irrecv(receivePin)` χρησιμοποιείται για τον ορισμό του εισερχόμενου σήματος ενώ ο αριθμός `receivePin` δηλώνει το πιν στο ποίο είναι συνδεδεμένο πάνω στον Arduino. Η δεχόμενη τιμή αποθηκεύεται μέσα στη μεταβλητή `results`.

3.2.2 Setup.

Στο επόμενο κομμάτι του προγράμματος έχουμε την θέσπιση αρχικών τιμών στις μεταβλητές που επιθυμούμε καθώς και εντολές για εκκίνηση διαδικασιών. Αναλυτικά:

1. `void setup()`
2. `{`
3. `Serial.begin(9600);`
4. `pinMode(motor1Pin1, OUTPUT);`
5. `pinMode(motor1Pin2, OUTPUT);`
6. `pinMode(motor2Pin1, OUTPUT);`
7. `pinMode(motor2Pin2, OUTPUT);`
8. `pinMode(enablePin, OUTPUT);`
9. `digitalWrite(enablePin, HIGH);`

```

10. s1=0;
11. g1=0;
12. g2=0;
13. irrecv.enableIRIn(); // Start the receiver
14. ir=0;
15. onoff=0;
16. dist1=41;
17. dist2=30;
18. x4539=0;
19. x5144=0;
20. }

```

Η πρώτη σειρά δηλώνει την αρχή του κομματιού του setup.

Στην Τρίτη γραμμή έχουμε την εκκίνηση λειτουργίας της σειριακής θύρας του Arduino καθώς και το επιθυμητό baud rate. Αυτό γίνεται για να αποτυπώσουμε τυχόν επιθυμητές μεταβλητές στην οθόνη του υπολογιστή για περαιτέρω χρήση.

Στις σειρές 4 έως 8 ορίζουμε τα αντίστοιχα πιν σαν εξόδους.

Στη σειρά 9 δίνουμε default τιμή high στο πιν enablepin. Αυτό αφορά το πιν ενεργοποίησης του ολοκληρωμένου κυκλώματος που οδηγεί τους κινητήρες.

Έπειτα στις σειρές 10 έως 19 δίνουμε αρχικές τιμές σε όσες μεταβλητές το χρειάζονται με εξαίρεση την σειρά 13 στην οποία δίνουμε την εντολή για εκκίνηση λειτουργίας του δέκτη υπερέθρων.

3.2.3 Void loop() και δέκτης υπερέθρων.

Με την εντολή "void loop()" ξεκινάει το κυρίως μέρος του προγράμματος.

Στην αρχή του προγράμματος έχω βάλει την απαραίτητη εντολή για το χειρισμό και τη λειτουργία του δέκτη υπερέθρων.

```

1. void loop()
2. { // RECEIVED IR
3. if (irrecv.decode(&results)) {
4. ir=(results.value);
5. irrecv.resume();
6. }

```


Στην πρώτη γραμμή υπάρχει η εντολή εκκίνησης του προγράμματος.

Στη δεύτερη γραμμή κάποια σχόλια τα οποία για να μην επηρεάζουν το πρόγραμμα πρέπει υποχρεωτικά να μπουν μετά //.

Στη γραμμή 3 υπάρχει η συνάρτηση (`irrecv.decode(&results)`). Η συγκεκριμένη συνάρτηση επιστρέφει `true` σε περίπτωση που εμφανιστεί σήμα (πατήσουμε πλήκτρο). Όταν δεχτεί το σήμα περνάει την τιμή του στη μεταβλητή `results` την οποία εγώ χρησιμοποιώ για να μεταφέρω σε μια άλλη μεταβλητή για χρήση παρακάτω. Τέλος η εντολή `irrecv.resume()`; προετοιμάζει το δέκτη για επόμενο σήμα.

Στη συνέχεια επεξεργάζομαι τα αποτελέσματα της παραπάνω συνάρτησης:

```
1. // ΕΠΕΚΣΕΡΓΑΣΙΑ IR
2. if (ir==16720605) {
3.   onoff=1;
4.   s1=85;
5.   s2=86;
6.   delay(100);
7.   ir=0;
8. }
9. if (ir==16753245) {
10.  onoff=0;
11.  s1=0;
12.  delay(100);
13.  ir=0;
14. }
15. if (ir==16761405) {
16.  ++dist1;
17.  ++dist2;
18.  delay(100);
19.  ir=0;
20. }
21. if (ir==16712445) {
22.  --dist1;
23.  --dist2;
24.  delay(100);
25.  ir=0;
26. }
27. if (ir==16748655) {
28.  ++s2;
```

```

29. ++s2;
30. delay(100);
31. ir=0;
32. }
33. if (ir==16754775) {
34. --s2;
35. --s2;
36. if (s1>s2) {
37. --s1;
38. --s1;
39. }
40. delay(100);
41. ir=0;
42. }
43. //TELOS IR

```

Μετά από ένα σχόλιο για διευκόλυνση μου στην πρώτη γραμμή, έχουμε συνεχώς περιπτώσεις if. Με τη συγκεκριμένη συνάρτηση συγκρίνω το εισερχόμενο σήμα από την συνάρτηση «irrecv.decode(&results)» με τις τιμές στις οποίες έχω θέσει διάφορες λειτουργίες.

Στις σειρές 2 έως 8 έχω θέσει τη λειτουργία για ενεργοποίηση του συστήματος προσαρμοζόμενου ελέγχου πορείας. Έτσι όταν ο δέκτης υπερύθρων δεχτεί το πλήκτρο με κωδικό 16720605, γίνεται 1 η μεταβλητή onoff ενώ θέτω ταυτόχρονα και τη default επιθυμητή ταχύτητα του συστήματος. Η εντολή delay(100) που ακολουθεί χρησιμεύει στο να «περιμένει» το πρόγραμμα 100ms έως ότου το πλήκτρο είναι και πάλι ελεύθερο και αμέσως μετά η μεταβλητή ir που δείχνει ποιο πλήκτρο πατήθηκε παίρνει ξανά την τιμή 0.

Αμέσως μετά στις σειρές 9 έως 14 έχουμε την εντολή απενεργοποίησης του συστήματος. Όταν πατηθεί το πλήκτρο με κωδικό 16753245 η μεταβλητή onoff θα πάρει την τιμή 0 ενώ θα μηδενιστεί και η ταχύτητα του οχήματος (s1). Ακολουθεί ξανά η εντολή delay(100) και ο μηδενισμός της μεταβλητής ir.

Στις σειρές 15 έως 20 έχουμε την εντολή χειρισμού του συστήματος που αφορά την αύξηση της επιθυμητής απόστασης από το προπορευόμενο όχημα. Οι εντολές ++dist1 και ++dist2 αυξάνουν κατά 1 την τιμή των μεταβλητών dist1 και dist2 αντίστοιχα.

Έπειτα, στις σειρές 21 έως 26 έχουμε παρόμοια λειτουργία με την παραπάνω. Αυτή τη φορά ωστόσο αντί να αυξάνουμε την επιθυμητή απόσταση την μειώνουμε κατά 1 μονάδα (εκατοστό).

Στις σειρές 27 έως 32 έχουμε άλλη μια λειτουργία χειρισμού που θα χρησιμοποιεί ο οδηγός για έλεγχο του συστήματος. Αυτή τη φορά, όταν πατηθεί το πλήκτρο με τον αντίστοιχο κωδικό, η τιμή της επιθυμητής ταχύτητας θα αυξάνεται κατά 1 μονάδα.

Τέλος, στις σειρές 33 έως 42 έχουμε τη μείωση της επιθυμητής ταχύτητας. Η διαφορά βρίσκεται στις σειρές 36 έως 39. Όταν η πραγματική ταχύτητα ξεπερνάει την επιθυμητή το πάτημα του πλήκτρου πρέπει να μειώνει κατά 1 και την αποθηκευμένη ταχύτητα. Αυτό επιτυγχάνουμε στις σειρές 36-39.

3.2.4 Φωτοαντιστάσεις και υπολογισμός απόστασης.

Ακολουθούν 2 εντολές οι οποίες διαβάζουν την τιμή που επιστρέφουν τα αισθητήρια των φωτοαντιστάσεων στο μπροστινό τμήμα της κατασκευής καθώς και μια εντολή η οποία μας δίνει την τιμή που επιστρέφει το αισθητήριο υπερήχων. Αυτές είναι:

1. `right_value = analogRead(right_ldr);`
2. `left_value = analogRead(left_ldr);`
3. `x=ultrasonic.Ranging(CM);`

Η συνάρτηση `analogRead()` διαβάζει την τιμή από το ορισμένο πιν του Arduino. Λόγω του ότι πρόκειται για αναλογικό σήμα, ο Arduino διαθέτει 6 αναλογικά πινς τα οποία όταν δέχονται ένα σήμα από 0 έως 5 Βολτ το μετατρέπουν σε αριθμό από το 0 έως το 1023.

Έτσι, διαβάζοντας τις τιμές αυτές μπορούμε να αντιληφθούμε την παρουσία των λωρίδων κάτω από τις φωτοαντιστάσεις πράγμα το ποίο σημαίνει ανάγκη στροφής των τροχών.

Η συνάρτηση `ultrasonic.Ranging(CM)` μετατρέπει το σήμα που επιστρέφει το αισθητήριο υπερήχων σε απόσταση, συγκεκριμένα σε εκατοστά του μέτρου και αποθηκεύει την τιμή σε μια μεταβλητή `x` για μελλοντική χρήση.

3.2.5 Σειριακή θύρα.

Στη συνέχεια συναντάμε κάποιες εντολές οι οποίες δεν επηρεάζουν με κάποιο τρόπο το πρόγραμμα ωστόσο είναι απαραίτητες σε εμάς για την βελτιστοποίηση του, παρουσιάζοντας στην οθόνη του υπολογιστή μας τις τιμές των παραμέτρων που θέλουμε να ελέγξουμε. Αυτές είναι:

1. `Serial.print(ultrasonic.Ranging(CM));`
2. `Serial.println("cm");`
3. `Serial.println(s1);`
4. `Serial.println(s3);`
5. `Serial.println(s4);`
6. `Serial.println(left_value);`
7. `Serial.println(right_value);`
8. `delay(100);`

Η εντολή `Serial.println()` εμφανίζει στη σειριακή θύρα του υπολογιστή την τιμή της μεταβλητής μέσα στις παρενθέσεις. Η εντολή `Serial.println()` κάνει ακριβώς το ίδιο πράγμα μόνο που εμφανίζει την τιμή στην επόμενη σειρά της θύρας. Η εντολή `delay(100)` έχει μπει για να εμφανίζονται οι μεταβλητές που θέλουμε κάθε 100ms για ευκολία στην όψη.

3.2.6 Στρίψιμο τροχών.

Στο επόμενο κομμάτι ορίζω την τάση που πρέπει να πάρει ο κινητήρας που ελέγχει τις μπροστινές ρόδες σε περίπτωση που οι φωτοαντιστάσεις αντιληφθούν τη λωρίδα και υπάρχει ανάγκη στροφής.

Σε αυτό το σημείο να δούμε λίγο πως ακριβώς ελέγχονται οι κινητήρες από τον Arduino. Ο Arduino διαθέτει 6 πινς με την ένδειξη PWM. Αυτά τα πινς μπορούν να χρησιμοποιηθούν για να δώσουμε αναλογικές τιμές στην έξοδο τους χρησιμοποιώντας την εντολή `analogWrite()`. Όταν εμφανιστεί η `analogWrite()` το πιν στο οποίο αναφέρεται θα παράγει ένα σταθερό τετραγωνικό παλμό με κύκλο εργασίας ανάλογο της τιμής που έχουμε δώσει επιτρέποντας μας να ελέγχουμε την ταχύτητα του κινητήρα αλλά και διάφορες άλλες λειτουργίες όπως τη φωτεινότητα ενός led. Παρακάτω βλέπουμε τον κώδικα:

```
1. if (onoff==1){
2.   if (right_value<220){
3.     if (left_value<500){
4.       s3=0;
5.       s4=0;
6.     }
7.   else if (left_value>500) {
8.     s3=0;
9.     s4=200;
10.  }
11. }
12. else if (right_value>220) {
13.   s3=200;
14.   s4=0;
15. }
```

Στη σειρά 1 ελέγχουμε εάν έχει πατηθεί το πλήκτρο on. Εάν αυτό έχει γίνει τότε το πρόγραμμα συνεχίζει ελέγχοντας τις εάν οι τιμές των φωτοαντιστάσεων είναι μέσα στα επιτρεπτά όρια πράγμα το

Εδώ να σημειώσουμε ότι η συνάρτηση millis() επιστρέφει τον χρόνο που έχει περάσει από τη στιγμή που ενεργοποιήθηκε ο Arduino σε milliseconds. Πρέπει να δηλωθεί σαν unsigned long καθώς παίρνει πολύ μεγάλες τιμές και θα υπερχειλίσει μετά από περίπου 50 ημέρες.

3.2.8 Αυξομείωση απόστασης.

Στη συνέχεια θα δούμε πως επιτυγχάνεται η αυξομείωση της επιθυμητής απόστασης μέσω της αυξομείωσης της ταχύτητας του οχήματος.

```
1. // AYKSOMEIWSH APOSTASHS
2. if (x>dist1) {
3.   if (s1<s2) {
4.     if (g1==0){
5.       t1=millis();
6.       ++g1;
7.     }
8.     p1=(millis()-t1);
9.     if (p1>200) {
10.      ++s1;
11.      g1=0;
12.    } }
13. analogWrite(motor1Pin1, s1);
14. analogWrite(motor1Pin2, 0);
15. analogWrite(motor2Pin1, s3);
16. analogWrite(motor2Pin2, s4);
17. }
18. else if (x<dist1&& x>dist2){
19.   analogWrite(motor1Pin1, s1);
20.   analogWrite(motor1Pin2, 0);
21.   analogWrite(motor2Pin1, s3);
22.   analogWrite(motor2Pin2, s4);
23. }
24. else if (x<dist2){
25.   if (s1>30){
26.     if (g2==0){
27.       t2=millis();
28.       ++g2;
29.     }
30.     p2=(millis()-t2);
31.     if (p2>200) {
```

```

32. --s1;
33. g2=0;
34. }
35. analogWrite(motor1Pin1, s1);
36. analogWrite(motor1Pin2, 0);
37. analogWrite(motor2Pin1, s3);
38. analogWrite(motor2Pin2, s4);
39. }}}
40. // TELOS AYKSOMEIWSIS

```

Σε αυτό το κομμάτι βλέπουμε 3 περιπτώσεις:

- Περίπτωση πρώτη: Η απόσταση του οχήματος μας από το προπορευόμενο είναι μεγαλύτερη από την επιθυμητή ορισμένη από τον οδηγό. Σε αυτή τη περίπτωση το πρόγραμμα αυξάνει την τιμή της ταχύτητας κατά 1 μονάδα κάθε 200ms εάν φυσικά αυτή δεν υπερβαίνει την ορισμένη από τον οδηγό. Δηλαδή, εάν το προπορευόμενο όχημα έχει ταχύτητα 80χλμ/ώρα και επιταχύνει στα 100, το πρόγραμμά μας θα επιταχύνει το όχημα μας μέχρι τα 85χλμ/ώρα η οποία είναι η default ρύθμιση και δε θα τα ξεπεράσει.
- Περίπτωση δεύτερη: Η απόσταση του οχήματος από το προπορευόμενο είναι ανάμεσα στην επιθυμητή απόσταση. Σε αυτή την περίπτωση η ταχύτητα παραμένει σταθερή.
- Περίπτωση τρίτη: Η απόσταση του οχήματος από το προπορευόμενο είναι μικρότερη από την επιθυμητή. Τότε, το πρόγραμμα θα μειώνει την ταχύτητα του οχήματος κατά 1 μονάδα κάθε 200ms έως ότου αυτή αυξηθεί ξανά στα επιθυμητά όρια.

3.2.9 Απενεργοποίηση.

Το τελευταίο κομμάτι αφορά τη διαδικασία απενεργοποίησης του συστήματος:

```

1. // APENERGOPOIHSH
2. if (onoff==0){
3. s1=0;
4. analogWrite(motor1Pin1, 0);
5. analogWrite(motor1Pin2, 0);
6. analogWrite(motor2Pin1, s3);
7. analogWrite(motor2Pin2, s4);
8. }}

```

Όταν πατηθεί το πλήκτρο off και η μεταβλητή onoff πάρει την τιμή 0, η ταχύτητα θα μηδενιστεί και αυτό θα προκαλέσει ελεύθερη κύλιση του οχήματος μέχρι να φτάσει σε πλήρη στάση. Φυσικά, το σύστημα θα συνεχίσει να λειτουργεί όσον αφορά την κατεύθυνση του οχήματος έτσι ώστε να παραμείνει στα όρια της πίστας και να μην ξεφύγει εκτός ελέγχου.

Κεφάλαιο 4ο: Μελλοντικές βελτιώσεις.

Συστήματα τα οποία βασίζονται στον ηλεκτρονικό έλεγχο χρησιμοποιούνται ολοένα και περισσότερο στη βιομηχανία αυτοκινήτων με σκοπό να προφέρουν ασφάλεια και άνεση στον οδηγό και τους επιβάτες, με συνεχώς αυξανόμενη χρήση αυτοματισμών. Υπάρχει αυξανόμενη τάση στις εφαρμογές οι οποίες βοηθούν τον οδηγό να κρατάει σταθερή ταχύτητα και απόσταση από προπορευόμενα οχήματα χωρίς να περιορίζονται από καιρικές συνθήκες, συνθήκες μπουτλιαρίσματος ή ακόμα και ακίνητων εμποδίων. Αυτά αναλύσαμε έως τώρα.

Ωστόσο, αν και τα συστήματα προσαρμοζόμενου ελέγχου πορείας έχουν φτάσει σε σημείο να αποτρέπουν ατυχήματα είτε ενεργοποιώντας τα συστήματα ασφαλείας του αυτοκινήτου είτε προειδοποιώντας τον οδηγό, σε βάθος χρόνου ελπίζουμε να αυτοματοποιηθεί πλήρως η οδήγηση ενός αυτοκινήτου ή και ομάδας αυτοκινήτων.

4.1 Cooperative Adaptive Cruise Control (CACC, Συνεργαζόμενος Προσαρμοζόμενος Έλεγχος Πορείας)

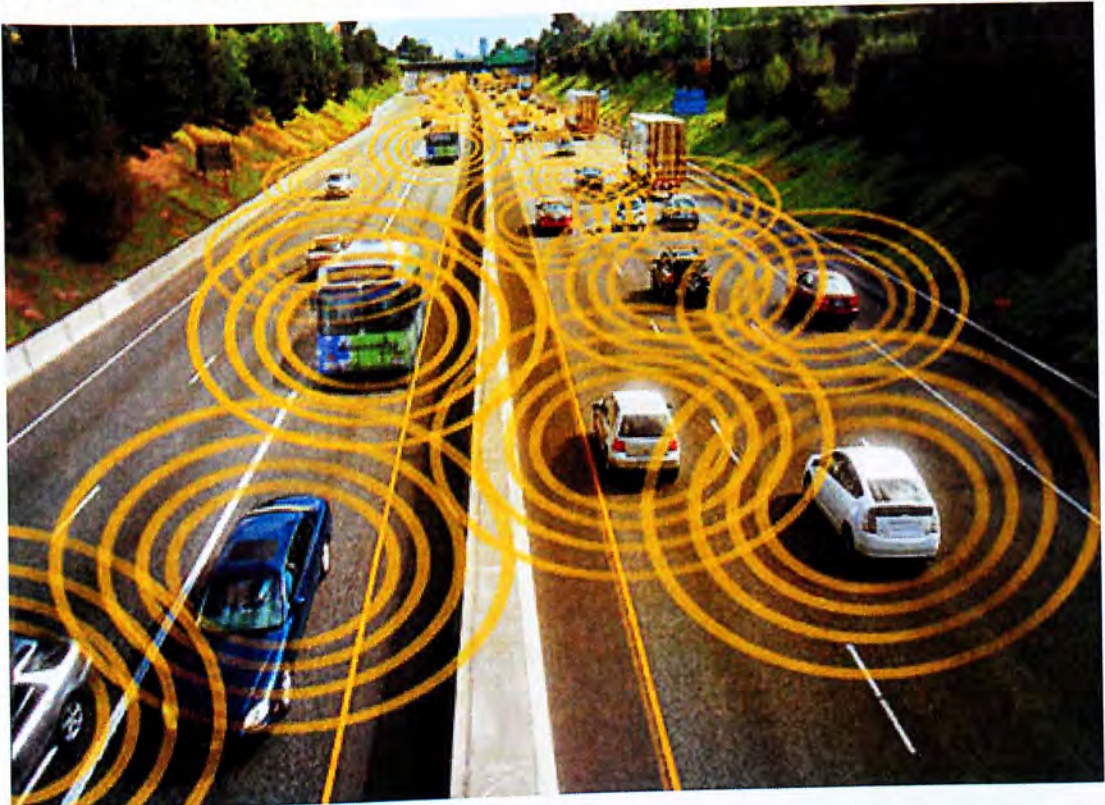
4.1.1 Τι είναι;

Το σύστημα συνεργαζόμενου προσαρμοζόμενου ελέγχου πορείας διατηρεί σταθερή την απόσταση από το προπορευόμενο όχημα ανάλογα με την επιθυμία του οδηγού ακριβώς όπως ο απλός προσαρμοζόμενος έλεγχος πορείας. Επιπροσθέτως, κάνοντας χρήση ενός συνδυασμού αισθητηρίων ραντάρ και κάμερας τοποθετημένα στο μπροστινό μέρος του οχήματος το CACC ανιχνεύει πότε ένα άλλο όχημα (το λεγόμενο όχημα-στόχος) αυξομειώνει την δική του ταχύτητα μέσω φρένων ή γκαζιού για να επιτύχει την παραμονή σε ασφαλή απόσταση από το δικό του προπορευόμενο όχημα. Το αυτοκίνητο δέχεται επίσης πληροφορίες μέσω GPS, όπως τοποθεσία, ταχύτητα και κατεύθυνση, οχημάτων που προπορεύονται καθώς εκπέμπει και πληροφορίες μέσω GPS στα οχήματα που ακολουθούν.

Αυτές οι πληροφορίες μπορούν να χρησιμοποιηθούν για να στηθεί ένα κομβίο αυτοκινήτων το οποίο θα ακολουθεί το πρώτο όχημα, διατηρώντας απόσταση ασφαλείας μεταξύ των μελών του. Κάθε όχημα στο κομβίο χρησιμοποιεί πληροφορίες που δέχεται μέσω GPS καθώς και από τα δικά του συστήματα ραντάρ και κάμερας για να ελέγξει το γκάζι και το φρένο ώστε να επιτευχθεί η απόσταση ασφαλείας. Με τη σειρά του, το όχημα εκπέμπει πληροφορίες σχετικά με την δική του κίνηση για χρήση από τα υπόλοιπα οχήματα.

Τα συστήματα CACC περιλαμβάνουν επίσης συμπληρωματικά στοιχεία τα οποία προσφέρουν στον οδηγό ασφάλεια και διευκόλυνση. Πολλαπλά χαρακτηριστικά, όπως προσαρμοζόμενος έλεγχος πορείας, παραμονή στο κέντρο της λωρίδας/αλλαγή λωρίδας και υποβοήθηση ταχύτητας στις στροφές, μπορεί να χρειαστεί να εφαρμόσουν διάφορες ποσότητες φρένου ή γκαζιού σε διαφορετική ισχύ και υπό διάφορες εξωτερικές συνθήκες. Οι αλληλεπιδράσεις μεταξύ των στοιχείων αυτών πρέπει να αντιμετωπίζεται με σεβασμό στην περίπτωση λάθους στον προγραμματισμό που επιδρά την ασφαλή πορεία του οχήματος.

Για παράδειγμα, εάν ένα από τα στοιχεία αυτά τείνει να επιταχύνει το όχημα ενώ ένα άλλο σύστημα κρίνει ότι χρειάζεται μείωση της ταχύτητας, ο εποπτικός ελεγκτής του οχήματος θα πρέπει να ξέρει αρκετές πληροφορίες σχετικά με τη γενική κατάσταση του περιβάλλοντος ώστε να καθορίσει ποια από τις ενέργειες είναι πιο κρίσιμη και πρέπει να πραγματοποιηθεί στέλνοντας ανάλογα σήματα στα περιφερειακά. Ένα χαρακτηριστικό που «ζητάει» επιβράδυνση (πχ καμπύλη ταχύτητας) μπορεί να είναι πιο κατάλληλη για τις συγκεκριμένες συνθήκες παρά ένα χαρακτηριστικό που ζητάει επιτάχυνση (πχ προσαρμοζόμενος έλεγχος πορείας). Σε αυτή την περίπτωση, ο επιβλέπων ελεγκτής του οχήματος θα επέμβει επιτρέποντας στο πρώτο στοιχείο να παρακάμψει την επιθυμία για επιτάχυνση προερχόμενη από το δεύτερο και να επιβραδύνει.



Cooperative Adaptive Cruise Control (βλέπε [22])

4.1.2 Υποσυστήματα.

Ένα σύστημα CACC αποτελείται από τα παρακάτω υποσυστήματα:

- 1) Αισθητήρας ραντάρ.
 - a. Ανίχνευση, ταυτοποίηση και προσδιορισμός θέσης οχήματος-στόχου.
 - b. Ταυτοποίηση οχήματος-στόχου μέσω αναμεταδότη με σκοπό την καθιέρωση διαύλου επικοινωνίας.
- 2) Τηλεπικοινωνία. (wi-fi)
 - a. Τηλεπικοινωνία με το όχημα-στόχος.
 - b. Τηλεπικοινωνία με το όχημα που ακολουθεί.
 - c. Τηλεπικοινωνία με τις υποδομές.
- 3) Ηλεκτρονικός έλεγχος γκαζιού.
 - a. Αυξομείωση ταχύτητας οχήματος δίνοντας η κόβοντας ισχύ.
- 4) Φρένο μέσω καλωδίου.
 - a. Αυξομείωση ταχύτητας του οχήματος χρησιμοποιώντας τα φρένα όταν υπάρχει ανάγκη επιβράδυνσης.
- 5) Αναμεταδότης ραντάρ.
 - a. Όταν ζητηθεί, δίνει την ταυτότητα του οχήματος σε οχήματα που ακολουθούν για την καθιέρωση επικοινωνίας μέσω δικτύου wi-fi μεταξύ τους.
- 6) Σύστημα GPS.
 - a. Διατηρεί ακριβείς πληροφορίες για τη θέση, την ταχύτητα και την κατεύθυνση του οχήματος.
 - b. Υποβοηθά το σύστημα ραντάρ στο να ξεχωρίζει τα οχήματα-στόχους από γνωστούς ακίνητους στόχους και επιδιορθώνει τις απώλειες από στροφές ή ανηφόρες στην πορεία.
 - c. Υποβοηθά τον ελεγκτή του οχήματος να διατηρήσει τη θέση στο κομβίο σε περίπτωση βλάβης του ραντάρ.
- 7) Αισθητήρας κάμερας.
 - a. Οπτική ταυτοποίηση προπορευόμενου οχήματος και εκτίμηση απόστασης και σχετικής ταχύτητας του.
- 8) Ελεγκτής αυτοκινήτου.
 - a. Συντονίζει όλα τα υποσυστήματα.

- b. Υπολογίζει την ταχύτητα του οχήματος, την ταχύτητα του προπορευόμενου και προσαρμόζει την ταχύτητα για τη διατήρηση ασφαλούς απόστασης.
- c. Διατηρεί την κατάσταση του οχήματος και πληροφορίες σχετικά με το περιβάλλον.
- d. Ελέγχει γκάζι και φρένο.
- e. Δέχεται πληροφορίες από το σύστημα ραντάρ.
- f. Στέλνει και δέχεται πληροφορίες μέσω wi-fi.

4.1.3 Πλεονεκτήματα-Μειονεκτήματα.

Υπάρχουν πολλαπλά πλεονεκτήματα από την εφαρμογή του CACC στα οχήματα του εμπορίου.

1. Εξοικονόμηση καυσίμου: Έως και 20% λιγότερη κατανάλωση καυσίμου λόγω της βέλτιστης χρήσης του γκαζιού από το σύστημα
2. Ασφάλεια: Μειωμένες συγκρούσεις μετωπικές καθώς και από πίσω λόγω των συστημάτων αποφυγής σύγκρουσης.
3. Μποτιλιάρισμα: Μειωμένη κίνηση στους δρόμους μέσω κατάλληλης αυξομείωσης της ταχύτητας όλων των οχημάτων σε ένα δρόμο.
4. Βέλτιστη χρήση αυτοκινητοδρόμων: Οι αποστάσεις ασφαλείας επιτρέπουν τη μέγιστη χρήση των αυτοκινητοδρόμων χωρίς αυτοί να υπερφορτώνονται.

Ωστόσο υπάρχουν και λίγα μειονεκτήματα όπως αυτό της περίπτωσης σφάλματος όπου μια σύγκρουση πολλαπλών οχημάτων είναι πολύ πιθανή. Επίσης το κόστος των συστημάτων αυτών είναι αρκετά μεγάλο διότι εκτός από τις εφαρμογές στα αυτοκίνητα πρέπει να διαμορφωθούν κατάλληλα και οι δρόμοι καθώς χρειάζεται προσθήκη ειδικών σημάτων.

Βιβλιογραφία

- [1] Ηλεκτρικές μηχανές (Γ. Πολίτης, Γ. Τσεκούρας)
- [2] Ηλεκτρονικά ισχύος (Γ. Πολίτης, Ι. Τσάλας)
- [3] C++ Προγραμματισμός (Δ. Κυτάγιας)
- [4] <http://www.iamaudi.com/audi-adaptive-cruise-control-acc-in-depth-review/>
- [5] <http://www.hido.or.jp/ITSHPe/wi/itshb/Preview.htm>
- [6] <http://auto.howstuffworks.com/cruise-control4.htm>
- [7] <http://delphi.com/manufacturers/auto/safety/active/adaptive-cruise-control/>
- [8] <http://townhall-talk.edmunds.com/direct/view/.f228735>
- [9] <http://www.xappsoftware.com/wordpress/2012/03/15/how-to-interface-the-hc-sr04-ultrasonic-ranging-module-to-arduino/>
- [10] http://en.wikipedia.org/wiki/H_bridge
- [11] http://www.cse.msu.edu/~chengb/RE-491/Projects/cacc_msu-ford.pdf
- [12] http://en.wikipedia.org/wiki/Autonomous_cruise_control_system
- [13] <http://www.jdpower.com/content/detail.htm?jdpaArticleId=222>
- [14] <http://cartech.about.com/od/Safety/a/How-Does-Adaptive-Cruise-Control-Work.htm>
- [15] http://en.wikipedia.org/wiki/Cruise_control
- [16] http://media.ford.com/article_display.cfm?article_id=2572
- [17] <http://communityofrobots.com/tutorial/kawal/how-use-ldr-arduino>
- [18] <http://learn.adafruit.com/ir-sensor>
- [19] <http://arduino.cc/>
- [20] http://media.ford.com/images/10031/Adaptive_Cruise_MKX.pdf
- [21] <http://www.youtube.com/watch?v=cMkorgujmw&feature=relmfu>
- [22] <http://doc.utwente.nl/77622/1/KleinWolterink11automated.pdf>
- [23] <http://www.youtube.com/watch?v=hyFnWw0w9ZQ>

[24] <http://www.honda-acura.net/forums/news-and-rumors-archives/71050-honda-develops-worlds-first-collision-mitigation-brake-system.html>

[25] http://www.euroncap.com/rewards/audi_pre_sense_front_plus.aspx

[26] <http://www.xappsoftware.com/wordpress/2012/03/15/how-to-interface-the-hc-sr04-ultrasonic-ranging-module-to-arduino/>