

ΠΡΟΓΡΑΜΜΑ ΜΕΤΑΠΤΥΧΙΑΚΩΝ ΣΠΟΥΔΩΝ
«ΡΟΜΠΟΤΙΚΗ, STEAM ΚΑΙ ΝΕΕΣ ΤΕΧΝΟΛΟΓΙΕΣ
ΣΤΗΝ ΕΚΠΑΙΔΕΥΣΗ»

Διπλωματική Εργασία

ΚΙΝΗΤΗ ΡΟΜΠΟΤΙΚΗ ΜΕ ARDUINO – Υλοποίηση και ανάπτυξη
εκπαιδευτικού υλικού

του

ΓΚΟΥΜΑ ΚΩΝΣΤΑΝΤΙΝΟΥ

Επιβλέπων Καθηγητής
Υφύλης Χρήστος

Υποβλήθηκε ως απαιτούμενο για την απόκτηση του μεταπτυχιακού διπλώματος
ειδίκευσης Ρομποτική, STEAM και Νέες Τεχνολογίες στην Εκπαίδευση
Θεσσαλονίκη, Φεβρουάριος 2023



Η παρούσα Διπλωματική Εργασία καλύπτεται στο σύνολό της νομικά από δημόσια άδεια πνευματικών δικαιωμάτων CreativeCommons:

Αναφορά Δημιουργού - Μη Εμπορική Χρήση - Παρόμοια Διανομή



Μπορείτε να:

- Μοιραστείτε: αντιγράψτε και αναδιανέμετε το παρόν υλικό με κάθε μέσο και τρόπο
- Προσαρμόστε: αναμείξτε, τροποποιήστε και δημιουργήστε πάνω στο παρόν υλικό

Υπό τους ακόλουθους όρους:

- Αναφορά Δημιουργού: Θα πρέπει να καταχωρίσετε αναφορά στο δημιουργό, με σύνδεσμο της άδειας, και με αναφορά αν έχουν γίνει αλλαγές. Μπορείτε να το κάνετε αυτό με οποιονδήποτε εύλογο τρόπο, αλλά όχι με τρόπο που να υπονοεί ότι ο δημιουργός αποδέχεται το έργο σας ή τη χρήση που εσείς κάνετε.
- Μη Εμπορική Χρήση: Δε μπορείτε να χρησιμοποιήσετε το υλικό για εμπορικούς σκοπούς.
- Παρόμοια Διανομή: Αν αναμείξετε, τροποποιήσετε, ή δημιουργήσετε πάνω στο παρόν υλικό, πρέπει να διανείμετε τις δικές σας συνεισφορές υπό την ίδια άδεια CreativeCommons όπως και το πρωτότυπο.

Αναλυτικές πληροφορίες νομικού κώδικα στην ηλεκτρονική διεύθυνση:

<https://creativecommons.org/licenses/by-nc-sa/4.0/legalcode>

Η παρούσα Διπλωματική Εργασία και τα συμπεράσματά της, σε οποιαδήποτε μορφή, αποτελούν συνιδιοκτησία του Τμήματος Μηχανικών Παραγωγής και Διοίκησης του Διεθνούς Πανεπιστημίου Ελλάδος και του φοιτητή. Οι προαναφερόμενοι διατηρούν το δικαίωμα ανεξάρτητης χρήσης και αναπαραγωγής (τμηματικά ή συνολικά) για διδακτικούς και ερευνητικούς σκοπούς. Σε κάθε περίπτωση πρέπει να αναφέρεται ο τίτλος, ο συγγραφέας, ο επιβλέπων και το τμήμα του ΔιΠαΕ.

Η έγκριση της παρούσας Διπλωματικής Εργασίας από το Τμήμα Μηχανικών Παραγωγής και Διοίκησης δεν υποδηλώνει απαραίτητως και αποδοχή των απόψεων του συγγραφέα εκ μέρους του Τμήματος.

Ο υπογεγραμμένος δηλώνω υπεύθυνα ότι η παρούσα Διπλωματική Εργασία είναι εξ' ολοκλήρου δικό μου έργο και συγγράφηκε ειδικά για τις απαιτήσεις του μεταπτυχιακού προγράμματος σπουδών του Τμήματος Μηχανικών Παραγωγής και Διοίκησης με τίτλο «Ρομποτική, STEAM και νέες Τεχνολογίες στην Εκπαίδευση».

Δηλώνω υπεύθυνα ότι κατά τη συγγραφή ακολούθησα την πρόπουσα ακαδημαϊκή δεοντολογία αποφυγής λογοκλοπής και έχω αποφύγει οποιαδήποτε ενέργεια που συνιστά παράπτωμα λογοκλοπής.

Κωνσταντίνος Γκούμας

25/01/2024



ΠΕΡΙΛΗΨΗ

Στην εργασία αυτή θα σχεδιαστεί ένα εκπαιδευτικό πρόγραμμα εκμάθησης της πλατφόρμας Arduino με τελικό σκοπό την κατασκευή μιας ρομποτικής κατασκευής με τη χρήση αυτού. Θεωρείται ότι οι μαθητές δεν έχουν πρότερη εμπειρία στον Arduino και αρχικά γίνεται μια σύντομη υπενθύμιση εννοιών του ηλεκτρισμού αναγκαίων για την κατανόηση λειτουργίας ηλεκτρονικών στοιχείων και του Arduino. Το πρόγραμμα αποτελείται από επτά μαθήματα προσαρμοσμένα στην ωριαία διάρκεια του μαθήματος Πληροφορικής της Γ' Γυμνασίου . Το κάθε ένα μάθημα αποτελεί μία ενότητα που θέτει τις βάσεις για να προχωρήσει ο μαθητής στο επόμενο και προοδευτικά να αποκτήσει δεξιότητα στον προγραμματισμό και τη χρήση του Arduino, της προγραμματιστικού περιβάλλοντος και των ηλεκτρονικών. Τα μαθήματα ξεκινάνε από τους κανόνες του ηλεκτρισμού, προχωράνε σε χτίσιμο βασικών ηλεκτρονικών κυκλωμάτων, έπειτα στην εισαγωγή τόσο στην πλακέτα και στο προγραμματιστικό περιβάλλον του Arduino και στα τρία τελευταία μαθήματα γίνεται η κατασκευή του αυτοκινούμενου οχήματος. Οι μαθητές χωρίζονται σε τέσσερις ομάδες και η κάθε ομάδα έχει εσωτερική δομή και ξεχωριστούς ρόλους. Το όλο project είναι μεν καθοδηγούμενο αλλά ενθαρρύνεται και η προσωπική πρωτοβουλία και επινοητικότητα των μαθητών μέσω της μελέτης υλικού που δίνεται στο τέλος κάθε μαθήματος. Παράλληλα μέσω της εργασίας σε ομάδα ενθαρρύνεται το ομαδικό πνεύμα ο αλληλοσεβασμός και η συνεργατικότητα. Στο τέλος με την περάτωση της ρομποτικής κατασκευής ο μαθητής όχι μόνο έχει εξασκηθεί και αφομοιώσει την γνώση πάνω στην πλατφόρμα Arduino αλλά έχει να επιδείξει και ένα αυτοκινούμενο όχημα με ικανότητα αποφυγής εμποδίων. Μία κατασκευή την οποία ο μαθητής αγγίζει συναρμολογεί και θέτει σε λειτουργία είναι μία επιβράβευση της προσπάθειας και του κόπου και χτίζει αυτοπεποίθηση στον χαρακτήρα του παιδιού.

ABSTRACT

This paper is dealing with the design of an educational programme in Arduino learning and competence. The final goal is the construction of a robotic apparatus utilizing Arduino. It is considered that the students have no prior experience in Arduino and initially the students are reminded of fundamental electricity concepts essential for the understanding of how Arduino and electronics function. The programme is consisting of seven lessons adjusted to the one hour long Third grade Middle School classes. Each lesson is a build up for the next one as it sets the foundation for the upcoming classes and an increasing competency in electronics Arduino usage and programming. The lessons begin with fundamental electricity and electronics principles, building simple circuits, continuing in using the Arduino board and its programming environment IDE and in the last three lessons concluding in the building of the self-driving vehicle. The students are divided in four groups with internal structure and separate roles. The whole project is guided but personal initiative and innovation is encouraged by providing study material at the ending of each lesson. Moreover, through team work group spirit mutual respect and collaboration are being promoted. Finally, in concluding the building of the robotic apparatus the student has not only absorbed knowledge in Arduino but also has a self-driving and obstacle avoiding vehicle to show for. An apparatus which the student touches, constructs and sets in full function rewards the student's efforts and strengthens confidence within his / her character.

ΠΙΝΑΚΑΣ ΠΕΡΙΕΧΟΜΕΝΩΝ

ΚΕΦΑΛΑΙΑ	ΣΕΛΙΔΕΣ
I. Σχεδιασμός Εκπαιδευτικού Προγράμματος	8
II. Διατύπωση Στόχων κ' Σκοπών	9
Ανάπτυξη ωρολογίου προγράμματος - Χρονοδιάγραμμα	10
ΚΕΦΑΛΑΙΟ 1 – ΜΑΘΗΜΑ 1 ^ο : Ο Ηλεκτρισμός	11
1.1 Το ηλεκτρικό φορτίο Q	11-12
1.2 Ο Νόμος του Coulomb	12-13
1.3 Το Ηλεκτρικό Πεδίο E	13-14
1.4 Το Δυναμικό V	14-16
1.5 Το Ηλεκτρικό Ρεύμα I	16-17
1.6 Η Ηλεκτρική Αντίσταση R	17-20
1.7 Ο Νόμος του Ohm	20-21
ΚΕΦΑΛΑΙΟ 2 – ΜΑΘΗΜΑ 2 ^ο : Ηλεκτρονικά Κυκλώματα	22
2.1 Βασικά Στοιχεία Ηλεκτρονικών Κυκλωμάτων	22-30
2.2 Κατασκευάζουμε τα πρώτα μας κυκλώματα	31-33
2.3 Ανάβοντας ένα LED	34-36
2.4 Δύο LED σε σειρά	36-37
2.5 Δυο LED σε παράλληλη σύνδεση	37-40
ΚΕΦΑΛΑΙΟ 3 – ΜΑΘΗΜΑ 3 ^ο : Εισαγωγή στον Arduino	41
3.1 Μικροελεγκτές	41-42
3.2 Η πλατφόρμα Arduino	42-44
3.3 Η πλακέτα Arduino UNO Rev 3	45-49
3.4 Το Προγραμματιστικό Περιβάλλον του Arduino IDE	50-51
3.5 Εισαγωγή στον Προγραμματισμό του Arduino	51-61
ΚΕΦΑΛΑΙΟ 4 – ΜΑΘΗΜΑ 4 ^ο : Προγραμματισμός Arduino – Εισαγωγή στο Project	62
4.1 Προγραμματίζοντας τον Arduino	62-67
4.2 Το Project Αυτοκινούμενου Οχήματος	67-73
ΚΕΦΑΛΑΙΟ 5 – ΜΑΘΗΜΑ 5 ^ο : Έλεγχος Κινητήρων με τον Arduino	74
5.1 Κινητήρες DC	74-85
5.2 Κινητήρες Servo	85-87
ΚΕΦΑΛΑΙΟ 6 – ΜΑΘΗΜΑ 6 ^ο : Η Μονάδα Μέτρησης Αποστάσεων	88
6.1 Ο Αισθητήρας Υπερήχων HC-SR04	89-92
6.2 Προγραμματισμός του Αισθητήρα HC-SR04	93-98
ΚΕΦΑΛΑΙΟ 7 – ΜΑΘΗΜΑ 7 ^ο : Ολοκλήρωση του Οχήματος	99
7.1 Ολοκλήρωση της Κατασκευής του Οχήματος	99-103
7.2 Προγραμματισμός του Οχήματος	104-108
ΒΙΒΛΙΟΓΡΑΦΙΑ	109-115
ΠΑΡΑΡΤΗΜΑ Α -Κώδικας του Οχήματος	116-118

ΠΙΝΑΚΑΣ ΔΙΑΓΡΑΜΜΑΤΩΝ ΚΑΙ ΠΙΝΑΚΩΝ

ΤΙΤΛΟΣ	ΣΕΛΙΔΑ
Ανάπτυξη Ωρολογίου Προγράμματος - Χρονοδιάγραμμα	10
Resistor Color Code	19

I. ΣΧΕΔΙΑΣΜΟΣ ΕΚΠΑΙΔΕΥΤΙΚΟΥ ΠΡΟΓΡΑΜΜΑΤΟΣ

ΕΠΙΣΚΟΠΗΣΗ ΠΡΟΓΡΑΜΜΑΤΟΣ

Το εκπαιδευτικό πρόγραμμα αυτό αφορά την εξοικείωση των μαθητών με την πλατφόρμα Arduino. Η open-source πλατφόρμα αυτή είναι ιδανική για την εισαγωγή των μαθητών σε αντικείμενα STEM καθώς περιλαμβάνει τόσο την χρήση ηλεκτρονικών μερών όσο και προγραμματισμού. Η πλατφόρμα Arduino μάλιστα δίνει την δυνατότητα στα παιδιά να κατασκευάσουν και να προγραμματίσουν δικά τους projects τόσο καθοδηγούμενα όσο και με δικά τους πρωτοβουλία, έμπνευση και δημιουργικότητα. Το πρόγραμμα χωρίζεται σε επτά μαθήματα συνολικής διάρκειας 7 ωρών στα πλαίσια του μαθήματος Πληροφορικής της Γ' Γυμνασίου. Το κάθε μάθημα (μια ώρα διδασκαλίας) είναι διάρκειας 45 λεπτών. Η τάξη αποτελείται από 20 μαθητές οι οποίοι θα χωριστούν σε 4 ομάδες των 5 ατόμων και σε κάθε ομάδα θα δοθεί kit με την πλακέτα Arduino και όλα τα απαραίτητα παρελκόμενα για την κατασκευή του αυτοκινούμενου οχήματος με δυνατότητα αποφυγής εμποδίων. Τα μαθήματα γίνονται σε αίθουσα πληροφορικής και κάθε μαθητής έχει δικό του Η/Υ με σύνδεση στο internet και τον editor για τον προγραμματισμό Arduino IDE.

Οι ομάδες χωρίζονται σε σχεδιαστικές και προγραμματιστικές υποομάδες δύο ατόμων η καθεμιά με ένα άτομο να εκτελεί χρέη συντονιστή. Οι μαθητές διαμοιράζονται σε ομάδες με μέριμνα του καθηγητή ο οποίος φροντίζει να μάθει το επίπεδο γνώσης του κάθε μαθητή στο Arduino, τον προγραμματισμό και τα ηλεκτρονικά κυκλώματα. Ο σκοπός είναι κάθε ομάδα να έχει ένα ή και περισσότερα άτομα που να γνωρίζουν κάτι από αυτά τα αντικείμενα. Τα μαθήματα γίνονται διαζώσης με παρουσία καθηγητή και μετά τα εισαγωγικά θεωρητικά μαθήματα ξεκινάει η κατασκευή του καθοδηγούμενου project αυτοκινούμενου οχήματος με δυνατότητα αποφυγής εμποδίων με χρήση Arduino. Οι διαλέξεις των μαθημάτων γίνονται με την χρήση PowerPoint. Στις επόμενες σελίδες γίνεται η αναλυτική παράθεση της θεωρίας που θα διδαχτούν οι μαθητές των πρακτικών που θα ακολουθήσουν για την κατασκευή του οχήματος και των βημάτων που καταλήγουν στην κατασκευή αυτοκινούμενου οχήματος με χρήση Arduino.

II. ΔΙΑΤΥΠΩΣΗ ΣΤΟΧΩΝ ΚΑΙ ΣΚΟΠΩΝ

ΣΤΟΧΟΙ

- Οι μαθητές μετά το πέρας του προγράμματος θα είναι ικανοί να αναγνωρίζουν τα διάφορα μέρη της πλατφόρμας να επεξηγούν και να αναλύουν την λειτουργία τους.
- Να αναλύουν την λειτουργία μιας κατασκευής και να συνθέτουν νοητικά την δικιά τους κατασκευή συνδυάζοντας, ερμηνεύοντας και επαναδιατυπώνοντας τη θεωρητική γνώση που απέκτησαν
- Να κατασκευάσουν και να συναρμολογούν μια δική τους εφαρμογή δοκιμάζοντας και πειραματίζοντας διάφορα μέρη, τρόπους λύσης και προγραμματισμού της διάταξης

ΣΚΟΠΟΙ

- Αρχικός στόχος του προγράμματος είναι να εξαφθεί το ενδιαφέρον των μαθητών και να αφυπνιστεί η δημιουργικότητα τους μέσω της ενασχόλησης με την κατασκευή ενός αυτοκινούμενου οχήματος με δυνατότητα αποφυγής εμποδίων με τη χρήση Arduino όπου τα αποτελέσματα και η εφαρμογή της γνώσης που έλαβαν είναι άμεσα παρατηρούμενα και χρησιμοποιούμενα
- Εξοικείωση με τις νέες τεχνολογίες και ενίσχυση της αυτοπεποίθησης των μαθητών στη χρήση αυτών
- Μέσω της διαχώρισης των μαθητών σε ομάδες με ανομοιογενές θεωρητικό και γνωστικό υπόβαθρο ενθαρρύνεται η ανάπτυξη πνεύματος συνεργασίας και αλληλοβοήθειας

Πίνακας 1 Ανάπτυξη Ωρολογίου Προγράμματος - Χρονοδιάγραμμα

	Τίτλος Μαθήματος	Ώρες διδασκαλίας	Μέθοδοι διδασκαλίας	Εκπαιδευτικά μέσα
1	Ο ΗΛΕΚΤΡΙΣΜΟΣ	1	Διάλεξη Επίδειξη	20 H/Y με εγκατεστημένο τον editor 4 κιτ Arduino που περιέχουν το board UNO R3 και όλα τα απαραίτητα παρελκόμενα για την κατασκευή του αυτοκινούμενου οχήματος Πίνακας Προτζέκτορας Φύλλο ερωτήσεων επί της ύλης και φύλλο ασκήσεων κατανόησης της ύλης
2	ΗΛΕΚΤΡΙΚΑ ΚΥΚΛΩΜΑΤΑ	1	Ερωτήσεις στα πλαίσια του μαθήματος	
3	ΕΙΣΑΓΩΓΗ ΣΤΟ ARDUINO	1	Πρακτική Εφαρμογή με καθοδήγηση	
4	ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΣ ARDUINO - ΠΡΩΤΗ ΕΠΑΦΗ ΜΕ ΤΟ PROJECT	1	Πρακτική Εφαρμογή χωρίς καθοδήγηση	
5	ΕΛΕΓΧΟΣ ΚΙΝΗΤΗΡΩΝ ΜΕ ΤΟΝ ARDUINO	1	(εργασία για το project στο σπίτι με κατευθύνσεις του καθηγητή)	
6	Η ΜΟΝΑΔΑ ΜΕΤΡΗΣΗΣ ΑΠΟΣΤΑΣΕΩΝ	1	Αξιολόγηση τελικής εφαρμογής	
7	ΟΛΟΚΛΗΡΩΣΗ ΤΟΥ ΟΧΗΜΑΤΟΣ - ΠΑΡΟΥΣΙΑΣΕΙΣ	1	Ανάλυση προβλημάτων και πιθανή αναζήτηση νέας γνώσης με επανασχεδιασμό και τελική εφαρμογή της λύσης	
	Σύνολο	7	Αλληλεπίδραση μαθητή με μαθητή και μαθητών με καθηγητή στη διαδικασία παρουσίασης	

ΚΕΦΑΛΑΙΟ 1

ΜΑΘΗΜΑ 1^ο: Ο Ηλεκτρισμός

1.1 Το Ηλεκτρικό Φορτίο Q

Το ηλεκτρικό φορτίο είναι μία ιδιότητα της ύλης που την καθιστά ικανή να αλληλοεπιδράσει με ηλεκτρομαγνητικά πεδία μεταξύ των φορέων του φορτίου με τις ηλεκτρικές δυνάμεις έλξης ή άπωσης. Μονάδα μέτρησης το Coulomb και συμβολίζεται με το γράμμα C. Το 1 Coulomb αντιστοιχεί σε ροή $6,24 \times 10^{18}$ φορτίων (θεωρούμε ηλεκτρονίων) που διέρχονται από μια διατομή ενός αγωγού υπό ρεύμα 1 Ampere. Κβαντισμένο μέγεθος, εμφανίζεται μόνο ως ακέραιο πολλαπλάσιο μίας στοιχειώδους ποσότητας $e = 1,602 \times 10^{-19}$ C που αποκαλείται στοιχειώδες φορτίο.

«Το 1 C είναι πολύ μεγάλη μονάδα φορτίου. Αν μπορούσαμε να φορτίσουμε δύο μικρές σφαίρες με 1 C την καθεμιά και τις τοποθετούσαμε έτσι ώστε τα κέντρα τους να απέχουν ένα μέτρο, τότε η ηλεκτρική δύναμη που θα ασκούσε η μια στην άλλη θα ήταν 109 N (σχεδόν ένα εκατομμύριο φορές μεγαλύτερη από το βάρος ενός ενήλικα)!!»[1]

Το φορτίο διαχωρίζεται σε αρνητικό και θετικό φορτίο (κατ' όνομα). Δεν υπάρχει κάποια θετική ή αρνητική ιδιότητα στο φορτίο είναι δυο διαφορετικές φύσεις του ίδιου μεγέθους. Ετερόσημα φορτία έλκονται ενώ ομόσημα απωθούνται. Από αυτή τη διπλή φύση του ηλεκτρικού φορτίου μπορέσαμε να αντιληφθούμε και τις ηλεκτρικές

δυνάμεις. Γύρω

από ένα ηλεκτρικό

φορτίο

δημιουργείται

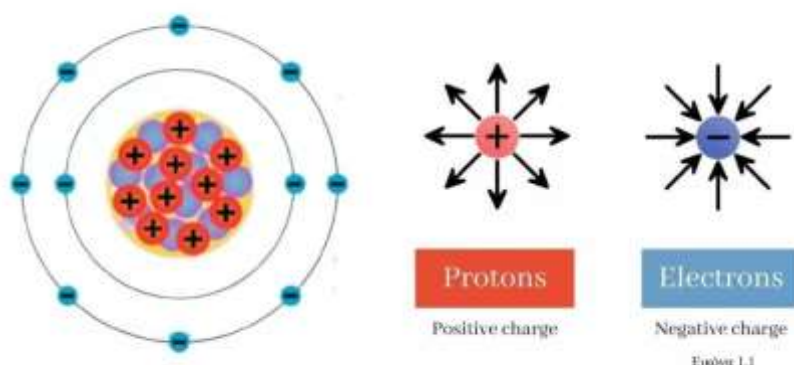
ηλεκτρικό πεδίο

που έλκει ή απωθεί

άλλα φορτία

ανάλογα το

«πρόσημο»



Εικόνα 1.1

τους[1]. Η δύναμη αυτή ονομάζεται Coulomb και θα την γνωρίσουμε αργότερα.

Κύριος φορέας του θετικού φορτίου είναι το πρωτόνιο και του αρνητικού το ηλεκτρόνιο. Σε απόλυτη τιμή τα φορτία αυτά είναι ίσα και φυσικά είναι φορείς του

στοιχειώδους φορτίου, το μόνο στο οποίο διαφέρουν είναι το πρόσημο τους το οποίο όπως είπαμε και παραπάνω επηρεάζει την αλληλεπίδραση με τα φορτία. Ένα σώμα λέγεται αρνητικά φορτισμένο όταν έχει πλεόνασμα αρνητικών φορτίων, θετικά όταν έχει πλεόνασμα θετικών και ουδέτερο όταν υπάρχει ισοζύγιο. Αν το δείτε από την προοπτική του ατόμου, το άτομο, που δεν έχει αλληλοεπιδράσει με κάποιο άλλο άτομο ή ακτινοβολία, περιέχει ίσο αριθμό αρνητικά φορτισμένων ηλεκτρονίων και θετικά φορτισμένων πρωτονίων. Ένα άτομο είναι λοιπόν ηλεκτρικά ουδέτερο σε κατάσταση ηρεμίας. Μακροσκοπικά ισχύει λοιπόν ένα σώμα φορτίζεται αρνητικά αν έχει πλεόνασμα ελεύθερων ηλεκτρονίων και θετικά αν έχει έλλειμμα[2]. Παρ' όλες τις μετακινήσεις του φορτίου από το ένα στο άλλο σώμα γίνεται αντιληπτό ότι το ηλεκτρικό φορτίο διατηρείται είτε σε ένα κλειστό σύστημα είτε σε ένα ανοιχτό σύστημα. Τα φορτία δεν εξαφανίζονται ούτε καταστρέφονται. Τα ηλεκτρόνια είναι κατά πολύ ελαφρύτερα των πρωτονίων και κινούνται σε συγκεκριμένες τροχιές γύρω από το πυρήνα που βρίσκονται σε σύνδεση με την ασθενή και την ισχυρή πυρηνική δύναμη τα θετικά φορτισμένα πρωτόνια και τα ουδέτερα νετρόνια[3].

1.2 Ο Νόμος του Coulomb

Ανάμεσα σε δύο φορτία στον χώρο ασκείται δύναμη αντιστρόφως ανάλογη του τετραγώνου της μεταξύ τους απόστασης και ανάλογη του γινομένου των φορτίων όπως φαίνεται στον τύπο της Εικόνας 1.2. Τα δύο φορτία μπορεί να είναι θετικά ή αρνητικά. Μπορεί να είναι σημειακά φορτία ή κατανομή φορτίων. Η δύναμη είναι ελκτική ή απωστική ανάλογα το πρόσημο των φορτίων. Η δύναμη είναι ελκτική αν τα φορτία είναι ετερόσημα και απωστική αν είναι ομόσημα. Είναι διανυσματικό


Coulomb's Law

$$F \propto \frac{q_1 q_2}{r^2}$$

$$F = k \frac{q_1 q_2}{r^2}$$

$$F = \frac{1}{4\pi\epsilon_0} \frac{q_1 q_2}{r^2} \text{ newton}$$

$k = 1/4\pi\epsilon_0$



Εικόνα 1.2

μέγεθος δηλαδή έχει μέγεθος (απόλυτη τιμή, κατεύθυνση και φορά). Μονάδες μέτρησης το Newton [1]. Στο σχήμα της Εικόνας 1.2 τα φορτία q_1 q_2 ασκούν το ένα στο άλλο δύναμη ελκτική ή απωστική ανάλογα με το αν είναι θετικά ή

αρνητικά φορτισμένα. Η κατεύθυνση της δύναμης θα έχει φορά και κατεύθυνση που

συμπίπτει με τον άξονα που συνδέει τα δύο φορτία. Η σταθερά k ονομάζεται σταθερά του Coulomb και είναι $k=9 \cdot 10^9$ με μονάδες $(N \cdot m^2)/C^2$. Η σταθερά ϵ_0 ονομάζεται ηλεκτρική διαπερατότητα του κενού με τιμή $8,85 \cdot 10^{-12}$ με μονάδες $C^2/N \cdot m^2$ [1].

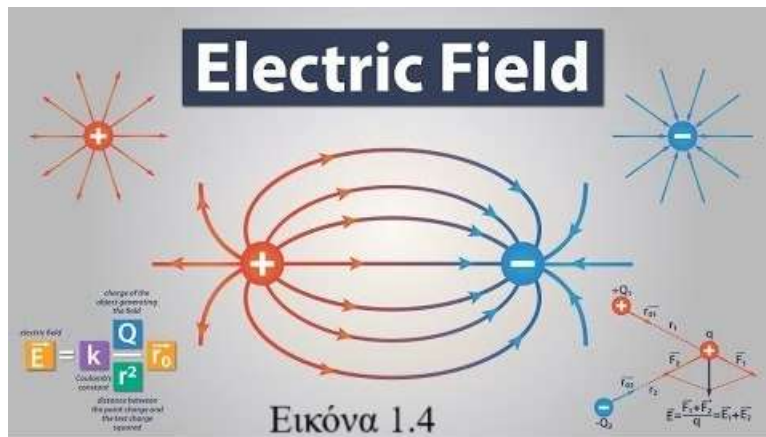
$$k_e = \frac{1}{4\pi\epsilon_0}$$

where

- k_e is the Coulomb's constant $= 8.987 \times 10^9 \text{ N} \cdot \text{m}^2 \cdot \text{C}^{-2}$
- ϵ_0 is the vacuum permittivity $= 8.854 \times 10^{-12} \text{ F} \cdot \text{m}^{-1}$

Εικόνα 1.3

1.3 Το Ηλεκτρικό Πεδίο E



Εικόνα 1.4

Ένα σημειακό φορτίο όπως της Εικόνας 1.4 (θετικό ή αρνητικό) ή μία κατανομή φορτίου στον χώρο δημιουργεί γύρω του ένα πεδίο με τις ανάλογες δυναμικές γραμμές. Η σύμβαση

είναι ότι οι δυναμικές γραμμές του ηλεκτρικού πεδίου εξέρχονται όταν πρόκειται για θετικό φορτίο και εισέρχονται όταν πρόκειται για αρνητικό. «Γενικά μια περιοχή του χώρου ονομάζεται ηλεκτρικό πεδίο, αν ασκούνται ηλεκτρικές δυνάμεις σε κάθε φορτισμένο σώμα που φέρνουμε μέσα σ' αυτή» [1]. Από τον τύπο της Εικόνας 1.4 παρατηρούμε ότι από τον τύπο του Ηλεκτρικού Πεδίου υπάρχει μόνο το ένα από τα δύο φορτία που είχαμε στον τύπο της δύναμης Coulomb. Το φορτίο Q είναι το φορτίο που προκαλεί το πεδίο γύρω του ενώ ένα φορτίο q το οποίο εισέρχεται στο πεδίο αυτό θα το ονομάσουμε δοκιμαστικό φορτίο. Παρατηρούμε ότι το ηλεκτρικό πεδίο εξαρτάται μόνο από το φορτίο Q [1]. Η ένταση ηλεκτρικού πεδίου ή απλά ηλεκτρικό πεδίο $E=F/Q$ δεν εξαρτάται από το οποιοδήποτε φορτίο εισέλθει στο πεδίο. Το ηλεκτρικό πεδίο είναι διανυσματικό μέγεθος με μέτρο, κατεύθυνση και φορά με μονάδες N/C . Από τη Εικόνα 1.4 βλέπουμε ότι το ηλεκτρικό πεδίο που προκαλείται από δύο φορτία $+Q_1$ και $-Q_2$ σε δοκιμαστικό φορτίο q είναι το διανυσματικό άθροισμα των επί μέρους

ηλεκτρικών πεδίων που προκαλούν ξεχωριστά τα Q_1 και Q_2 στο φορτίο Q . Από τον τύπο βλέπουμε ότι και το ηλεκτρικό πεδίο E είναι αντιστρόφως ανάλογο της απόστασης του φορτίου Q από το δοκιμαστικό φορτίο q .

1.4 Το Δυναμικό V

$$\begin{array}{ll}
 \textit{Electric Force} & F = \frac{kq_1 q}{r^2} = \textit{Newtons} \\
 \textit{Electric Field} & E = \frac{F}{q} = \frac{kq_1}{r^2} = \textit{Newtons/Coulomb} \\
 \textit{Potential Energy} & U = Fr = \frac{kq_1 q}{r} = \textit{Joules} \\
 \textit{Potential} & V = \frac{Fr}{q} = \frac{kq_1}{r} = \textit{Volts}
 \end{array}$$

Εικόνα 1.5

Στην Εικόνα 1.5 βλέπουμε συγκεντρωμένους τους τύπους των μεγεθών που θα χρησιμοποιήσουμε σε αυτή την ενότητα του μαθήματος. Παρατηρούμε το μέγεθος U την δυναμική ενέργεια. Έστω ότι q_1 είναι το σταθερό σημειακό φορτίο που δημιουργεί ηλεκτρικό πεδίο E και q το δοκιμαστικό φορτίο που εισέρχεται στο πεδίο E . Αν και τα δύο φορτία είναι θετικά τότε για να φέρουμε το φορτίο q από μία θέση στην οποία το q δεν επηρεάζεται από το πεδίο E προς μία εγγύτερη προς το q_1 και εντός της επίδρασης του πεδίου. Η κίνηση από ένα σημείο προς ένα άλλο σημείο εντός πεδίου απαιτεί δαπάνη έργου (θυμηθείτε από τη Φυσική ότι Έργο = Δύναμη*Απόσταση) [4]. Από τον τύπο της ηλεκτρικής δυναμικής ενέργειας στην εικόνα 1.5 παρατηρούμε μια ομοιότητα μεταξύ έργου και ηλεκτρικής δυναμικής ενέργειας κατ' αντιστοιχία με το μηχανικό έργο και την μηχανική δυναμική ενέργεια. Θα λέμε ότι ένα φορτίο έχει ηλεκτρική δυναμική ενέργεια όταν έχει τη δυνατότητα να παραγάγει έργο λόγω της θέσης του εντός του πεδίου E . Ένα φορτίο q λοιπόν εντός του πεδίου E σε θέση έστω A έχει δυναμική ενέργεια $U_A=(kq_1q)/r_A$ όπου r_A η απόσταση του q_1 από το q . Αν θέλουμε να πάμε το φορτίο q από το σημείο A στο σημείο B τότε η διαφορά δυναμικής ενέργειας από το A στο B είναι απλά $U_B - U_A$ [4]. Η ηλεκτρική ενέργεια είναι μέγεθος βαθμωτό. Έχει δηλαδή μόνο μέτρο.

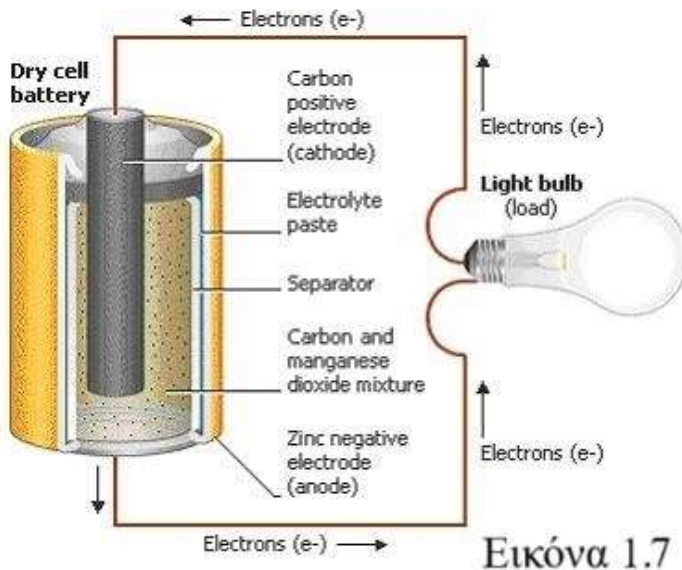
Το Δυναμικό V όπως βλέπουμε από τον τύπο της Εικόνας 1.5 έχει μια αντιστοιχία με την δυναμική ενέργεια U που έχει το πεδίο E με την δύναμη Coulomb F . Είναι

δηλαδή ανεξάρτητο του δοκιμαστικού φορτίου q ή μπορούμε ακόμα να θεωρήσουμε ότι το φορτίο q έχει την τιμή $+1$. Ένα μοναδιαίο φορτίο $q = 1 \text{ C}$ δηλαδή μέσα σε ένα πεδίο E με δυναμικό V έχει δυναμική ενέργεια $U=V \cdot q=V$ [4]. Το δυναμικό είναι και αυτό βαθμωτό μέγεθος. Στα επόμενα μαθήματα και στα ηλεκτρονικά κυκλώματα περισσότερο ενδιαφέρον έχει η διαφορά δυναμικού μεταξύ δύο σημείων $V_{AB}=V_B - V_A$ η αλλιώς τάση. Η διαφορά δυναμικού μεταξύ δύο σημείων A και B ($V_{AB}= V_B-V_A$) εντός ηλεκτρικού πεδίου για δοκιμαστικό φορτίο q ισούται με το λόγο του έργου της δύναμης που ασκεί στο δοκιμαστικό φορτίο το ηλεκτρικό πεδίο προς το φορτίο q . Αν κάνουμε την αντιστοίχιση με μία πηγή τάσης (μπαταρία) «Ονομάζουμε ηλεκτρική τάση ή διαφορά δυναμικού (V πηγής) μεταξύ των δύο πόλων μιας ηλεκτρικής πηγής το πηλίκο της ενέργειας που προσφέρεται από την πηγή σε ηλεκτρόνια (Ηλεκτρική) συνολικού φορτίου (q) όταν διέρχονται από αυτήν προς το φορτίο q » [1] όπου $E_{\text{ηλεκτρική}}$ ο συγγραφέας εννοεί την ενέργεια U . Μονάδα μέτρησης της τάσης είναι το Volt.



Στην Εικόνα 1.6 βλέπουμε το υδραυλικό ανάλογο μιας πηγής τάσης. Η πηγή τάσης παρέχει μία «πίεση» [1] η οποία θέτει τα ηλεκτρόνια σε κίνηση μέσα σε ένα κύκλωμα. Στην εικόνα η ενέργεια που δίνει ο άνθρωπος στην χειροκίνητη αντλία μετατρέπεται σε

κινητική των μορίων του νερού. Τα μόρια του νερού μετακινούνται μέσω της αντλίας από θέση χαμηλού δυναμικού (κατ' επέκταση χαμηλής δυναμικής ενέργειας) σε θέση υψηλού δυναμικού (υψηλής δυναμικής ενέργειας). Η χημική ενέργεια της μπαταρίας προκαλεί την κίνηση των ηλεκτρονίων από χαμηλό σε υψηλό δυναμικό με αποτέλεσμα την κίνηση των ηλεκτρονίων μέσα στο κύκλωμα (ηλεκτρικό ρεύμα).

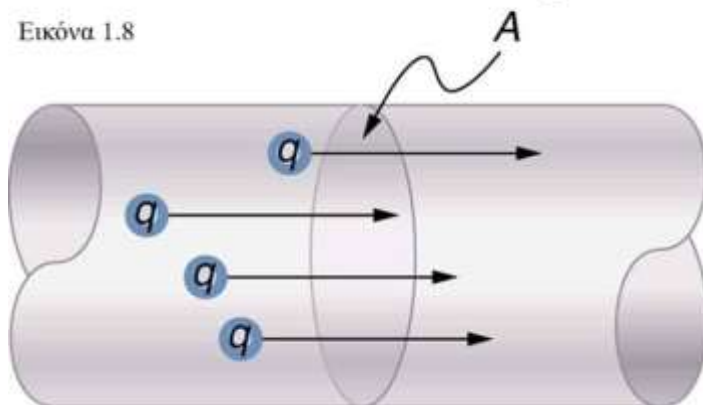


Ας δούμε ένα παράδειγμα πηγής τάσης. Μια από τις πιο κοινές πηγές τάσης είναι η μπαταρία. Είναι μία πηγή τάσης που χρησιμοποιείται σε μεγάλο αριθμό καθημερινών συσκευών. Ποια είναι η λειτουργία της μπαταρίας; Τι

εννοούμε όταν λέμε «μπαταρία 1,5 Volt»; Η ηλεκτροχημική μπαταρία αποτελείται από δύο μεταλλικά ηλεκτρόδια η άνοδος αρνητικά φορτισμένη και η κάθοδος θετικά ηλεκτρισμένη και τον ηλεκτρολύτη (οξύ ή αλκαλικός). Η χημική αντίδραση του ηλεκτρολύτη με τα ηλεκτρόδια απελευθερώνει ηλεκτρόνια σε μεγαλύτερη ποσότητα στο ένα μέταλλο (με αποτέλεσμα φορτίζεται θετικά) απ' ότι στο άλλο. Όταν η μπαταρία συνδέεται με αγωγό μέσα σε κάποιο στοιχείο ξεκινά και η ροή των ηλεκτρονίων. Όταν λέμε ότι μία μπαταρία είναι 1,5 Volt αυτό σημαίνει ότι η μπαταρία παρέχει 1,5 Joules ενέργειας σε φορτίο (ηλεκτρόνια) 1 Coulomb[6]

Current = flow of charge

Εικόνα 1.8



1.5 Το Ηλεκτρικό Ρεύμα I

Ορίζεται ως η ροή ομόσημων φορτίων μέσα σε ένα αγωγό υλικό και συμβολίζεται με το αγγλικό I. Στα μαθήματα μας θα θεωρούμε ρεύμα τη ροή ηλεκτρονίων μέσα σε ένα

αγωγό υλικό χωρίς αυτό να σημαίνει ότι μόνο μια ροή ηλεκτρονίων θεωρείται ρεύμα. Ρεύμα γενικότερα είναι η προσανατολισμένη ροή ομόσημων φορτίων μέσα σε ένα μέσο (είτε αγωγός είτε μονωτής είτε ημιαγωγός) [1]. Το ηλεκτρικό ρεύμα μέσα σε έναν αγωγό φυσικά δεν είναι μία αυθόρμητη διαδικασία αλλά αποτέλεσμα εφαρμογής μιας τάσης στον αγωγό. Σκεφτείτε τη σύνδεση μίας μπαταρίας με ένα

λαμπτήρα με σύρμα. Η σύνδεση των συρμάτων στα άκρα της μπαταρίας άμεσα προκαλεί εντός του αγωγού ένα πεδίο και κατά συνέπεια μία δύναμη προς τα ελεύθερα ηλεκτρόνια. Γνωρίζετε από την καθημερινότητα σας ότι μία μπαταρία έχει ένα θετικό και ένα αρνητικό πόλο. Ο θετικός πόλος της μπαταρίας ονομάζεται και άνοδος ενώ ο αρνητικός ονομάζεται κάθοδος. Τα ηλεκτρόνια στην πραγματικότητα κινούνται προς τον θετικά φορτισμένο πόλο (εξ' ου και το όνομα άνοδος) και απομακρύνονται από τον αρνητικά φορτισμένο πόλο (κάθοδος). Για ιστορικούς λόγους έχει επικρατήσει η ροή των ηλεκτρονίων να σημειώνεται σε ένα κύκλωμα ως ροή από τον θετικό στον αρνητικό πόλο[1]. Η πηγή τάσης λοιπόν παρέχει την απαραίτητη διαφορά δυναμικού για την κίνηση των ηλεκτρονίων και την παραγωγή του επιθυμητού αποτελέσματος. Τα ηλεκτρόνια που συμμετέχουν στην ροή είναι τα ηλεκτρόνια των εξωτερικών στοιβάδων του αγωγού (ηλεκτρόνια σθένους).

Το ηλεκτρικό ρεύμα είναι το ποσό φορτίου που ρέει σε έναν αγωγό στη μονάδα του χρόνου. *«Ορίζουμε την ένταση (I) του ηλεκτρικού ρεύματος που διαρρέει έναν αγωγό ως το φορτίο (q) που διέρχεται από μια διατομή του αγωγού σε χρονικό διάστημα (t) προς το χρονικό διάστημα.»* [1]. Στην Εικόνα 1.8 οπότε το ρεύμα είναι ο λόγος του συνολικού φορτίου q που διέρχεται από τη διατομή A για ένα συγκεκριμένο χρόνο t προς τον χρόνο αυτό $I=q/t$. Η μονάδα του ηλεκτρικού ρεύματος είναι το Ampere - A (Coulomb / seconds). Ρεύμα 1 Ampere ισοδυναμεί με ροή ενός Coulomb ανά second ή $6,2 \times 10^{18}$ ηλεκτρόνια/δευτερόλεπτο. Το υδραυλικό ανάλογο του ηλεκτρικού ρεύματος είναι η ροή των μορίων του νερού μέσα σε ένα σωλήνα. Το ρεύμα των μορίων του νερού μέσα από μία διατομή A είναι ο λόγος του αριθμού των μορίων του νερού που περνάνε μέσα από την διατομή A για χρόνο t προς τον χρόνο αυτό.

1.6 Η Ηλεκτρική Αντίσταση R

Η ηλεκτρική αντίσταση είναι η αντίσταση στην ροή του ηλεκτρικού ρεύματος και συμβολίζεται με το γράμμα R και οι μονάδες της είναι το Ohm (Ω). Τα ηλεκτρόνια που ρέουν μέσα στον αγωγό συναντούν αντίσταση καθώς συγκρούονται με τα άτομα του κρυσταλλικού πλέγματος του αγωγού *«η αντίσταση είναι ένα μέτρο της δυσκολίας που προβάλλει ένας αγωγός στη διέλευση του ηλεκτρικού ρεύματος μέσα από αυτόν»*[1] *«ονομάσαμε ηλεκτρικό ρεύμα που διαρρέει ένα μεταλλικό σύρμα την προσανατολισμένη κίνηση των ελεύθερων ηλεκτρονίων του. Όταν στα άκρα του σύρματος εφαρμόσουμε μια διαφορά*

δυναμικού, στο εσωτερικό του δημιουργείται ένα ηλεκτρικό πεδίο. Το πεδίο αυτό ασκεί ηλεκτρικές δυνάμεις στα ελεύθερα ηλεκτρόνια του μετάλλου, οι οποίες προκαλούν αύξηση της ταχύτητάς τους κατά μήκος του σύρματος. Όμως κατά την κίνησή τους τα ελεύθερα ηλεκτρόνια συγκρούονται με τα ιόντα του μετάλλου τα οποία ταλαντώνονται γύρω από ορισμένες σταθερές θέσεις. Σε κάθε τέτοια σύγκρουση ένα μέρος της κινητικής ενέργειας του ηλεκτρονίου μεταφέρεται στο ιόν. Έτσι η ταχύτητα του ηλεκτρονίου μειώνεται. Αμέσως μετά τη σύγκρουση, η δύναμη του ηλεκτρικού πεδίου προκαλεί εκ νέου αύξηση της ταχύτητας του ηλεκτρονίου μέχρι την επόμενη σύγκρουση, οπότε επαναλαμβάνεται η ίδια διαδικασία»[1]. Οι φυσικές ιδιότητες ενός μέσου επηρεάζουν και την αντίσταση ή την αγωγιμότητα του όπως το ατομικό πλέγμα του, η θερμοκρασία, οι διαστάσεις κτλ. Ένα στοιχείο μεγαλύτερου μήκους παρουσιάζει και μεγαλύτερη αντίσταση από ένα στοιχείο από το ίδιο υλικό μικρότερου μήκους. Ένα στοιχείο μεγαλύτερης διατομής παρουσιάζει μικρότερη αντίσταση από ένα στοιχείο ίδιου υλικού αλλά μικρότερης διατομής. Το ίδιο στοιχείο παρουσιάζει αυξημένη αντίσταση όσο αυξάνεται και η θερμοκρασία του κοκ.

Υλικά από τα οποία το ρεύμα διέρχεται ευκολά ονομάζονται αγωγοί (κυρίως μέταλλα χαλκός, αλουμίνιο, χρυσός κτλ.). Υλικά από τα οποία το ρεύμα δεν διέρχεται ονομάζονται ή διέρχεται πολύ δύσκολα ονομάζονται μονωτές (γυαλί, ξύλο, πλαστικό κτλ.)[3]. Το υδραυλικό ανάλογο της αντίστασης είναι η ροή του νερού μέσα σε σωλήνα με μειούμενη διατομή σε κάποιο σημείο με αποτέλεσμα την δυσχερέστερη ροή.



Εικόνα 1.9

Ο πιο κοινός τύπος αντιστάτη που θα χρησιμοποιηθεί στα κυκλώματα των επόμενων μαθημάτων είναι αυτός της Εικόνας 1.9. Αποτελείται από δύο ακροδέκτες που συνδέουν το στοιχείο στο κύκλωμα και

το κύριο σώμα του αντιστάτη. Τα υλικά από τα οποία κατασκευάζονται οι αντιστάτες είναι ανάλογα της επιθυμητής αντίστασης που χρειάζεται να έχει. Ο αντιστάτης είναι

από τα λεγόμενα «παθητικά στοιχεία» δηλαδή ένα στοιχείο που καταναλώνει

ενέργεια. Οι

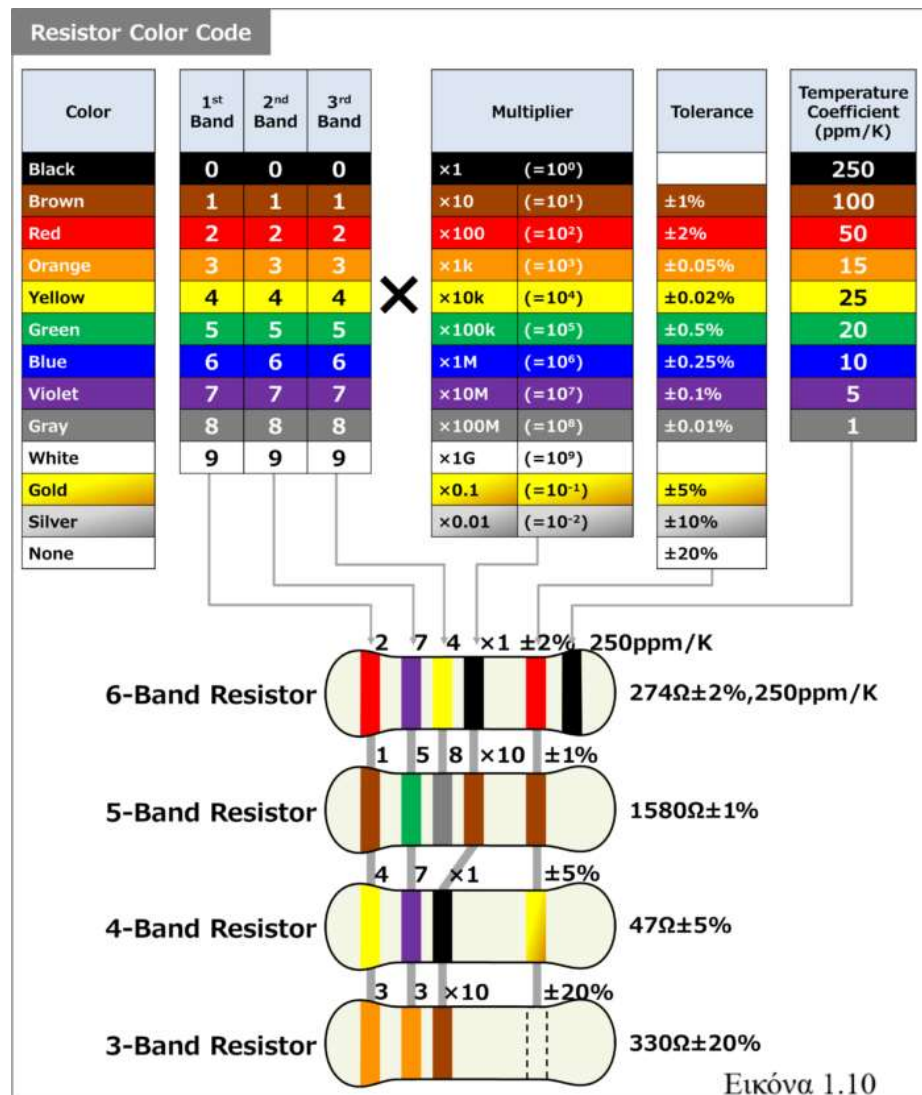
χρωματιστές

λωρίδες

δείχνουν

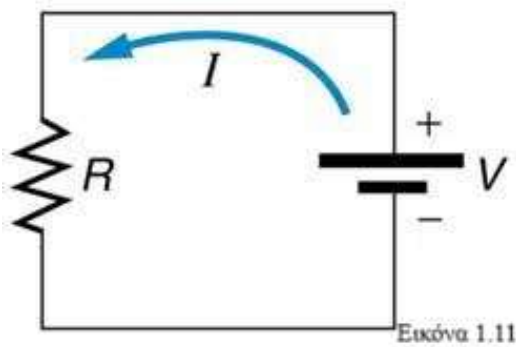
την τιμή της

αντίστασης.



Εικόνα 1.10

Στην Εικόνα 1.10 έχουμε ένα πίνακα που επεξηγεί την χρωματική κωδικοποίηση των αντιστατών. Είναι ένα διεθνώς αναγνωρισμένο πρότυπο και ακολουθείται από όλους τους κατασκευαστές. Οι χρωματικές λωρίδες μπορεί να είναι 3, 4, 5 ή 6. Ανάλογα το χρώμα τον αριθμό τους και τη θέση τους δηλώνουν μία τιμή ή παράμετρο της αντίστασης όπως η απόκλιση της και η εξάρτηση της από τη θερμοκρασία. Τα band 1, 2, 3 δείχνουν το πόσα ψηφία θα έχει η αριθμητική τιμή της αντίστασης. Οι αντιστάτες 6 και 5 χρωμάτων έχουν ακρίβεια τριών ψηφίων στην τιμή τους ενώ αυτοί των 3 και 4 χρωμάτων δύο. Το επόμενο χρώμα δηλώνει τον δεκαδικό πολλαπλασιαστή. Το πολλαπλάσιο του 10 δηλαδή που θα πολλαπλασιαστεί τα 3 ή 2 προηγούμενα ψηφία. Tolerance είναι το σφάλμα στην τιμή που παρέχει ο κατασκευαστής και temperature coefficient η μεταβολή της τιμής με τη μεταβολή της θερμοκρασίας [5].



1.7 Ο Νόμος του Ohm

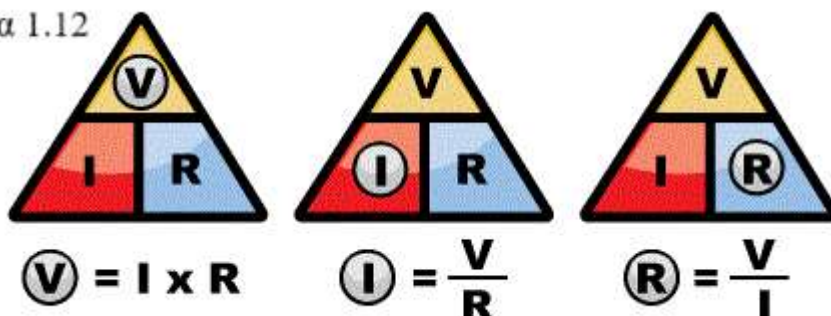
Σε ένα κλειστό κύκλωμα Εικόνα 1.11 έχουμε μια πηγή τάσης V , μία αντίσταση R και το ρεύμα που διαρρέει το κύκλωμα I .

Το ρεύμα που I διαρρέει την αντίσταση είναι ο λόγος της τάσης στα άκρα της αντίστασης προς την τιμή της αντίστασης

$$I = V/R$$

«Η ένταση (I) του ηλεκτρικού ρεύματος που διαρρέει ένα μεταλλικό αγωγό είναι ανάλογη της διαφοράς δυναμικού (V) που εφαρμόζεται στα άκρα του.»[1]. Ο νόμος του Ohm χρησιμοποιείται ευρέως στην ανάλυση κυκλωμάτων αλλά δεν είναι κατάλληλος για όλες τις περιπτώσεις. Από τον τύπο του νόμου καταλαβαίνουμε ότι η σχέση είναι γραμμική. Υπάρχουν όμως ηλεκτρονικά στοιχεία που η αντίσταση τους δεν είναι σταθερή αλλά μεταβάλλεται με τη μεταβολή της τάσης που εφαρμόζεται στα άκρα τους. Τέτοια στοιχεία είναι οι κρυσταλλοδιόδοι, οι ηλεκτρονικές λυχνίες και τα τρανζίστορ[1].

Εικόνα 1.12



Στην εικόνα 1.12 παρατηρείστε τις τρεις διαφορετικές μορφές του νόμου του Ohm. Οι τρεις μορφές αυτές είναι ισοδύναμες.

Υλικό για μελέτη στο σπίτι

Γραφική αναπαράσταση του νόμου του OHM

<http://photodentro.edu.gr/aggregator/lo/photodentro-lor-8521-10486>

Θεωρία του νόμου του Ohm

http://ebooks.edu.gr/ebooks/v/html/8547/2226/Fysiki_G-Gymnasiou_html-empl/index2.html

Άσκηση 1

έχουμε τα φορτία q_1 q_2 σε απόσταση r .

- i) Αν $q_1=3$ C και $q_2=5$ C και $r=50$ cm να υπολογίσετε τη δύναμη Coulomb
- ii) Να υπολογίσετε το δυναμικό στο σημείο που βρίσκεται το φορτίο q_2 αν αυτό θεωρείται το δοκιμαστικό φορτίο και το q_1 φορτίο αναφοράς

Προσοχή στις μετατροπές των μονάδων

Άσκηση 2

Στο κύκλωμα της Εικόνας 1.11 με αντίσταση R ρεύμα I και τάση V να υπολογίσετε:

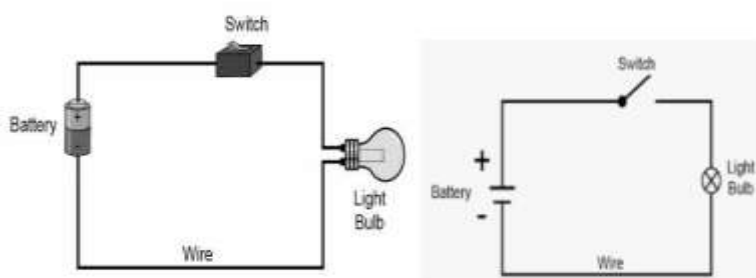
- i) Αν η τάση στα άκρα της αντίστασης είναι 8 V και το ρεύμα που διαρρέει το κύκλωμα είναι 2 A να βρείτε την αντίσταση R
- ii) Αν η αντίσταση είναι 5 Ω και το ρεύμα είναι 3 A να βρείτε την τάση στα άκρα της αντίστασης

ΚΕΦΑΛΑΙΟ 2

ΜΑΘΗΜΑ 2^Ο: Ηλεκτρονικά Κυκλώματα

2.1 Βασικά Στοιχεία Ηλεκτρονικών Κυκλωμάτων

Ένα ηλεκτρονικό κύκλωμα με απλούς όρους είναι μία συλλογή ηλεκτρικών στοιχείων που συνδέονται μεταξύ τους με αγωγή μέσα. Θυμηθείτε το παράδειγμα



της δημοτικής σας εκπαίδευσης αυτό του απλού κυκλώματος που αποτελούνταν από μια μπαταρία, ένα λαμπτήρα, καλώδια και ένα διακόπτη.

Εικόνα 2.1 Στην Εικόνα 2.1

βλέπουμε μια πηγή τάσης συνδεδεμένη με αγωγίμο υλικό (μεταλλικό σύρμα) με διακόπτη και λαμπτήρα. Ο διακόπτης μπορεί απλά να είναι μια απλά μία διακοπή στην συνέχεια του καλωδίου το οποίο ξανασυνδέεται κατά το δοκούν. Το κύκλωμα όπως φανερώνει και η λέξη που το περιγράφει είναι μία κυκλική διαδρομή ρεύματος από μία πηγή προς το διάφορα ηλεκτρονικά στοιχεία μέσω των καλωδίων ή γενικά αγωγιμών μέσω. Στο κύκλωμα μας της Εικόνας 2.1 ο λαμπτήρας ονομάζεται και φορτίο που είναι η γενική ονομασία των ηλεκτρονικών στοιχείων στα οποία συνδέεται μία πηγή.

Το διάγραμμα κυκλώματος (schematic) είναι μία πιο πρακτική μορφή γραφικής απεικόνισης αυτού. Στην Εικόνα 2.1 δίπλα από το κύκλωμα με την μπαταρία τον διακόπτη και τον λαμπτήρα βλέπετε το schematic του κυκλώματος που είναι μια μορφή απεικόνισης κυκλωμάτων που κάνει πιο εύκολη την ανάγνωση και τον καθορισμό της λειτουργίας τους. Σε ένα κύκλωμα συνδέεται μια πηγή ρεύματος ή τάσης σε ηλεκτρονικά στοιχεία (αντιστάτες, πυκνωτές, δίοδοι κ.α.) με σκοπό να εκτελεστεί κάποια λειτουργία. Στη συνέχεια του μαθήματος η κάθε ομάδα θα αναπτύξει ορισμένα απλά κυκλώματα κατά τη διάρκεια του μαθήματος.



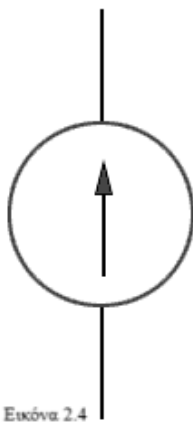
Εικόνα 2.2 Η καθημερινότητα σας είναι η κοινή μπαταρία. Οι πηγές

τάσης στη θεωρία κυκλωμάτων χωρίζονται σε ιδανικές και μη ιδανικές. Η ιδανική πηγή τάσης δεν εμφανίζει αντίσταση (εσωτερική αντίσταση) ενώ η μη ιδανική (πραγματική) πηγή τάσης εμφανίζει εσωτερική αντίσταση. Το σύμβολο της Εικόνας 2.2 απεικονίζει το σύμβολο που χρησιμοποιείται στο schematics για τις πηγές τάσης (συνεχούς). Να σημειωθεί εδώ ότι τόσο η τάση όσο και το ρεύμα διακρίνονται σε συνεχές (DC) και εναλλασσόμενο (AC) και το ίδιο και οι πηγές τάσης και ρεύματος.



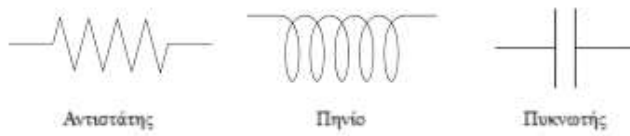
Εικόνα 2.3

Η εναλλασσόμενη τάση μεταβάλλεται με συγκεκριμένη συχνότητα (συνήθως) ως προς το πλάτος και την πόλωση ενώ το εναλλασσόμενο ρεύμα με μεταβάλλεται ως προς το πλάτος του και την κατεύθυνση διάδοσης. Το συνεχές ρεύμα ρέει σε ένα αγωγό προς συγκεκριμένη κατεύθυνση ενώ το εναλλασσόμενο αλλάζει περιοδικά κατεύθυνση και πλάτος. Η εικόνα 2.3 δείχνει το σύμβολο μίας πηγής εναλλασσόμενης τάσης. Χαρακτηριστική πηγή συνεχούς τάσης είναι η κοινή μπαταρία που χρησιμοποιείτε κάθε μέρα.



Εικόνα 2.4

Στην Εικόνα 2.4 βλέπουμε το σύμβολο μιας πηγής ρεύματος. Μια πηγή ρεύματος παρέχει μια σταθερή ποσότητα ρεύματος ανεξάρτητα του φορτίου που επιβάλλεται στην πηγή[8]. Για να το επιτύχουμε αυτό επιλέγουμε η αντίσταση του φορτίου του κυκλώματος μας να είναι πολύ μικρότερη της εσωτερικής αντίστασης της πηγής τάσης που χρησιμοποιούμε [9]. Έτσι έχουμε μια πηγή ρεύματος που παρέχει σταθερό ρεύμα. Θυμηθείτε τον



Εικόνα 2.5

ρεύμα που διαρρέει το κύκλωμα ανάλογα με το φορτίο.

Στην Εικόνα 2.5 βλέπετε τα σύμβολα των πιο απλών ηλεκτρονικών στοιχείων που θα συναντήσετε σε ένα κύκλωμα. Κατά σειρά παρουσιάζονται ο αντιστάτης που γνωρίσαμε στο πρώτο μάθημα, το πηνίο και ο πυκνωτής. Είναι και τα τρία παθητικά



Εικόνα 2.6

εκτελέσει και άλλες λειτουργίες όπως του διαιρέτη τάσης.

Στην Εικόνα 2.6 βλέπετε ένα πυκνωτή. Ο πυκνωτής αποτελείται από δύο αγώγιμα φύλλα τοποθετημένα το ένα έναντι του άλλου. Όταν στα άκρα του πυκνωτή εφαρμοστεί μία τάση μεταξύ των φύλλων δημιουργείται πεδίο με τα φύλλα να φορτίζονται με φορτίο $+Q$ και $-Q$ αντίστοιχα και ο πυκνωτής μπορεί να αποθηκεύσει ενέργεια [10]. Τα αγώγιμα φύλλα είναι μονωμένα μεταξύ τους με κάποιο υλικό (διηλεκτρικό). Όταν στον πυκνωτή δεν εφαρμόζεται πλέον τάση τότε η ενέργεια αυτή διοχετεύεται στο κύκλωμα με τη μορφή (σύντομης) κυκλοφορίας ρεύματος. Ο πυκνωτής επιτρέπει στο εναλλασσόμενο ρεύμα να διέλθει αλλά όχι στο συνεχές. Χρησιμοποιείται για αποθήκευση ενέργειας, αποκοπή ανεπιθύμητων σημάτων κ.α.

νόμο του Ohm $R=V/I$. Από τον νόμο αυτό συμπεραίνουμε ότι μια πηγή τάσης παρέχει σταθερή τάση με το να αυξομειώνει το

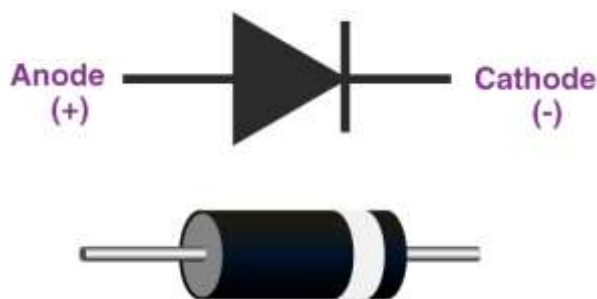
στοιχεία. Ο αντιστάτης όπως μάθαμε περιορίζει τη ροή του ρεύματος σε ένα κύκλωμα και μπορεί να



Εικόνα 2.7

Το πηνίο (Εικόνα 2.7) Είναι ένα ηλεκτρονικό στοιχείο που αποτελείται από αγωγίμο σύρμα (συνήθως χάλκινο) τυλιγμένο σε σπείρα γύρω από μεταλλικό πυρήνα (όχι πάντα). Το φαινόμενο που χαρακτηρίζει τη λειτουργία του πηνίου είναι αυτό της *επαγωγής*.

Γνωρίζουμε ότι γύρω από έναν αγωγό που διαρρέεται από ρεύμα δημιουργείται μαγνητικό πεδίο. Η επαγωγή είναι η τάση των αγωγών να αντισταθούν στην αλλαγή του ρεύματος που τους διαρρέει (ή δεν τους διαρρέει). Σύμφωνα με το νόμο του Faraday της επαγωγής όταν μέσα από κλειστό κύκλωμα διέλθει μεταβαλλόμενο μαγνητικό πεδίο τότε θα εμφανιστεί στο κύκλωμα τάση. Ο νόμος του Lenz μας λέει ότι το ρεύμα που θα δημιουργηθεί στο κύκλωμα αυτό έχει τέτοια φορά έτσι ώστε να αντισταθεί στην εναλλαγή του μαγνητικού φορτίου. Έτσι λοιπόν το πηνίο δεν επιτρέπει στο (διαρκώς) εναλλασσόμενο ρεύμα να διέλθει από αυτό ενώ αφήνει το συνεχές να περάσει. Σε κυκλώματα τα πηνία χρησιμοποιούνται συχνά για αποκοπή ρευμάτων DC.

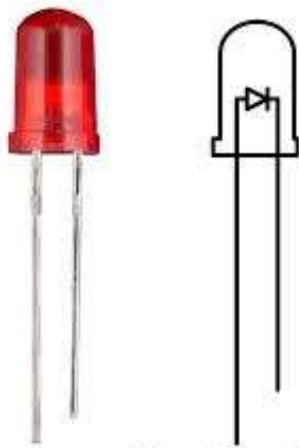


Εικόνα 2.8

Στην Εικόνα 2.8 βλέπετε το σύμβολο μίας διόδου. Η δίοδος είναι ένα ηλεκτρονικό στοιχείο που έχει την ιδιότητα να αφήνει το ρεύμα να ρέει μόνο προς μία κατεύθυνση (Το σύμβολο που ομοιάζει με βέλος

δείχνει και την κατεύθυνση). Οι δίοδοι μπορούν να χρησιμοποιηθούν για ρύθμιση τάσης, αποκοπή απότομων αυξήσεων τάσης κ.α. Μια άλλη

λειτουργία μίας διόδου είναι η μετατροπή εναλλασσόμενου ρεύματος σε συνεχές. Αφού η διάδος αφήνει μόνο τη μία κατεύθυνση να περνάει τότε κόβεται η μία συνιστώσα του εναλλασσόμενου ρεύματος ενώ επιτρέπεται στην άλλη. Υπάρχουν δύο κύρια είδη διόδων, αυτές του κενού και οι p-n που χρησιμοποιεί τις ιδιότητες των ημιαγωγών. Το υδραυλικό ανάλογο της διόδου είναι μία βαλβίδα που επιτρέπει τη ροή του νερού προς μία κατεύθυνση [10].



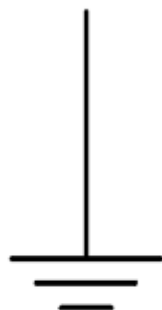
Εικόνα 2.9

Στην Εικόνα 2.9 βλέπετε μια δίοδο LED (πραγματική και σύμβολο που χρησιμοποιείται στα schematic των κυκλωμάτων). Το όνομα LED προέρχεται από τα αρχικά της φράσης Light Emitting Diode και όπως φαίνεται και από το όνομα της είναι μία διάδος ημιαγωγών p-n που εκπέμπει φως είτε λευκό είτε μονοχρωματικό. Οι ημιαγωγοί είναι υλικά με αγωγιμότητα κάτω των αγωγών αλλά μεγαλύτερη των μονωτών.

Συμπεριφέρονται ως αγωγοί ή μονωτές ανάλογα των προσμίξεων στο κρυσταλλικό πλέγμα τους και τις εξωτερικές συνθήκες όπως η θερμοκρασία. Οι σύγχρονες διόδους φτιάχνονται από ημιαγωγούς. Οι ημιαγωγοί κατά μεγάλο ποσοστό κατασκευάζονται από πυρίτιο το οποίο με κατάλληλες προσμίξεις αποκτά διαφορετικά χαρακτηριστικά. Το άτομο του πυριτίου έχει στην εξωτερική στοιβάδα του 4 ηλεκτρόνια. Λέμε ότι ένας ημιαγωγός είναι τύπου p όταν στο κρυσταλλικό πλέγμα του πυριτίου γίνεται πρόσμιξη με κάποιο στοιχείο (πχ βόρειο) το οποίο έχει 3 ηλεκτρόνια στην εξωτερική στοιβάδα του. Στους δεσμούς που θα δημιουργηθούν μεταξύ των ατόμων του πυριτίου και του βορίου θα υπάρχει έλλειμμα ενός ηλεκτρονίου. Αυτό το σημείο ονομάζεται οπή και συμπεριφέρεται ως σωματίδιο φορέας θετικού φορτίου. Λέμε ότι ένας ημιαγωγός είναι τύπου n όταν στο κρυσταλλικό πλέγμα του πυριτίου γίνεται πρόσμιξη με κάποιο στοιχείο (πχ φώσφορο) το οποίο έχει 5 ηλεκτρόνια στην εξωτερική στοιβάδα του. Στους δεσμούς που θα δημιουργηθούν μεταξύ των ατόμων του πυριτίου και του φωσφόρου θα υπάρχει πλεόνασμα ενός ηλεκτρονίου το οποίο κινείται ελεύθερα. Οι προσμίξεις αυτές είναι σε μικρά ποσοστά στο πλέγμα του πυριτίου. Αν λοιπόν φέρουμε σε επαφή έναν ημιαγωγό p με έναν ημιαγωγό n έχουμε μια επαφή p-n όπου τα ηλεκτρόνια συμπληρώνουν θέσεις οπών και οι οπές

μετακινούνται προς τον n ημιαγωγό. Αυτό ονομάζεται διάχυση και δημιουργείται μία ενδογενής τάση. Όταν τώρα η επαφή αυτή συνδεθεί σε μία πηγή τάσης η επαφή μπορεί να παίζει τον ρόλο του μονωτή ή του αγωγού ανάλογα με την πολικότητα και την τιμή της τάσης. Αλλαγές στην αγωγιμότητα του ημιαγωγού μπορεί να επιφέρει και η μεταβολή της θερμοκρασίας του.

Ένα LED έχει θετικό (μακρύς ακροδέκτης - άνοδος) και αρνητικό πόλο (κοντύτερος ακροδέκτης - κάθοδος). Η λειτουργία του LED εξαρτάται από τη σωστή σύνδεση του στο κύκλωμα (σωστή πολικότητα - το ρεύμα να ρέει από την άνοδο στην κάθοδο - σύνδεση του θετικού πόλου της πηγής στον μακρύ ακροδέκτη του LED και του αρνητικού πόλου στον κοντύτερο ακροδέκτη). Η τάση λειτουργίας των περισσότερων LED κυμαίνεται από 1,6V έως 3,5V. Αναλόγως της τάσης το LED έχει το αντίστοιχο χρώμα. (Κόκκινο - 1.6 V, Πορτοκαλί - 2 V, Κίτρινο - 2.1 V, Πράσινο - 1.9 V, Μπλε - 2.4 V, Υπεριώδες - 3 V, Λευκό - 3.5V [11]. Έχουν κατασκευαστεί μονοχρωματικά LED, LED υπέρυθρων, RGB LED που αναπαράγουν την πλειονότητα των χρωμάτων. Η ένταση του φωτός είναι ανάλογη του ρεύματος που καταναλώνει το LED αλλά εδώ θα πρέπει να γίνει μία διάκριση στο ότι τα LED έχουν ανώτερες και κατώτερες τιμές για τις οποίες μπορεί να λειτουργήσει. Υψηλό ρεύμα μπορεί να καταστρέψει το LED και γι' αυτό συνδέεται μαζί με κατάλληλο αντιστάτη που περιορίζει τη ροή του ρεύματος. Η σύνδεση αυτή θα πρέπει να είναι σε σειρά όπως θα δούμε παρακάτω. Σε γενικές γραμμές τα LEDs λειτουργούν σε ρεύμα 20mA με ανώτερη τιμή τα 30mA.

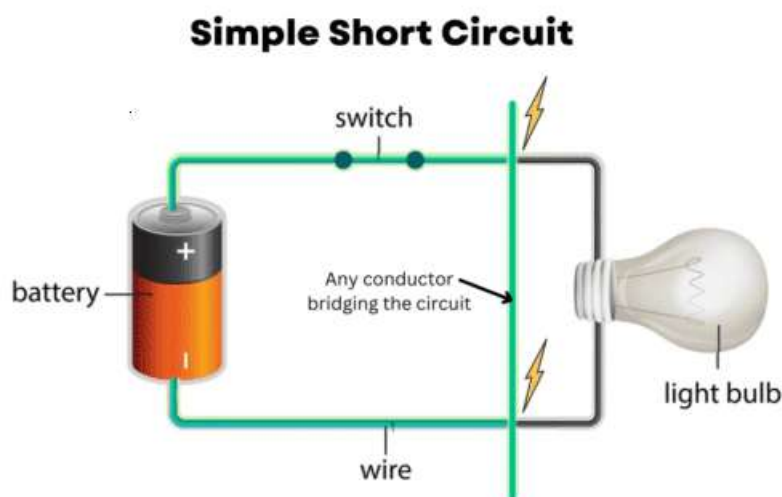


Εικόνα 2.10

Στην εικόνα 2.10 η γείωση είναι ένα σημείο αναφοράς σε ένα κύκλωμα του οποίου την τάση θεωρούμε 0 V. Σε μεγάλες ηλεκτρικές εγκαταστάσεις όπως της κατοικίας σας η γείωση είναι κυριολεκτικά ένα αγωγίμο μέταλλο ικανής διατομής συνδεδεμένο στο έδαφος. Σε ένα κύκλωμα το να πούμε ότι ένα σημείο έχει τάση 9V δεν μας δίνει πολλές πληροφορίες. Η τάση στα κυκλώματα έχει νόημα μεταξύ δύο σημείων του κυκλώματος. Η

γείωση εξυπηρετεί αυτόν τον σκοπό χωρίς να είναι πάντα το σημείο ως προς το οποίο μετρείται η τάση σε ένα στοιχείο του. Το 0 V της τάσης της γείωσης δεν είναι

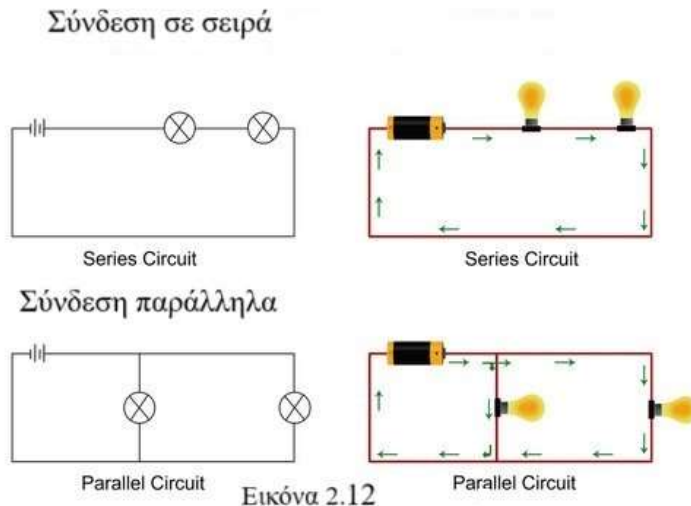
απόλυτο. Γείωση μπορεί να θεωρείται σε ένα schematic κυκλώματος ο αρνητικός πόλος μίας πηγής. Έτσι λοιπόν από το να σχεδιάζονται όλες οι γραμμές - καλώδια που συνδέονται στο - της πηγής σχεδιάζεται η σύνδεση στη γείωση. Σε άλλα κυκλώματα μπορεί να συναντήσετε σύμβολα αρνητικών και θετικών πόλων καθώς και σύμβολα γείωσης [12].



The electricity will primarily take the path of least resistance, creating a shorter circuit, and eliminating the light bulb
Εικόνα 2.11

Το βραχυκύκλωμα είναι μια σύνδεση όπου το ηλεκτρικό ρεύμα διαρρέει το κύκλωμα με πολύ μικρή αντίσταση. Ένα πρακτικό παράδειγμα θα ήταν η σύνδεση των πόλων μίας μπαταρίας με ένα απλό χάλκινο καλώδιο. Η χαμηλή

αντίσταση του σύρματος επιτρέπει υψηλή ροή ρεύματος και ελάχιστη πτώση τάσης κατά μήκος του καλωδίου. Το βραχυκύκλωμα μπορεί να προκαλέσει καταστροφή τόσο της μπαταρίας όσο και του καλωδίου για αυτό πρέπει να δίνετε προσοχή στον σχεδιασμό και στην δοκιμή ενός κυκλώματος με την χρήση κάποιας ασφάλειας για παράδειγμα [13]. Βραχυκύκλωμα μπορεί να προκύψει και από σχεδιαστικό σφάλμα όσο και από αστοχία υλικού.



Στην σύνδεση σε σειρά (Εικόνα 2.12) τα ηλεκτρικά στοιχεία συνδέονται κατά τέτοιο τρόπο έτσι ώστε ο ακροδέκτης του ενός στοιχείου να αποτελεί συνέχεια του προηγούμενου. Στην σύνδεση σε σειρά τα

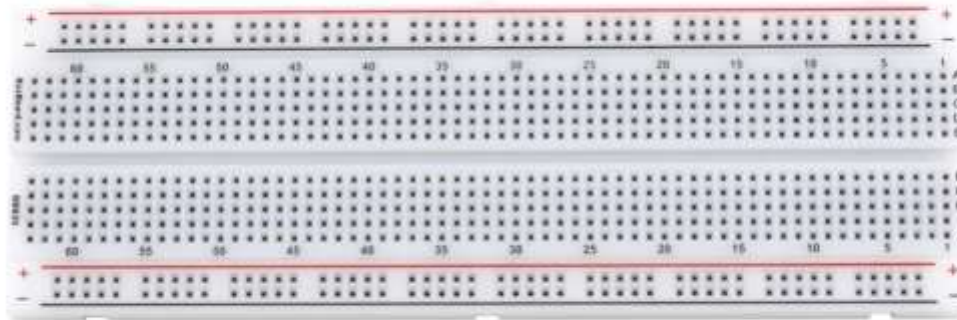
ηλεκτρικά στοιχεία διαρρέονται από το ίδιο ρεύμα αλλά έχουν διαφορετική τάση. Ας θεωρήσουμε ότι οι λαμπτήρες έχουν αντίσταση R_1 και R_2 . Η συνολική αντίσταση δύο αντιστάσεων R_1 , R_2 σε σειρά είναι $R = R_1 + R_2$. Ανατρέξτε στον νόμο του Ohm. Προφανώς $R < R_1 + R_2$ άρα με δύο λαμπτήρες το κύκλωμα θα διαρρέεται από ρεύμα μικρότερο σε τιμή από το αν στο κύκλωμα υπήρχε μόνο ένας λαμπτήρας. Κάντε στο σπίτι το εξής πείραμα. Κατασκευάστε δύο κυκλώματα: Ένα με ένα λαμπτήρα και ένα με δύο λαμπτήρες. Σε ποιο κύκλωμα λαμποκοπούν πιο έντονα οι λαμπτήρες; Τι συμπέρασμα βγάξετε για το ρεύμα που διαρρέει το ένα και το άλλο κύκλωμα; Μια χρήσιμη εφαρμογή της σύνδεσης αντιστάτων σε σειρά είναι ο Διαιρέτης Τάσης. Έστω ότι έχουμε ένα κύκλωμα με 2 αντιστάσεις R_1 και R_2 σε σειρά και πηγή τάσης V . Από τον νόμο του Ohm ξέρουμε ότι το ρεύμα θα είναι $I = V / (R_1 + R_2)$. Η τάση στα άκρα της R_2 θα είναι $V_2 = I \cdot R_2$ αφού διαρρέονται από το ίδιο ρεύμα. Άρα $V_2 = V \cdot \{R_2 / (R_1 + R_2)\}$ όπου προφανώς $V_2 < V$. Άρα αν από μία τάση V θέλω να πάρω μια μικρότερη τάση στα άκρα μίας εκ των αντιστάσεων το μόνο που έχει να κάνει είναι να επιλέξεις τους κατάλληλους αντιστάτες[14].

Στην παράλληλη σύνδεση οι ακροδέκτες των λαμπτήρων είναι κατά μία έννοια κοινοί όπως φαίνεται στην Εικόνα 2.12. Το ίδιο θα συνέβαινε αν στη θέση των λαμπτήρων ήταν κάποιο άλλο ηλεκτρονικό εξάρτημα. Στην παράλληλη σύνδεση τα στοιχεία έχουν ίση τάση στα άκρα τους αλλά τα διαρρέει διαφορετικό ρεύμα. Η συνολική αντίσταση δύο αντιστάσεων R_1 , R_2 σε παράλληλη σύνδεση είναι $1/R = 1/R_1 + 1/R_2 = (R_1 + R_2) / R_1 \cdot R_2$. Τα ρεύματα που διαρρέουν τις R_1 και R_2 αντίστοιχα είναι $I_1 = V / R_1$ και $I_2 = V / R_2$. Από τη φύση της συνδεσμολογίας στην Εικόνα 2.12 καταλαβαίνετε εύκολα ότι στην σύνδεση με σειρά αν αφαιρεθεί ένας

λαμπτήρας τότε ο άλλος σβήνει και αυτός καθότι από το κύκλωμα δεν διέρχεται ρεύμα. Στην παράλληλη σύνδεση αν αφαιρεθεί ένας λαμπτήρας τότε δεν σταματά να λειτουργεί ο άλλος καθότι δεν συνδέονται με κοινό αγωγό στο κύκλωμα.

2.2 Κατασκευάζουμε τα πρώτα μας κυκλώματα

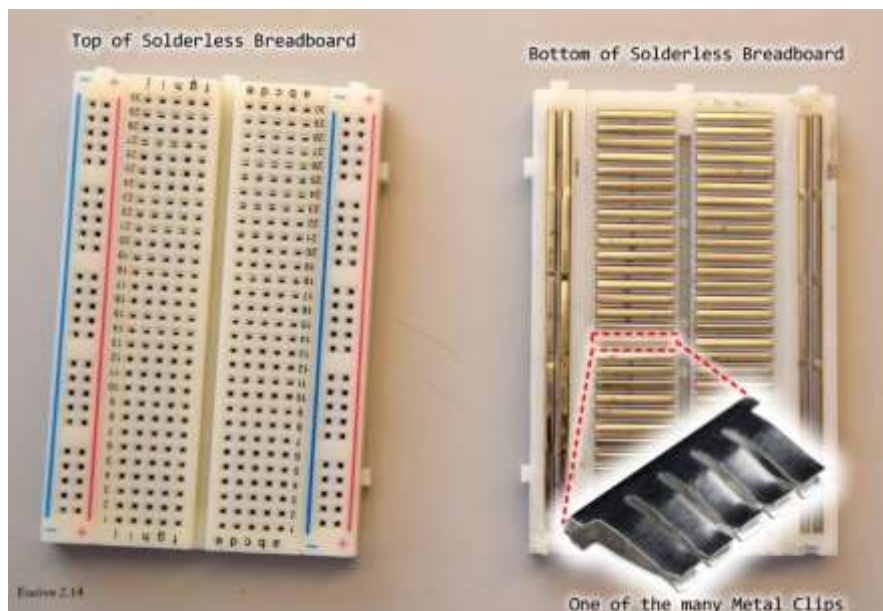
Θα ξεκινήσουμε την κατασκευή των πρώτων απλών κυκλωμάτων κάνοντας μια επίδειξη της συνδεσμολογίας κατά σειρά και παράλληλα. Στη συνέχεια θα Το Raster ή Breadboard είναι ένα πολύ χρήσιμο εργαλείο για την κατασκευή και την δοκιμή



Εικόνα 2.13

κυκλωμάτων και ηλεκτρονικών στοιχείων. Στην Εικόνα 2.13 βλέπετε ένα τυπικό Raster του εμπορίου. Το raster μας βοηθάει να κατασκευάζουμε κυκλώματα χωρίς να χρειάζεται να έχουμε εργαλείο συγκόλλησης με αποτέλεσμα σε περίπτωση λάθους να μπορούμε να το διορθώσουμε γρήγορα και χωρίς κόστος. Η εξωτερική του επιφάνεια είναι από πλαστικό και εσωτερικά βρίσκονται τα μεταλλικά αγώγιμα στοιχεία. Η διάτρητη επιφάνεια του raster δείχνει τις θέσεις στις οποίες τοποθετούνται οι ακροδέκτες των καλωδίων σύνδεσης και των διάφορων ηλεκτρονικών στοιχείων.

Στο εσωτερικό του το raster αποτελείται από μεταλλικά κλιπ 5 θέσεων (Εικόνα 2.14) στα οποία «κουμπώνουν» οι ακροδέκτες των στοιχείων του κυκλώματος μας.



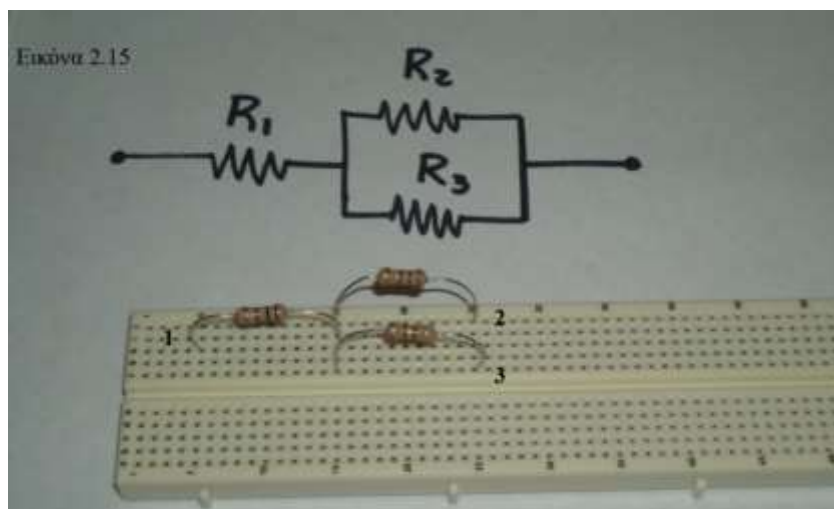
Εικόνα 2.14

Θεωρείστε τα κλιπ αυτά ως τα καλώδια ενός κυκλώματος που συνδέουν τα ηλεκτρονικά στοιχεία [15]

μεταξύ τους για αυτό λοιπόν Οι ακροδέκτες ενός ηλεκτρονικού στοιχείου που έχουν μπει στο ίδιο κλιπ καθιστούν το στοιχείο βραχυκυκλωμένο. Οι οπές δηλαδή μίας σειράς έχουν την ίδια τάση οπότε από αν για παράδειγμα συνδεθεί μία αντίσταση με τους ακροδέκτες της να είναι σε οπές της ίδιας σειράς τότε από αυτή δεν θα διέλθει ρεύμα. Στην περίπτωση ενός LED που έχει πολύ μικρή αντίσταση τότε μπορεί η δίοδος ακόμα και να καταστραφεί. Δεν συνδέουμε λοιπόν τους ακροδέκτες ενός στοιχείου στην ίδια σειρά διότι θα είναι βραχυκυκλωμένο ή θα είναι ουδέτερο στοιχείο στο κύκλωμα. Θα είναι δηλαδή στοιχείο που δεν θα συμμετέχει στη λειτουργία του κυκλώματος. Τα διάφορα ηλεκτρονικά στοιχεία συνδέονται με τους ακροδέκτες τους σε διαφορετικές σειρές οι οποίες καταλαβαίνουμε ότι βρίσκονται σε διαφορά δυναμικού διότι και οι σειρές είναι σε διαφορά δυναμικού. Υπονοείται εδώ ότι οι σειρές αυτές έχουν συνδεθεί σε μία πηγή τάσης. Το raster απλά διευκολύνει τις συνδέσεις δεν παρέχει την τάση από μόνο του.

Οι σειρές είναι συνήθως αριθμημένες από το 1-30 οριζόντια (ανάλογα το μέγεθος του raster) και από a έως j κάθετα. Οι σειρές όπως βλέπουμε χωρίζονται στη μέση δεν συνδέονται αγωγή. Μερικά raster χωρίζονται και στην μέση των στηλών + και - με αυτό να επισημαίνεται από ένα κενό στην μέση της κόκκινης και μπλε γραμμής. Έχουμε 10 θέσεις σε κάθε σειρά 5 από κάθε πλευρά - σε κάθε θέση (οπή) μπορεί να συνδεθεί και από ένα στοιχείο (ο ακροδέκτης του στοιχείου). Φυσικά μπορούμε να συνδέσουμε την σειρά της μίας πλευράς με σειρά της άλλης πλευράς με καλώδιο ή κάποιο στοιχείο του κυκλώματος. Τα πλαϊνά μεγάλα και οριζόντια μεταλλικά κλιπ ονομάζονται ράγες τροφοδοσίας (power rails). Επισημαίνονται με - η μπλε γραμμή και με + η κόκκινη γραμμή. Χρησιμοποιούνται για την τροφοδοσία του κυκλώματος με τάση ή και τη γείωση. Οι ράγες τροφοδοσίας προφανώς παρέχουν στο κύκλωμα την ίδια τάση από όλα τα σημεία τους. Τα σύμβολα + και - δεν συμβολίζουν κάποια υποχρέωση σύνδεσης με σωστή πολικότητα. Ας υποθέσουμε ότι συνδέουμε μια μπαταρία στο raster μέσω των power rails. Θα μπορούσαμε να συνδέσουμε τον αρνητικό πόλο στο rail με σύμβολο + και το αντίστροφο χωρίς να είναι κάτι «απαγορευτικό» για το raster αφού τα + και - απλά εξυπηρετούν την ευκολία αναφοράς στο τι ακριβώς συνδέουμε στο raster. Το κενό στη μέση του raster εξυπηρετεί στη χρήση microchip όπου οι ακροδέκτες δεν θα μπορούσαν να συνδεθούν σε οπές ίδιας σειράς διότι προφανώς θα είναι βραχυκυκλωμένοι. Ένα τέτοιο microchip είναι το L293D H Bridge που μπορούμε να το χρησιμοποιήσουμε

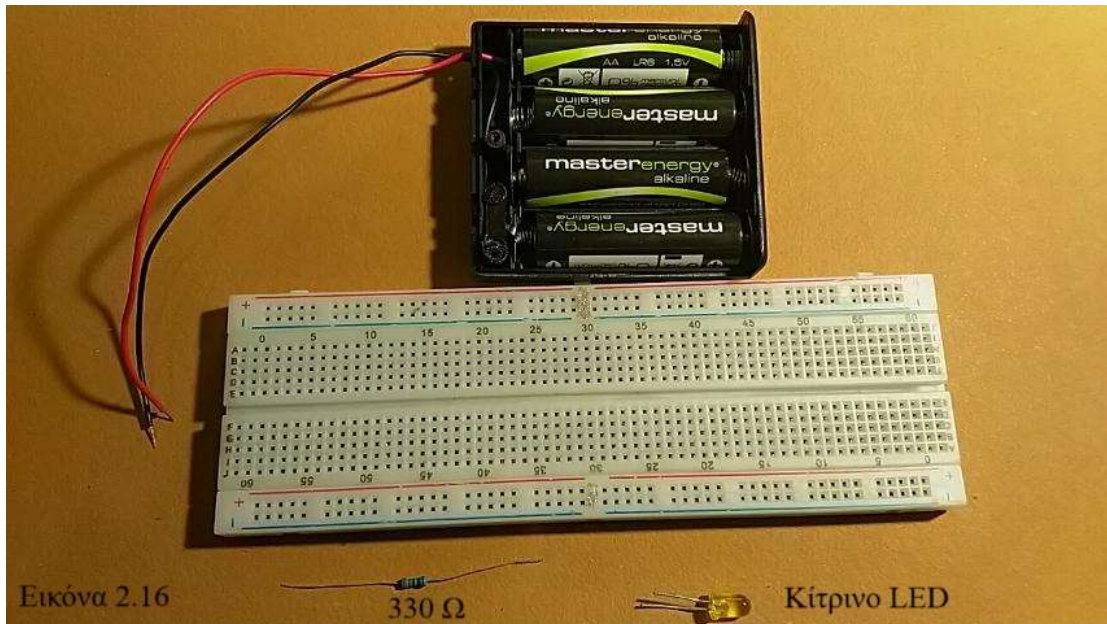
για έλεγχο DC κινητήρων με τον Arduino. Στην κάθε πλευρά του έχει 8 ακροδέκτες οι οποίοι εφαρμόζουν στις οπές της κάθε πλευράς του raster. Η αλφαβητική σήμανση των στηλών βοηθά στην σωστή τοποθέτηση των στοιχείων ενός σύνθετου κυκλώματος και στην χαρτογράφησή τους κατά κάποιο τρόπο.



Στην Εικόνα 2.15 βλέπουμε ένα παράδειγμα του τί σημαίνει σύνδεση σε σειρά και σύνδεση παράλληλα. Οι αντιστάσεις 2 και 3 είναι σε παράλληλη

σύνδεση – οι ακροδέκτες τους είναι συνδεδεμένοι στις ίδιες σειρές δηλαδή οι αντιστάσεις έχουν την ίδια τάση στα άκρα τους (διαφορά τάσης ανάμεσα στις σειρές που είναι συνδεδεμένοι οι ακροδέκτες). Το κλιπ που συνδέονται οι ακροδέκτες των αντιστάσεων είναι το «απλό σύρμα σύνδεσης» όπως το βλέπετε στο σκαρίφημα των αντιστάσεων R_2 και R_3 . Αν ο ένας από τους δύο αντιστάτες είχε και τους δύο ακροδέκτες σε οπές διαφορετικών σειρών τότε οι αντιστάτες δεν θα ήταν συνδεδεμένοι ενώ αν είχε τον ένα ακροδέκτη σε σειρά ίδια με τον άλλο ακροδέκτη ενώ ο άλλος ακροδέκτης ήταν σε διαφορετική σειρά από τον αντίστοιχο του άλλου αντιστάτη τότε θα ήταν συνδεδεμένοι σε σειρά (οι αντιστάτες). Παράδειγμα σύνδεσης σε σειρά είναι η αντίσταση 1 με την αντίσταση 2 ή 3. Ο ένας ακροδέκτης της μίας είναι συνδεδεμένος στην ίδια σειρά με τον αντίστοιχο της άλλης ενώ οι άλλοι ακροδέκτες σε οποιαδήποτε άλλη. Έτσι το ρεύμα περνάει από τον ένα αντιστάτη (ανάλογα την φορά του) και μέσω της κοινής τους επαφής στο κλιπ στο οποίο συνδέονται περνάει και από την επόμενη.

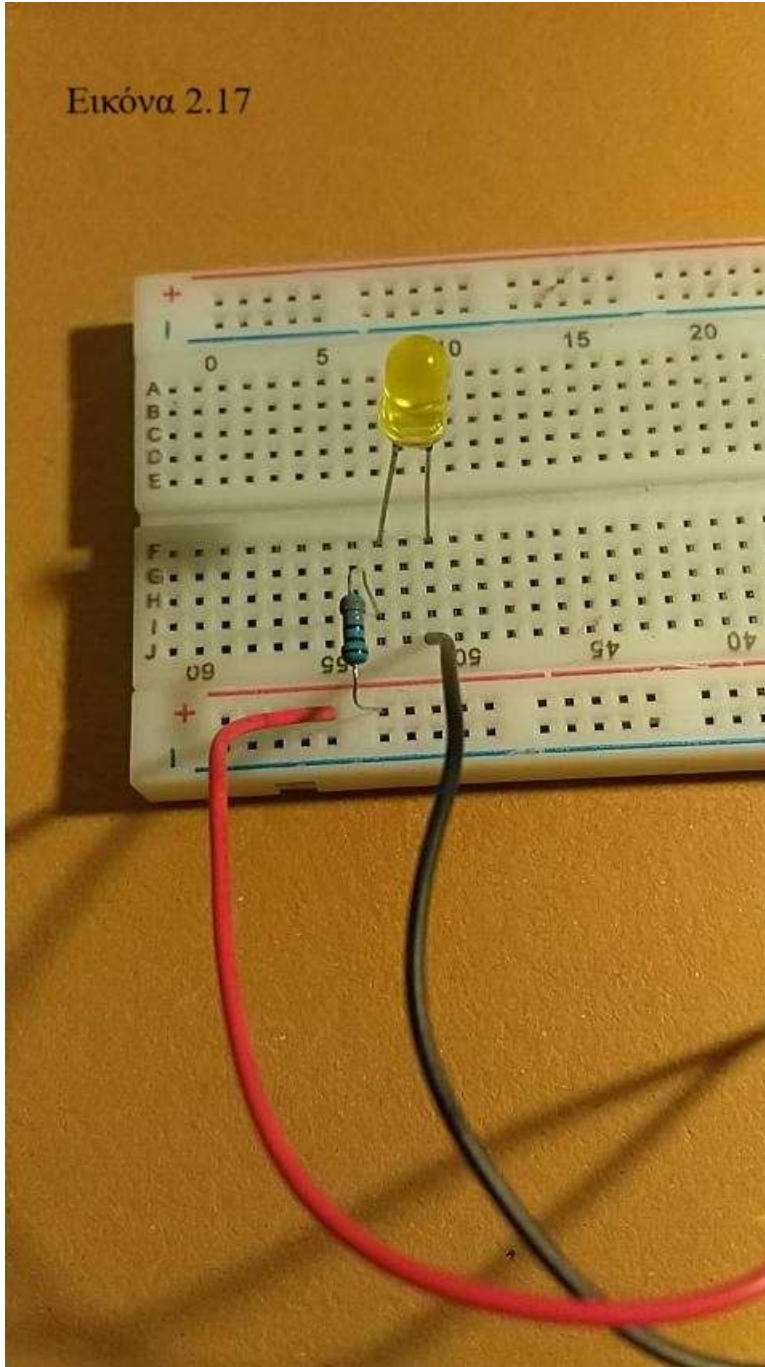
2.3 Ανάβοντας ένα LED



Το kit κάθε ομάδας περιλαμβάνει:

- 4 μπαταρίες AA 1,5 Volt η καθεμία συνδεδεμένες σε μπαταριοθήκη με διακόπτη ON/OFF. Θεωρείστε το κόκκινο καλώδιο ως άνοδο της πηγής και το μαύρο καλώδιο την κάθοδο της πηγής. Η πηγή (μπαταριοθήκη) αποτελείται από 4 επί μέρους πηγές (τις μπαταρίες) οι οποίες είναι συνδεδεμένες σε σειρά (αρνητικός πόλος με θετικό πόλο). Η συνολική τάση είναι $4 \cdot 1,5 \text{ Volt} = 6 \text{ Volts}$
- Raster
- Αντίσταση 330 Ohm. Η αντίσταση χρησιμεύει για την προστασία του LED από υπερβολικό ρεύμα που θα έχει σαν συνέπεια την καταστροφή του
- LED κίτρινου χρώματος Παρατηρήστε ότι ο μακρύτερος ακροδέκτης είναι η άνοδος και ο κοντύτερος η κάθοδος.

Εικόνα 2.17



σύνδεσης;)

ΕΡΩΤΗΜΑ: Θα μπορούσαν το κόκκινο καλώδιο, η αντίσταση και η άνοδος του LED να συνδεθούν στην ίδια σειρά του raster;

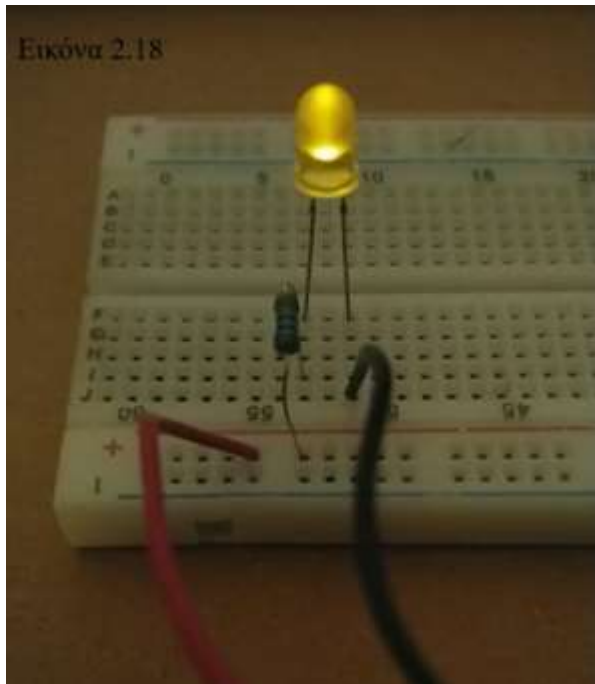
ΑΠΑΝΤΗΣΗ: Όχι – αν η πηγή ήταν συνδεδεμένη στην ίδια σειρά με την άνοδο του LED και την αντίσταση τότε το ρεύμα θα παράκαμπε την αντίσταση και θα κατέστρεφε το LED. Η συνδεσμολογία αυτή αναγκάζει το ρεύμα να διέλθει μέσα από την αντίσταση και προστατεύει το LED.

- Συνδέστε το LED σε δυο διαφορετικές σειρές για να αποφύγουμε το βραχυκύκλωμα

- Συνδέστε τον ένα ακροδέκτη της αντίστασης στην ίδια σειρά με την άνοδο του LED και τον άλλο ακροδέκτη στην ράγα τροφοδοσίας +

- Συνδέστε το κόκκινο καλώδιο της πηγής στην ράγα +

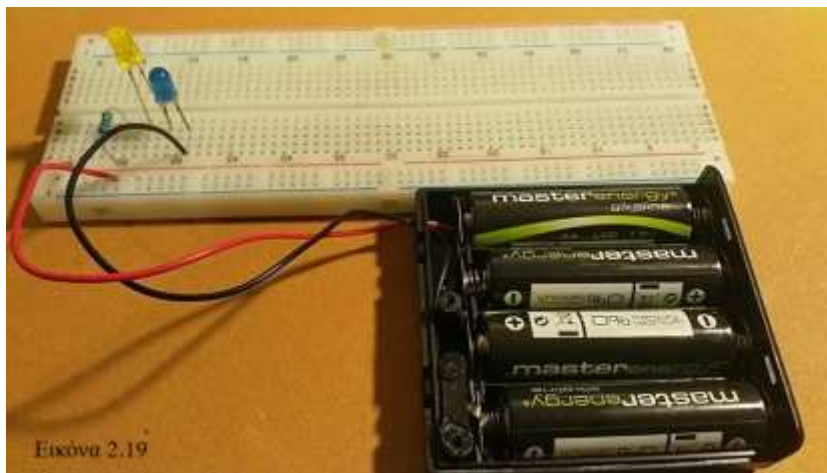
- Συνδέστε το μαύρο καλώδιο της πηγής στην ίδια σειρά με την κάθοδο του LED (θα ήταν ισοδύναμη η σύνδεση του μαύρου καλωδίου στην ράγα – και μετά η σύνδεση της σειράς της καθόδου με τη ράγα – με καλώδιο



Το κύκλωμα είναι έτοιμο. Γυρίστε τον διακόπτη στη θέση ON και παρατηρήστε το αποτέλεσμα.

Τι θα περιμένατε ως προς την φωτεινότητα του LED αν ο αντιστάτης ήταν 220 Ω αντί για 330 Ω;

2.4 Δύο LED σε σειρά



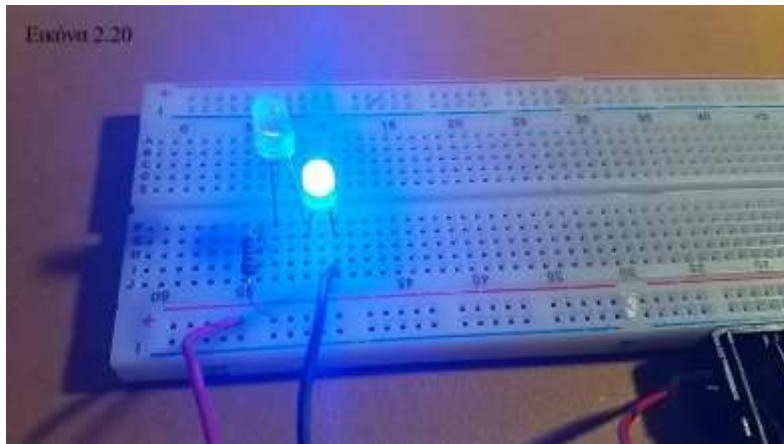
Στην ίδια συνδεσμολογία με την προηγούμενη ενότητα προσθέτουμε ένα μπλε LED. Το μπλε LED συνδέεται σε σειρά με το κίτρινο LED. Η

συνδεσμολογία των δύο LED σε σειρά επιτυγχάνεται με τη σύνδεση της καθόδου του κίτρινου LED με την άνοδο του μπλε LED. Δεν έχει κάποια σημασία το χρώμα του LED, η σύνδεση θα μπορούσε να ήταν και αντίστροφη.

ΕΡΩΤΗΜΑ: Γιατί δεν χρησιμοποιούμε και μια δεύτερη αντίσταση για την προστασία του μπλε LED;

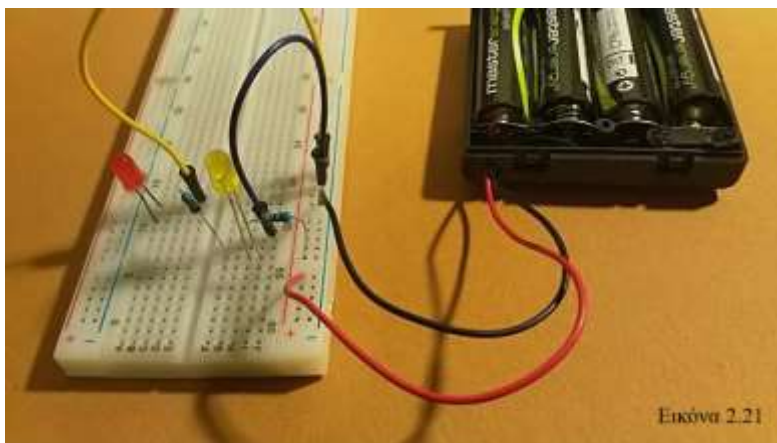
ΑΠΑΝΤΗΣΗ: Η αντίσταση και τα δύο LED είναι συνδεδεμένα σε σειρά μεταξύ τους οπότε το ρεύμα που διαρρέει το κύκλωμα είναι το ίδιο. Τα LED προστατεύονται εξίσου από την αντίσταση οπότε μια δεύτερη αντίσταση θα ήταν περιττή ή και θα

εμπόδιζε την καλή λειτουργία του κυκλώματος. Θα δούμε παρακάτω στην παράλληλη σύνδεση αν χρειάζεται ή όχι δεύτερη αντίσταση.



Το κύκλωμα είναι έτοιμο.

2.5 Δυο LED σε παράλληλη σύνδεση



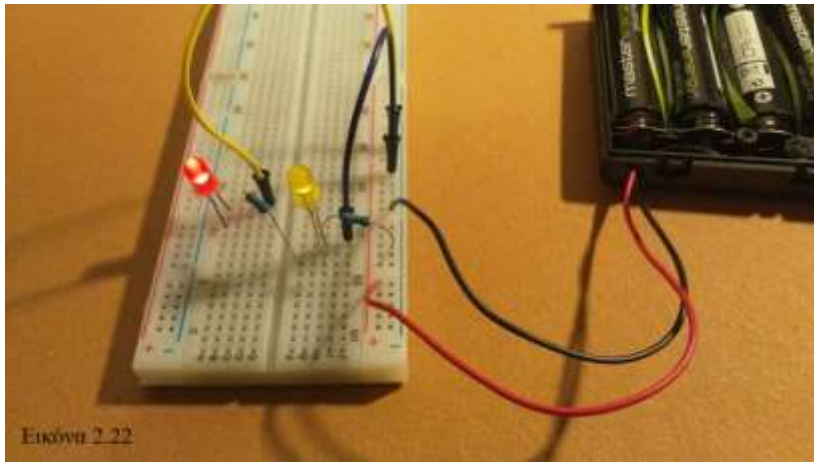
Το κύκλωμα μας γίνεται λίγο πιο πολύπλοκο.

- Συνδέστε την άνοδο της πηγής στην κόκκινη ράγα τροφοδοσίας (+)
- Συνδέστε την κάθοδο της πηγής στην μπλε ράγα

τροφοδοσίας(-). Με αυτόν τον τρόπο μπορούμε να συνδέσουμε πολλαπλά στοιχεία με τις δύο μπάρες τροφοδοσίας

- Η αντίσταση 330 Ohm συνεχίζει να συνδέει την άνοδο του κίτρινου LED με την μπάρα τροφοδοσίας (άνοδο της πηγής)
- Μια δεύτερη αντίσταση των 330 Ohm συνδέει την άνοδο του κίτρινου LED με την άνοδο του κόκκινου LED. Αυτό συμβαίνει διότι το ρεύμα που διαρρέει τα δύο LED δεν είναι κοινό αλλά σε κάθε LED ρέει το ανάλογο με την αντίσταση του ρεύμα. Δηλαδή στον κλάδο του κυκλώματος που ανήκει το κάθε LED κυκλοφορεί ρεύμα ανάλογο της αντίστασης που συναντά. Εδώ το ρεύμα καθορίζεται από τον αντιστάτη και την έστω μικρή αντίσταση του LED. Οι δυο αντιστάσεις παίζουν τον ρόλο και του καλωδίου σύνδεσης

- Οι κάθοδοι και των δύο LED συνδέονται μέσω καλωδίων σύνδεσης με την μπλε μπάρα τροφοδοσίας δηλαδή την κάθοδο της πηγής (μαύρο καλώδιο του μπαταριοθήκης). Η – μπάρα τροφοδοσίας μπορεί να θεωρηθεί ως τάση αναφοράς ή γείωση όπως θα δούμε στα επόμενα
- Η αυλάκωση στο κέντρο γνωρίζουμε ότι χωρίζει τις σειρές του raster γι' αυτό και ο δεύτερος αντιστάτης συνδέει τις δύο πλευρές



Εικόνα 2.22

Στην Εικόνα 2.22 βλέπετε το κύκλωμα σε λειτουργία. Η φωτεινότητα των LED είναι περίπου ίδια αναλόγως της αντίστασης και της τάσης λειτουργίας του LED

Υλικό για μελέτη στο σπίτι

Γενική θεωρία κυκλωμάτων

<https://www.youtube.com/watch?v=ZrPorg3pY4k>

Οπτικοποιημένα παραδείγματα σύνδεσης ηλεκτρονικών στοιχείων σε σειρά και παράλληλα

<http://photodentro.edu.gr/aggregator/lo/photodentro-lor-8521-1599>

<http://photodentro.edu.gr/aggregator/lo/photodentro-lor-8521-1623>

<http://photodentro.edu.gr/aggregator/lo/photodentro-lor-8521-6207>

Εκτενής παρουσίαση σχετικά με την χρήση του raster

<https://www.youtube.com/watch?v=krpFYF3NRds>

Μελέτη κυκλωμάτων με τη χρήση του Tinkercad

Το Tinkercad είναι ένα εικονικό περιβάλλον που σου δίνει την δυνατότητα να φτιάξεις και να δεις τη λειτουργία κυκλωμάτων χωρίς να χρειάζεται η αγορά κάποιου κιτ

<https://www.tinkercad.com/join>

Αφού κάνετε εγγραφή (Create a personal account) και δημιουργήσετε τον λογαριασμό σας επιλέξτε Circuits -> Create your first Circuits Design

Μελέτη κυκλωμάτων με τη χρήση του Tinkercad

Οι επιλογές και τα κυκλώματα που μπορείτε να δημιουργήσετε και να δοκιμάσετε είναι πολλαπλά

Ενδεικτικά θα μπορούσατε να μελετήσετε τα κυκλώματα LC και RL κατά τα καθοδηγητικά παραδείγματα που θα δοθούν παρακάτω αλλά και όποιο άλλο κύκλωμα επιθυμείτε

<http://photodentro.edu.gr/aggregator/lo/photodentro-lor-8521-10569> (RL)

<http://photodentro.edu.gr/aggregator/lo/photodentro-educationalvideo-8522-261> (LC)

Εάν δεν έχετε εμπειρία στην κατασκευή και δοκιμή κυκλωμάτων με το Tinkercad μπορείτε να συμβουλευτείτε τον επίσημο οδηγό:

<https://www.tinkercad.com/blog/official-guide-to-tinkercad-circuits>

Άσκηση 1^η

1. Οι μπαταρίες της μπαταριοθήκης είναι σε σειρά και αποδίδουν 1,5 Volt η καθεμία. Πόση θα είναι η συνολική τάση των 5 όμοιων μπαταριών των 1,5V;
2. Τα LED αποδίδουν καλύτερα όταν τα διαρρέει ρεύμα 15 έως 18 mA. Η ανώτερη ασφαλής τιμή ρεύματος είναι 20 mA. Πάνω από αυτή τη τιμή υπάρχει κίνδυνος καταστραφεί το LED. Το LED έχει μικρή αντίσταση και τη θεωρούμε αμελητέα. Ποια η τιμή του ρεύματος που διαρρέει το κύκλωμα μας (Θεωρήστε ως τάση την τιμή που υπολογίσατε στο 1^ο ερώτημα και αντίσταση τα 330Ω);

Άσκηση 2^η

Στο κύκλωμα της Εικόνας 2.22 να σκεφτείτε και να σχεδιάσετε στο Tinkercad έναν διαφορετικό τρόπο σύνδεσης των αντιστάσεων των LED έτσι ώστε να μην συνδέονται με την ράγα τροφοδοσίας (αντίσταση κίτρινου LED) ή τις σειρές του άλλου μισού του raster (αντίσταση κόκκινου LED). Μπορείτε να χρησιμοποιήσετε οποιοδήποτε εξάρτημα κρίνετε απαραίτητο για τον επανασχεδιασμό του κυκλώματος.

ΚΕΦΑΛΑΙΟ 3

ΜΑΘΗΜΑ 3^ο: Εισαγωγή στο Arduino

3.1 Μικροελεγκτές

Οι μικροελεγκτές είναι ολοκληρωμένα κυκλώματα (integrated circuit - IC) που χρησιμοποιούνται για κάποια συγκεκριμένη εφαρμογή και είναι σχεδιασμένα για να εκτελούν συγκεκριμένες λειτουργίες [17]. Ο μικροελεγκτής είναι ουσιαστικά μία



μικρογραφία του γνωστού από την καθημερινότητα μας ηλεκτρονικού υπολογιστή χωρίς αυτό να σημαίνει ότι ο μικροελεγκτής έχει όλες τις δυνατότητες του υπολογιστή. Οι μικροελεγκτές δεν είναι ένα απομονωμένο τσιπ το οποίο έχουμε προγραμματίσει να εκτελέσει μία λειτουργία.

Οι μικροελεγκτές

συλλέγουν δεδομένα μέσω αισθητηρίων και άλλων διεπαφών τα επεξεργάζονται και εκτελούν την ανάλογη λειτουργία. Οι μικροελεγκτές είναι σχεδιασμένοι έτσι ώστε να εκτελούν επαναλαμβανόμενα ένα πρόγραμμα μέσω του μικροεπεξεργαστή τους, [16] με αποτέλεσμα η χρήση τους να είναι οικονομική και να προωθεί την επέκταση του Internet of Things (IoT). Έχουν μικρές απαιτήσεις σε ρεύμα και μνήμη και είναι πολύ χρήσιμοι για την περάτωση απλών διαδικασιών (π.χ. άναμμα ή σβήσιμο LED) για τις οποίες θα ήταν ασύμφορη η χρήση μίας μονάδας ηλεκτρονικού υπολογιστή. Οι μικροελεγκτές θα μπορούσαν να θεωρηθούν μια πιο περιορισμένη μορφή ηλεκτρονικού υπολογιστή συμπιεσμένου σε ένα ολοκληρωμένο κύκλωμα αφού τα βασικά μέρη που αποτελούν ένα υπολογιστή υπάρχουν και σε ένα μικροελεγκτή (CPU, RAM, I/O, Flash Memory, EEPROM). Μπορούμε να κάνουμε παραλληλισμούς από τα γνωστά για εμάς στους Η/Υ και στους μικροελεγκτές. Η

Κεντρική Υπολογιστική Μονάδα (CPU) είναι υπεύθυνη για τους αριθμητικούς και λογικούς υπολογισμούς και την εκτέλεση των εντολών ενώ δέχεται και επεξεργάζεται τα εισερχόμενα δεδομένα. Η RAM είναι η γνωστή προσωρινή μνήμη γρήγορης ανάκλησης. Αποθηκεύει δεδομένα προσωρινά τα οποία χάνονται όταν απενεργοποιηθεί ο μικροελεγκτής. Η Flash Memory είναι ένα είδος μνήμης που τα δεδομένα που αποθηκεύει διατηρούνται και μετά την απενεργοποίηση αλλά εγγράφει δεδομένα σε τμήματα της μνήμης και όχι σε κάθε byte ξεχωριστά. Η EEPROM (Electrical Erasable Programmable Read-Only Memory) είναι και αυτή μνήμη μακρόχρονης αποθήκευσης δεδομένων με τη διαφορά ότι μπορεί να εγγράψει και να σβήσει byte στην μνήμη στον απαιτούμενο χώρο για 1 byte και όχι σε ολόκληρους τομείς της μνήμης. Η EEPROM είναι μακροβιότερη αλλά και πιο ακριβή από την μνήμη Flash. Οι θύρες Εισόδου Εξόδου είναι οι «πύλες» επικοινωνίας του μικροεπεξεργαστή με το περιβάλλον. Θα μπορούσαμε να τις χαρακτηρίσουμε και περιφερειακές μονάδες του μικροεπεξεργαστή. Ο μικροελεγκτής δέχεται δεδομένα από το περιβάλλον του. Αυτό μπορεί να είναι μία μέτρηση θερμοκρασίας από ένα αισθητήρα, μια μέτρηση φωτεινότητας κτλ. Μπορεί να δεχτεί μία εισαγωγή δεδομένων από το πληκτρολόγιο κοκ. Ο μικροεπεξεργαστής αναλύει αυτά τα δεδομένα, τα επεξεργάζεται και τελικά εκτελεί την διαδικασία σύμφωνα με τις εντολές για τις οποίες έχει προγραμματιστεί. Οι εντολές αυτές προφανώς υλοποιούνται μέσω των πυλών εξόδου όπου ο μικροεπεξεργαστής εκτελεί μία λειτουργία ανάλογα με τις εντολές που έχει λάβει. Μπορεί για παράδειγμα να κινήσει κάποιον κινητήρα ή να ενεργοποιήσει ένα LED.

3.2 Η πλατφόρμα Arduino

Η πλατφόρμα Arduino χρησιμοποιείται για την κατασκευή project ηλεκτρονικών κυκλωμάτων και βρίσκει μία πλειάδα εφαρμογών στην εκπαίδευση, σε κάποιες εφαρμογές στην βιομηχανία, στην έρευνα ακόμα και στα χόμπι και την δημιουργική απασχόληση. Η πλατφόρμα Arduino είναι ειδικότερα αναπόσπαστο κομμάτι της STEM εκπαίδευσης και δίνει την δυνατότητα στο εκπαιδευτικό προσωπικό να εκπαιδεύσει τους μαθητές και τις μαθήτριες στους μικροελεγκτές, τα ηλεκτρονικά κυκλώματα, τον προγραμματισμό, το Internet of Things. Μπορεί να εξάψει στους μαθητές την δημιουργικότητα, τον αναλυτικό τρόπο σκέψης, την φαντασία και την εφευρετικότητα ενώ παράλληλα μέσω της εργασίας σε ομάδες μαθαίνουν την

αλληλοβοήθεια, τον συντονισμό, την συνεργατικότητα κ.α. Η πλατφόρμα Arduino αποτελείται τόσο από το κομμάτι Hardware (πλακέτα και παρελκόμενα) όσο και από κομμάτι Software (κώδικας που αποθηκεύεται στον μικροεπεξεργαστή της πλακέτας) που δίνει τις οδηγίες στο Arduino να εκτελέσει συγκεκριμένες λειτουργίες επαναλαμβανόμενα ανάλογα των σημάτων εισόδου όπως ένταση ηλιακής ακτινοβολίας, υγρασία ατμόσφαιρας ακόμα και μία ανάρτηση σε μέσα κοινωνικής δικτύωσης.



Εικόνα 3.2

Η σύνδεση με τον ηλεκτρονικό υπολογιστή γίνεται με καλώδιο USB με το κατάλληλο βύσμα στην πλακέτα. Η πλατφόρμα Arduino αναπτύχθηκε το 2003 από μία ακαδημαϊκή ομάδα στο Interaction Design Institute Ivrea και είχε σαν σκοπό να κάνει εύκολη και οικονομικά συμφέρουσα την εισαγωγή των φοιτητών

στον αυτοματισμό. Η ιδέα αυτή με τα χρόνια επεκτάθηκε και εισχώρησε και στις κατώτερες βαθμίδες εκπαίδευσης αφού η πλατφόρμα είναι ιδιαίτερα προσβάσιμη και εύκολη στην κατανόηση ακόμα και σε αυτές τις ηλικίες [18]. Ένα άλλο προτέρημα του Arduino είναι ο open source χαρακτήρας του. Είναι open source όχι μόνο σε επίπεδο software αλλά και hardware. Το σχετικά χαμηλό κόστος του Arduino μπορεί να είναι ακόμα πιο χαμηλό με την αγορά μίας πλακέτας off brand η οποία δεν υστερεί σε λειτουργικότητα ή δυνατότητες από την αυθεντική πλακέτα. Τα δε παρελκόμενα (αισθητήρες, καλώδια, κινητήρες κτλ.) είναι επίσης οικονομικά και ευρέως διαδεδομένα στο εμπόριο. Ο κώδικας προγραμματισμού είναι και αυτός open source με πληθώρα έτοιμων προγραμμάτων και βιβλιοθηκών κώδικα να είναι διαθέσιμα στο διαδίκτυο και στο προγραμματιστικό περιβάλλον.

Τα κυριότερα πλεονεκτήματα του Arduino είναι[19]:

- Χαμηλό κόστος: Ειδικά για τους στενούς σχολικούς προϋπολογισμούς η πλατφόρμα Arduino είναι μία πολύ προσιτή επιλογή για τον εξοπλισμό ενός εργαστηρίου για τα μαθήματα Πληροφορικής και Τεχνολογίας
- Συμβατό με όλα τα λειτουργικά συστήματα: Το προγραμματιστικό περιβάλλον και η γλώσσα προγραμματισμού του Arduino είναι συμβατά τόσο με τα Windows όσο και με Linux και MacOS
- Ευκολία προγραμματισμού: η γλώσσα προγραμματισμού του Arduino είναι απλή, ευκολονόητη και κατάλληλη για εκμάθηση σε μαθητές Γυμνασίου
- Open Source: πρακτική που βοηθάει όχι μόνο στην διάδοση αλλά και στην δημιουργία community και υποστήριξης καθώς και στην διατήρηση του χαμηλού κόστους στο hardware. Η γλώσσα Wiring με την οποία προγραμματίζεται η πλακέτα για να εκτελέσει κάποια διαδικασία και να επικοινωνήσει με τα διάφορα περιφερειακά (αισθητήρες, κινητήρες, LED κτλ.) είναι μια παραλλαγή της προγραμματιστικής γλώσσας C++, open source, κατανοητή και απλή στη χρήση

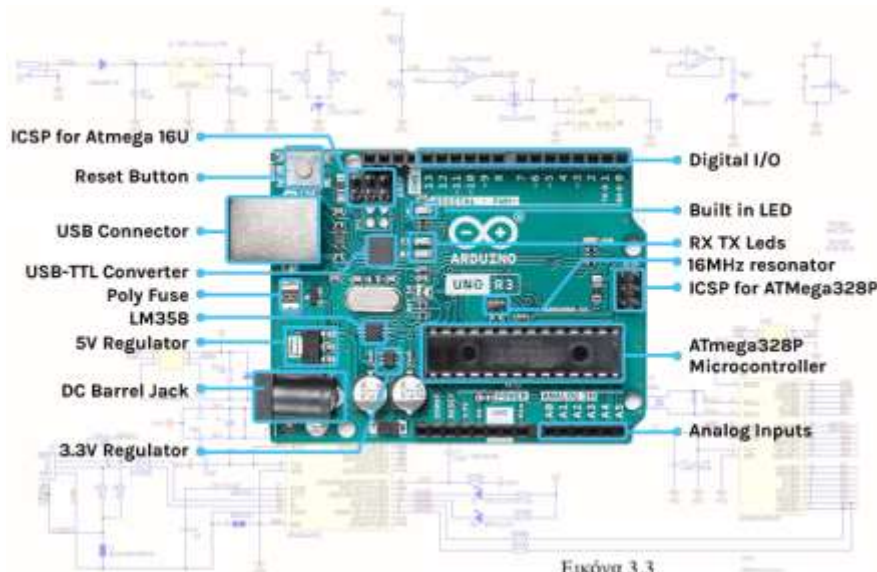
Το Arduino έχει το δικό του προγραμματιστικό περιβάλλον Arduino IDE (Integrated Development Environment) το οποίο είναι διαθέσιμο τόσο για download όσο και για online εργασία. Ο κώδικας αυτός είναι εύκολα κατανοητός και φυσικά επιδέχεται αλλαγών ανάλογα με τις ιδιαίτερες συνθήκες και λειτουργίες που επιθυμεί ο προγραμματιστής να εκτελεί σύστημα που μέρος του αποτελεί το Arduino. Η κοινότητα του Arduino είναι μεγάλη, ενεργή και παρέχει λύσεις και προοπτική στα νέα μέλη της. Το Arduino υστερεί σε βιομηχανικές εφαρμογές καθότι εκεί απαιτούνται μικροελεγκτές με ισχυρότερη υπολογιστική δύναμη, πολύπλοκο κώδικα και αντοχή του υλικού σε ακραίες περιβαλλοντικές συνθήκες. Στην βιομηχανία το Arduino χρησιμοποιείται για να ανιχνευθούν σήματα τα οποία με τη σειρά τους τα οποία ενεργοποιούν τον Arduino για να δώσει το ανάλογο σήμα έναρξης σε κάποια άλλη συσκευή. Η ευκολία χρήσης, το χαμηλό κόστος και το μεγάλο εύρος εφαρμογών καθιστούν το Arduino ιδανικό για το project μας που θα βασιστεί στον έλεγχο κίνησης, απόστασης και αυτόνομης κίνησης.

3.3 Η πλακέτα Arduino UNO Rev 3

Ανά τα έτη έχουν εκδοθεί πολλές πλακέτες (ή αλλιώς board κατά την Αγγλική ορολογία) Arduino με διαφορετικά χαρακτηριστικά όπως το Leonardo, Mega, RedBoard, LilyPad. Οι διαφορετικές πλακέτες εξυπηρετούν και διαφορετικές σκοπούς ή είναι εξελιγμένες μορφές προηγούμενων εκδόσεων. Το Arduino Pro για παράδειγμα είναι κατάλληλο για βιομηχανικές εφαρμογές καθώς δέχεται υψηλότερες τάσεις και είναι πιο ανθεκτική κατασκευή ενώ το LilyPad είναι κατάλληλο για wearable εφαρμογές (συσκευές μέτρησης παλμών της καρδιάς με δυνατότητα επικοινωνίας με smartphone κ.α.). Σε αυτό το project θα ασχοληθούμε με το board UNO Rev3 που είναι από τα πιο κοινά board λόγω του χαμηλού κόστους του και της σταθερότητας του. Είναι από τα πιο κοινά και ευρέως χρησιμοποιούμενα board και μπήκε στην κυκλοφορία περίπου το 2010 στο Maker Faire στη Νέα Υόρκη [20]. Οι προηγούμενες εκδόσεις του εξελίσσονταν έτσι ώστε να γίνονται όλο και πιο χρηστικές και αξιόπιστες. Ένα παράδειγμα είναι το βύσμα USB όπου σε προηγούμενες εκδόσεις είχε τοποθετηθεί microUSB το οποίο όμως αποδείχτηκε ευαίσθητο και αντικαταστάθηκε από το συνηθισμένο USB. Συνδέεται στον υπολογιστή μέσω καλωδίου USB (USB Connector) ενώ παράλληλα έχει δυνατότητα να συνδεθεί με πηγή DC (μπαταρία) μέσω του βύσματος (Barrel Jack).

Το board λειτουργεί μέσα σε ένα περιθώριο τάσης εισόδου από 6 έως 20 Volt (οριακές ασφαλείς τιμές λειτουργίας). Τάσεις κάτω των 6 Volt δεν είναι επαρκείς ενώ άνω των 20 μπορούν να επιφέρουν βλάβη στο board. Η ιδανική τιμή τάσης εισόδου είναι 9 V ενώ το περιθώριο καλής λειτουργίας είναι τα 7-12 V. Το board λειτουργεί στα 5 Volt. Το board μπορεί να παρέχει τάσεις 5 και 3,3 Volt. Στους ακροδέκτες αυτούς μπορούν να συνδεθούν αισθητήρια από τα οποία το board μπορεί να συλλέξει δεδομένα και να εκτελέσει τις προγραμματισμένες λειτουργίες. Ο «εγκέφαλος» του Arduino είναι το τσιπ ATmega328P. Το τσιπ αυτό έχει 32 kB Flash μνήμη, 1 B EEPROM [20], 23 θύρες Input / Output, εσωτερικό ρολόι 20 MHz και 2 kB RAM. Λειτουργεί σε τάσεις μεταξύ 1,8-5,5 Volts. Μπορεί να λειτουργήσει σε θερμοκρασίες από -40 έως +85 °C Ο μικροελεγκτής ATmega328P μπορεί να τοποθετηθεί στους αντίστοιχους ακροδέκτες του board και να αφαιρεθεί κατά το δοκούν αφού τα «ποδαράκια» του δεν είναι κολλημένα στους ακροδέκτες. Μια από τις λειτουργίες

του τσιπ είναι να φορτώνεται σε αυτό ο κώδικας που αναπτύξαμε και να εκτελείται συνεχόμενα. Εκεί είναι όπως είπαμε και στα προηγούμενα ο εγκέφαλος του board.



Εικόνα 3.3

νευρικό του σύστημα. Σε αυτούς τους ακροδέκτες συνδέονται τα διάφορα παρελκόμενα (LEDs, αισθητήρες κ.α.) και παράλληλα μπορούν να αποδώσουν στο κύκλωμα κάποιο σήμα εξόδου[22]. Μπορούν να χρησιμοποιηθούν δηλαδή και ως είσοδος και ως έξοδος εξ ου και το όνομα του Input/Output. Είναι ακροδέκτες θηλυκού τύπου.

Πιο αναλυτικά, συνολικά το board έχει 28 ακροδέκτες που χωρίζονται σε τρεις ομάδες: τους ακροδέκτες τροφοδοσίας που αποδίδουν 5 Volt, 3,3 Volt και γείωση GND όπου χρειάζεται. Είδαμε στο προηγούμενο μάθημα πως συνδέουμε με καλώδιο σύνδεσης την τάση τροφοδοσίας στο raster. Οι ακροδέκτες Analog Input A0-A5 μπορούν να χρησιμοποιηθούν και αυτές ως έξοδοι και μάλιστα ως ψηφιακοί έξοδοι και με τις ίδιες εντολές με τις ψηφιακές όπως θα δούμε παρακάτω. Οι ακροδέκτες αυτοί όπως και οι ψηφιακές δηλώνονται ως είσοδοι ή έξοδοι σε κάποιο σημείο του sketch μας. Οι αναλογικοί ακροδέκτες Analog Input μπορούν να χρησιμοποιηθούν και ως αναλογικοί είσοδοι. Μπορούν δηλαδή να λάβουν αναλογικό σήμα και να το μεταδώσουν στον μικροελεγκτή του board. Ας πάρουμε το παράδειγμα ενός αισθητήρα φωτεινότητας. Ο αισθητήρας αυτός μετατρέπει την φωτεινή ακτινοβολία που λαμβάνει από το περιβάλλον του σε τάση. Αυτός ο αισθητήρας παράγει συνεχείς τιμές και όχι ψηφιακές. Δηλαδή μπορεί να αποδώσει οποιαδήποτε τιμή ανάμεσα σε ένα διάστημα αριθμών και όχι μόνο 0 ή 5 Volt, TRUE ή FALSE 0 ή 1. Το board βέβαια δεν μπορεί να διαβάσει οποιαδήποτε τιμή τάσης οσοδήποτε υψηλή. Υπάρχει ένα όριο μέχρι 5 Volt. Πως όμως μπορεί ο μικροελεγκτής να επεξεργαστεί αυτές τις

Αν το τσιπ ATmega328P είναι ο εγκέφαλος του board τότε οι ακροδέκτες – θύρες Digital I/O (Input/Output) και Analog Input είναι το

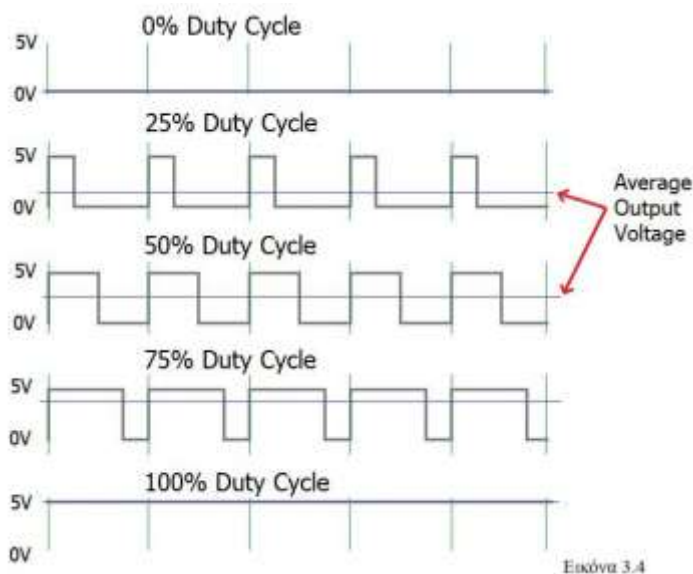
συνεχείς τιμές τάσης από 0 έως 5 Volt. Ο μικροεπεξεργαστής έχει την μονάδα ADC (Analog Digital Converter) που «μεταφράζει» τις τιμές αυτές σε μία ψηφιακή τιμή που μπορεί να την διαβάσει ένα sketch του Arduino. Η μονάδα αυτή δεν κάνει τίποτα άλλο από το να αντιστοιχεί τις τιμές 0 – 5 Volt σε έναν από τους ακέραιους αριθμούς από 0 – 1023. Το 0 αντιστοιχεί στα 0 Volt και το 1023 στα 5 Volt άρα για κάθε 4,8 mVolt αντιστοιχεί ένας αριθμός από 0 έως 1023.

Από τούς 28 ακροδέκτες οι 20 χρησιμοποιούνται για Είσοδο / Έξοδο. Οι 6 είναι οι Analog Input και οι 14 Digital Input / Output. Οι Digital ακροδέκτες 0-13 (Digital Input/Output) χρησιμοποιούνται για να αποδώσουν αλλά και να λάβουν κάποιο ψηφιακό σήμα όπως να ανάψουν ένα LED ή να λάβουν σήμα από ένα διακόπτη για το αν είναι κλειστός ή ανοιχτός. Θα παρατηρήσατε ότι οι ψηφιακοί ακροδέκτες 3,5,6,9,10,11 έχουν δίπλα τους το σύμβολο \sim . Οι ακροδέκτες αυτοί μπορούν να χρησιμοποιηθούν και για να παρέχουν μία αναλογική έξοδο (σκεφτείτε μια συνεχώς μειούμενη τάση που σβήνει ένα LED σταδιακά). Είναι δηλαδή ψηφιακοί ακροδέκτες που μπορούν με την τεχνική PWM που θα γνωρίσουμε παρακάτω να λειτουργήσουν ως αναλογικοί έξοδοι. Μπορούν να παρέχουν δηλαδή μία συνεχή τιμή εξόδου και όχι μια διακριτή. Στο παράδειγμα ενός κινητήρα δηλαδή μπορούν να αποδώσουν μία οποιαδήποτε τιμή εντός ενός διαστήματος αντί για το «ψηφιακό» 1 ή 0, μέγισυμυ ταχύτητα – μηδενική ταχύτητα. Σκεφτείτε επίσης μια συνεχώς μειούμενη τάση που σβήνει ένα LED σταδιακά και μετά να το ανάβει πάλι σταδιακά σε αντιδιαστολή με το ON – OFF ανάβω ένα LED και μετά το σβήνω.

Αναφερθήκαμε στην προηγούμενη διαφάνεια στους ψηφιακούς ακροδέκτες 3,5,6,9,10,11 που μπορούν να χρησιμοποιηθούν και ως αναλογική έξοδος (να παρέχουν δηλαδή αναλογικό σήμα ως έξοδο - Analog Output). Το σήμα που παρέχουν αυτοί οι ακροδέκτες είναι κατ' ουσία ψηφιακό αλλά λειτουργεί ως αναλογικό με την τεχνική PWM. Ο ψηφιακός παλμός που αποδίδει μια ψηφιακή έξοδος έχει δυο τιμές 5 Volt και 0 Volt (ON - OFF). Με την τεχνική PWM μπορούμε να ελέγξουμε την χρονική διάρκεια για την οποία ο ψηφιακός παλμός είναι 5 V(ON) ή 0V (OFF) και κατά προέκταση το μέσο πλάτος του παλμού – τάσης (average output voltage). Δηλαδή αν έχουμε έναν ακροδέκτη να παρέχει τάση 5 Volt για 500msec και τάση 0 V για άλλα 500msec τότε στο 1 sec ο ακροδέκτης αυτός θα είναι σαν να αποδίδει 2,5 Volt τάσης. Οι χρόνοι αυτοί είναι χονδρικοί και δεν σημαίνει ότι η

διαμόρφωση του σήματος γίνεται με αυτές τις χρονικές διάρκειες. Παίζοντας λοιπόν με την χρονική διάρκεια που ο ψηφιακός ακροδέκτης δίνει 0 ή 5 Volt το board μπορεί να αποδώσει οποιαδήποτε τάση μεταξύ 0 και 5 Volt προσομοιάζοντας μια πηγή σταθερής τάσης. Μπορούμε να χρησιμοποιήσουμε την τεχνική αυτή στον Arduino για τους ψηφιακούς ακροδέκτες που προαναφέραμε με τον να χρησιμοποιήσουμε την εντολή `analogWrite()` την οποία θα γνωρίσουμε αργότερα στον προγραμματισμό του

Arduino



Εικόνα 3.4

Ας αναφέρουμε προς το παρόν ότι η εντολή αυτή δέχεται τιμές 0 έως 255 με το 0 να αντιστοιχεί σε μηδενική τάση δηλαδή «αναλογικό» σήμα 0V (0% duty cycle – κύκλος εργασίας). Στην τιμή 255 η λειτουργία ON – 5 V είναι συνεχής δηλαδή το αναλογικό σήμα είναι

συνεχώς 5V (100% duty cycle) [23]. Οι ενδιάμεσες τιμές των 0-255 προσομοιάζουν όλες τις ενδιάμεσες τιμές τάσης 0-5 Volt όπως βλέπετε στην Εικόνα 3.4. Στην περίπτωση ενός LED δηλαδή η χρήση της εντολής `analogWrite(0)` αντιστοιχεί σε σβηστό LED ενώ η εντολή `analogWrite(255)` σε συνεχώς αναμμένο με οποιεσδήποτε ενδιάμεσες τιμές να αυξομειώνουν την φωτεινή ένταση του LED. Μπορούμε να αυξομειώσουμε την ένταση με μία δομή επανάληψης που θα δούμε αργότερα.

Το board μπορεί να συνδεθεί με καλώδιο USB στον υπολογιστή το οποίο παρέχει και την τάση λειτουργίας αλλά και την επικοινωνία με τον περιβάλλον προγραμματισμού του Arduino. Το sketch που αναπτύσσουμε «φορτώνεται» στο board μέσω του καλωδίου. Εάν χρειάζεται η πλακέτα να είναι σε λειτουργία σε μεγάλη απόσταση από τον υπολογιστή μπορούμε με το καλώδιο USB να φορτώσουμε τον κώδικα στην πλακέτα το οποίο θα μείνει αποθηκευμένο. Στη συνέχεια συνδέουμε μια πηγή (μπαταρία 9 Volt) στο βύσμα DC Barrel Jack και το ολοκληρωμένο κύκλωμα μας (Μικροελεγκτής του Arduino και το κύκλωμα που συνδέεται σε αυτόν) λειτουργεί αυτόνομα και εκτελεί συνέχεια τον κώδικα που του έχουμε αποθηκεύσει μέσω του

καλωδίου USB. Γενικά προτιμούμε επαναφορτιζόμενες μπαταρίες διότι το board ξοδεύει γρήγορα τις μπαταρίες.



Στην εικόνα 3.5 βλέπουμε μία σειρά αισθητηρίων που συνδέονται στο board για να εκτελέσουν εργασίες ή να λάβουν μετρήσεις τις οποίες επιθυμούμε. Στην εικόνα βλέπουμε αισθητήρες

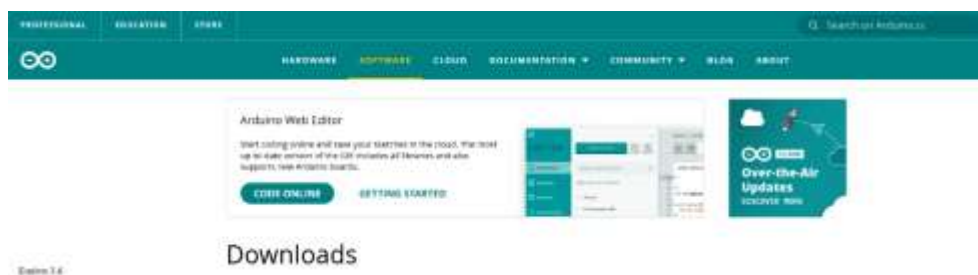
υγρασίας και θερμοκρασίας, εκπομπούς υπέρυθρων, laser, μετρητές απόστασης υπερήχων κα. Στην κατασκευή του οχήματος μας θα χρησιμοποιήσουμε αισθητήρα υπερήχων για την μέτρηση αποστάσεων. Έχετε στο νου σας ότι όσο περισσότερα αισθητήρια και μηχανισμούς συνδέουμε στο board τόσο μεγαλώνουν και οι απαιτήσεις σε ρεύμα.

3.4 Το Προγραμματιστικό Περιβάλλον του Arduino IDE

Το πρώτο μας βήμα στον προγραμματισμό του μικροελεγκτή Arduino είναι φυσικά το να εξοικειωθούμε με το προγραμματιστικό περιβάλλον και τις εντολές της γλώσσας προγραμματισμού Wiring του Arduino. Στα επόμενα μαθήματα δεν θα εξαντλήσουμε το αντικείμενο του προγραμματισμού Arduino αλλά θα εντυπώσουμε αρκετά στην αρχιτεκτονική της γλώσσας Wiring του Arduino και σε βασικές εντολές της που θα μας χρειαστούν για την ανάπτυξη του αυτοκινούμενου οχήματος με δυνατότητα αποφυγής εμποδίων. Ας κάνουμε το πρώτο βήμα με την εγγραφή στην πλατφόρμα του Arduino:

<https://www.arduino.cc/>

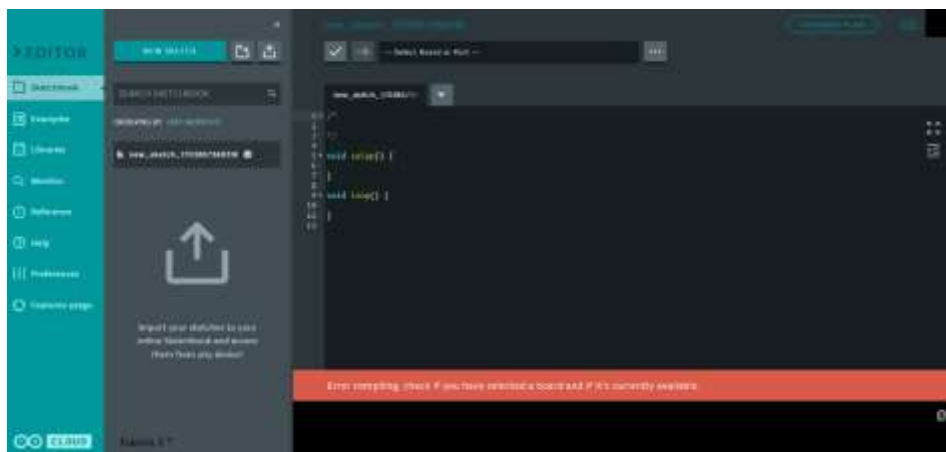
Μπορούμε να κατεβάσουμε το προγραμματιστικό περιβάλλον του Arduino (Arduino Integrated Development Environment) στον υπολογιστή μας και να δουλέψουμε offline ή να γράψουμε κώδικα απ' ευθείας στο online προγραμματιστικό περιβάλλον. Παρέχονται υπηρεσίες cloud οι οποίες μπορούν να αναβαθμιστούν επί πληρωμή. Στο cloud του Arduino μπορούμε να αποθηκεύσουμε τα sketch μας και να έχουμε πρόσβαση σε αυτά από οποιοδήποτε τερματικό. Στο cloud του Arduino μπορούμε να γράψουμε κώδικα, να τον αποθηκεύσουμε ως έτοιμο πρόγραμμα (sketch), να δημιουργήσουμε εικονικά αντίγραφα των συσκευών που θα κατασκευάσουμε και πολλά άλλα. Στον editor περιλαμβάνονται έτοιμα παραδείγματα που περιλαμβάνουν τον κώδικα και την ανάλογη συνδεσμολογία του κυκλώματος, επεξήγηση των εντολών της γλώσσας Wiring του Arduino και φυσικά τον compiler



που
επαληθ
εύει και
εντοπιζ
ει λάθη
στον

κώδικα που γράψαμε. Σε αυτό το μάθημα θα χρησιμοποιήσουμε τον online editor. Από το μενού software επιλέξτε code online (Εικόνα 3.6).

Η επιλογή αυτή ανοίγει τον editor και μπορείτε να ξεκινήσετε να γράφετε κώδικα (Εικόνα 3.7). Ήδη βλέπετε ότι ο κώδικας χρησιμοποιεί χρωματική διαφοροποίηση



στις
διάφορες
λέξεις της
γλώσσας
είτε αυτές
είναι
δεσμευμέ
νες είτε
είναι

μεταβλητές είτε είναι ονομασία συναρτήσεων. Αυτό είναι μία πολύ βολική πρακτική για την ευκρίνεια του κώδικα και την ευκολία στον καθορισμό της λειτουργίας του της αποσφαλμάτωσης κλπ.

3.5 Εισαγωγή στον Προγραμματισμό του Arduino

Ανοίξτε στον editor ένα νέο sketch – Θα παρατηρήσετε τις λέξεις void setup() και loop(), Η setup() είναι μια υπορουτίνα στην οποία γίνονται οι αρχικές δηλώσεις μεταβλητών, βιβλιοθηκών, ο ορισμός των ακροδεκτών ως είσοδος (INPUT) ή έξοδος (OUTPUT) κτλ. Εκτελείται μία φορά στην αρχή της εκτέλεσης του προγράμματος ή όταν πιεστεί το κουμπί reset. Η loop() είναι η υπορουτίνα στην οποία εκτελείται το κύριο σώμα του κώδικα δηλαδή οι διάφορες λειτουργίες του προγράμματος. Εκτελείτε συνέχεια σε κύκλους μέχρι το board να απενεργοποιηθεί. Όταν το board ενεργοποιηθεί ξανά τότε ο κώδικας που περιέχεται στη loop() ξεκινά να εκτελείται ξανά [24]. Η δεσμευμένη λέξη void συναντάται μόνο μπροστά από συναρτήσεις και δείχνει ότι η συνάρτηση δεν επιστρέφει δεδομένα στην συνάρτηση η οποία την κάλεσε. [24]. Μια συνάρτηση είναι και αυτή μία υπορουτίνα και εκτελεί μία ή περισσότερες λειτουργίες που θέλουμε να εκτελεστούν κατά τη λειτουργία του sketch. Έχουν δικιά τους ονομασία και συνοδεύονται από τις παρενθέσεις () και μπορούν να κληθούν σε κάποιο σημείο του sketch όταν χρειάζεται. Φτιάχνουμε δικές μας συναρτήσεις για να κάνουμε τον κώδικα λιγότερο συμπυκνωμένο και σαφή για την λειτουργία του. Ταυτόχρονα περιορίζονται τα λάθη κατά το γράψιμο του κώδικα.

Οι μεταβλητές στον κώδικα Arduino μοιάζουν πολύ με τις μεταβλητές σε μία μαθηματική εξίσωση. Μπορεί να είναι ένα γράμμα, μία λέξη (μη δεσμευμένη, μια πρόταση, ένας (επιτρεπόμενος) χαρακτήρας. Επιλέγουμε το όνομα μιας μεταβλητής με την εύκολη αναφορά στο τι ελέγχει αυτή η μεταβλητή και με το που χρησιμοποιείται αυτή η μεταβλητή κατά νου. Η μεταβλητή ουσιαστικά αντιπροσωπεύει ένα τόπο στην μνήμη όπου αποθηκεύονται δεδομένα είτε εισερχόμενα είτε εξερχόμενα. Όπως για παράδειγμα το σήμα ενός αισθητήρα που θα μεταφραστεί σε μία τάση και μετέπειτα σε μία τιμή (εισερχόμενα δεδομένα) ή η τάση που θα παρέχει ένας ακροδέκτης του board (εξερχόμενα δεδομένα). Στις επόμενες παραγράφους του μαθήματος θα χρησιμοποιήσουμε μια μεταβλητή με το όνομα `variable1`.

Οι κυριότεροι τύποι μεταβλητών [24],[25] που συναντούμε στα sketch είναι οι:

int: Χρησιμοποιείται για ακέραιους αριθμούς τόσο αρνητικούς όσο και θετικούς (-32768 έως 32767). Ο πιο συνηθισμένος τύπος μεταβλητής

long: Προέκταση του τύπου `int` σε μεγαλύτερο εύρος τιμών (-2,147,483,648 έως 2,147,483,647). Μέγεθος 32 bit – 4 byte

short: Ακέραιοι αριθμοί θετικοί και αρνητικοί. Εύρος τιμών από -32,768 έως 32,767. Μέγεθος 16 bits – 2 bytes

float: Χρησιμοποιείται για δεκαδικούς αριθμούς. Εύρος τιμών -3.4028235E+38 έως 3.4028235E+38. Χρησιμοποιεί μνήμη για 32 bits (4 bytes). Οι υπολογισμοί με δεκαδικούς αριθμούς στη γλώσσα Wiring θέλουν προσοχή καθώς τα αποτελέσματα μπορεί να αποκλίνουν σε ακρίβεια [24].

boolean ή **bool:** μεταβλητή Λογικής. Παίρνει 2 τιμές, `true` ή `false`. Είναι προτιμότερο να χρησιμοποιείται η διατύπωση `bool`

byte: Ακέραιος θετικός αριθμός (0 έως 255)

char: Χρησιμοποιείται για μεταβλητές χαρακτήρων. Στην δήλωση της μεταβλητής χρησιμοποιούνται μονά εισαγωγικά αν πρόκειται να καταχωρηθεί ένα και μόνο γράμμα και διπλά αν είναι μια σειρά χαρακτήρων. Μπορούν να γίνουν και αριθμητικές πράξεις με τις `char` μεταβλητές καθώς στο ASCII πρότυπο οι μονοί χαρακτήρες γράμματα A,B,C,D..... αντιστοιχούν σε αριθμούς «'A' + 1 has the value 66, since the ASCII value of the capital letter A is 65» [24]

array: Πίνακας που αποθηκεύει (ομοειδείς) μεταβλητές. Η δήλωση των πινάκων γίνεται με τον εξής τρόπο:

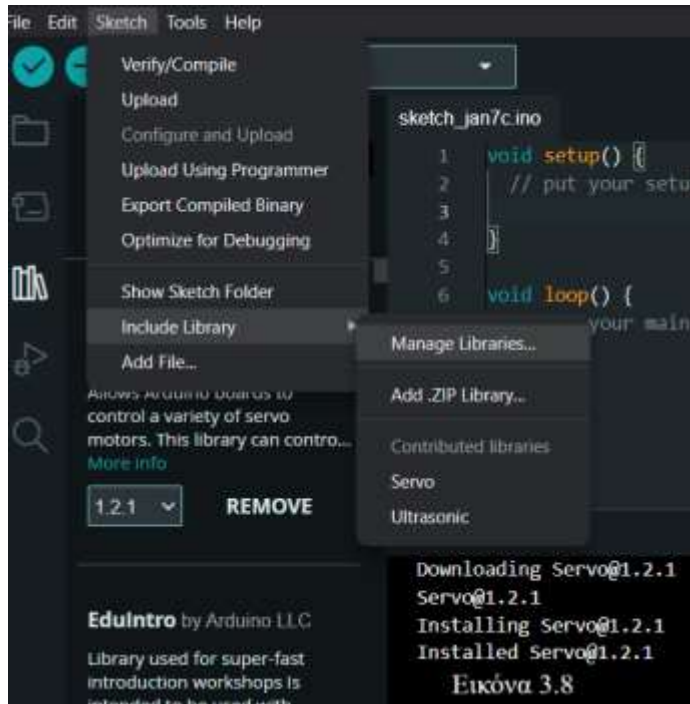
```
«int myArray[10]={9, 3, 2, 4, 3, 2, 7, 8, 9, 11};»[24]
```

Η πρώτη θέση σε ένα πίνακα είναι η 0 και όχι η 1 έτσι όταν θέλουμε να καταχωρήσουμε μία τιμή σε ένα πίνακα στη θέση 0 το κάνουμε με τον εξής τρόπο:

```
MyArray[0]=2
```

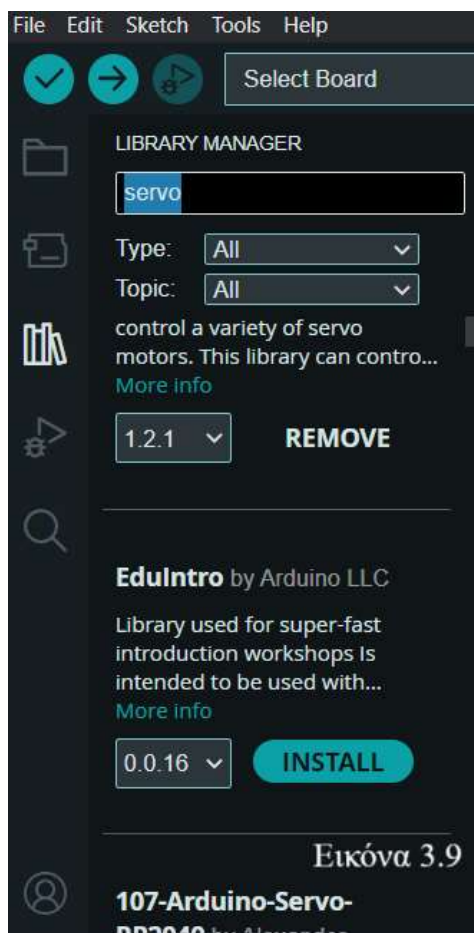
Οι μεταβλητές μπορούν να χωριστούν σε global και local. Οι global μεταβλητές χρησιμοποιούνται από όλο τον κώδικα του sketch ενώ οι local μόνο από τις συναρτήσεις μέσα στις οποίες ορίζονται. Χοντρικά μεταβλητές που ορίζονται δηλαδή έξω από τις συναρτήσεις setup(), loop() και οποιαδήποτε άλλη είναι μεταβλητές global [27]

Οι βιβλιοθήκες είναι στην ουσία έτοιμος κώδικας ενσωματωμένος μέσα στο IDE τον οποίο μπορούμε να επικαλεστούμε στο πρόγραμμα μας για να μας επιτρέψει να εκτελέσουμε κάποια λειτουργία. Αντί να γράφουμε τον ίδιο κώδικα ξανά και ξανά στα sketch μας μπορούμε να χρησιμοποιήσουμε μια έτοιμη βιβλιοθήκη με τον απαραίτητο κώδικα για να εκτελέσει μια λειτουργία. Έτσι δεκάδες ή και εκατοντάδες γραμμές κώδικα μπορούν να αντικατασταθούν από μια γραμμή κώδικα που επικαλείται την κατάλληλη βιβλιοθήκη. Στα επόμενα μαθήματα στο project που θα αναπτύξουμε θα χρησιμοποιήσουμε τις απαραίτητες βιβλιοθήκες για να μπορούμε να χειριστούμε τον αισθητήρα υπερήχων μέτρησης απόστασης HC-SR04 και κινητήρα servo. Στον κώδικα (Sketch) που θα αναπτύξουμε για το project θα χρησιμοποιήσουμε τις βιβλιοθήκες Ultrasonic και Servo. Η βιβλιοθήκη Ultrasonic έχει τον απαραίτητο κώδικα για την λειτουργία του αισθητήρα υπερήχων και η Servo για τον χειρισμό κινητήρα Servo [26]. Όταν αποκτήσουμε άνεση με τον προγραμματισμό Arduino θα είστε ικανοί να δημιουργήσετε δικές σας βιβλιοθήκες.



Για να εγκαταστήσετε μια βιβλιοθήκη ακολουθήστε τα βήματα της Εικόνας 3.8. Ανοίξτε τον editor επιλέξτε Sketch – Include Library – Manage Libraries

Εικόνα 3.8



Στον editor επιλέγουμε Sketch – Include Library – Manage Libraries. Η βιβλιοθήκη είναι έτοιμη για χρήση στο sketch μας.

Εικόνα 3.9

Ας δούμε ορισμένες βασικές εντολές που θα μας χρησιμεύσουν στο project αυτοκινούμενου οχήματος που θα κατασκευάσουμε στα επόμενα μαθήματα
#include <Servo.h> Με αυτή την γραμμή κώδικα συμπεριλαμβάνεται η βιβλιοθήκη Servo που θα μας χρησιμεύσει στον χειρισμό του κινητήρα servo.

int variable1 = 2; Με αυτή την πρόταση η μεταβλητή variable1 δηλώνεται ως ακέραιος (int) και καταχωρείται σε αυτή η τιμή 2

pinMode(variable1,OUTPUT); Από την προηγούμενη δήλωση η τιμή της μεταβλητής είναι 2. Με αυτή τη δήλωση ο ακροδέκτης 2 ορίζεται ως έξοδος – OUTPUT. Το ερωτηματικό στο τέλος της δήλωσης δηλώνει το τέλος της δήλωσης. Η μεταβλητή variable1 στην προηγούμενη δήλωση πήρε την τιμή 2 κατά συνέπεια ο ακροδέκτης 2 της πλακέτας ορίζεται ως έξοδος. Θα μπορούσε να συνδεθεί ένας κινητήρας, ένα buzzer, στοιχεία δηλαδή που εκτελούν κάποια ενέργεια. Οι δηλώσεις αυτές μπαίνουν μέσα στη συνάρτηση setup() [28].

pinMode(variable1,INPUT); Στην πρόταση αυτή η μεταβλητή (δηλαδή ο ακροδέκτης του board) τώρα δηλώνεται ως είσοδος. Στον ακροδέκτη 2 θα μπορούσε να συνδεθεί για παράδειγμα ένας διακόπτης.

Δομές ελέγχου

Σε αυτή την ενότητα θα δούμε κάποιες δομές ελέγχου και επανάληψης της ροής του sketch

```
if (... > 10){  
    .....; //Εκτέλεση εντολών  
    .....;  
}
```

else if (... <= 10)

Η πρόταση αυτή είναι μία δομή ελέγχου της ροής του προγράμματος. Αν η συνθήκη μέσα στην παρένθεση ισχύει τότε το πρόγραμμα εκτελεί τις επόμενες εντολές. Αν δεν ισχύει το πρόγραμμα προχωράει στον έλεγχο της συνθήκης στην πρόταση else if. Προσέξτε ότι οι εντολές δεν βρίσκονται στην ίδια ευθεία με την δεσμευμένη λέξη if. Αυτή η πρακτική διευκολύνει την κατανόηση του προγράμματος. Προσέξτε επίσης

ότι οι εντολές περιέχονται σε brackets { }. Πρέπει να είστε προσεκτικοί στην τοποθέτηση των εντολών μέσα σε brackets καθώς μη σωστή τοποθέτηση τους ή παράληψη τους είναι συντακτικό λάθος και δεν προχωράει η εκτέλεση του κώδικα. Όταν πατήσετε μεταγλώττιση θα εμφανιστεί μήνυμα λάθους. Η δομή αυτή είναι μία εξέλιξη της δομής **if** όπου η συνθήκη μέσα στην παρένθεση ελέγχεται αν είναι αληθής ή όχι και προχωράει στην εκτέλεση των εντολών ή όχι. Η if else if δίνει την δυνατότητα ελέγχου πολλαπλών συνθηκών και σεναρίων δράσης.

switch (μεταβλητή) {

case τιμή1:

 εντολές

break;

case τιμή2:

 εντολές

break;

}

Στην δομή αυτή η μεταβλητή είναι κάποιος τύπος ακεραίας μεταβλητής int, long κτλ. Αν η μεταβλητή είναι ίση με κάποια από τις τιμή1 ή τιμή2 τότε εκτελούνται οι αντίστοιχες εντολές. Η εντολή **break;** σταματάει τη ροή του προγράμματος και συνεχίζει έξω από τη δομή switch break.

Δομές επανάληψης

Οι δομές επανάληψης είναι και αυτές δομές ελέγχου με τη διαφορά ότι οι εντολές που περιέχουν στα brackets τους εκτελούνται επαναλαμβανόμενα για συγκεκριμένο ή αόριστο αριθμό επαναλήψεων. Θα δούμε δυο από τις πιο χαρακτηριστικές δομές επανάληψης

while (συνθήκη) {

.....

}

Αν και όσο ικανοποιείται η συνθήκη (boolean) οι εντολές εντός των bracket εκτελούνται. Αν δεν ικανοποιείται η συνθήκη τότε το πρόγραμμα προχωρά στις επόμενες γραμμές κώδικα.

Μια παραλλαγή της while είναι η:

```
do {  
    .....  
}  
while (συνθήκη);
```

Το πρόγραμμα εκτελεί τις εσωτερικές γραμμές εντολών μέχρι να μην ικανοποιείται μια λογική συνθήκη όπως ένας μετρητής να ξεπεράσει μία τιμή. Οι εντολές θα εκτελεστούν τουλάχιστον μία φορά αφού η συνθήκη θα ελέγχεται αν είναι true ή false στο τέλος της δομής [24]

```
for (int i = 0; i <= 255; i++) {  
    .....;  
}
```

Αυτή η δομή εκτελεί τις εντολές που περιέχονται στα brackets για όσες επαναλήψεις ορίζει ο μετρητής μέσα στην παρένθεση. Εδώ ο μετρητής i ξεκινάει από το 0 μέχρι το 255 με βήμα 1 δηλαδή οι εντολές εκτελούνται 256 φορές. Παρατηρήστε ότι μπορούμε να δηλώσουμε την μεταβλητή int εντός της παρένθεσης της δομής for.

delay(1000);

Η εντολή αυτή καθυστερεί την ροή εκτέλεσης του κώδικα για χρόνο όσο το όρισμα μέσα στην παρένθεση. Εδώ το 1000 σημαίνει 1000 millisecond (1 δευτερόλεπτο)

digitalWrite(ακροδέκτης, HIGH ή LOW)

Η εντολή αυτή δέχεται ως πρώτο όρισμα τον αριθμό κάποιου ακροδέκτη (pin) και θέτει τον ακροδέκτη σε τιμή HIGH 5Volt ή LOW 0 Volt. Σε αυτά τα μαθήματα και στο project θα έχουμε δηλώσει τον συγκεκριμένο ακροδέκτη ως OUTPUT με την εντολή pinMode()

digitalRead(ακροδέκτης)

Η εντολή αυτή δέχεται ως όρισμα τον αριθμό κάποιου ψηφιακού pin και επιστρέφει την τιμή είτε HIGH 5Volt είτε LOW 0 Volt

analogWrite(ακροδέκτης, τιμή)

Η εντολή αυτή αφορά τους ψηφιακούς ακροδέκτες με δυνατότητα PWM (3,5,6,9,10,11 για το board UNO). Το πρώτο όρισμα είναι ο αριθμός του ακροδέκτη. Οι τιμές που μπορεί να λάβει η τιμή είναι ακέραιοι θετικοί από 0 έως 255 με το 0 να είναι η μη λειτουργία και το 255 η συνεχής λειτουργία. Η εντολή αυτή δεν αφορά τους αναλογικούς ακροδέκτες A0-A5.

analogRead(ακροδέκτης)

Αφορά τους αναλογικούς ακροδέκτες A0 - A5 και ως όρισμα δέχεται τον αριθμό του ακροδέκτη (ή τη μεταβλητή που αντιστοιχεί στον ακροδέκτη). Το board μετατρέπει το αναλογικό σήμα (τάση από 0 – 5 Volt) σε ακέραιες μονάδες - ψηφία από 0 (0V) – 1023(5V). Η κάθε μονάδα αντιστοιχεί δηλαδή σε $5/1024=0,0049$ V ή 4,9 mV [24].

```
void strofiaristera(){  
}
```

Αυτό το παράδειγμα αφορά το πώς κωδικοποιείται μια συνάρτηση – υπορουτίνα μέσα στο sketch. Φυσικά οι μεταβλητές έχουν δηλωθεί σε κάποιο σημείο του Sketch. Η κλήση της υπορουτίνας μπορεί να γίνει σε οποιοδήποτε σημείο με την αναφορά του ονόματος της υπορουτίνας με την απλή αναφορά του ονόματος της συνάρτησης. Σε αυτό το παράδειγμα: strofiaristera()

Λογικοί Τελεστές

Οι λογικοί τελεστές στην γλώσσα προγραμματισμού των sketch του Arduino χρησιμοποιούνται σε δομές ελέγχου ροής του προγράμματος όπως η for, η while κλπ. Η τιμή που μπορούν να πάρουν είναι FALSE ή TRUE. Ας δούμε τον καθένα ξεχωριστά. Υποθέτουμε ότι μέσα στο όρισμα μίας δομής if else συγκρίνονται με κάποιο λογικό τελεστή ορισμένες προτάσεις όπως η σύγκριση των μεταβλητών a , b, c μεταξύ τους

&& (AND) - Ελέγχει αν ικανοποιούνται (TRUE) όλες οι συνθήκες μίας λογικής πρότασης [24] Παράδειγμα: (a>b) && (b<c)

|| (OR) – Ελέγχει αν τουλάχιστον μία από τις προτάσεις είναι TRUE [24]

Παράδειγμα: $(a > b) \parallel (b < c)$. Αν και οι δύο είναι TRUE τότε επιστρέφει TRUE. Αν και οι δύο είναι FALSE επιστρέφει FALSE

! (NOT) – Ελέγχει αν η πρόταση είναι FALSE. Αν είναι FALSE επιστρέφει TRUE. Αν η πρόταση είναι TRUE επιστρέφει FALSE. Παράδειγμα: $!(a > b)$ [24]

Τελεστές Σύγκρισης

Οι τελεστές σύγκρισης στην γλώσσα προγραμματισμού των sketch του Arduino χρησιμοποιούνται σε δομές ελέγχου ροής του προγράμματος όπως η for η while κλπ. Η τιμή που μπορούν να πάρουν είναι FALSE ή TRUE. Ας δούμε τον καθένα ξεχωριστά

== Έλεγχος ισότητας (να μην συγχέεται με το = που είναι καταχώρηση τιμής σε μία μεταβλητή)

!= Έλεγχος μη ισότητας

> Μεγαλύτερο από

< Μικρότερο από

>= Μεγαλύτερο ή ίσο από

<= Μικρότερο ή ίσο από

Μαθηματικοί Τελεστές

% Ακέραιο Υπόλοιπο. Δίνει το ακέραιο υπόλοιπο μιας διαίρεσης. Στην διαίρεση $4/3$ θα δώσει 1, $8/4$ θα δώσει 0 κοκ. Ο διαιρέτης θα πρέπει να είναι int και μη μηδενικός ενώ ο διαιρετέος μπορεί να είναι int, float, double [24]

/ Διαίρεση

* Πολλαπλασιασμός

- Αφαίρεση

+ Πρόσθεση

= Καταχώρηση τιμής σε μεταβλητή (όχι σύγκριση)

Υλικό για μελέτη στο σπίτι

Παρακολουθήστε τα ελληνόφωνα βίντεο #1 - #8 και #12-#14 σχετικά με το Arduino

<https://www.youtube.com/watch?v=SIU7ADUIgtI&list=PLw-emLuQ05QZJ3nSWVp4I8IMEQBYSgAxF>

Κατατοπιστικό αγγλόφωνο site σχετικά με το Arduino

<https://learn.sparkfun.com/tutorials/what-is-an-arduino/all>

Παρουσίαση των κυριότερων αισθητήρων του Arduino

<https://www.thegeekpub.com/wiki/list-of-arduino-sensors-and-modules/>

Οδηγός Arduino για μαθητές γυμνασίου

<http://3gym-igoum.thesp.sch.gr/images/Arduino/book.pdf>

Ερωματολόγιο

1. Τι είδους δηλώσεις γίνονται στις υπορουτίνες setup() και loop(); Να αναφέρετε κάποιες εντολές ή γραμμές κώδικα που θα εντάσσαμε στην κάθε υπορουτίνα.
2. Ποια η σημασία της δεσμευμένης λέξης void που συναντάμε σε ένα sketch;
3. Οι δεσμευμένες λέξεις της γλώσσας του Arduino είναι case sensitive ή όχι; Να ερευνήσετε το ερώτημα και να απαντήσετε ανάλογα.
4. Μπορούμε να χρησιμοποιήσουμε την εντολή analogWrite() με όρισμα κάποιον από τους αναλογικούς ακροδέκτες A0-A5;
5. Οι αναλογικοί ακροδέκτες μπορούν να χρησιμοποιηθούν ως ψηφιακοί; Αν ναι με ποιες εντολές μπορούμε να χρησιμοποιήσουμε τους αναλογικούς ακροδέκτες ως ψηφιακούς;
6. Τι τύπο μεταβλητής θα χρησιμοποιούσατε για να καταχωρούνται σε αυτή τόσο θετικές όσο και αρνητικές τιμές;
 - i) Int
 - ii) Byte
 - iii) boolean

ΚΕΦΑΛΑΙΟ 4

ΜΑΘΗΜΑ 4^ο : Εισαγωγή στον προγραμματισμό του Arduino – Εισαγωγή στο Project

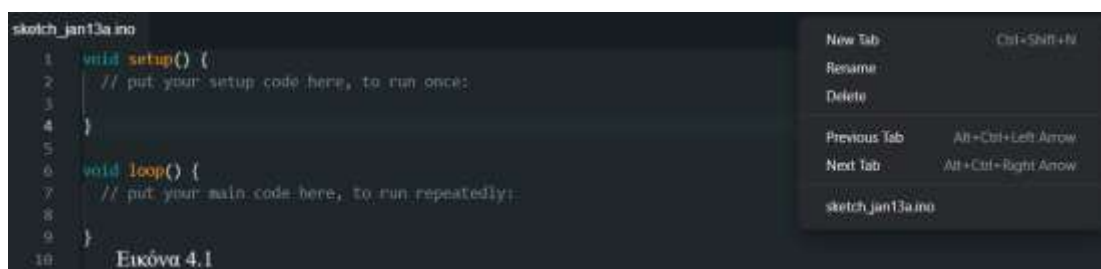
Στο μάθημα αυτό θα εργαστούμε σε δύο διαφορετικές ενότητες:

1. Στην ενότητα 4.1 η κάθε ομάδα θα γράψει τον κώδικα για δυο απλά προγράμματα και θα τα ανεβάσει στο board του kit μέσω του editor που είναι εγκατεστημένος στον υπολογιστή στον οποίο εργάζεται η κάθε ομάδα
2. Στην ενότητα 4.2 η κάθε ομάδα θα αναγνωρίσει τα κομμάτια που περιέχονται σε κάθε kit και θα ενημερωθεί για τη λειτουργία τους. Οι υποομάδες κάθε ομάδας θα αναλάβουν πρωτοβουλία για τον σχεδιασμό, τον προγραμματισμό και την υλοποίηση της κατασκευής του οχήματος

4.1 Προγραμματίζοντας τον Arduino

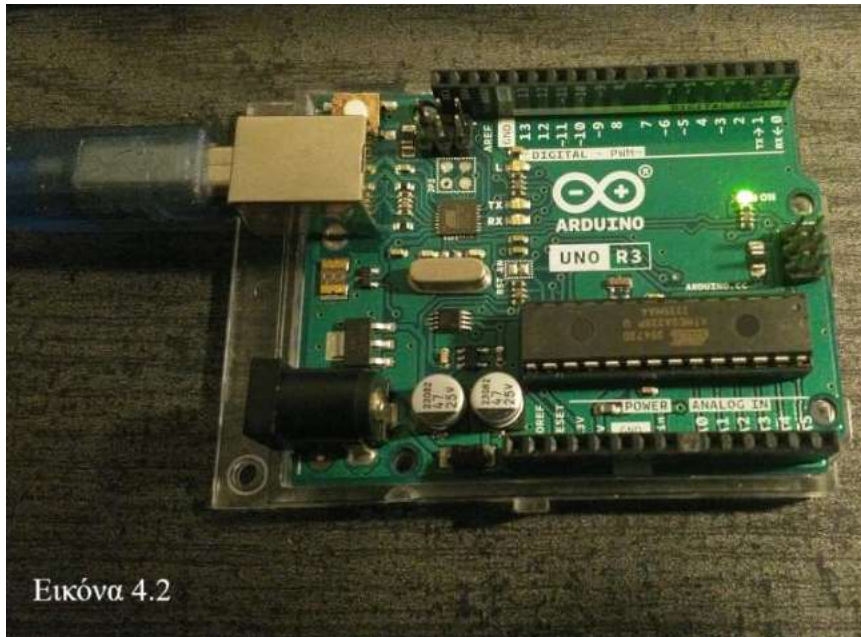
“Hello World”

Το πιο απλό πρόγραμμα το οποίο μπορούμε να γράψουμε στον editor είναι το Hello World. Αυτό το sketch απλά θα εμφανίσει το λεκτικό Hello World στο display του IDE.



Ανοίξτε τον editor (Εικόνα 4.1). Τα σχόλια που βλέπουμε να ακολουθούν τους χαρακτήρες // αγνοούνται από τον compiler του IDE και είναι πολύ χρήσιμα στο να επεξηγούν τα διάφορα κομμάτια κώδικα. Χρησιμοποιούμε τα // για σχόλια μίας γραμμής και /* */ για σχόλια περισσότερων γραμμών (τα σχόλια είναι ενδιάμεσα στα δύο σύμβολα). Από το εικονίδιο ... πάνω δεξιά μπορούμε να αλλάξουμε όνομα στο sketch. Ας το ονομάσουμε Hello_World. Τα αρχεία κώδικα Arduino δεν μπορούν να

έχουν κενά στο όνομα τους και έχουν την κατάληξη .ino. Το αρχείο θα αποθηκευτεί αυτόματα σε φάκελο που θα δημιουργηθεί επίσης αυτόματα από το IDE στον υπολογιστή σας.



Εικόνα 4.2

Επόμενο βήμα είναι η σύνδεση του board στον υπολογιστή.

Αφού συνδέσουμε το βύσμα του καλωδίου USB στο βύσμα του board και το βύσμα USB του καλωδίου στη

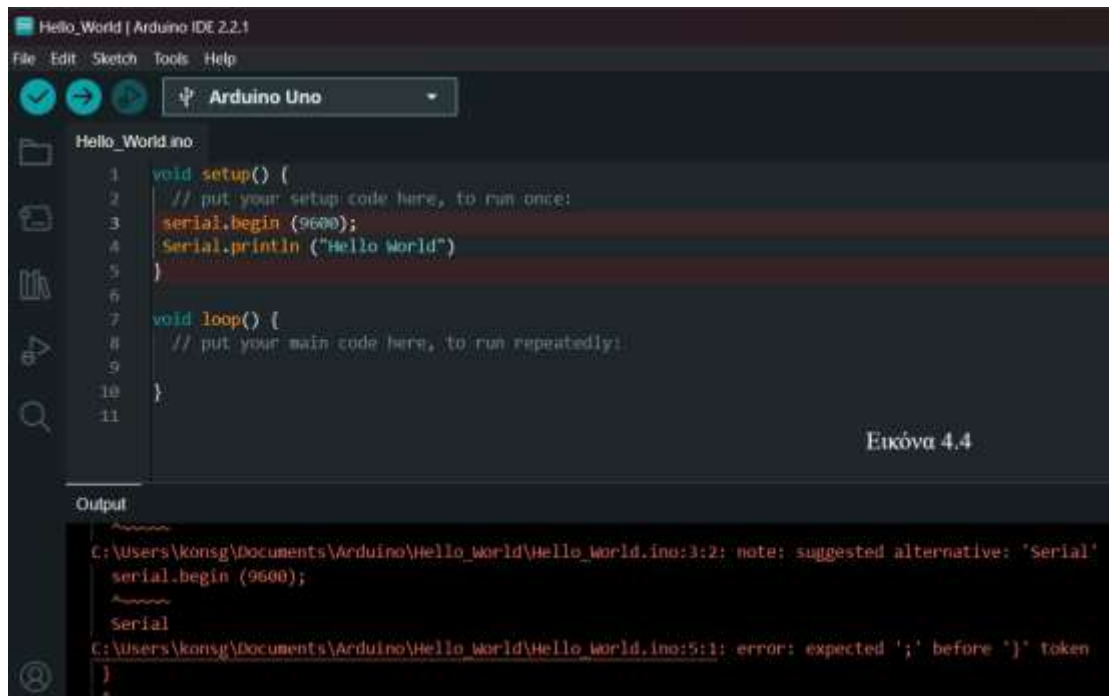
αντίστοιχη θύρα του υπολογιστή το board τίθεται σε λειτουργία όπως βλέπουμε και από το ενσωματωμένο αναμμένο LED του board (ON) στην Εικόνα 4.2. Το board είναι έτοιμο για «ανέβασμα» του κώδικα.




Εικόνα 4.3

Από το παράθυρο Select Board επιλέγουμε το board μας που το IDE αναγνώρισε αυτόματα (Εικόνα 4.3). Θα μπορούσαμε να είχαμε περισσότερα από ένα board συνδεδεμένα στον

υπολογιστή με το κάθε board να εμφανίζεται στην ανάλογη θύρα (COM).




Γράψτε τον κώδικα ακριβώς όπως τον βλέπετε στην Εικόνα 4.4 και πατήστε το πλήκτρο  Verify. Η διαδικασία αυτή είναι γνωστή ως compiling – μεταγλώττιση και εντοπίζει λάθη στον κώδικα. Ο IDE εντόπισε 2 λάθη και τα επισημαίνει τόσο με highlight στη γραμμή που έγινε το λάθος όσο και στην επεξήγηση στο παράθυρο Output. Στην Εικόνα 4.4 βλέπετε ότι υπάρχουν 2 λάθη. Στην σειρά 4 (αρίθμηση γραμμών από αριστερά) λείπει ένα ; και στη σειρά 3 η λέξη serial είναι γραμμένη λάθος. Η λέξη θα έπρεπε να είναι γραμμένη Serial. Η γλώσσα Wiring είναι case sensitive, πρέπει να προσέχετε την σύνταξη με κεφαλαία ή μικρά όταν γράφετε κώδικα. Μέχρι να γίνουν οι διορθώσεις ο κώδικας δεν φορτώνεται στο board.

Σημείωση: Ένα συντακτικά ορθό sketch δεν σημαίνει ότι θα εκτελεί τις λειτουργίες που επιθυμούμε επακριβώς.

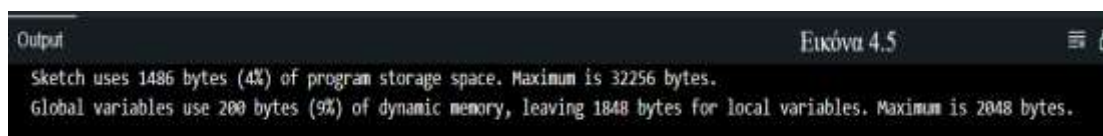
Η εντολή Serial.begin(9600) καθορίζει τον ρυθμό επικοινωνίας (σειριακή) μεταξύ του υπολογιστή και του board. Η επικοινωνία αυτή καθορίζεται σε bits/second (baud rate). Υπάρχει η δυνατότητα και για άλλους ρυθμούς σειριακής επικοινωνίας [29].


Η εντολή Serial.println() εμφανίζει στο Serial Monitor μία φράση ή την τιμή μιας μεταβλητής. Μια παραλλαγή είναι η Serial.print() που εκτυπώνει στην οθόνη σε μία γραμμή ενώ η Serial.println() εκτυπώνει τα ορίσματα της παρένθεσης σε μία γραμμή και η επόμενη εκτύπωση όταν ξανακληθεί η εντολή θα γίνει σε επόμενη γραμμή.

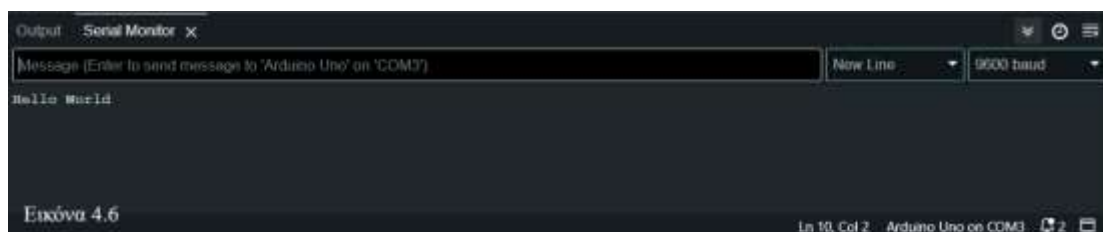
Σημειώστε ότι ένα λεκτικό για να εμφανιστεί στην οθόνη πρέπει να είναι μέσα σε διπλά εισαγωγικά “ ” ενώ η τιμή μίας μεταβλητής όχι. Η εμφάνιση λεκτικού και μεταβλητής απαιτεί το + ανάμεσα τους (“ Hello World ” + variable). Η εντολή Serial.print(var, format) εμφανίζει την μεταβλητή var σε αριθμητικό σύστημα της επιλογής μας (δυναδικό - BIN, δεκαδικό - DEC, οκταδικό - OCT κτλ) [30].

Το πλήκτρο  φορτώνει τον κώδικα στο board και το sketch εκτελείται.

Όταν δεν υπάρχει κάποιο λάθος στον κώδικα εμφανίζεται το κείμενο της Εικόνας 4.5



Πατώντας το πλήκτρο  εμφανίζεται η σειριακή οθόνη στο κάτω μέρος του IDE



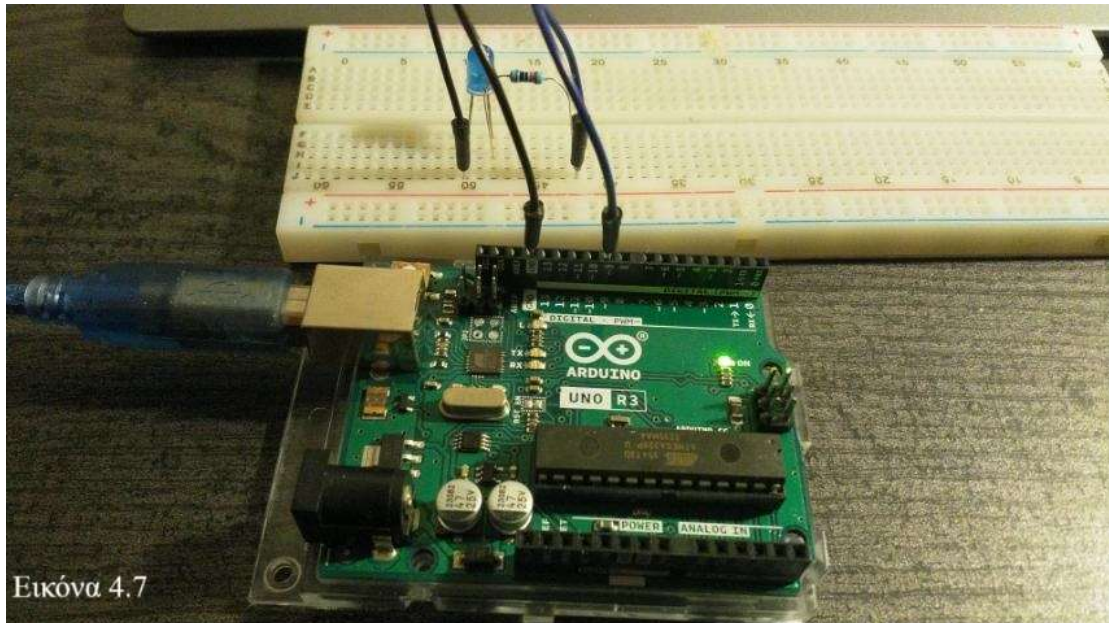
Ολοκληρώσατε το πρώτο επιτυχημένο sketch.

Αυξομείωση έντασης LED

Στο sketch που θα γράψετε θα αυξομειώσετε την ένταση του LED του kit σας. Θα δείτε τη λειτουργία της τεχνικής PWM, της δομής for και της εντολής analogWrite() στη πράξη.

Για την κατασκευή του κυκλώματος θα χρειαστούμε:

1. Board
2. Καλώδιο USB
3. Raster
4. Αντίσταση 220 Ohm
5. LED
6. Καλώδια σύνδεσης



Εικόνα 4.7

Υλοποιήστε το κύκλωμα της Εικόνας 4.7

Παρατηρήστε ότι:

Η κάθοδος του LED συνδέεται με τον ακροδέκτη GND του board

Η άνοδος του LED συνδέεται με την αντίσταση προστασίας των 220 Ohm και με τον ακροδέκτη 9 του board που έχει δυνατότητα PWM. Ένας ακροδέκτης χωρίς τη δυνατότητα PWM δεν θα μπορούσε να αυξομειώσει την ένταση του LED καθώς δέχεται δύο μόνο τιμές HIGH – ON και LOW – OFF.

Συνδέστε και τον απαραίτητο αντιστάτη των 220 Ohm για την προστασία του LED.[31],[32]

```

led_fading.ino
1  int akrlEd = 9;
2  int fading;
3  //στη γραμμή 1 ορίζεται η μεταβλητή akrlEd ως int και καταχωρούμε την τιμή 9 η οποία σε επόμενη δήλωση θα αποτελέσει την pin 9
4  //στη γραμμή 2 ορίζεται η μεταβλητή fading που είναι το μέγεθος που θα αυξομειώνει την ένταση του LED σταδιακά κατά μία σταθερή ποσότητα */
5  void setup() {
6    pinMode(akrlEd, OUTPUT); //ορίζουμε την pin 9 ως OUTPUT
7  }
8
9
10 void loop() { // κύριο σώμα του κώδικα που θα εκτελείται συνεχώς
11   for (fading = 0 ; fading <= 255 ; fading += 9 ) { // η μεταβλητή fading πηγαίνει από την τιμή 0 έως 255 με βήμα 9
12     digitalWrite (9, fading); // η εντολή αυτή ανάβει το LED με την ένταση που ορίζεται από την δομή for
13     delay(50); // η καθυστέρηση αυτή κάνει την μεταβολή της έντασης πιο αμείλιχτη και εύκολα παρατηρήσιμη
14   }
15
16   for (fading = 255 ; fading >= 0 ; fading -= 9) { // στην δομή αυτή γίνεται η αντίστροφη διαδικασία από την προηγούμενη δομή
17     digitalWrite (9, fading);
18     delay(50);
19   }
20 }

```

Εικόνα 4.8

Ο κώδικας για το sketch είναι στην Εικόνα 4.8.

Μετονομάστε το sketch σε led_fading

Γραμμή 1 ορίζεται η μεταβλητή `arkled` ως `int` και καταχωρούμε την τιμή 9 η οποία σε επόμενη δήλωση θα αποτελέσει την `pin` 9

Γραμμή 2 ορίζεται η μεταβλητή `fading` που είναι το μέγεθος που θα αυξομειώνει την ένταση του LED σταδιακά της μία σταθερή ποσότητα

Γραμμή 9 ορίζουμε την `pin` 9 ως OUTPUT

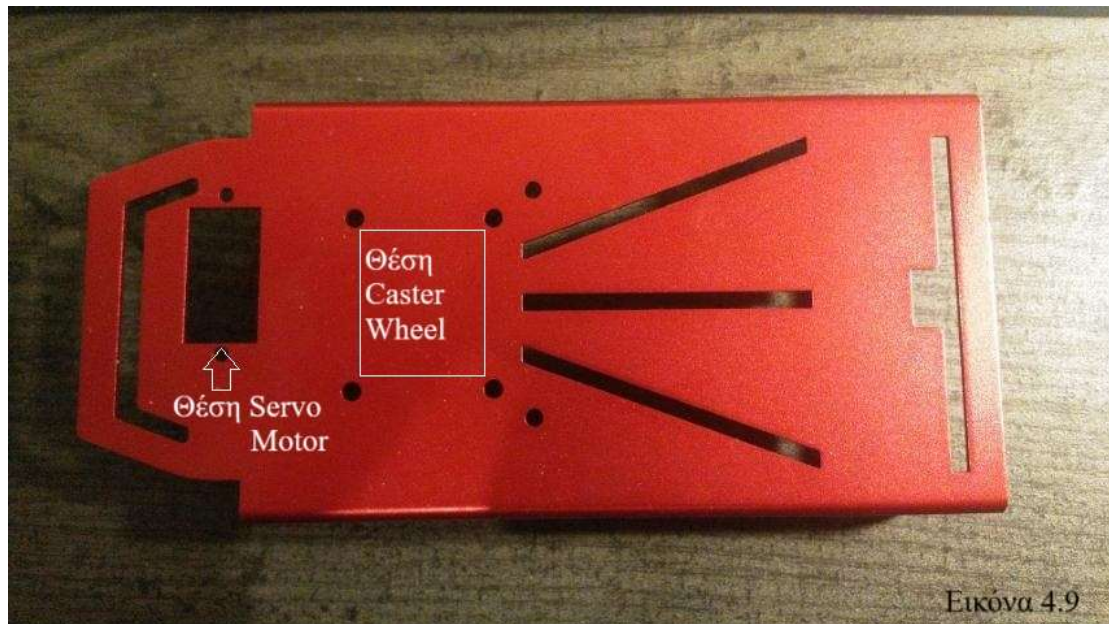
Γραμμή 11 η μεταβλητή `fading` πηγαίνει από την τιμή 0 έως 255 με βήμα 9. Το LED σταδιακά αυξάνει την φωτεινότητα του μέχρι την ανώτερη τιμή.

Γραμμή 12 η εντολή αυτή ανάβει το LED με την ένταση που ορίζεται από την δομή `for`

Γραμμή 13 η καθυστέρηση αυτή κάνει την μεταβολή της έντασης πιο ομαλή και εύκολα παρατηρήσιμη

Γραμμή 16 στην δομή αυτή γίνεται η αντίστροφη διαδικασία από την προηγούμενη δομή. Το LED σταδιακά μειώνει την φωτεινότητα του μέχρι να σβήσει.

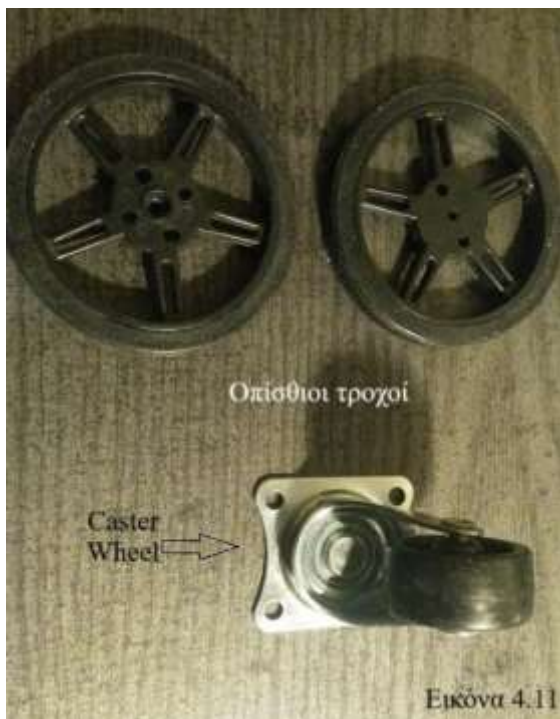
4.2 Το Project Αυτοκινούμενου Οχήματος





Εικόνα 4.10

Στις Εικόνες 4.9 και 4.10 βλέπετε το σασί αυτοκινούμενου οχήματος της FEETECH – Παρατηρήστε τις θέσεις όπου βιδώνουν ο κινητήρας servo – ο εμπρόσθιος τροχός (Caster Wheel) - οι οπίσθιοι τροχοί. Στο kit του σασί περιέχονται ο εμπρόσθιος τροχός (caster) οι οπίσθιοι τροχοί και δύο κινητήρες DC σε μορφή micro servo για τους οπίσθιους τροχούς.

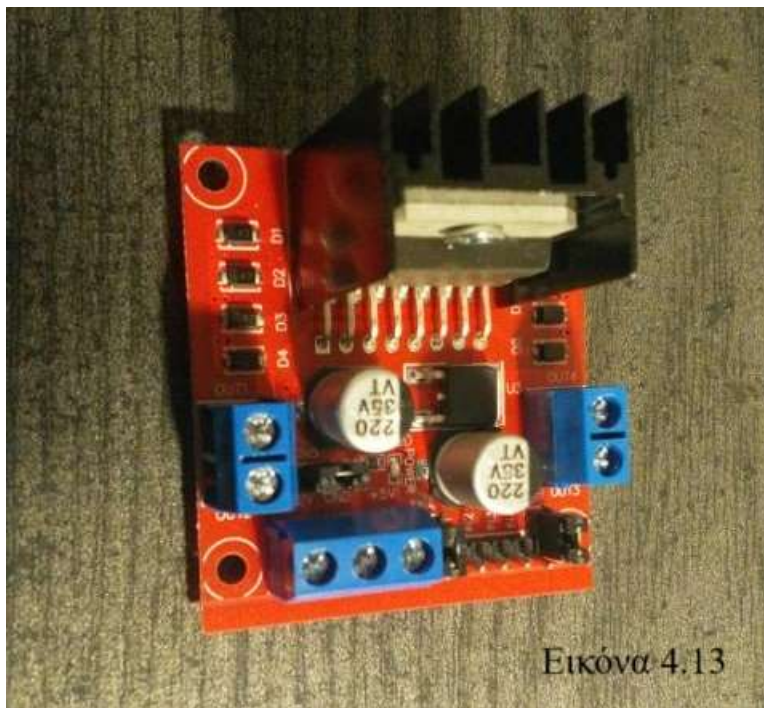


Εικόνα 4.11

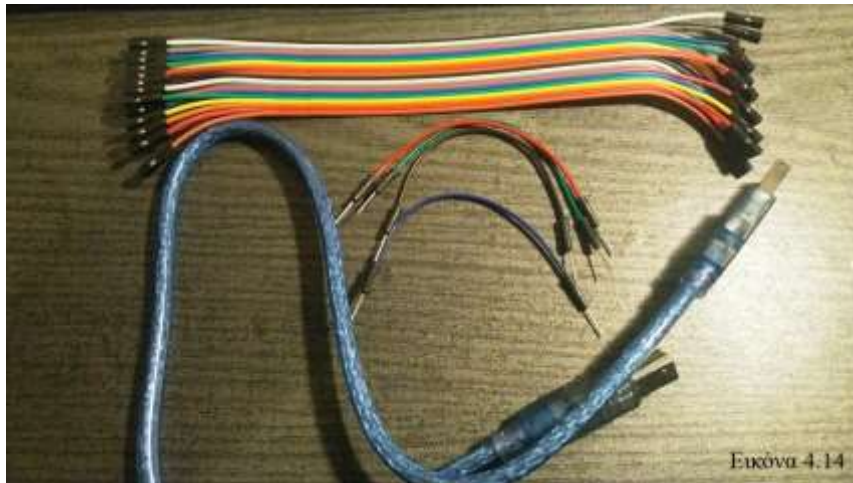
Εικόνα 4.11: Τροχοί που θα χρησιμοποιηθούν στην κατασκευή. Οι οπίσθιοι τροχοί βιδώνουν στους κινητήρες DC και ο εμπρόσθιος βιδώνει στο σασί



Εικόνα 4.12: Κινητήρες DC (κινούν τους οπίσθιους τροχούς) και κινητήρας Servo (κινεί την μονάδα μέτρησης αποστάσεων)

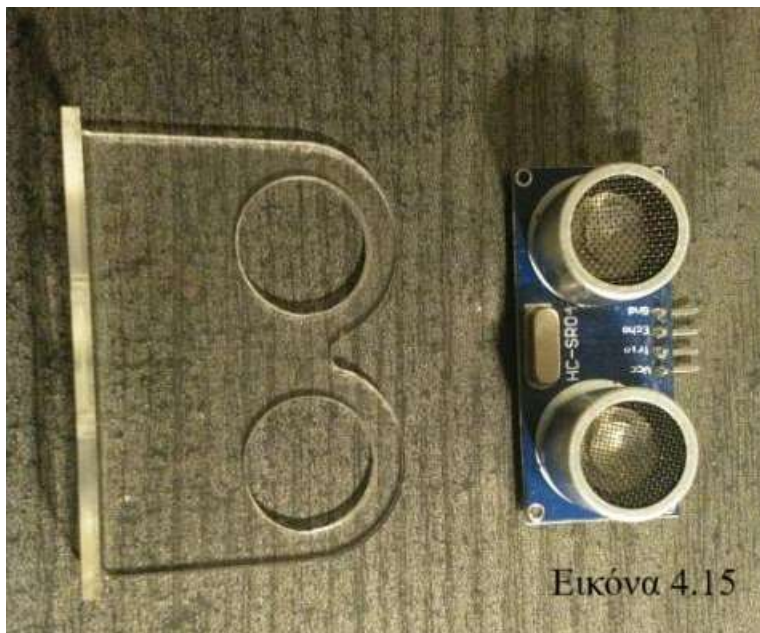


Εικόνα 4.13: Οδηγός L298N για κινητήρες DC της OEM. Έχει τη δυνατότητα να ελέγξει δύο κινητήρες DC. Στην κατασκευή μας θα ελέγχει τους δύο κινητήρες DC που κινούν τους οπίσθιους τροχούς του οχήματος



Εικόνα 4.14:
Καλώδια
σύνδεσης. Κατά
σειρά από άνω
προς κάτω
Θηλυκό σε
θηλυκό
Αρσενικό σε
αρσενικό

Καλώδιο USB για τη σύνδεση του board στον υπολογιστή. Τα θηλυκά καλώδια χρησιμοποιούνται για προέκταση συνδέσεων και ευκολία στην κατασκευή.



Εικόνα 4.15:
Αισθητήρας υπερήχων
και βάση στήριξης του
αισθητήρα. Έχει τη
δυνατότητα μέτρησης
αποστάσεων μέχρι 4
μέτρα. Μαζί με τον
κινητήρα servo θα
αποτελέσει την μονάδα
μέτρησης αποστάσεων.



Εικόνα 4.16: Το board του κιτ μας UNO Rev 3. Η πλαστική βάση χρησιμεύει τόσο για τη στήριξη όσο και για την προστασία του board από βραχυκυκλώματα σε περίπτωση που το κάτω μέρος του ακουμπήσει σε κάτι μεταλλικό.

Εικόνα 4.16



Εικόνα 4.17: Πηγή τάσης για την κατασκευή (μπαταρία 9 V). Θα συνδεθεί στο board μέσω του barrel jack για μπορούμε να αποσυνδέσουμε το καλώδιο USB από το board και το όχημα να μπορέσει να εκτελέσει την κίνηση του ελεύθερα.

Εικόνα 4.17

Περιγραφή του Οχήματος

Στα επόμενα μαθήματα θα κατασκευάσετε και θα προγραμματίσετε όχημα που θα εκτελεί τις εξής λειτουργίες:

- Θα είναι αυτοκινούμενο
- Θα μπορεί να εντοπίζει και να αποφεύγει εμπόδια αλλάζοντας την πορεία του

Το όχημα αυτό θα έχει τα εξής χαρακτηριστικά:

- Οι εντολές που θα προγραμματίσετε θα εκτελούνται από το board του kit της κάθε ομάδας το οποίο θα είναι ενσωματωμένο στο όχημα
- Ο εντοπισμός των εμποδίων θα γίνεται από την μονάδα μέτρησης αποστάσεων η οποία θα μετρά τρεις αποστάσεις (εμπρόσθια, αριστερά, δεξιά) πριν γίνει η επιλογή της νέας πορείας από τον κώδικα που θα αναπτύξουμε
- Ο αισθητήρας υπερήχων θα εκτελεί μετρήσεις στις τρεις αυτές κατευθύνσεις με την τοποθέτηση του σε κινητήρα servo και θα αποτελέσουν την μονάδα μέτρησης αποστάσεων και θα έχουν ξεχωριστή πηγή τροφοδοσίας

Υλικό για μελέτη στο σπίτι

Παρακολουθήστε τα βίντεο #9-#11 και #15-#17 με τις κατασκευές και παράλληλα αναπτύξτε τις στο Tinkercad

<https://www.youtube.com/watch?v=9hFVhGTYB-4&list=PLw-emLuQ05QZJ3nSWVp4I8IMEQBYSgAxF&index=16>

Για πιο προχωρημένα project μπορείτε να διαβάσετε το επίσημο Arduino Project Book

<https://www.uio.no/studier/emner/matnat/ifi/IN1060/v21/arduino/arduino-projects-book.pdf>

Άσκηση 1

Στο sketch Hello World γράψαμε την εντολή `Serial.println("Hello world");` εντός της συνάρτησης `setup()`. Παρατηρήστε ότι η φράση `Hello world` εμφανίστηκε μόνο μία φορά στο `Serial Monitor`

i) Γιατί συμβαίνει αυτό;

ii) Εντός ποιας συνάρτησης θα έπρεπε να γραφτεί η εντολή αν θέλαμε να εκτελείτε συνέχεια;

iii) Να γράψετε κώδικα στο IDE έτσι ώστε η φράση να εμφανίζεται συνέχεια

iv) Αν θέλατε η φράση `Hello world` να εμφανίζεται ανά 1 δευτερόλεπτο ποια εντολή θα χρησιμοποιούσατε; Να γράψετε τον αντίστοιχο κώδικα

Άσκηση 2

Να αναπτύξετε ένα sketch που να ανάβει και να σβήνει εναλλάξ δύο LED. Η εντολή που θα χρησιμοποιήσετε για την λειτουργία των LED είναι η `digitalWrite()` η οποία θα θέτει τα LED σε κατάσταση HIGH ή LOW. Να ρυθμίσετε, κατά την κρίση σας, την διάρκεια που θα είναι τα LED σε κατάσταση OFF ή ON. Να δοκιμάσετε τη λειτουργία του κώδικα σας στο αντίστοιχο κύκλωμα που θα κατασκευάσετε είτε στο Tinkercad είτε στο προσωπικό σας board

Άσκηση 3

Η κάθε ομάδα θα πρέπει να ξεκινήσει την έρευνα για την ανάπτυξη του οχήματος τόσο σε σχεδιαστικό κατασκευαστικό επίπεδο όσο και σε σχεδιαστικό. Μελετήστε τα διαθέσιμα υλικά του kit και σχεδιάστε ένα αρχικό γενικό πλάνο κατασκευής και λειτουργίας. Ο συντονιστής θα πρέπει να φέρει σε επαφή τις ομάδες να αξιολογεί το υλικό που αναζητήσανε και να αναζητήσει και αυτός σχετικό υλικό. Στο διαδίκτυο υπάρχει πληθώρα ανάλογων project με το δικό σας από τα οποία μπορείτε να αντλήσετε ιδέες.

Ενδεικτικά παρατίθενται κάποια παραδείγματα:

<https://www.instructables.com/Arduino-Self-Driving-Car/>

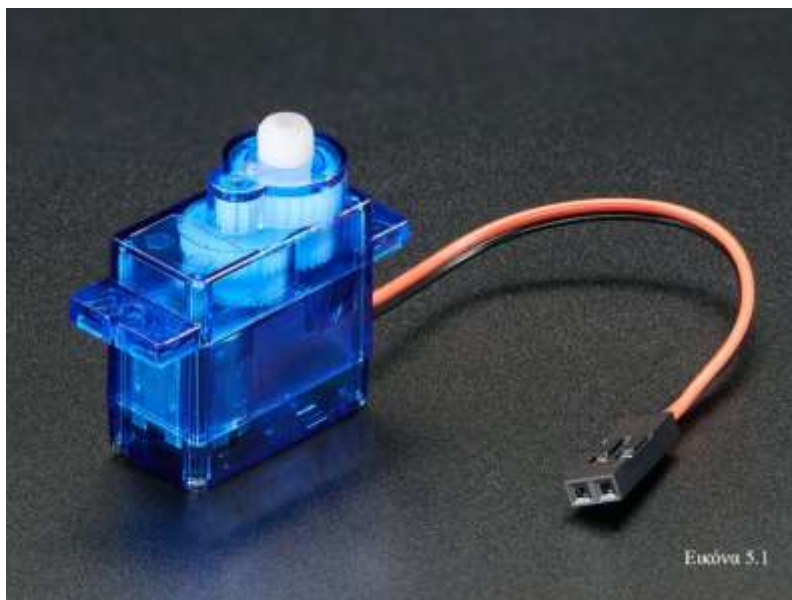
https://www.sciencebuddies.org/science-fair-projects/project-ideas/Robotics_p042/robotics/arduino-self-driving-car

ΚΕΦΑΛΑΙΟ 5

ΜΑΘΗΜΑ 5^ο: Έλεγχος κινητήρων με τον Arduino

- Στο μάθημα αυτό θα ασχοληθούμε με τους 2 τύπους κινητήρων οι οποίοι θα μας χρησιμεύσουν και στην κατασκευή του αυτοκινούμενου οχήματος.
- Θα γνωρίσουμε τους κινητήρες DC (Direct Current) και τους κινητήρες servo και πως μπορούμε να τους ελέγξουμε με τον Arduino και τον οδηγό L298N.
- Θα αναπτύξουμε sketch για να εξοικειωθούμε με την λειτουργία και τις εντολές κίνησης των κινητήρων.
- Οι 2 κινητήρες DC θα χρησιμοποιηθούν για την κίνηση των οπίσθιων τροχών και ο κινητήρας servo για την κίνηση του αισθητήρα υπερήχων που θα «σκανάρει» τον χώρο και θα μετρά αποστάσεις
- Σταδιακά θα αρχίσουμε να κατασκευάζουμε το όχημα μας και να παίρνει μορφή

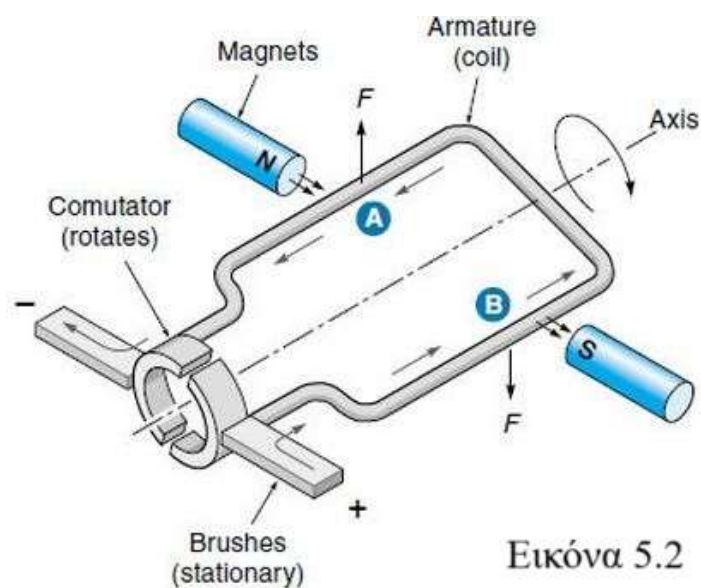
5.1 Κινητήρες DC



Στο όχημα που θα κατασκευάσετε οι κινητήρες DC έχουν την μορφή micro servo (Εικόνα 5.1) αλλά η λειτουργία τους είναι πανομοιότυπη με τον κοινό κινητήρα DC που ξέρετε από την καθημερινότητα σας.

Η τάση λειτουργίας του κινητήρα είναι τα 4-6 Volt. Παρατηρήστε ότι έχει δύο καλώδια που συνδέονται με τους πόλους του μοτέρ. Αυτό σημαίνει ότι η αλλαγή φοράς της περιστροφής του κινητήρα γίνεται με απλή αντιστροφή της πολικότητας της πηγής ή με αντίστροφη σύνδεση των καλωδίων του κινητήρα. Μέγιστη ταχύτητα

περιστροφής οι 130 στροφές ανά λεπτό ανάλογα την τάση τροφοδοσίας του κινητήρα.



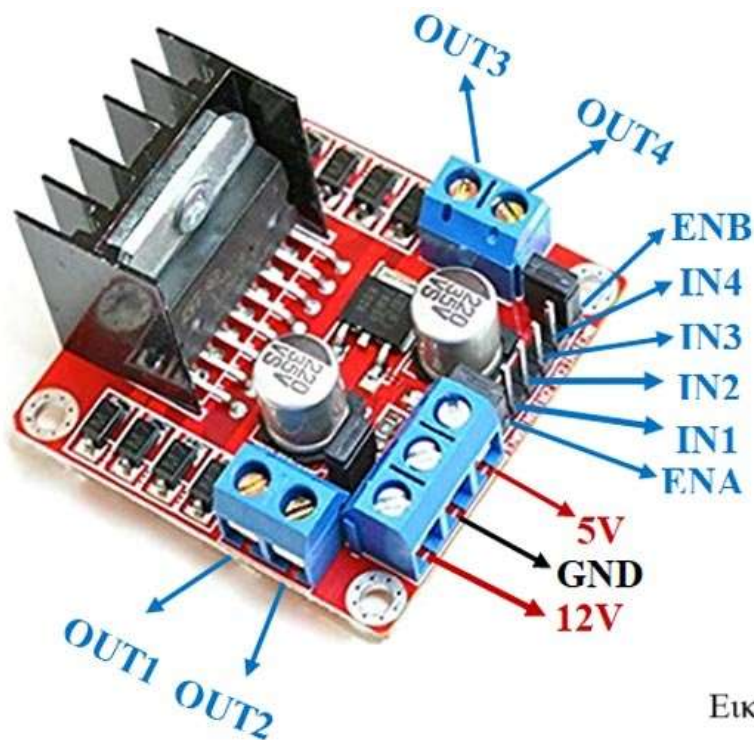
Εικόνα 5.2

Η λειτουργία του κινητήρα DC βασίζεται στη δύναμη Lorentz - η δύναμη δηλαδή που ασκείται σε έναν αγωγό που διαρρέεται από ρεύμα όταν αυτός βρίσκεται μέσα σε ένα πεδίο [33]. Η φορά της δύναμης F φαίνεται στην Εικόνα 5.2 και ασκείται στο πλαίσιο το οποίο προφανώς κινεί τον

άξονα του κινητήρα και αυτός με την σειρά του τον οπίσθιο τροχό. Ο άξονας του κινητήρα DC έχει τη μορφή γραναζιού και φέρει οπή στο κέντρο του όπου βιδώνεται ο τροχός (Εικόνα 5.1). Η φορά της F μπορεί να βρεθεί εμπειρικά με τον κανόνα του αριστερού χεριού - αν ο μέσος δείχνει τη φορά του ρεύματος ο δείκτης τη φορά του μαγνητικού πεδίου ο αντίχειρας θα δείξει τη φορά της δύναμης

Ο οδηγός κινητήρων L289N

Αποφεύγουμε να συνδέσουμε κινητήρα DC απ' ευθείας στο board διότι υπάρχει πιθανότητα ζημιάς στο board. Υπάρχουν τρόποι να ελεγχθεί κινητήρας DC με Arduino με τα κατάλληλα κυκλώματα με τρανζίστορ και δίοδο. Για το όχημα μας θα χρησιμοποιήσουμε τον οδηγό L289N που είναι μια πιο πρακτική λύση για τον έλεγχο των κινητήρων. Με τον οδηγό L289N μπορούμε να ελέγξουμε τόσο την φορά όσο και την ταχύτητα περιστροφής του κινητήρα. Ο οδηγός έχει τη δυνατότητα να ελέγξει ταυτόχρονα δύο κινητήρες DC τάσης 5 – 35 Volt με μέγιστη τιμή ρεύματος 2 Ampere [34].



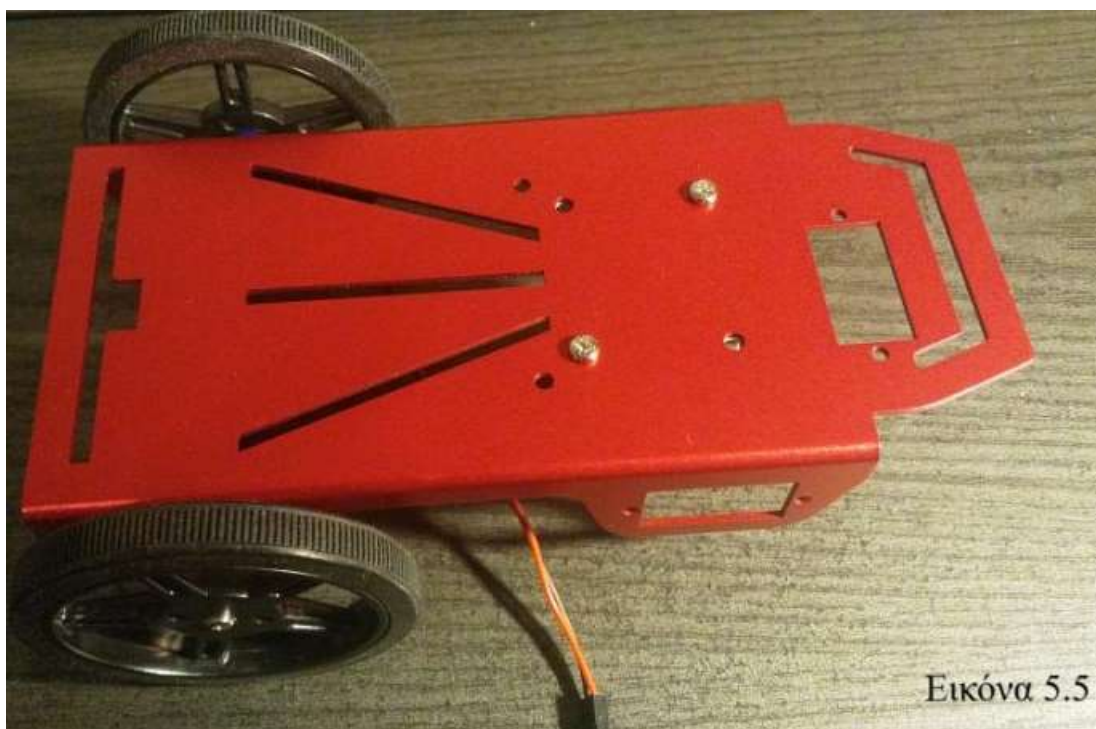
Εικόνα 5.3

Στον οδηγό μπορούμε να συνδέσουμε και τους δυο κινητήρες DC. Τα καλώδια των κινητήρων θα συνδεθούν στους ακροδέκτες OUT1 – OUT2 για τον ένα κινητήρα και OUT3 – OUT4 για τον άλλο. Τα καλώδια των κινητήρων ενθυλακώνονται στις οπές των ακροδεκτών OUT1 έως OUT4 και στερεώνονται με το σφίξιμο της βίδας του κάθε ακροδέκτη. Ορίζουμε την πολικότητα κάθε κινητήρα και τα καλώδια συνδέονται με την σωστή πολικότητα στο σετ OUT1 – OUT2 είτε στο OUT3 – OUT4 ανάλογα με τον τύπο κίνησης που θέλουμε να εκτελέσει το όχημα [35].

Οι ακροδέκτες OUT παρέχουν στους κινητήρες το ρεύμα για να εκτελέσουν κίνηση και μπορούν να ρυθμίσουν την ταχύτητα και τη φορά περιστροφής (αν ο OUT1 είναι ενεργός ο κινητήρας κινείται προς μία κατεύθυνση, αν ο OUT2 είναι ενεργός ο κινητήρας κινείται στην αντίστροφη κατεύθυνση). Μπορούμε να ελέγξουμε την ταχύτητα περιστροφής με την τεχνική PWM και τον αντίστοιχο κώδικα `analogWrite()`. Οι ακροδέκτες IN συνδέουν τον οδηγό με το board και πιο συγκεκριμένα συνδέονται στους ακροδέκτες του board. Καταλαβαίνουμε ότι οι ακροδέκτες IN1 - IN2 αντιστοιχούν στους ακροδέκτες OUT1 – OUT2. Ο ακροδέκτης IN1 δηλαδή θα συνδεθεί με κάποιο pin του board και θα μεταφέρει το σήμα που θα λάβει στον OUT1 για να εκτελέσει ο κινητήρας την εντολή που έχει γραφεί για τον συγκεκριμένο κινητήρα στο sketch.

Οι ακροδέκτες 5V, GND και 12 V αφορούν την τροφοδοσία του οδηγού. Μπορούμε είτε να συνδέσουμε μια πηγή 12 V (ή μικρότερη) στον ακροδέκτη 12V (το ένα καλώδιο στα 12V και το άλλο στο GND) και έτσι να τροφοδοτήσουμε τον οδηγό. Το board σε αυτή τη συνδεσμολογία θα συνδεθεί με καλώδιο από τον ακροδέκτη 5V (σειρά ακροδεκτών POWER) στον ακροδέκτη 5V του οδηγού – με αυτόν τον τρόπο τροφοδοτείται και το board. Τέλος θα συνδεθούν με καλώδιο και οι ακροδέκτες GND οδηγού με GND board (σειρά ακροδεκτών POWER). Ο οδηγός δηλαδή τώρα παρέχει ρεύμα και για τη λειτουργία του board. Μια εναλλακτική συνδεσμολογία μπορεί να εφαρμοστεί όταν προτιμούμε να τροφοδοτήσουμε απευθείας το board και όχι μέσω του οδηγού. Τώρα απλά συνδέουμε με καλώδια GND με GND και 5V με 5V σε board και οδηγό αντίστοιχα. Με αυτή τη συνδεσμολογία είναι το board που παρέχει ρεύμα για τη λειτουργία του οδηγού. Θα εφαρμόσουμε τη δεύτερη συνδεσμολογία στο όχημα μας αφού το board θα έχει δικιά του τροφοδοσία από την μπαταρία των 9 Volt.

Κατασκευή του Οχήματος

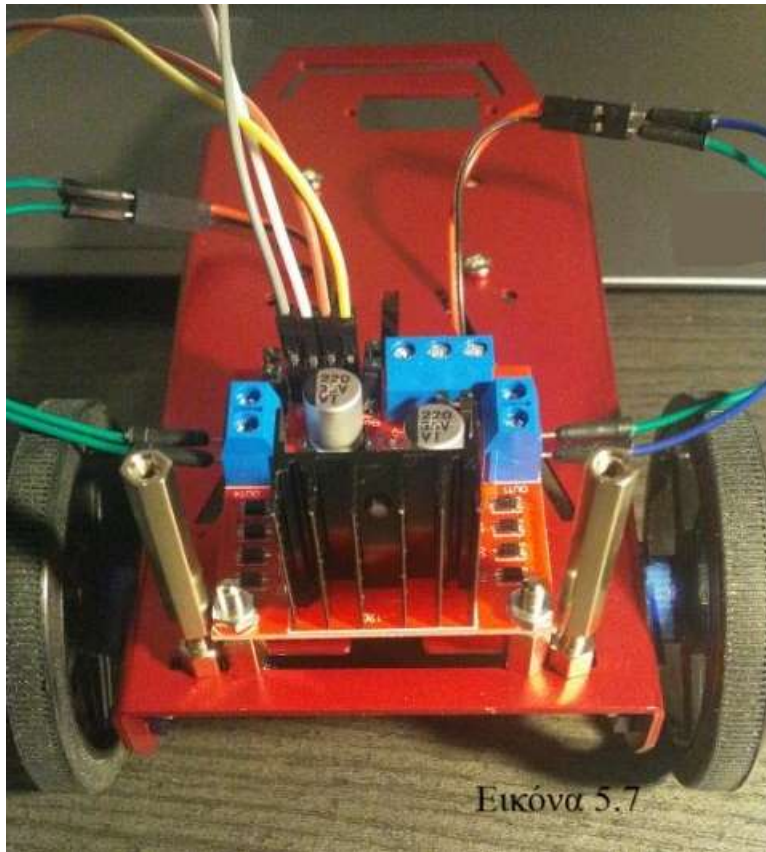


Εικόνα 5.5



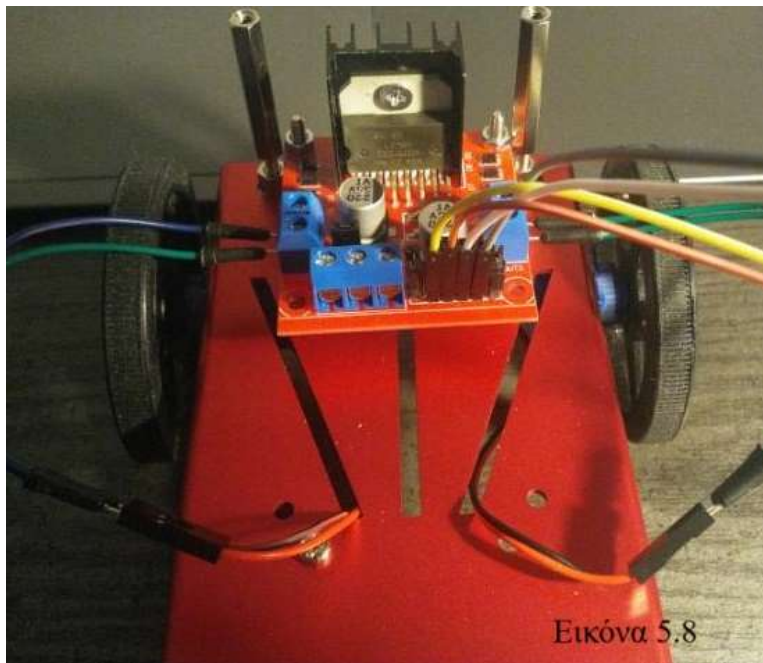
Εικόνα 5.6

Στις Εικόνα 5.5 και Εικόνα 5.6 ξεκινάμε με το πρώτο στάδιο κατασκευής του οχήματος. Οι κινητήρες DC βιδώνονται πάνω στο σασί με βίδες και παξιμάδια στην εσωτερική πλευρά (Εικόνα 5.6) Στη συνέχεια βιδώνονται και οι τροχοί στην οπή του άξονα των τροχών. Παράλληλα βιδώνεται και ο εμπρόσθιος τροχός στο σασί.



Μοντάρετε τον οδηγό στο σασί με τις βίδες τα παξιμάδια και τους μικρούς αποστάτες του κιτ (Εικόνα 5.7). Οι αποστάτες φροντίζουν ο οδηγός να μην ακουμπά στο μεταλλικό σασί και έτσι να αποφεύγονται τα βραχυκυκλώματα. Ο οδηγός όπως και το board έχουν από την κάτω πλευρά τους εκτεθειμένα αγωγία στοιχεία του κυκλώματος τους και γι

αυτό αποφεύγουμε την επαφή με μεταλλικά σώματα. Οι μεγάλοι αποστάτες βιδώνονται και αυτοί δίπλα από τους μικρούς. Θα χρησιμεύσουν για την στερέωση του board. Παράλληλα στους ακροδέκτες IN1 – IN4 τοποθετούνται θηλυκά καλώδια



σύνδεσης για να συνδεθούν αργότερα στα pin του board.

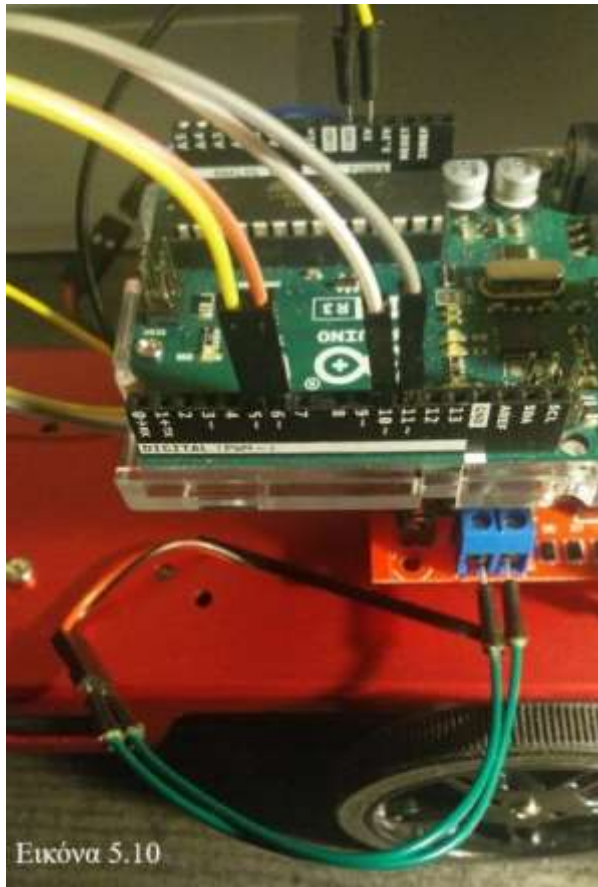
Στην Εικόνα 5.8 βλέπετε τη σύνδεση των ακροδεκτών OUT1-OUT4 με καλώδια σύνδεσης με τους ακροδέκτες των κινητήρων. Συνδέστε τους OUT1 και OUT2 με τον ένα κινητήρα και τους OUT3 - OUT4 με

τον άλλο κινητήρα. Σημειώστε σε ποιον ακροδέκτη συνδέσατε το κόκκινο και μαύρο καλώδιο κάθε κινητήρα. Αυτό θα μας χρησιμεύσει στον προγραμματισμό του

οχήματος έτσι ώστε οι κινήσεις που θα εκτελεί να είναι συντονισμένες και σωστά προγραμματισμένες. Συνδέστε καλώδια στους ακροδέκτες GND και 5V του οδηγού.



Συνδέστε το board στους μεγάλους αποστάτες την τροφοδοσία (μπαταρία 9V) στο βύσμα. Η μπαταρία επικολλάται με ταινία διπλής όψης στο κάτω μέρος του σασί (Εικόνα 5.9).



Εικόνα 5.10

Τα καλώδια συνδέονται κατά τον ακόλουθο τρόπο της Εικόνας 5.10 (δεν είναι ο μοναδικός τρόπος):

Μικρό κίτρινο καλώδιο : Συνδέει τον ακροδέκτη 5V του οδηγού (τροφοδοσία) με τον ακροδέκτη 5V του board

Μικρό μαύρο καλώδιο : Συνδέει τον ακροδέκτη GND του οδηγού με τον GND του board (γείωση)

Κίτρινο καλώδιο : Συνδέει τον IN1 στον ακροδέκτη 5 του board

Πορτοκαλί καλώδιο : Συνδέει τον IN2 στον ακροδέκτη 6 του board

Λευκό καλώδιο : Συνδέει τον IN3 στον ακροδέκτη 10 του board

Γκρι καλώδιο : Συνδέει τον IN4 στον ακροδέκτη 11 του board

Παρατηρήστε ότι οι ακροδέκτες που ελέγχουν τους κινητήρες (5,6,10,11) υποστηρίζουν PWM. Θα δούμε αργότερα ποιες οι δυνατότητες από αυτή την επιλογή.

Προγραμματισμός Βασικών Κινήσεων του Οχήματος

Στα επόμενα θα προγραμματίσουμε το όχημα να εκτελεί κάποιες βασικές κινήσεις (Εμπρός, Πίσω, Αριστερά, Δεξιά). Θέλουμε το όχημα να εκτελέσει κίνηση: Εμπρός για 1 δευτερόλεπτο - Πίσω για 1 δευτερόλεπτο – Αριστερά για 1 δευτερόλεπτο – Δεξιά για 1 δευτερόλεπτο

```

const int kinitiras1_f= 5; // το κόκκινο καλώδιο του κινητήρα 1 (δεξιάς από την οπίσθια πλευρά) συνδέεται στον ακροδέκτη 5
const int kinitiras1_b= 6; // το μαύρο καλώδιο του κινητήρα 1 (δεξιάς από την οπίσθια πλευρά) συνδέεται στον ακροδέκτη 6
const int kinitiras2_f=10; // το μαύρο καλώδιο του κινητήρα 2 (αριστερός από την οπίσθια πλευρά) συνδέεται στον ακροδέκτη 10
const int kinitiras2_f=11; // το κόκκινο καλώδιο του κινητήρα 2 (αριστερός από την οπίσθια πλευρά) συνδέεται στον ακροδέκτη 11
void setup() {
  // put your setup code here, to run once:
  pinMode(kinitiras1_f,OUTPUT); // όλα οι ακροδέκτες των κινητήρων δηλώνονται ως OUTPUT
  pinMode(kinitiras1_b,OUTPUT);
  pinMode(kinitiras2_f,OUTPUT);
  pinMode(kinitiras2_b,OUTPUT);
}

```

Εικόνα 5.11

Οι μεταβλητές `kinitiras1_f`, `kinitiras1_b` κτλ. Δηλώνονται ως `const` (σταθερές) γιατί δεν αλλάζουν μέσα στη ροή του `sketch`. Παράλληλα όλα τα `pin` δηλώνονται ως έξοδοι – OUTPUT. Τα `f` και `b` στις μεταβλητές `kinitiras1` σημαίνουν `forward` και `backward` δηλαδή εμπρός και πίσω για να φανερώσουν την κίνηση που ορίζει το κάθε `pin`. Προσέχουμε να συνδέσουμε την σωστή μεταβλητή με το σωστό `pin` ανάλογα την κίνηση που θέλουμε να εκτελέσει ο κινητήρας. Η μεταβλητή `kinitiras1_b` θέλουμε να κινεί τον κινητήρα 1 προς τα πίσω. Άρα ο ακροδέκτης 6 στον οποίο συνδέεται μέσω του οδηγού η μεταβλητή `kinitiras1_b` θα έχει τιμή HIGH όταν θέλουμε ο κινητήρας 1 να κινηθεί προς τα πίσω. Οι ίδιες αντιστοιχίες ισχύουν και για τις μεταβλητές `kinitiras1_f`, `kinitiras2_f`, `kinitiras2_b` (Εικόνα 5.11).

```
void loop() {
  // put your main code here, to run repeatedly:
  forward();
  delay(1000);
  backward();
  delay(1000);
  left();
  delay(1000);
  right();
  delay(1000);
  stop();
  delay(2000);
}
```

Εικόνα 5.12

Οι συναρτήσεις `forward()` [εμπρόσθια κίνηση], `backward()` [όπισθεν], `left()` [αριστερή επιτόπια στροφή], `right()` [δεξιά

επιτόπια στροφή], `stop()` [στάση] (Εικόνα 5.12) κωδικοποιούνται εκτός της `loop()` στις επόμενες γραμμές. Η `loop()` καλεί την καθεμία για τον χρόνο που ορίζει η κάθε `delay()`.

```

void forward() { // ΕΜΠΡΟΣ
    digitalWrite(kinitiras1_f, HIGH);
    digitalWrite(kinitiras1_b, LOW);
    digitalWrite(kinitiras2_f, LOW);
    digitalWrite(kinitiras2_b, HIGH);
}

void backward() { // ΠΙΣΩ
    digitalWrite(kinitiras1_f, LOW);
    digitalWrite(kinitiras1_b, HIGH);
    digitalWrite(kinitiras2_f, HIGH);
    digitalWrite(kinitiras2_b, LOW);
}

void right() { // ΔΕΞΙΑ
    digitalWrite(kinitiras1_f, LOW);
    digitalWrite(kinitiras1_b, HIGH);
    digitalWrite(kinitiras2_f, LOW);
    digitalWrite(kinitiras2_b, HIGH);
}

```

Εικόνα 5.13

Στην Εικόνα 5.13 βλέπετε τον κώδικα για τις συναρτήσεις που καλούνται στη loop(). Η γεωμετρία και η συνδεσμολογία της κατασκευής μας ορίζουν τον τρόπο με τον οποίο γράφουμε τον κώδικα. Για παράδειγμα, για να πάει το όχημα εμπρός οι δύο κινητήρες πρέπει να λειτουργήσουν με αντίθετη φορά δηλαδή πολικότητα. Στο κύκλωμα μας αυτό σημαίνει ότι στην κίνηση προς τα εμπρός το κόκκινο καλώδιο του κινητήρα 1 τροφοδοτείται με τάση (HIGH) ενώ το μαύρο όχι (LOW).

Στον κινητήρα 2 η τροφοδοσία είναι

αντίστροφη. Σκεφτείτε ότι ενώ σε κάθετη θέση οι κινητήρες (με τροφοδοσία ίδιας πολικότητας) κινούν τον τροχό προς ίδια κατεύθυνση στην οριζόντια θέση με την οποία βιδώνονται στο σασί κινούνται αντίστροφα. Άρα για να κινηθεί το όχημα εμπρός πρέπει οι κινητήρες να κινούνται αντίρροπα. Στο κύκλωμα μας αυτό σημαίνει ότι στην κίνηση προς τα εμπρός το κόκκινο καλώδιο του κινητήρα 1 τροφοδοτείται με τάση (HIGH) ενώ το μαύρο όχι (LOW). Στον κινητήρα 2 η τροφοδοσία είναι αντίστροφη. Στην δεξιά στροφή θέλουμε ο κινητήρας 1 να κάνει κίνηση προς τα εμπρός ενώ ο κινητήρας 2 κίνηση προς τα πίσω. Στην αριστερή στροφή εκτελούνται οι αντίστροφες κινήσεις από αυτές της. Θα παρατηρήσατε ότι χρησιμοποιούμε την

```

void left() { // ΑΡΙΣΤΕΡΑ
    digitalWrite(kinitiras1_f, HIGH);
    digitalWrite(kinitiras1_b, LOW);
    digitalWrite(kinitiras2_f, HIGH);
    digitalWrite(kinitiras2_b, LOW);
}

void stop() { // ΣΤΑΣΗ
    digitalWrite(kinitiras1_f, LOW);
    digitalWrite(kinitiras1_b, LOW);
    digitalWrite(kinitiras2_f, LOW);
    digitalWrite(kinitiras2_b, LOW);
}

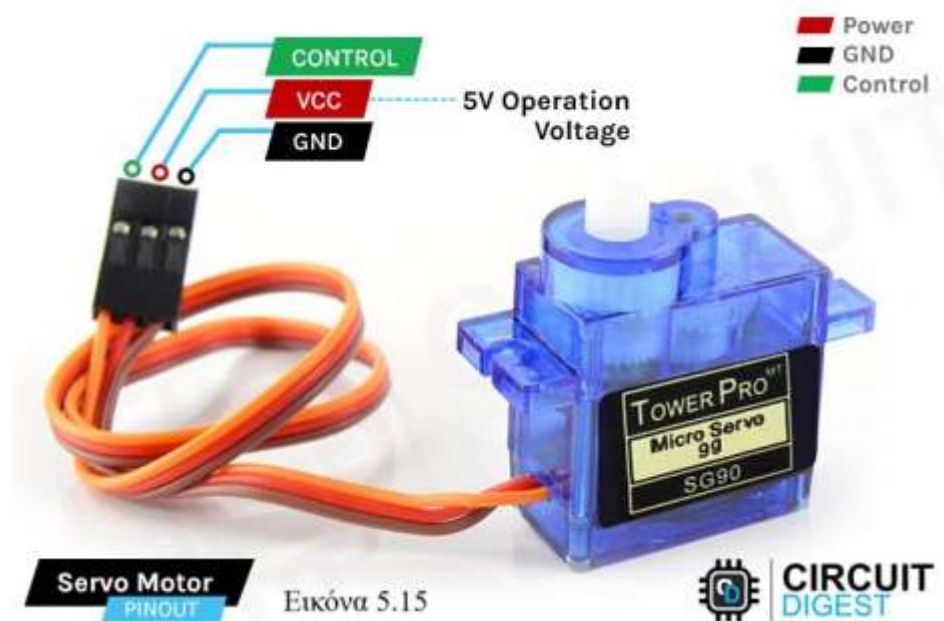
```

Εικόνα 5.14

εντολή digitalWrite(). Θυμηθείτε ότι επιλέξαμε όλα τα pin που χρησιμοποιήσαμε έχουν δυνατότητα PWM άρα μπορούμε να τις χρησιμοποιήσουμε και ως αναλογικές εξόδους και να χρησιμοποιήσουμε την εντολή analogWrite(). Η analogWrite() μας επιτρέπει να ελέγξουμε την ταχύτητα του

κινητήρα και να έχουμε μία πιο ομαλή κίνηση (ταχύτητα πχ 150 αντί για 255-HIGH που είναι η maximum τιμή). Στο project μας θα χρησιμοποιήσουμε την digitalWrite() (Εικόνα 5.14).

5.2 Κινητήρες Servo

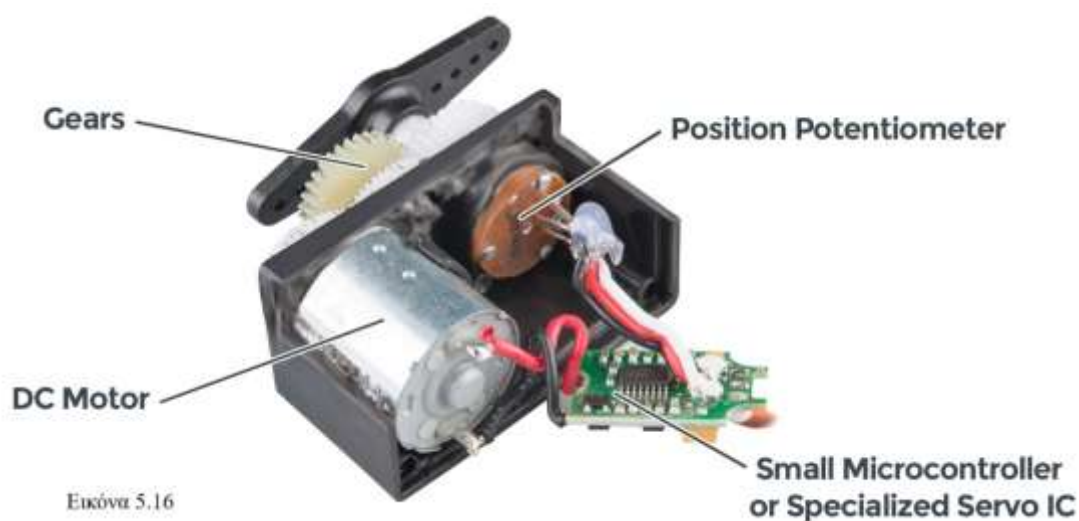


Εικόνα 5.15

Ένας
κινητήρας
servo
(Εικόνα
5.15)
διαφέρει
ως προς
ένα
κινητήρα
DC αρχικά
ως προς το
εύρος

κίνησης του. Χρησιμοποιείται για κινήσεις ακριβείας παρά για συνεχή κίνηση. Στο όχημα θα τον χρησιμοποιήσουμε για να μετακινήσουμε τον αισθητήρα υπερήχων σε τρεις συγκεκριμένες θέσεις (0, 90, 180 μοίρες) ούτως ώστε ο αισθητήρας να μετρήσει την απόσταση από εμπόδια στις θέσεις αυτές. Το καφέ καλώδιο αντιστοιχεί στη γείωση GND και θα συνδεθεί με κάποιο από τα GND pins του board. Το κίτρινο καλώδιο CONTROL αντιστοιχεί στο σήμα που θα πάρει ο κινητήρας για να κινηθεί προς μία θέση. Θα συνδεθεί δηλαδή σε κάποιο ψηφιακό ακροδέκτη του board με δυνατότητα PWM. Το κόκκινο καλώδιο Vcc αντιστοιχεί στην τροφοδοσία του servo με 5V [36]. Η τάση λειτουργίας ενός servo motor κυμαίνεται από τα 4,8 – 6 Volts όταν είναι στάσιμο (idle current)[37]. Χρησιμοποιεί περισσότερο ρεύμα όσο κινείται ή όσο συναντά κάποιο εμπόδιο. Όσο κινείται ελεύθερα παρουσιάζεται η λεγόμενη και κατανάλωση λειτουργίας (running current)[37]. Η παύση της κίνησης (stall current) λόγω κάποιου εμποδίου μπορεί να αυξήσει την τιμή του ρεύματος στα 600 mA που μπορεί να προκαλέσουν βλάβη στο board [37]. Στο όχημα μας ο κινητήρας servo θα φέρει και θα κινεί σε συγκεκριμένες θέσεις τον αισθητήρα υπερήχων και θα

μπορούσαμε να τους θεωρήσουμε μία κοινή μονάδα που θα προγραμματίσουμε στο επόμενο μάθημα.



Εικόνα 5.16

Ο κινητήρας servo αποτελείται από γρανάζια έναν κινητήρα DC, ένα ολοκληρωμένο κύκλωμα και ένα δυναμόμετρο (Εικόνα 5.16). Τα γρανάζια αυξάνουν τη ροπή του κινητήρα servo [38]. Το εσωτερικό κύκλωμα ελέγχει την ταχύτητα και την φορά της περιστροφής του κινητήρα DC. Το δυναμόμετρο είναι ένας μεταβαλλόμενος αντιστάτης που συνδέεται με το τελικό γρανάζι κίνησης. Η αλλαγή της αντίστασης διαβάζεται από το κύκλωμα και έτσι καθορίζεται η ακριβής θέση του γραναζιού. Τυπικά ένας κινητήρας servo όπως αυτός που θα χρησιμοποιήσετε στο όχημα αναμένει ένα σήμα με εντολή να κινηθεί κάθε 20 msec και το πλάτος αυτού του σήματος καθορίζει και την θέση προς την οποία θα κινηθεί [38].

Υλικό για μελέτη στο σπίτι

Κινητήρες DC

<https://www.allaboutcircuits.com/projects/control-a-motor-with-an-arduino/>

<https://uk.rs-online.com/web/content/discovery/ideas-and-advice/dc-motors-guide>

Οδηγός L289N

<https://projecthub.arduino.cc/lakshyajhalani56/l298n-motor-driver-arduino-motors-motor-driver-l298n-7e1b3b>

<https://howtomechatronics.com/tutorials/arduino/arduino-dc-motor-control-tutorial-l298n-pwm-h-bridge/>

Κινητήρες Servo

<https://ardtech.webnode.gr/l/pos-leitoyrgei-enas-servo-kinitiras/>

<https://circuitdigest.com/microcontroller-projects/interface-servo-motor-with-arduino>

Άσκηση 1

Να αναπτύξετε στο Tinkercad ή στο προσωπικό σας kit κυκλώματα ελέγχου κινητήρων DC με ή χωρίς Arduino. Να παραδώσετε τρία sketch στο επόμενο μάθημα

Ορισμένα παραδείγματα:

https://www.youtube.com/watch?v=zu_ZOpwJLkY

<https://www.youtube.com/watch?app=desktop&v=eTe2ZmOmkAU>

<https://www.tinkercad.com/things/hEfKwa82dfb-dc-motor>

<https://www.tinkercad.com/things/1VYZZJKspEV-motor-control-using-a-l298n-h-bridge>

Άσκηση 2

Μελετήστε το κύκλωμα της κατασκευής σας και προτείνετε τις απαραίτητες αλλαγές στην καλωδίωση αν:

- i) Θέλουμε ο οδηγός να τροφοδοτήσει το board
- ii) Στον κώδικα της εικόνας 5.13 θέλουμε για τον κινητήρα 2 στην συνάρτηση forward() η μεταβλητή kinitiras2_f έχει σαν τιμή το HIGH και όχι το LOW
- iv) Αν γίνει η αλλαγή του προηγούμενου ερωτήματος να γράψετε τις αλλαγές στον κώδικα και για τις υπόλοιπες συναρτήσεις

Άσκηση 3

Να αναπτύξετε κώδικα έτσι ώστε το όχημα να εκτελεί κίνηση σε τετράγωνο. Θα χρησιμοποιήσετε τις συναρτήσεις forward() και left() ή right(). Ο χρόνος που θα γράψετε στην συνάρτηση delay() δεν χρειάζεται να είναι ο ακριβής χρόνος έτσι ώστε το όχημα να εκτελέσει στροφή 90 μοιρών

ΚΕΦΑΛΑΙΟ 6

ΜΑΘΗΜΑ 6^ο: Η Μονάδα Μέτρησης Αποστάσεων

Σε αυτό το μάθημα θα αναπτύξουμε τη μονάδα μέτρησης αποστάσεων από την οποία το όχημα θα έχει την ικανότητα να αποφεύγει εμπόδια και να εκτελεί αυτόνομη κίνηση. Η μονάδα αυτή θα αποτελείται από ένα κινητήρα servo τον οποίο γνωρίσαμε στο προηγούμενο μάθημα και ένα αισθητήρα υπερήχων HC-SR04 σταθεροποιημένο πάνω στον κινητήρα μέσω του κατάλληλου βραχίονα. Θα προγραμματίσουμε τη μονάδα έτσι ώστε να κινείται σε τρεις διαφορετικές θέσεις (Αρχή στις 90 μοίρες – 0 μοίρες – 180 μοίρες – επιστροφή στις 90 μοίρες) και να παίρνει μετρήσεις απόστασης. Στον προγραμματισμό της μονάδας θα χρησιμοποιηθούν οι βιβλιοθήκες Servo και Ultrasonic που απλοποιούν τον κώδικα που χρειάζεται για τον χειρισμό του κινητήρα Servo και του αισθητήρα αντίστοιχα.

Ας ξεκινήσουμε με την κατασκευή της μονάδας



Εικόνα 6.1Α

Βιδώστε την βάση στήριξης του αισθητήρα στον βραχίονα του servo και στερεώστε τις βίδες με παξιμάδια. Προσέξτε οι βίδες να μην εμποδίζουν την κίνηση του servo ούτε να ακουμπάνε σε μεταλλικά μέρη του αισθητήρα. Περάστε τα αντηχεία του αισθητήρα στις υποδοχές της βάσης στήριξης. Προσέξτε οι ακροδέκτες του αισθητήρα να τοποθετηθούν προς τα άνω. Στερεώστε τον αισθητήρα στη βάση με μονωτική ταινία όπως φαίνεται στην Εικόνα 6.1Α. Εφαρμόστε καλώδια σύνδεσης στους ακροδέκτες του αισθητήρα (τέσσερα τον αριθμό). Κουμπώστε τον βραχίονα με την βάση στήριξης και τον αισθητήρα στο εξωτερικό γρανάζι του servo. Η μονάδα μέτρησης αποστάσεων είναι έτοιμη και θα στερεωθεί με βίδες στην κατάλληλη υποδοχή για servo του σασί.

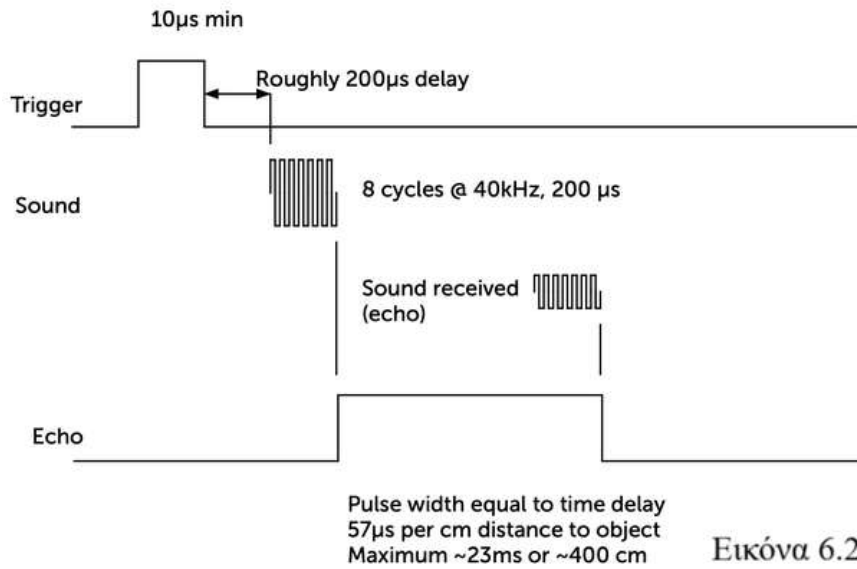
6.1 Ο Αισθητήρας Υπερήχων HC-SR04



Εικόνα 6.1B

Οι υπέρηχοι είναι κύματα ήχου σε υψηλές συχνότητες οι οποίες δεν γίνονται αντιληπτές από το ανθρώπινο αυτί. Το ανθρώπινο αυτί ανταποκρίνεται σε συχνότητες 20Hz – 20 kHz ενώ οι υπέρηχοι είναι

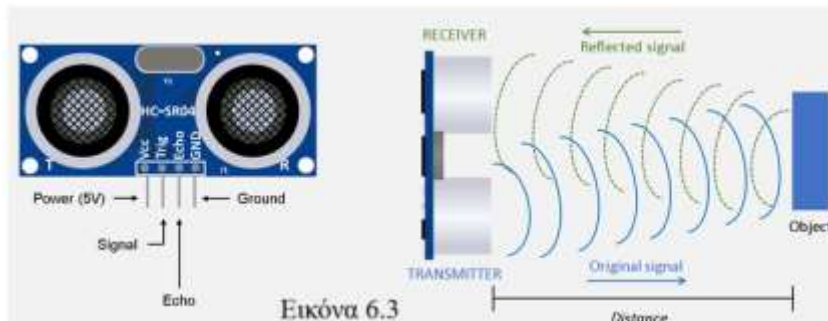
κύματα ήχου με συχνότητα μεγαλύτερη των 20kHz [39]. Οι υπέρηχοι ταξιδεύουν στο μέσο διάδοσης με ταχύτητα του ήχου (για τον αέρα θα θεωρήσουμε ως ταχύτητα του ήχου τα 340 m/sec). Η ταχύτητα διάδοσης επηρεάζεται από το μέσο διάδοσης. Άλλη η ταχύτητα διάδοσης στον αέρα άλλη εντός στερεού ή υγρού. Σε γενικές γραμμές, όσο πιο πυκνό το μέσο τόσο πιο γρήγορα διαδίδεται ο ήχος ενώ κατά τα γνωστά από τη Φυσική ο ήχος δεν διαδίδεται στο κενό. Στο project μας το μέσο διάδοσης είναι ο αέρας και η λειτουργία του αισθητήρα βασίζεται στο φαινόμενο της κυματικής ανάκλασης. Στην Εικόνα 6.1B βλέπετε τον αισθητήρα HC-SR04 που χρησιμοποιήθηκε στην μονάδα μέτρησης αποστάσεων. Τα βασικά μέρη του αισθητήρα είναι ο εκπομπός Transmitter από τον οποίο εκπέμπεται το ηχητικό σήμα ανακλάται στο αντικείμενο και λαμβάνεται από τον δέκτη Receiver[40]. Τα T και R που βλέπετε στον αισθητήρα επισημαίνουν ποιο από τα δύο αισθητήρια είναι ο εκπομπός και ποιο ο δέκτης. Αντίστοιχα, τα pins Trig και Echo αφορούν τον εκπομπό και τον δέκτη και συνδέουν τα αισθητήρια με δύο ψηφιακά pin του board.



Εικόνα 6.2

Ο εκπομπός αφού λάβει το σήμα από τον ακροδέκτη Trig των 10 microsecond εκπέμπει έναν υπέρηχο (8 παλμοί) με συχνότητα 40 kHz. Ο δέκτης

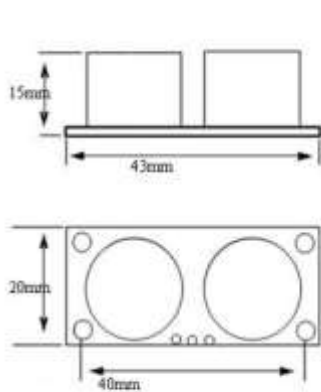
ενεργοποιείται με την εκπομπή του υπέρηχου και παραμένει ενεργός μέχρι να λάβει τον υπέρηχο πίσω (ECHO). Αν δεν λάβει κάποιο σήμα απενεργοποιείται αυτόματα (μετά από 38 ms) [40]. Η απόσταση υπολογίζεται από τον χρόνο που χρειάστηκε από την εκπομπή μέχρι τη λήψη /2. Χρησιμοποιείται ο γνωστός από τη Φυσική τύπος απόσταση = ταχύτητα x χρόνος.



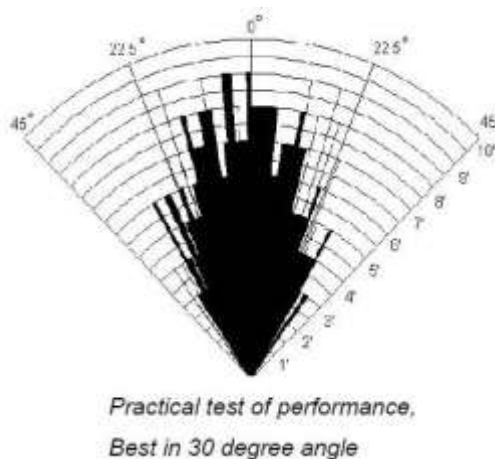
Εικόνα 6.3

Πιο συγκεκριμένα:
 $Distance = (U_{\text{ήχου}} * T) / 2$. Η ταχύτητα ήχου είναι γνωστή στα

340m/sec. Το γινόμενο ταχύτητας - χρόνου από τον γνωστό τύπο που διδαχτήκατε στο μάθημα της Φυσικής διαιρείται δια 2 διότι ο χρόνος T είναι ο χρόνος που χρειάζεται το σήμα να φτάσει στο αντικείμενο και να επιστρέψει στον αισθητήρα (είναι δηλαδή διπλάσιος του χρόνου για να φτάσει το αντικείμενο όπως φαίνεται στην Εικόνα 6.3).

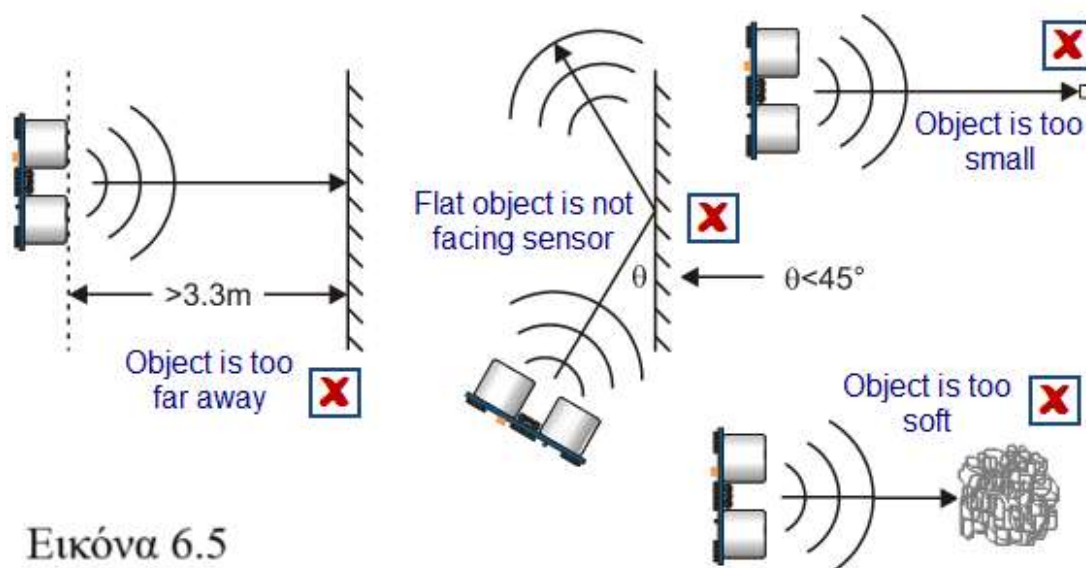


Εικόνα 6.4



Ο αισθητήρας είναι πιο αποτελεσματικός σε μία γωνία περίπου 30 μοιρών ως προς τον άξονα που περνάει από το κέντρο του.

Μπορεί να μετρήσει αποστάσεις από 2cm έως περίπου 400cm με ακρίβεια 3mm. Στην Εικόνα 6.4 βλέπετε το σχετικό διάγραμμα με το βεληνεκές του αισθητήρα και την «βεντάλια» που η ανίχνευση είναι αποτελεσματική. Καταλαβαίνετε γιατί ο αισθητήρας χρειάζεται να σκανάρει τον χώρο σε τρεις θέσεις και δεν γιατί δεν είναι αρκετό να σκανάρει μόνο στην εμπρόσθια πλευρά του έτσι ώστε να καθοριστεί το που έχει το περιθώριο για να στρίψει.



Εικόνα 6.5

Από την Εικόνα 6.5 βλέπετε ότι ο αισθητήρας δεν μπορεί να δώσει ακριβή (ή καθόλου) μέτρηση όταν:

- Ο αισθητήρας είναι υπό γωνία μεγαλύτερη από 45 μοίρες σε σχέση με το αντικείμενο
- Το αντικείμενο είναι πολύ μικρό
- Το αντικείμενο είναι πολύ μακριά

- Το αντικείμενο έχει ακανόνιστη ή μαλακή επιφάνεια με αποτέλεσμα ο ήχος να μην ανακλάται

Στην δοκιμή της λειτουργίας του οχήματος θα φροντίσετε τα εμπόδια να είναι λεία σταθερά και συμπαγή ούτως ώστε οι μετρήσεις του αισθητήρα να ανταποκρίνονται στην πραγματικότητα.



Εικόνα 6.6

Στην Εικόνα 6.6 βλέπετε την επεξήγηση των ακροδεκτών του αισθητήρα.

VCC: Τροφοδοσία του αισθητήρα.

Συνδέεται στον ακροδέκτη 5V του board ή σε όποια άλλη τροφοδοσία είναι διαθέσιμη

VCC- Connects to 5V of positive voltage for power

Trig- A pulse is sent here for the sensor to go into ranging mode for object detection

Echo- The echo sends a signal back if an object has been detected or not. If a signal is returned, an object has been detected. If not, no object has been detected.

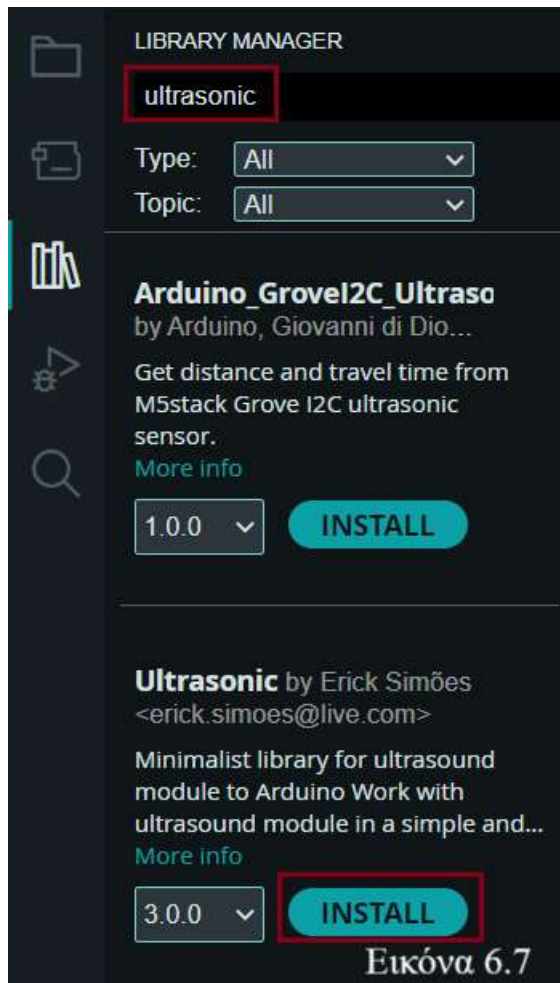
GND- Completes electrical pathway of the power.

Trig: Δίνει το σήμα στον εκπομπό. Συνδέεται σε ένα digital pin του board

ECHO: Δέχεται το ανακλώμενο σήμα. Συνδέεται σε ένα digital pin του board

GND: Γείωση. Συνδέεται σε αντίστοιχο GND pin του board

6.2 Προγραμματισμός του Αισθητήρα HC-SR04



Στην περίπτωση του αισθητήρα HC-SR04 δεν χρειάζεται κάθε φορά να γράφουμε κώδικα για τον ορισμό κάθε λειτουργίας των ακροδεκτών. Υπάρχει η απλή βιβλιοθήκη Ultrasonic (Erik Simoes) [41] που μπορούμε να την επικαλεστούμε στον Editor και να χρησιμοποιήσουμε τις εντολές της. Ανοίξτε τον editor και στον Library Manager αναζητήστε την βιβλιοθήκη και πατήστε INSTALL (Εικόνα 6.7). Η βιβλιοθήκη είναι έτοιμη προς χρήση. Η επίκληση της βιβλιοθήκης στο sketch γίνεται με την εντολή `#include <Ultrasonic.h>`

Πληκτρολογήστε τον κώδικα που θα δείτε στις επόμενες εικόνες. Θα γράψετε

κώδικα έτσι ώστε στην οθόνη του IDE να εμφανίζεται η απόσταση ανά διαστήματα.

```
#include <Ultrasonic.h> // εισαγωγή της βιβλιοθήκης Ultrasonic
#include <Servo.h> // εισαγωγή της βιβλιοθήκης Servo
Ultrasonic ultrasonic(12, 13); //δημιουργία του αντικείμενου Ultrasonic
int apostasi;
```

Εικόνα 6.8

Η εντολή `Ultrasonic ultrasonic(12, 13);` (Εικόνα 6.8) δημιουργεί το αντικείμενο Ultrasonic και ορίζει τον ακροδέκτη Trig στο pin 12 και τον ECHO στο pin 13. Το αντικείμενο θα μπορούσαμε να το θεωρήσουμε σαν ένα είδος μεταβλητής της βιβλιοθήκης Ultrasonic [42]. Με την εντολή `apostasi = ultrasonic.read();` καταχωρείται στην μεταβλητή `apostasi` η απόσταση του εμποδίου σε cm. Την μεταβλητή `apostasi` μπορούμε να τη δηλώσουμε είτε ως `int` είτε ως `float`.

```
void setup() {  
  // put your setup code here, to run once:   
  Serial.begin(9600);  
}
```

Πληκτρολογήστε τον κώδικα που βλέπετε στις εικόνες και συνδέστε τον αισθητήρα στο

board. Προσδιορίζουμε το bit rate της σειριακής επικοινωνίας board - υπολογιστή (Εικόνα 6.9).

```
void loop() {  
  // put your main code here, to run repeatedly:  
  apostasi = ultrasonic.read();  
  Serial.println(apostasi);  
  delay(500);  
}
```

Εικόνα 6.10

Η εντολή
ultrasonic.read() είναι
ο τρόπος με τον οποίο
το αντικείμενο
ultrasonic

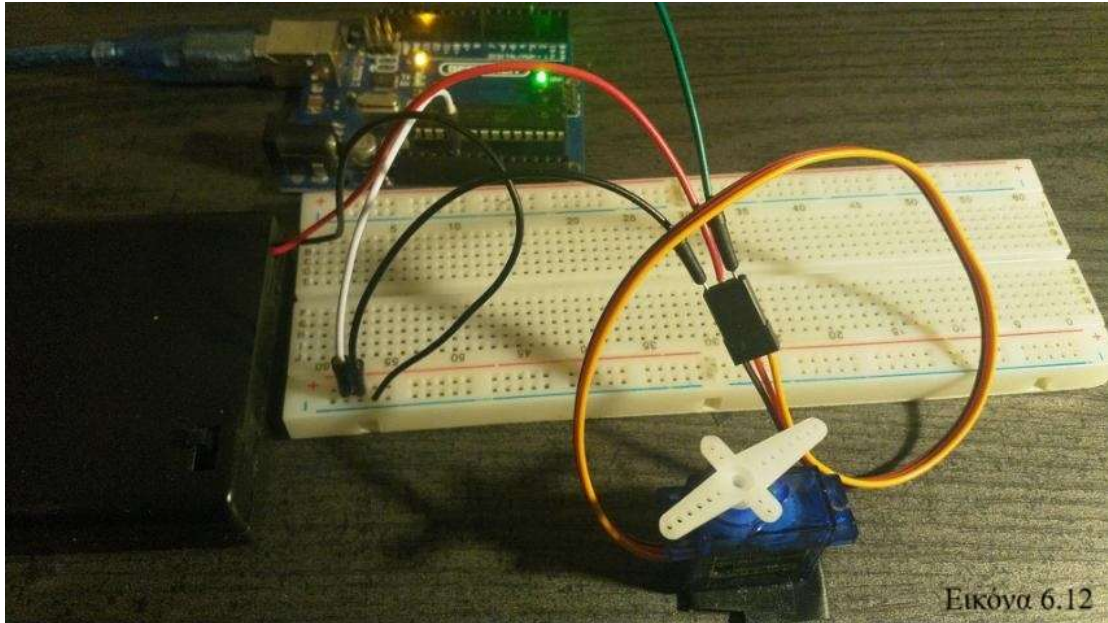
προσδιορίζει την απόσταση από το εμπόδιο. Στη οθόνη (serial monitor) θα εμφανίζεται η απόσταση ανά 500 msec (Εικόνα 6.10).

```
Message (Enter to send message to 'Arduino Uno' on 'COM3')  
---  
357  
357  
60  
66
```

Εικόνα 6.11

Στο serial monitor (Εικόνα 6.11) εμφανίζεται η απόσταση ανά μισό δευτερόλεπτο. Η τιμή 357 είναι η μέγιστη τιμή που μπορεί να μετρήσει ο αισθητήρας και εμφανίζεται είτε όταν το αντικείμενο είναι μακριά είτε σε απότομες αλλαγές της κατεύθυνσης του αισθητήρα. Η default τιμή της απόστασης που δείχνει ο αισθητήρας (λόγω της συγκεκριμένης βιβλιοθήκης) είναι σε εκατοστά. Υπάρχει και η δυνατότητα μέτρησης σε ίντσες.

Προγραμματισμός κινήσεων του servo



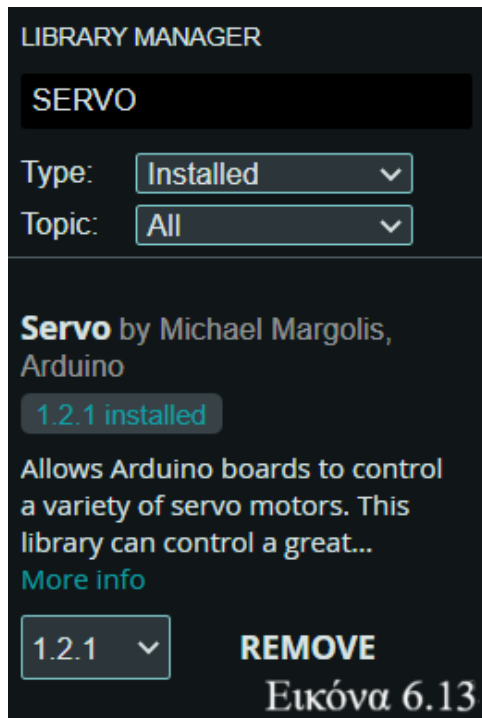
Υλοποιήστε το κύκλωμα της εικόνας 6.12. Σημειώστε ότι ο κινητήρας έχει υψηλές απαιτήσεις σε κατανάλωση ρεύματος (ένα micro servo μπορεί να έχει απαίτηση ρεύματος ακόμα και 1Ampere[38]) και η σύνδεση του servo απευθείας στο pin 5V του board μπορεί να επιφέρει καταστροφή του pin γι' αυτό φροντίστε να:

Συνδέσετε τον αρνητικό πόλο της πηγής του servo (μαύρο καλώδιο), το GND του servo (μαύρο καλώδιο), και το GND του board (λευκό καλώδιο) στη ράγα – του raster

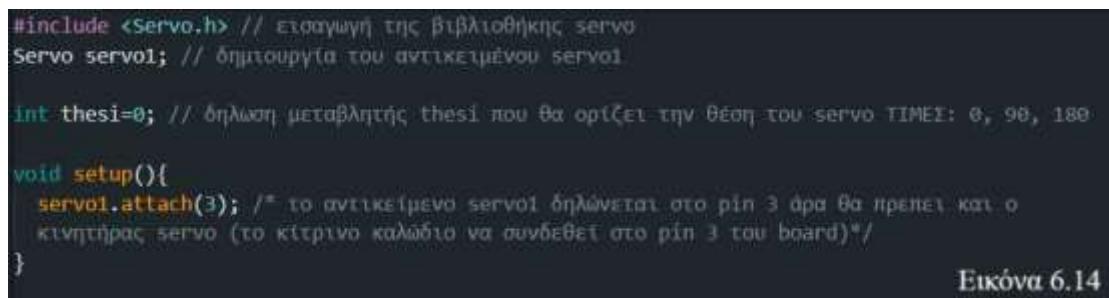
Εξαιτίας των μεγάλων ενεργειακών απαιτήσεων του servo στο όχημα θα συνδέσουμε τον servo σε δικιά του παροχή (pack μπαταριών 6 V) ανεξάρτητη της παροχής του board

Συνδέστε το CONTROL (κίτρινο καλώδιο) του servo με το pin 3 του board μέσω αρσενικού καλωδίου (πράσινο καλώδιο).

Το κύκλωμα είναι έτοιμο. Προχωρήστε στον προγραμματισμό της κίνησης.



Για να ελέγξουμε ένα servo θα χρειαστεί να εγκαταστήσουμε στον editor την βιβλιοθήκη Servo από τον Library Manager. Κατά τη γνωστή διαδικασία κατεβάστε την βιβλιοθήκη Servo στον IDE αν δεν είναι ήδη εγκατεστημένη (Εικόνα 6.13).



Όπως και στον αισθητήρα ultrasonic πρέπει να δημιουργήσουμε το αντίστοιχο αντικείμενο και να συμπεριλάβουμε στο sketch την βιβλιοθήκη με την εντολή #include <Servo.h> [43]

Η εντολή servo1.attach(3) είναι ο τρόπος με τον οποίο δηλώνεται ότι το αντικείμενο servo1 συνδέεται στο pin 3 του board [43]

Η μεταβλητή thesi που δηλώσαμε στην αρχή του προγράμματος δεν μας χρησιμεύει έως τώρα αλλά θα σας χρησιμεύσει για την επίλυση της άσκησης σας


```
void loop(){
  servo1.write(0); //με την εντολή αυτή ο servo πάει στη θέση 0 (0 μοίρες)
  delay(500); // καθυστέρηση 500 msec
  servo1.write(90); //με την εντολή αυτή ο servo πάει στη θέση 90 (90 μοίρες)
  delay(500); // καθυστέρηση 500 msec
  servo1.write(180); //με την εντολή αυτή ο servo πάει στη θέση 180 (180 μοίρες)
  delay(500); // καθυστέρηση 500 msec
}
```

Εικόνα 6.15

Η εντολή `servo1.write()` είναι ο τρόπος με τον οποίο το αντικείμενο `servo1` μετακινείται προς την θέση την οποία του ορίζουμε και με την εντολή `delay(500)` η κίνηση προς αυτές τις θέσεις γίνεται πιο ομαλή και ευδιάκριτη . Το sketch είναι έτοιμο. Μεταγλωττίστε το, φορτώστε το στο board και παρατηρήστε την κίνηση.

Υλικό για μελέτη στο σπίτι

Για τον αισθητήρα υπερήχων:

<https://www.youtube.com/watch?v=n-gJ00GTsNg>

<https://projecthub.arduino.cc/Isaac100/getting-started-with-the-hc-sr04-ultrasonic-sensor-7cabel>

Σε αυτό το site μπορείτε να δείτε κάποιες πιο εξειδικευμένες εντολές για τη λειτουργία του αισθητήρα χωρίς τη χρήση βιβλιοθήκης

Για τους κινητήρες servo και τον έλεγχό τους:

<https://www.circuitbasics.com/controlling-servo-motors-with-arduino/>

<https://www.youtube.com/watch?v=kUHmYKWwuWs>

Άσκηση 1

Στο επόμενο μάθημα θα ολοκληρώσουμε το αυτοκινούμενο όχημα. Αρχικά, θέλουμε ο αισθητήρας υπερήχων να τοποθετηθεί στην εμπρόσθια πλευρά του οχήματος και προφανώς να κοιτάει προς τα εμπρός και να ελέγχει στη συνέχεια τις αποστάσεις στην δεξιά και αριστερή πλευρά.

Για να σιγουρέψουμε ότι κινητήρας servo και ο αισθητήρας είναι στην κατάλληλη θέση (90 μοίρες – μέσο της κίνησης) να γράψετε κώδικα που ο κινητήρας servo να τοποθετείται στις 90 μοίρες.

Να θεωρήσετε ότι η μονάδα θα τοποθετηθεί στην αντίστοιχη οπή servo του σασί.

Άσκηση 2

Είδαμε στο μάθημα τον κώδικα για να κινήσουμε τη μονάδα στις θέσεις 0, 90, 180 με τον κινητήρα servo. Να γράψετε κώδικα έτσι ώστε η μετάβαση σε αυτές τις θέσεις να γίνεται ομαλά και όχι απότομα. Να χρησιμοποιήσετε την δομή επανάληψης for και να δοκιμάσετε τον κώδικα σας στο Tinkercad.

Συμβουλευτείτε την σελίδα:

<https://docs.arduino.cc/learn/electronics/servo-motors/>

Επιλέξτε την κατάλληλη μέθοδο

Άσκηση 3

Έχοντας υπόψιν τον κώδικα που έχουμε αναπτύξει μέχρι τώρα να γράψετε ένα sketch που να εκτελεί όλες τις απαραίτητες λειτουργίες του project (κίνηση, αποφυγή εμποδίων). Να χρησιμοποιήσετε την δομή ελέγχου if else if.

ΚΕΦΑΛΑΙΟ 7

ΜΑΘΗΜΑ 7^ο: Ολοκλήρωση του Οχήματος

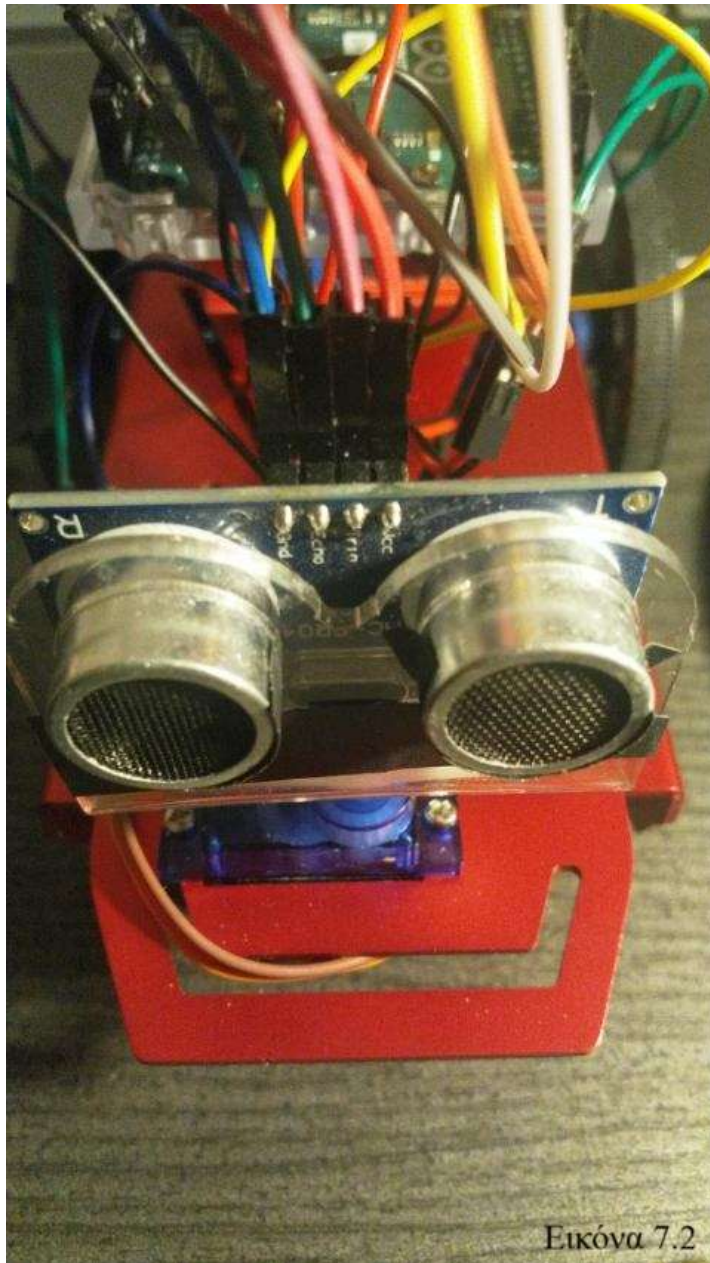
Στο τελικό αυτό μάθημα θα κάνουμε τις τελευταίες προσθήκες στο όχημα μας τόσο σε υλικό όσο και κώδικα. Η κάθε ομάδα ανάλογα με την πρόοδο της θα τελειώσει την κατασκευή και θα συμπληρώσει τον κώδικα (εάν υπολείπεται σε κάτι). Ο κώδικας που θα παρουσιαστεί στον προτζέκτορα είναι μία από τις λύσεις που θα μπορούσαν να δοθούν στο πρόβλημα της κατασκευής. Αν κάποια ομάδα έχει κάποια διαφορετική προσέγγιση στην κατασκευή και τον προγραμματισμό του οχήματος φυσικά και θα ακολουθήσει το δικό της πλάνο και θα επιδείξει την λειτουργία του οχήματος στην τάξη

Ο κώδικας και η προσέγγιση κάθε ομάδας θα παρουσιαστεί μετά το τέλος της ενότητας 7.1 στην υπόλοιπη ώρα του μαθήματος. Κάθε ομάδα θα δεχτεί ερωτήσεις από τον καθηγητή και τις άλλες ομάδες.

7.1 Ολοκλήρωση της Κατασκευής του Οχήματος

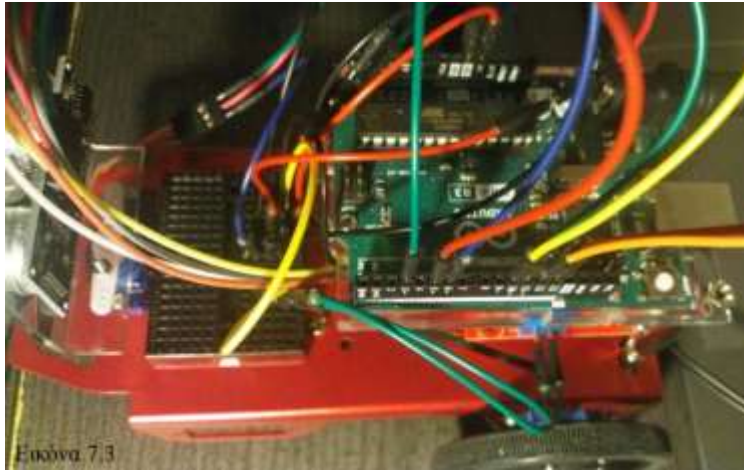


Στο όχημα έχουμε πολλά εξαρτήματα που τροφοδοτούνται με 5 V και απαιτούν σύνδεση σε γείωση – οδηγός – αισθητήρας υπέρηχων – Servo. Γι' αυτό το λόγο επικολλούμε το mini raster (Εικόνα 7.1) που έχετε στο κιτ σας πάνω στο σασί. Τώρα μπορούμε να συνδέσουμε δύο σειρές του raster με τους ακροδέκτες 5V (κόκκινο καλώδιο) και GND (μαύρο καλώδιο) του board και τα εξαρτήματα αυτά να συνδέονται στο raster. Συνηθίζουμε στις γειώσεις να χρησιμοποιούμε μαύρα καλώδια για λόγους εύκολης αναγνώρισης του εκάστοτε καλωδίου.

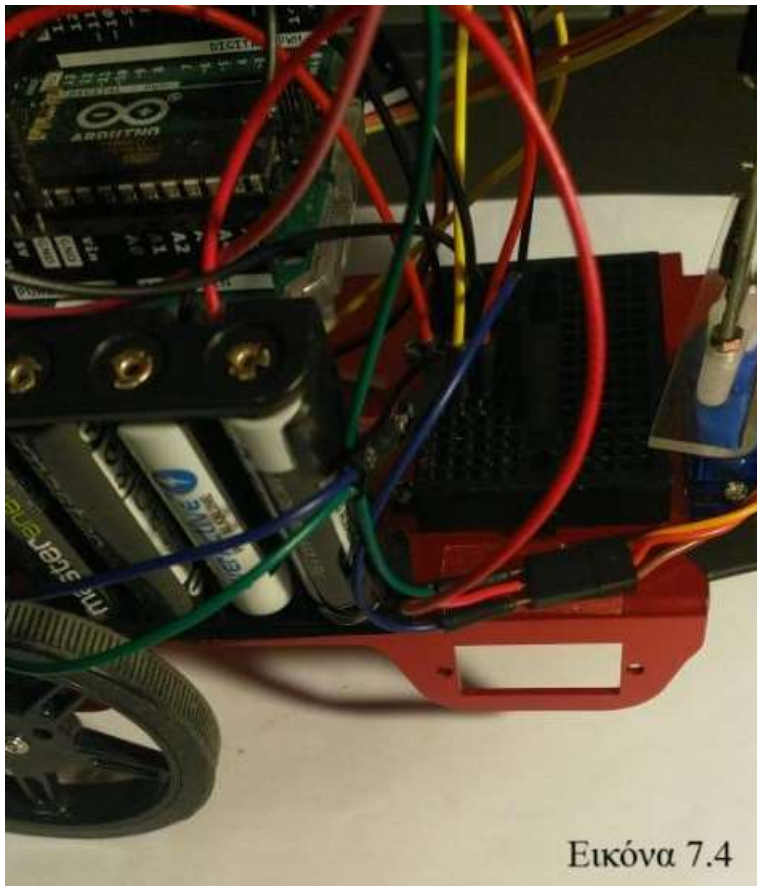


Εικόνα 7.2

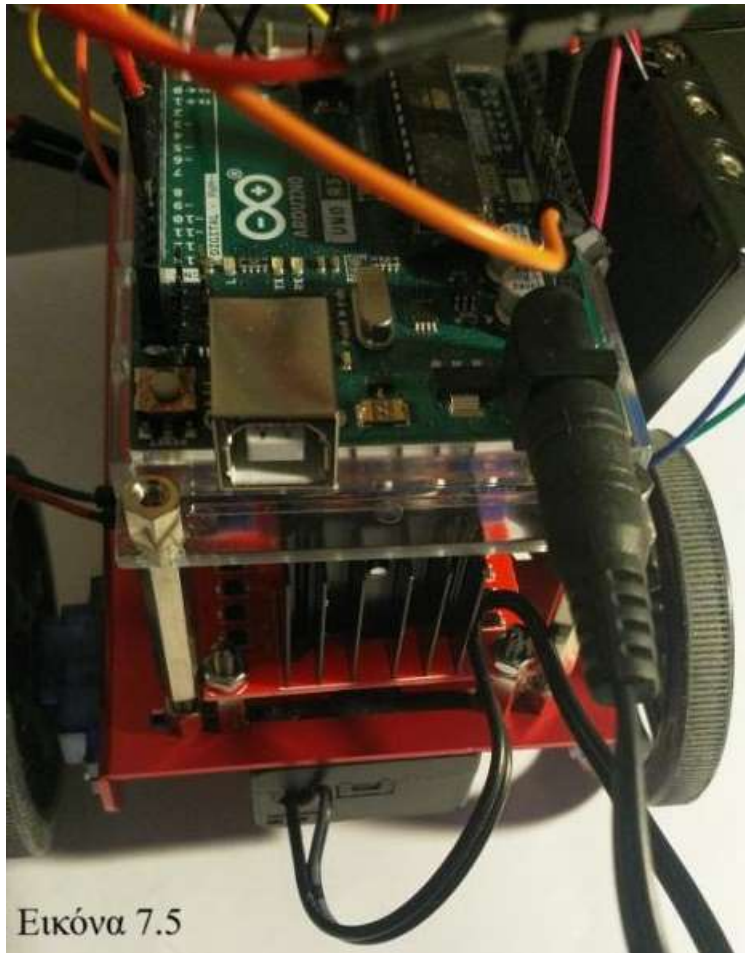
Η μονάδα μέτρησης αποστάσεων (servo – βραχίονας servo – αισθητήρας) βιδώνεται στην υποδοχή για servo του σασί. Οι ακροδέκτες Vcc και GND του αισθητήρα συνδέονται στο mini raster στις αντίστοιχες γραμμές όπου έχουν συνδεθεί και τα καλώδια 5V και GND που προέρχονται από το raster (Εικόνα 7.2). Τα καλώδια για τους ακροδέκτες Trig και ECHO συνδέονται στα pin 12 και 13 αντίστοιχα.



Συνδέστε το GND του servo στη γείωση του mini raster (Μπλέ καλώδιο που συνδέεται με το καφέ καλώδιο του servo Εικόνα 7.3)

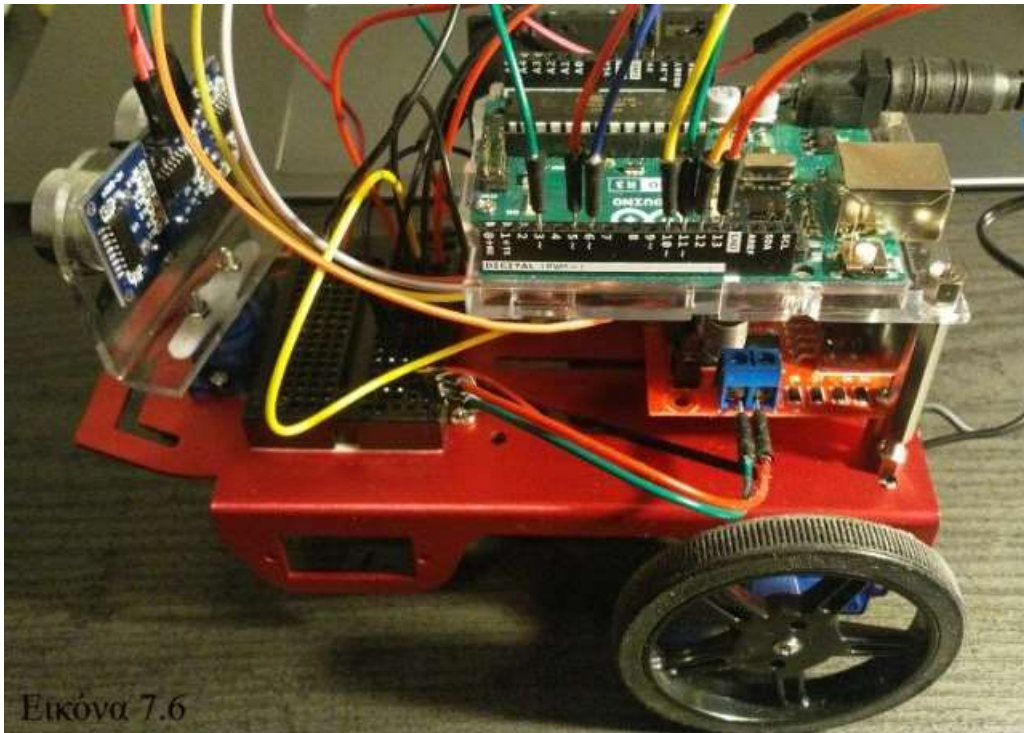


Ο κινητήρας Servo έχει δική του τροφοδοσία (4 μπαταρίες AAA συνολικής τάσης 6V Εικόνα 7.4). Συνδέστε το κόκκινο καλώδιο της μπαταριοθήκης στο πορτοκαλί καλώδιο του servo και το μαύρο καλώδιο της μπαταριοθήκης στο GND του mini raster.



Η τροφοδοσία του board γίνεται με μπαταρία 9V σε κατάλληλη θήκη και με το βύσμα συμβατό με barrel jack όπως έχουμε ήδη δει στο μάθημα 5. Έχουμε επίσης κολλήσει τη θήκη με τη μπαταρία 9V στο κάτω μέρος του σασί ανάμεσα στους κινητήρες των οπίσθιων τροχών με ταινία διπλής όψης. Συνδέστε το βύσμα στο barrel jack (Εικόνα 7.5).

ΤΟ ΟΧΗΜΑ ΕΙΝΑΙ ΕΤΟΙΜΟ – ΠΕΡΑΣΤΕ ΣΤΗΝ ΟΛΟΚΛΗΡΩΣΗ ΤΟΥ ΚΩΔΙΚΑ



Εικόνα 7.6

7.2 Προγραμματισμός του Οχήματος

Στο 5ο μάθημα προγραμματίσατε το όχημα στο να εκτελεί βασικές κινήσεις με την βοήθεια του οδηγού L289N και τις κατάλληλες εντολές και συνδεσμολογία. Στο 6ο μάθημα γνωρίσατε τον κινητήρα servo και τον αισθητήρα υπερήχων καθώς και εξασκηθήκατε στον προγραμματισμό τους. Στις ασκήσεις 2 και 3 του 6ου μαθήματος χρησιμοποιήσατε τη δομή ελέγχου if...else if και την επαναληπτική δομή for. Θα δούμε στα επόμενα πως ακριβώς λειτουργούν. Σε αυτό το μάθημα θα συνδυαστούν οι κώδικες πάνω στους οποίους εξασκηθήκατε για να καταλήξετε στο επιθυμητό αποτέλεσμα. Ο κώδικας που θα σας παρουσιαστεί στις επόμενες διαφάνειες φυσικά επιδέχεται τροποποιήσεων ανάλογα με τις ανάγκες, το πλάνο και την έρευνα κάθε ομάδας

Αρχικά σιγουρευτείτε ότι στον editor είναι εγκατεστημένες οι βιβλιοθήκες Servo και Ultrasonic. Αν όχι τις αναζητούμε και τις εγκαθιστούμε από τον Library Manager. Τα εξαρτήματα του οχήματος σας είναι ενεργοβόρα. Όταν θέλετε να δοκιμάσετε τη λειτουργία του κώδικα σας και είναι δυνατό να προτιμάτε το καλώδιο σύνδεσης με τον υπολογιστή για την ενεργοποίηση και τη λειτουργία του οχήματος παρά τις μπαταρίες. Ο κώδικας έχει επεξηγηματικά σχόλια, αν παρ' όλα αυτά έχετε κάποια απορία μην διστάσετε να ρωτήσετε. Αγνοήστε τα κομμάτια κώδικα που έχετε έτοιμα και εκτελούν την ίδια λειτουργία. Συμπληρώστε τον κώδικα που πιθανά σας λείπει έτσι ώστε το όχημα να εκτελεί όλες τις λειτουργίες

```
1 #include <ultrasonic.h> // εγκατάσταση της βιβλιοθήκης ultrasonic
2 #include <Servo.h> // εγκατάσταση της βιβλιοθήκης Servo
3
4 Ultrasonic ultrasonic(12, 13); //δημιουργία των αντικειμένων Ultrasonic
5 Servo servo1; // δημιουργία του αντικειμένου servo1
6
7 int thes1 = 90; // δήλωση μεταβλητής που θα ορίζει την θέση του servo PWM: 0, 90, 180
8 int apostasi; // απόσταση εμπόδιου μπροστά
9 int deksia; // απόσταση εμπόδιου δεξιά
10 int aristera; // απόσταση εμπόδιου αριστερά
11
12 const int kinitiras1_f = 5; // το κόκκινο καλώδιο του κινητήρα 1 (δεξιάς από την οπίσθια πλευρά) συνδέεται στον ακροδέκτη 5
13 const int kinitiras1_b = 6; // το μαύρο καλώδιο του κινητήρα 1 (δεξιάς από την οπίσθια πλευρά) συνδέεται στον ακροδέκτη 6
14 const int kinitiras2_b = 10; // το μαύρο καλώδιο του κινητήρα 2 (αριστεράς από την οπίσθια πλευρά) συνδέεται στον ακροδέκτη 10
15 const int kinitiras2_f = 11; // το κόκκινο καλώδιο του κινητήρα 2 (αριστεράς από την οπίσθια πλευρά) συνδέεται στον ακροδέκτη 11*
```

Εικόνα 7.7

Στην εικόνα 7.7. Εισάγουμε τις βιβλιοθήκες Ultrasonic Servo στο sketch, δημιουργούνται τα αντικείμενα ultrasonic και servo1 που αναφέρονται στον αισθητήρα και στον κινητήρα servo αντίστοιχα. Στη μεταβλητή thes1 καταχωρείται η θέση του servo, στη μεταβλητή apostasi η απόσταση εμπρόσθιου εμπόδιου μπροστά,

στη μεταβλητή `deksia` η απόσταση εμποδίου δεξιά του οχήματος και τέλος στην `aristera` η απόσταση εμποδίου αριστερά του οχήματος. Οι μεταβλητές `kinitiras1_f`, `kinitiras1_b`, `kinitiras2_f`, `kinitiras2_b` δηλώνονται ως `const` δηλαδή δεν μεταβάλλονται. Προσέξτε τη συνδεσμολογία των καλωδίων που αναφέρεται στα σχόλια.

```
18 void setup() {
19   // put your setup code here, to run once:
20   pinMode(kinitiras1_f, OUTPUT); // Όλοι οι ακροδέκτες των κινητήρων δηλώνονται ως OUTPUT
21   pinMode(kinitiras1_b, OUTPUT);
22   pinMode(kinitiras2_f, OUTPUT);
23   pinMode(kinitiras2_b, OUTPUT);
24   servo1.attach(3); /* το αντικείμενο servo1 δηλώνεται στο pin 3 άρα θα πρέπει και ο
25   κινητήρας servo (το κίτρινο καλώδιο να συνδέθει στο pin 3 του board)*/
26   servo1.write(90);
27 }
```

Εικόνα 7.8

Στην Εικόνα 7.8. Οι μεταβλητές των κινητήρων DC `kinitiras1_f`, `kinitiras1_b`, `kinitiras2_f`, `kinitiras2_b` δηλώνονται ως `OUTPUT` ενώ ο `servo` δηλώνεται στη θέση 3 μέσω της εντολής `servo1.attach(3)` [43]. Ο κινητήρας `servo` πηγαίνει στην θέση 90 (ο αισθητήρας βλέπει μπροστά) με την εντολή `servo1.write(90)`. Φροντίστε ο βραχίονας να είναι σωστά τοποθετημένος στο γρανάζι του `servo` έτσι ώστε ο αισθητήρας όντως να βλέπει μπροστά.

```
29 void loop() {
30   // put your main code here, to run repeatedly:
31   apostasi = ultrasonic.read(); // μέτρηση απόστασης σε CM
32   delay(100);
33   servo1.write(90); // ο servo πηγαίνει στη θέση 90
34   delay(100);
35   if (apostasi > 15){ // αν η απόσταση είναι μεγαλύτερη των 15cm το όχημα συνεχίζει ευθεία
36     forward();
37 }
```

Εικόνα 8A

Στην Εικόνα 8A. Γίνεται ο πρώτος έλεγχος της εμπρόσθιας απόστασης με την εντολή `ultrasonic.read()`. Ο `servo` είναι ήδη στη θέση 90 αλλά για αποφυγή αποκλίσεων τον ξαναπάμε στη θέση 90 στην αρχή κάθε εκτέλεσης του κώδικα που περιέχεται στη `loop()`. Γίνεται επίσης ο πρώτος έλεγχος της εμπρόσθιας απόστασης με τη δομή `if`. Αν η απόσταση είναι μεγαλύτερη των 15 cm το όχημα συνεχίζει την πορεία του.

```

39 else if (apostasi <= 15){ //αν η απόσταση είναι μικρότερη από 15cm
40     stop(); // στάση μέχρι τον υπολογισμό των αποστάσεων
41     for(thesi = 0; thesi < 180; thesi += 1){ //δομή επανάληψης - ο αισθητήρας σκάνει από 0 - 180 μοίρες με βήμα 1
42         servo1.write(thesi); // ο servo πάει στη θέση που ορίζει η δομή επανάληψης
43         delay(10); //συμμόνη 10 ms για πιο ομαλή μετάβαση της μονάδας μέτρησης
44     }
45
46     aristera = ultrasonic.read(); //μετράει την απόσταση από εμπόδια στη αριστερή πλευρά και καταχωρεί την τιμή στη μεταβλητή
47     delay(100);
48     for(thesi = 180; thesi >= 0; thesi -=1){ // δομή επανάληψης - ο αισθητήρας σκάνει από 180 - 0 μοίρες με βήμα -1
49         servo1.write(thesi); // ο servo πάει στη θέση που ορίζει η δομή επανάληψης
50         delay(10); //συμμόνη 10 ms για πιο ομαλή μετάβαση της μονάδας μέτρησης
51     }
52
53     deksia = ultrasonic.read(); //μετράει την απόσταση από εμπόδια στην δεξιά πλευρά και καταχωρεί την τιμή στη μεταβλητή
54     delay(100);
55     servo1.write(90); // η μονάδα επιστρέφει στην εμπρόσθια θέση 90
56     delay(100);
57
58     if (aristera < deksia){ //έλεγχος για το ποια μεταβλητή είναι μεγαλύτερη
59         right(); //στροφή δεξιά αν aristera < deksia
60         delay(600); // καθυστέρηση έτσι ώστε το όχημα να στρίψει επαρκώς
61     }
62     else if (aristera > deksia){ //καρτίωση aristera > deksia
63         left(); //στροφή αριστερά
64         delay(600); // καθυστέρηση έτσι ώστε το όχημα να στρίψει επαρκώς
65     }

```

Εικόνα 7.9

Στην Εικόνα 7.9. Αν το όχημα έχει απόσταση από εμπόδιο μικρότερη από 15 cm (έλεγχος από την else if) το όχημα σταματάει, ο κώδικας καλεί την stop(). Με την δομή for που μεταβάλλει τη τιμή της μεταβλητής thesi η μονάδα μέτρησης αποστάσεων πηγαίνει από τη θέση 0 στη θέση 180 με βήμα 1 μοίρα μέσω της εντολής servo1.write(thesi). Η μονάδα παίρνει μέτρηση από την απόσταση εμποδίων από την αριστερή πλευρά του με την εντολή ultrasonic.read() [43]. Η απόσταση καταχωρείται στην μεταβλητή aristera. Με την δεύτερη δομή for γίνεται η αντίστροφη κίνηση της μονάδας (προς τη θέση 0) όπου η μονάδα παίρνει μέτρηση απόστασης από την δεξιά πλευρά και την καταχωρεί στην μεταβλητή deksia. Προσέξτε ότι η διατύπωση thesi = 180 εμφανίζεται μόνο στη δεύτερη δομή for και όχι και στην πρώτη διότι τότε θα υπήρχε ασάφεια και λάθος ως προς ποια εντολή θα έπρεπε να εκτελέσει ο κώδικας. Οι δύο δομές for δεν είναι τίποτα άλλο από την λειτουργία sweep του κινητήρα servo [37].

Στην γραμμή 58 ξεκινάει μια δεύτερη δομή if – else if. Η δομή είναι εσωτερική της πρώτης else if στη γραμμή 39. Εδώ τώρα ξεκινάει ο έλεγχος για το ποια μεταβλητή (aristera, deksia) είναι μεγαλύτερη και στη συνέχεια το όχημα αλλάζει κατεύθυνση προς την πλευρά με την ελεύθερη διαδρομή καλώντας την αντίστοιχη συνάρτηση.

```

58   if (aristera < deksia){ // έλεγχος για το ποιά μεταβλητή είναι μεγαλύτερη
59       right(); //στροφή δεξιά αν aristera < deksia
60       delay(600); // καθυστέρηση έτσι ώστε το όχημα να στρίψει επαρκώς
61   }
62   else if (aristera > deksia){ //περίπτωση aristera > deksia
63       left(); // στροφή αριστερά
64       delay(600); // καθυστέρηση έτσι ώστε το όχημα να στρίψει επαρκώς
65   }
66   else if (aristera <=15 && deksia <=15){ /*αν aristera<=15 ΚΑΙ deksia<=15
67       (όσο περίπου το μήκος του οχήματος)*/
68       backward(); //οπισθοχώρηση - δεν υπάρχει χώρος για στροφή
69       delay(1000);
70       left(); //στροφή αριστερά
71       delay(600);
72   }
73 }
74 } // τέλος επανάληψης loop()

```

Εικόνα 7.10

Στην Εικόνα 7.10. Αν η deksia είναι μεγαλύτερη τότε το όχημα καλεί την right() και κινείται δεξιά για 600 msec. Σε αντίθετη περίπτωση κινείται αριστερά με την left() για τον ίδιο χρόνο. Αν και (&&) η aristera και η deksia είναι μικρότερες ή ίσες των 15 εκατοστών τότε το όχημα δεν έχει χώρο να στρίψει και οπισθοχωρεί για 1 sec και στρίβει αριστερά.

```

76 // ΑΚΟΛΟΥΘΕΙ Ο ΟΡΙΣΜΟΣ ΤΩΝ ΣΥΝΑΡΤΗΣΕΩΝ ΚΙΝΗΣΗΣ ΠΟΥ ΕΪΔΑΜΕ ΣΤΟ 5ο ΜΑΘΗΜΑ
77 void forward() { // ΕΜΠΡΟΣ
78     digitalWrite(kinitiras1_f, HIGH);
79     digitalWrite(kinitiras1_b, LOW);
80     digitalWrite(kinitiras2_f, LOW);
81     digitalWrite(kinitiras2_b, HIGH);
82 }
83
84 void backward() { // ΠΙΣΩ
85     digitalWrite(kinitiras1_f, LOW);
86     digitalWrite(kinitiras1_b, HIGH);
87     digitalWrite(kinitiras2_f, HIGH);
88     digitalWrite(kinitiras2_b, LOW);
89 }
90
91 void right() { // ΔΕΞΙΑ
92     digitalWrite(kinitiras1_f, LOW);
93     digitalWrite(kinitiras1_b, HIGH);
94     digitalWrite(kinitiras2_f, LOW);
95     digitalWrite(kinitiras2_b, HIGH);
96 }

```

Εικόνα 7.11

Στην Εικόνα 7.11. Γίνεται η δήλωση των συναρτήσεων κίνησης forward(), backward(), right() που αναπτύξατε στο 5^ο Μάθημα.

```

98 void left() { // ΑΡΙΣΤΕΡΑ
99     digitalWrite(kinitiras1_f, HIGH);
100    digitalWrite(kinitiras1_b, LOW);
101    digitalWrite(kinitiras2_f, HIGH);
102    digitalWrite(kinitiras2_b, LOW);
103 }
104
105 void stop() { // ΣΤΑΣΗ
106     digitalWrite(kinitiras1_f, LOW);
107     digitalWrite(kinitiras1_b, LOW);
108     digitalWrite(kinitiras2_f, LOW);
109     digitalWrite(kinitiras2_b, LOW);
110 }

```

Εικόνα 7.12

Στην Εικόνα 7.12. Γίνεται η δήλωση των συναρτήσεων κίνησης left(), stop() που αναπτύξατε στο 5ο Μάθημα.

Πατήστε Upload και σχολιάστε η κάθε ομάδα τα αποτελέσματα

Σε τυχόν μήνυμα σφάλματος διορθώστε το και ξαναπατήστε Upload

ΣΥΓΧΑΡΗΤΗΡΙΑ

ΚΑΤΑΣΚΕΥΑΣΑΤΕ ΤΟ ΠΡΩΤΟ ΣΑΣ ΟΜΑΔΙΚΟ PROJECT

Η ΚΑΘΕ ΟΜΑΔΑ ΕΧΕΙ 5 ΛΕΠΤΑ ΝΑ ΠΑΡΟΥΣΙΑΣΕΙ ΤΟ ΟΧΗΜΑ ΤΗΣ ΣΕ ΛΕΙΤΟΥΡΓΙΑ ΚΑΙ ΝΑ ΕΠΙΣΗΜΑΝΕΙ ΚΑΙΝΟΤΟΜΙΕΣ ΚΑΙ ΔΙΑΦΟΡΕΤΙΚΕΣ ΠΡΟΣΕΓΓΙΣΕΙΣ ΣΤΟΝ ΚΩΔΙΚΑ ΚΑΙ ΤΗΝ ΚΑΤΑΣΚΕΥΗ ΤΟΥ ΟΧΗΜΑΤΟΣ

ΚΑΘΕ ΟΜΑΔΑ ΕΧΕΙ ΔΙΚΑΙΩΜΑ ΝΑ ΚΑΝΕΙ ΑΠΟ ΜΙΑ ΕΡΩΤΗΣΗ ΣΤΗΝ ΟΜΑΔΑ ΠΟΥ ΠΑΡΟΥΣΙΑΖΕΙ ΜΕΤΑ ΤΟ ΤΕΛΟΣ ΤΗΣ ΠΑΡΟΥΣΙΑΣΗΣ

ΒΙΒΛΙΟΓΡΑΦΙΑ

- [1] Αντωνίου Α., Δημητριάδης Π., Καμπούρης Κ., Παπαμιχάλης Κ, Παπατσιμπα Λ. «Φυσική Γ' ΓΥΜΝΑΣΙΟΥ.» ebooks.edu.gr.
http://ebooks.edu.gr/ebooks/v/html/8547/2226/Fysiki_G-Gymnasiou_html-empl/index1.html
- [2] Editors of Encyclopedia Britannica. «Electric Charge.» britannica.com.
<https://www.britannica.com/science/electric-charge>
- [3] Dejan. «What is Electric Charge and How Electricity Work.» Howtomechatronics.com.
<https://howtomechatronics.com/learn/electricity/electric-charge-how-electricity-works/>
- [4] W. McAllister. «Electric Potential, Voltage.» khanacademy.org.
<https://www.khanacademy.org/science/electrical-engineering/ee-electrostatics/ee-fields-potential-voltage/a/ee-electric-potential-voltage>
- [5] «Resistor Color Code Calculator and Chart—4 Band, 5 Band, or 6 Band Resistors». allaboutcircuits.com. <https://www.allaboutcircuits.com/tools/resistor-color-code-calculator/>
- [6] The Engineering Mindset. *How Batteries Work - Battery electricity working principle* (Jul 19, 2020). [Online Video]. Available:
<https://www.youtube.com/watch?v=PXNKkcB0pI4>
- [7] Γ. Πλευριτάκης. «Πηγές ρεύματος και τάσης.» learnaboutelectronics.gr.
<https://learnelectronics.gr/%CF%80%CE%B7%CE%B3%CE%AD%CF%82-%CF%81%CE%B5%CF%8D%CE%BC%CE%B1%CF%84%CE%BF%CF%82-%CE%BA%CE%B1%CE%B9-%CF%84%CE%AC%CF%83%CE%B7%CF%82/>
- [8] «ΗΛΕΚΤΡΙΚΑ ΚΥΚΛΩΜΑΤΑ Ι Κεφάλαιο 2 Στοιχεία Ηλεκτρικών Κυκλωμάτων (Κυκλωματικά Στοιχεία).» eclass.uowm.gr.
https://eclass.uowm.gr/modules/document/file.php/ECE373/%CE%98%CE%95%CE%A9%CE%A1%CE%99%CE%91/%CE%A3%CE%B7%CE%BC%CE%B5%CE%B9%CF%8E%CF%83%CE%B5%CE%B9%CF%82%20%CE%B4%CE%B9%CE%B1%CE%BB%CE%AD%CE%BE%CE%B5%CF%89%CE%BD/CH02LN_2022_23.pdf

- [9] Γ. Λιαπέρδος. «Εισαγωγή στην Ηλεκτρονική.» liaperdos.gr.
https://liaperdos.gr/public/intro_to_electronics/xhtml/chapter_01.xhtml#section_01_1_0
- [10] M. Rainer. «How Electronic Components Work.» blog.mide.com.
<https://blog.mide.com/how-electronic-components-work>
- [11] «LED (Light-emitting diode) explained.» soldered.com.
<https://soldered.com/learn/led-light-emitting-diode-explained/>
- [12] N. Davis. «An Introduction to Ground: Earth Ground, Common Ground, Analog Ground, and Digital Ground.» allaboutcircuits.com.
<https://www.allaboutcircuits.com/technical-articles/an-introduction-to-ground/>
- [13] Gadgetronicx. *What is SHORT CIRCUIT - Explained with Example | Basics of Electronics* (Jul 17, 2021). [Online Video]. Available:
<https://www.youtube.com/watch?v=dNnTubgY2gs>
- [14] G. Rizzoni, Θεωρία και Εφαρμογές της Επιστήμης του Ηλεκτρολόγου Μηχανικού, Πρώτος Τόμος: Ανάλυση Κυκλωμάτων και Σημάτων Αθήνα: Εκδόσεις Παπαζήση, 2005, pp. 1-175
- [15] South San Francisco Public Library. «Using a Breadboard.» eleccelerator.com.
https://eleccelerator.com/ssfpl_robotics_class_2018/connectingmoreparts.htm
- [16] T. Foxworth. «Hardware Fundamentals: What Exactly is a Microcontroller?.» freecodecamp.org. <https://www.freecodecamp.org/news/hardware-fundamentals-what-exactly-is-a-microcontroller-8a502a3650dc/>
- [17] M. Gudino. «Introduction to Microcontrollers.» arrow.com.
<https://www.arrow.com/en/research-and-events/articles/engineering-basics-what-is-a-microcontroller>
- [18] A. Armenta. «Introduction to Arduino: History, Hardware, and Software.» control.com.
<https://control.com/technical-articles/introduction-to-arduino-history-hardware-and-software/>
- [19] «What is Arduino?.» arduino.cc.
<https://www.arduino.cc/en/Guide/Introduction>
- [20] Arduino Team. «One Board to Rule Them All: History of the Arduino UNO.» blog.arduino.cc. <https://blog.arduino.cc/2021/12/09/one-board-to-rule-them-all-history-of-the-arduino-uno/>
- [21] «UNO R3.» docs.arduino.cc. <https://docs.arduino.cc/hardware/uno-rev3>

- [22] «A Closer Look Inside the Arduino Uno.» circuitcrush.com.
<https://www.circuitcrush.com/look-inside-arduino/>
- [23] «Basics of PWM (Pulse Width Modulation).» docs.arduino.cc.
<https://docs.arduino.cc/learn/microcontrollers/analog-output>
- [24] «Language Reference.» arduino.cc.
<https://www.arduino.cc/reference/en/>
- [25] b_e_n. «Data Types in Arduino.» learn.sparkfun.com.
<https://learn.sparkfun.com/tutorials/data-types-in-arduino/all>
- [26] «Installing Libraries.» docs.arduino.cc.
<https://docs.arduino.cc/software/ide-v1/tutorials/installing-libraries>
- [27] Derek Banas. *Arduino Programming* (Aug 15, 2018). [Online Video]. Available:
https://www.youtube.com/watch?v=QO_Jlz1qpDw&t=189
- [28] E. Πουλάκης, *Προγραμματίζοντας με τον Μικροελεγκτή Arduino*, Ηράκλειο: Creative Commons, 2015
<https://users.sch.gr/manpoul/docs/arduino/ProgrammingArduino.pdf>
- [29] «Serial.begin().» arduino.cc.
<https://www.arduino.cc/reference/en/language/functions/communication/serial/begin/>
- [30] «Serial.println().» Arduino.cc.
<https://www.arduino.cc/reference/en/language/functions/communication/serial/println/>
- [31] Bas on Tech. *FADING LED ON BREADBOARD WITH PWM - Arduino tutorial #3* (Mar 2, 2018). [Online Video]. Available:
<https://www.youtube.com/watch?v=1fhRf-0RxyQ>
- [32] S. Fitzgerald & M. Shiloh, *The Arduino Projects Book*, Torino: Arduino LLC, 2012
- [33] «Working Principle of DC Motor.» robu.in.
<https://robu.in/working-principle-of-dc-motor/>
- [34] Dejan. «L298N Motor Driver – Arduino Interface, How It Works, Codes, Schematics.» howtomechatronics.com.
<https://howtomechatronics.com/tutorials/arduino/arduino-dc-motor-control-tutorial-l298n-pwm-h-bridge/>
- [35] Project Weekend. L298N | How to Control DC Motor with Arduino | Motor Speed and Direction Control (May 15, 2022). [Online Video]. Available:
<https://www.youtube.com/watch?v=5RMwtuV7BM0>

- [36] cornelam. «Arduino Servo Motors.» instructables.com.
<https://www.instructables.com/Arduino-Servo-Motors/>
- [37] «Servo Motor Basics with Arduino.» docs.arduino.cc.
<https://docs.arduino.cc/learn/electronics/servo-motors/>
- [38] «Servos Explained.» sparkfun.com.
<https://www.sparkfun.com/servos>
- [39] «Ultrasound Physics.» fusic-sy.co.uk. <https://fusic-sy.co.uk/education/ultrasound-physics/>
- [40] «How to Build an HC-SR04 Distance Sensor Circuit.»
learningaboutelectronics.com.
<https://www.learningaboutelectronics.com/Articles/HC-SR04-ultrasonic-distance-sensor-circuit.php>
- [41] «HCSR04 Ultrasonic Sensor.» Arduino.cc.
<https://www.arduino.cc/reference/en/libraries/hcsr04-ultrasonic-sensor/>
- [42] «Ultrasonic.» github.com.
<https://github.com/ErickSimoes/Ultrasonic>
- [43] «Servo.» github.com
<https://github.com/arduino-libraries/Servo>
- mathima 7
<https://www.instructables.com/Arduino-Self-Driving-Car/>
<https://docs.arduino.cc/learn/electronics/servo-motors/> μαθημα 5

ΠΗΓΕΣ ΕΙΚΟΝΩΝ

ΚΕΦΑΛΑΙΟ 1 - ΜΑΘΗΜΑ 1^ο: Ο ΗΛΕΚΤΡΙΣΜΟΣ

*Εικόνες για τις οποίες δεν αναφέρεται πηγή παρήχθησαν από τον συγγραφέα

Εικόνα 1.1: <https://howtomechatronics.com/learn/electricity/electric-charge-how-electricity-works/>

Εικόνα 1.2: <https://slideplayer.com/slide/9697602/>

Εικόνα 1.3: <https://www.electricity-magnetism.org/electrostatics/coulombs-law/coulomb-constant/>

Εικόνα 1.4: <https://howtomechatronics.com/learn/electric-field/>

Εικόνα 1.5: https://sdsu-physics.org/physics180/physics180B/Chapters/electric_potential.htm

Εικόνα 1.6: <https://docplayer.gr/4331889-Kefalaio-2-ilektriko-reyma-meros-1-o.html>

Εικόνα 1.7: https://www.researchgate.net/figure/Graphite-rod-from-the-Dry-cell-battery_fig1_295898491

Εικόνα 1.8: https://www.dit.uoi.gr/e-class/modules/document/file.php/113/04_%CE%97%CE%9C.pdf

Εικόνα 1.9: https://www.planitikos.gr/2012/04/blog-post_5251.html

Εικόνα 1.10: <https://electrical-information.com/resistor-color-code/>

Εικόνα 1.11: <http://pc164.materials.uoi.gr/dpapageo/courses/emo/1-ohm.pdf>

Εικόνα 1.12: <https://www.guyhowto.com/ohms-law-explained-with-formula-and-examples/>

ΜΑΘΗΜΑ 2^ο: ΗΛΕΚΤΡΟΝΙΚΑ ΚΥΚΛΩΜΑΤΑ

Εικόνα 2.1: <https://unacademy.com/content/jee/study-material/physics/electric-circuit-and-electrical-symbols/>

Εικόνα 2.2: <https://www.onlinewebfonts.com/icon/527221>

Εικόνα 2.3: <https://www.shutterstock.com/cs/image-vector/alternating-current-symbol-ac-voltage-source-1786279412>

Εικόνα 2.4: https://en.m.wikipedia.org/wiki/File:Current_Source.svg

Εικόνα 2.5: https://stock.adobe.com/gr_en/search?k=resistor+symbol

Εικόνα 2.6: <https://resources.pcb.cadence.com/blog/2023-electrolytic-capacitor-use-and-design>

Εικόνα 2.7: <https://info.triadmagnetics.com/blog/chokes-vs-inductors-whats-the-difference>

Εικόνα 2.8: <https://byjus.com/physics/diodes/>

Εικόνα 2.9: <https://soldered.com/learn/led-light-emitting-diode-explained/>

Εικόνα 2.10: <https://www.allaboutcircuits.com/technical-articles/an-introduction-to-ground/>

Εικόνα 2.11: <https://www.ahouseonarock.com/what-is-a-short-circuit/>

Εικόνα 2.12: <https://stock.adobe.com/search?k=series+circuit>

Εικόνα 2.13: <https://www.amazon.com/BB830-Solderless-Plug-BreadBoard-tie-Points/dp/B0040Z4QN8>

Εικόνα 2.14:

https://eleccelerator.com/ssfpl_robotics_class_2018/connectingmoreparts.htm

Εικόνα 2.15: https://www.researchgate.net/figure/The-first-series-parallel-configuration-on-the-breadboard-configuration-2_fig2_45926012

ΜΑΘΗΜΑ 3^ο: ΕΙΣΑΓΩΓΗ ΣΤΟ ARDUINO

Εικόνα 3.1: <https://www.freecodecamp.org/news/hardware-fundamentals-what-exactly-is-a-microcontroller-8a502a3650dc/>

Εικόνα 3.2:

<https://electrobot.gr/image/cache/catalog/products/Arduino/Boards/UNO%20R3/arduino-uno-r3-600x800.jpg>

Εικόνα 3.3: <https://circuitdigest.com/article/everything-you-need-to-know-about-arduino-uno-board-hardware>

Εικόνα 3.4: <https://circuitdigest.com/tutorial/what-is-pwm-pulse-width-modulation>

Εικόνα 3.5: https://cdn-media-1.freecodecamp.org/images/1*tJrDBAK3Gi1gd3EucYqsnw.jpeg

Εικόνα 3.6: <https://www.arduino.cc/en/software>

Εικόνα 3.7: <https://create.arduino.cc/editor>

ΜΑΘΗΜΑ 5^ο: ΕΛΕΓΧΟΣ ΚΙΝΗΤΗΡΩΝ ΜΕ ΤΟΝ ARDUINO

Εικόνα 5.1: <https://www.adafruit.com/product/2941>

Εικόνα 5.2: <https://i.stack.imgur.com/s37EI.jpg>

Εικόνα 5.3: <https://components101.com/modules/l293n-motor-driver-module>

Εικόνα 5.15: <https://circuitdigest.com/microcontroller-projects/interface-servo-motor-with-arduino>

Εικόνα 5.16: <https://3dprint.com/301411/stepper-motors-vs-servo-motors-what-makes-a-better-fdm-3d-printer/>

ΜΑΘΗΜΑ 6^ο: Η ΜΟΝΑΔΑ ΜΕΤΡΗΣΗΣ ΑΠΟΣΤΑΣΕΩΝ

Εικόνα 6.1: <https://topelectronics.gr/images/detailed/3/hc-sr04-02.jpg>

Εικόνα 6.2: <https://www.quora.com/How-do-you-explain-the-PulseIn-function-with-Arduino-and-ultrasonic-sensor-configurations-Arduino-Uno-programming-C-timers-programming-sequence-Arduino>

Εικόνα 6.3: <https://medium.com/@bharanid410/iot-project-7c2a508b4dfe>

Εικόνα 6.4: <https://www.tinyosshop.com/hc-sr04-ultrasonic-sensor>

Εικόνα 6.5: <https://forum.arduino.cc/t/hc-sr04-and-newping-shows-crap/594466>

Εικόνα 6.6: <https://www.learningaboutelectronics.com/Articles/HC-SR04-ultrasonic-distance-sensor-circuit.php>

ΠΑΡΑΡΤΗΜΑ Α – Κώδικας του Οχήματος

```
#include <Ultrasonic.h> // εισαγωγή της βιβλιοθήκης Ultrasonic
#include <Servo.h> // εισαγωγή της βιβλιοθήκης Servo

Ultrasonic ultrasonic(12, 13); //δημιουργία του αντικείμενου
Ultrasonic
Servo servo1; // δημιουργία του αντικειμένου servo1

int thesi = 90; // δηλωση μεταβλητής που θα ορίζει την θέση του servo
TIMES: 0, 90, 180
int apostasi; // απόσταση εμποδίου μπροστά
int deksia; // απόσταση εμποδίου δεξιά
int aristera; // απόσταση εμποδίου αριστερά

const int kinitiras1_f = 5; // το κόκκινο καλώδιο του κινητήρα 1
(δεξιός απο την οπίσθια πλευρά) συνδέεται στον ακροδέκτη 5
const int kinitiras1_b = 6; // το μαύρο καλώδιο του κινητήρα 1
(δεξιός απο την οπίσθια πλευρά) συνδέεται στον ακροδέκτη 6
const int kinitiras2_b = 10; // το μαύρο καλώδιο του κινητήρα 2
(αριστερός απο την οπίσθια πλευρά) συνδέεται στον ακροδέκτη 10
const int kinitiras2_f = 11; // το κόκκινο καλώδιο του κινητήρα 2
(αριστερός απο την οπίσθια πλευρά) συνδέεται στον ακροδέκτη 11*/

void setup() {
  // put your setup code here, to run once:
  pinMode(kinitiras1_f, OUTPUT); // Όλοι οι ακροδέκτες των κινητήρων
δηλώνονται ως OUTPUT
  pinMode(kinitiras1_b, OUTPUT);
  pinMode(kinitiras2_f, OUTPUT);
  pinMode(kinitiras2_b, OUTPUT);
  servo1.attach(3); /* το αντικείμενο servo1 δηλώνεται στο pin 3 άρα
θα πρέπει και ο
κινητήρας servo (το κίτρινο καλώδιο να συνδεθεί στο pin 3 του
board)*/
  servo1.write(90);
}

void loop() {
  // put your main code here, to run repeatedly:
  apostasi = ultrasonic.read(); // μέτρηση απόστασης σε CM
  delay(100);
  servo1.write(90); // ο servo πηγαίνει στη θέση 90
  delay(100);
  if (apostasi > 15){ // αν η απόσταση είναι μεγαλύτερη των 15cm το
οχήμα συνεχίζει ευθεία
```

```

forward();
}

else if (apostasi <= 15){ //αν η απόσταση είναι μικρότερη από 15cm
stop(); // στάση μέχρι τον υπολογισμό των αποστάσεων
for(thesi = 0; thesi < 180; thesi += 1){ //δομή επανάληψης - ο
αισθητήρας σκανάρει από 0 - 180 μοίρες με βήμα 1
servo1.write(thesi); // ο servo πάει στη θέση που ορίζει η δομή
επανάληψης
delay(10); //αναμονή 10 ms για πιο ομαλή μετάβαση της μονάδας
μέτρησης
}

aristera = ultrasonic.read(); //μετράει την απόσταση από εμπόδια
στη αριστερή πλευρά και καταχωρεί την τιμή στη μεταβλητή
delay(100);
for(thesi = 180; thesi >= 0; thesi-=1){ // δομή επανάληψης - ο
αισθητήρας σκανάρει από 180 - 0 μοίρες με βήμα -1

servo1.write(thesi); // ο servo πάει στη θέση που
ορίζει η δομή επανάληψης
delay(10); //αναμονή 10 ms για πιο ομαλή μετάβαση της μονάδας
μέτρησης
}

deksia = ultrasonic.read(); //μετράει την απόσταση από εμπόδια στην
δεξιά πλευρά και καταχωρεί την τιμή στη μεταβλητή
delay(100);
servo1.write(90); // η μονάδα επιστρέφει στην εμπρόσθια θέση 90
delay(100);

if (aristera < deksia){ // έλεγχος για το ποιά μεταβλητή είναι
μεγαλύτερη
right(); //στροφή δεξιά αν aristera < deksia
delay(600); // καθυστέρηση έτσι ώστε το όχημα να στρίψει επαρκώς

}
else if (aristera > deksia){ //περίπτωση aristera > deksia
left(); // στροφή αριστερά
delay(600); // καθυστέρηση έτσι ώστε το όχημα να στρίψει επαρκώς
}
else if (aristera <=15 && deksia <=15){ /*αν aristera<=15 ΚΑΙ
deksia<=15
(όσο περίπου το μήκος του οχήματος)*/
backward(); //οπισθοχώρηση - δεν υπάρχει χώρος για στροφή
delay(1000);
left(); //στροφή αριστερά
delay(600);
}

```

```

    }
}
} // τέλος επανάληψης loop()

// ΑΚΟΛΟΥΘΕΙ Ο ΟΡΙΣΜΟΣ ΤΩΝ ΣΥΝΑΡΤΗΣΕΩΝ ΚΙΝΗΣΗΣ ΠΟΥ ΕΪΔΑΜΕ ΣΤΟ 5ο
ΜΑΘΗΜΑ
void forward() { // ΕΜΠΡΟΣ
    digitalWrite(kinitiras1_f, HIGH);
    digitalWrite(kinitiras1_b, LOW);
    digitalWrite(kinitiras2_f, LOW);
    digitalWrite(kinitiras2_b, HIGH);
}

void backward() { // ΠΙΣΩ
    digitalWrite(kinitiras1_f, LOW);
    digitalWrite(kinitiras1_b, HIGH);
    digitalWrite(kinitiras2_f, HIGH);
    digitalWrite(kinitiras2_b, LOW);
}

void right() { // ΔΕΞΙΑ
    digitalWrite(kinitiras1_f, LOW);
    digitalWrite(kinitiras1_b, HIGH);
    digitalWrite(kinitiras2_f, LOW);
    digitalWrite(kinitiras2_b, HIGH);
}

void left() { // ΑΡΙΣΤΕΡΑ
    digitalWrite(kinitiras1_f, HIGH);
    digitalWrite(kinitiras1_b, LOW);
    digitalWrite(kinitiras2_f, HIGH);
    digitalWrite(kinitiras2_b, LOW);
}

void stop() { // ΣΤΑΣΗ
    digitalWrite(kinitiras1_f, LOW);
    digitalWrite(kinitiras1_b, LOW);
    digitalWrite(kinitiras2_f, LOW);
    digitalWrite(kinitiras2_b, LOW);
}
}

```