



ΤΕΧΝΟΛΟΓΙΚΟ  
ΕΚΠΑΙΔΕΥΤΙΚΟ  
ΙΔΡΥΜΑ  
—■—  
ΤΕΙ ΗΠΕΙΡΟΥ

**ΤΕΧΝΟΛΟΓΙΚΟ ΕΚΠΑΙΔΕΥΤΙΚΟ ΙΔΡΥΜΑ ΗΠΕΙΡΟΥ**

**ΣΧΟΛΗ ΤΕΧΝΟΛΟΓΙΚΩΝ ΕΦΑΡΜΟΓΩΝ**

**ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ Τ.Ε.**

## **ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ**

**Επιβλέπων:** Αγγέλης Κωνσταντίνος, Αναπληρωτής Καθηγητής

---

*«Μελέτη και υλοποίηση συστήματος ελέγχου  
βηματικού κινητήρα με χρήση ARDUINO»*

---

**Σιατής Χρήστος**

**ΑΜ: 8366**

## **ΠΕΡΙΛΗΨΗ**

*Σε αυτή την εργασία θα ασχοληθούμε με την ανάπτυξη ενός συστήματος ελέγχου βηματικού κινητήρα (stepper motor) με τη χρήση ενός μικροελεγκτή Arduino Uno.*

*Θα αναλύσουμε τα χαρακτηριστικά και άλλων ηλεκτροκινητήρων με σκοπό να δούμε τις διαφορές τους και να παρουσιάσουμε τις ιδιότητες και τις ιδιαιτερότητές τους.*

# **ΠΕΡΙΕΧΟΜΕΝΑ**

## **Κεφάλαιο 1 – ΗΛΕΚΤΡΟΚΙΝΗΤΗΡΕΣ**

- 1.1. Εισαγωγή στους ηλεκτροκινητήρες
- 1.2. Χαρακτηριστικά ηλεκτρικών κινητήρων
- 1.3. Ηλεκτρικοί βηματικοί κινητήρες (Stepper motors)
- 1.4. Ηλεκτρικοί κινητήρες σέρβο (R/C motors)
- 1.5. Ηλεκτροκινητήρες υψίσουχοι
- 1.6. Επαγωγικοί κινητήρες και inverter

## **Κεφάλαιο 2 - ΜΙΚΡΟΕΛΕΓΚΤΕΣ**

- 2.1 Ιστορία μικροεπεξεργαστών-μικροελεγκτών
- 2.2 Εξέλιξη μικροεπεξεργαστών
- 2.3 Διαφορές μικροεπεξεργαστή-μικροελεγκτή
- 2.4 Κατηγορίες μικροελεγκτών
- 2.5 Κατασκευαστές μικροελεγκτών
- 2.6 Γλώσσες προγραμματισμού μικροελεγκτών

## **Κεφάλαιο 3 – ARDUINO**

- 3.1 Εισαγωγή στο Arduino
- 3.2 Ιστορικό του Arduino
- 3.3 Δυνατότητες και πλεονεκτήματα του Arduino
- 3.4 Εκδόσεις Arduino
- 3.5 Εφαρμογές Arduino
- 3.6 Arduino Shields
- 3.7 Βασικά χαρακτηριστικά του Arduino Uno
- 3.8 Γλώσσα Arduino

## **Κεφάλαιο 4 – Υλοποίηση συστήματος ελέγχου**

- **ΑΝΑΦΟΡΕΣ**

# **ΚΕΦΑΛΑΙΟ 1 - ΗΛΕΚΤΡΟΚΙΝΗΤΗΡΕΣ**

## 1.1 Εισαγωγή στους ηλεκτροκινητήρες

Ο ηλεκτρικός κινητήρας ή ηλεκτροκινητήρας, είναι διάταξη που χρησιμοποιείται για την μετατροπή της ηλεκτρικής ενέργειας σε μηχανική !!!

Η αρχή λειτουργίας του ηλεκτρικού κινητήρα είναι η δύναμη [1] Laplace. Όταν ένας αγωγός από τον οποίο διαρρέει ηλεκτρικό ρεύμα βρεθεί μέσα σε ένα μαγνητικό πεδίο ασκείται πάνω του δύναμη ίση με:

$$F = I * \lambda * B * \eta \mu \phi$$

Όπου:

- $I$  = Ένταση Ρεύματος
- $\lambda$  = Μήκος Αγωγού
- $B$  = Ένταση Μαγνητικού πεδίου
- $\phi$  = η γωνία που σχηματίζει ο αγωγός με τη διεύθυνση των δυναμικών γραμμών (  $B$  )

Οι ηλεκτρικοί κινητήρες αποτελούνται από:

- Τον δρομέα

Ο δρομέας αποτελείται από τον ηλεκτροφόρο αγωγό ο οποίος είναι τοποθετημένος σε πυκνές περιελίξεις (σπείρες) ώστε να περιέχει όσο μεγαλύτερο μήκος αγωγού γίνεται για δεδομένο όγκο.

- Τον στάτη

Ο στάτης αποτελείται από μόνιμους ή τεχνητούς μαγνήτες οι οποίοι δημιουργούν το μαγνητικό πεδίο.

- Τις ψήκτρες

Οι ψήκτρες έρχονται σε επαφή με τον δρομέα τροφοδοτώντας τον με ρεύμα.

Τα απαραίτητα στοιχεία για κάθε ηλεκτροκινητήρα τα οποία και τον προσδιορίζουν εμπορικά είναι:

1. Η απαιτούμενη τάση για την τροφοδοσία του σε βολτ (V),

2. Το είδος της απαιτούμενης τάσης, συνεχές ή εναλλασσόμενο ρεύμα (DC ή AC) και στη 2η περίπτωση, μονοφασικό (1PH) ή τριφασικό (3PH),
3. Η συχνότητα του εναλλασσόμενου ρεύματος, εφόσον πρόκειται για ηλεκτροκινητήρα AC και προφανώς σε κύκλους ανά δευτερόλεπτο κ/δ (c/s) ή Hertz. Πολλές φορές χρησιμοποιείται το σύμβολο ( ~ ) αντί του κ/δ.
4. Η ισχύς του κινητήρα σε Βατ ή ίππους (W ή HP)
5. Η ένταση του ρεύματος σε αμπέρ που διαρρέει τον κινητήρα, και
6. Η αποκτώμενη ταχύτητα περιστροφής του άξονα του κινητήρα σε στροφές ανά λεπτό (RPM).

Όλα τα παραπάνω στοιχεία φέρονται χαραγμένα, από τους κατασκευαστές, σε ειδική ενσωματωμένη στον ηλεκτροκινητήρα πινακίδα, καθώς και ο αριθμός της έγκρισης του Υπουργείου Βιομηχανίας για εμπορική διάθεση ή άλλα σύμβολα πιστοποίησης ασφαλούς λειτουργίας.

Οι ηλεκτροκινητήρες διακρίνονται σε "συνεχούς ρεύματος" (DC motors) και σε "εναλλασσόμενου ρεύματος" (AC motors).

Η χρήση των ηλεκτροκινητήρων σήμερα είναι ευρύτατα διαδεδομένη, ξεκινώντας από τις οικιακές συσκευές (ψυγείο, κλιματιστικό, μπλέντερ) συνεχίζοντας στις βιομηχανικές (τόρνοι, ανυψωτικές διατάξεις, αντλίες) και καταλήγοντας στις μετακινήσεις (ηλεκτρικοί σιδηρόδρομοι, υβριδικά αυτοκίνητα) μπορούμε εύκολα να κατανοήσουμε ότι είναι άρρηκτα συνδεδεμένες με την καθημερινή ζωή, σε πολλές περιπτώσεις ακόμα και σε βιοτικής σημασίας πρωτογενείς ανάγκες (τεχνητή καρδιά, ρομποτικοί βραχίονες νευροχειρουργικής, αντλίες συσκευών αιμοκάθαρσης, αυτοκινούμενες αναπηρικές καρέκλες). Οι δε χρησιμοποιούμενοι στις βιομηχανίες είναι εκατοντάδων ίππων.

## 1.2 Χαρακτηριστικά ηλεκτρικών κινητήρων

- **Τάση λειτουργίας**, μετράται σε Volts και είναι η τάση που πρέπει να έχει το ηλεκτρικό ρεύμα τροφοδοσίας ώστε ο κινητήρας να λειτουργεί σωστά.
- **Ένταση του ρεύματος**, μετράται σε Amps και είναι η ένταση του ρεύματος που διαρρέει τον κινητήρα όταν αυτός λειτουργεί. Η ένταση αυτή είναι ανάλογη του φορτίου του κινητήρα. Η ελάχιστη τιμή της αντιστοιχεί στην ελεύθερη περιστροφή του κινητήρα. Εάν ο κινητήρας έχει φορτίο η τιμή αυτή αυξάνεται. Για κάποια τιμή του φορτίου ο κινητήρας σταματά να περιστρέφεται, λόγω της μεγάλης αντίστασης, οπότε και η τιμή της έντασης μεγιστοποιείται. Η μέγιστη αυτή τιμή είναι ένα από τα σημαντικά χαρακτηριστικά του κινητήρα που πρέπει να είναι γνωστό για τη σωστή επιλογή της τροφοδοσίας του.
- **Ταχύτητα**, στους σέρβο κινητήρες μετράται σε στροφές ανά λεπτό και στους βηματικούς κινητήρες σε βήματα ανά δευτερόλεπτο. Πρόκειται για την ταχύτητα περιστροφής του άξονα του κινητήρα, όταν ο κινητήρας λειτουργεί υπό κανονική ηλεκτρική τάση και με δεδομένο φορτίο.
- **Ροπή**, μετράται σε Nm (Newton meters) και είναι η ροπή που παράγει ο κινητήρας στις διάφορες ταχύτητες περιστροφής του. Η μέγιστη τιμή ροπής κάθε κινητήρα, που ονομάζεται ροπή ακινητοποίησης, είναι η ροπή που παράγει όταν το φορτίο που αντιμετωπίζει είναι τόσο μεγάλο, ώστε να τον ακινητοποιεί.

## 1.3 Ηλεκτρικοί βηματικοί κινητήρες (Stepper motors)



Εικόνα 1.Βηματικός κινητήρας

Ο βηματικός κινητήρας είναι ένας σταθερός μετατροπέας [2] παραγωγής δύναμης, όπου η δύναμη ορίζεται ως το γινόμενο της ροπής με την ταχύτητα. Αυτό σημαίνει ότι η ροπή των μηχανών είναι αντίστροφη της ταχύτητας. Για να βοηθήσουμε να καταλάβουμε γιατί η δύναμη ενός βηματικού κινητήρα είναι ανεξάρτητη από την ταχύτητα, πρέπει να κατασκευάσουμε (μεταφορικά) έναν ιδανικό βηματικό κινητήρα.

Ένας ιδανικός βηματικός κινητήρας θα είχε μη μηχανική τριβή, η ροπή του θα ήταν ανάλογη προς τις αμπέρ-στροφές και το μόνο ηλεκτρικό χαρακτηριστικό του θα ήταν η αυτεπαγωγή. Οι αμπέρ-στροφές απλά σημαίνουν ότι η ροπή είναι ανάλογη προς τον αριθμό στροφών του καλωδίου στο στάτη της μηχανής πολλαπλασιαζόμενο με το ρεύμα που περνά μέσω εκείνων των στροφών του καλωδίου.

Οποτεδήποτε υπάρχουν στροφές ενός καλωδίου γύρο από ένα μαγνητικό υλικό όπως ο σίδηρος στο στάτη της μηχανής, τότε αποκτάει μια ηλεκτρική ιδιότητα, την αυτεπαγωγή. Η αυτεπαγωγή περιγράφει την ενέργεια που αποθηκεύεται σε ένα μαγνητικό πεδίο όταν περάσει ρεύμα μέσω μιας σπείρας αυτού του καλωδίου.

Η αυτεπαγωγή ( $L$ ) έχει μια ιδιότητα που ονομάζετε επαγωγική άεργη αντίσταση, η οποία για τους σκοπούς αυτής της συζήτησης μπορεί να θεωρηθεί ως αντίσταση ανάλογη προς τη συχνότητα και επομένως της ταχύτητας του κινητήρα.

Σύμφωνα με το νόμο του Ohm, το ρεύμα είναι ίσο με την τάση που διαρρέει μια αντίσταση προς την αντίσταση αυτή. Σε αυτήν την



περίπτωση αντικαθιστούμε την επαγωγική άεργη αντίσταση με την αντίσταση στο νόμο του Ohm και καταλήγουμε πως το ρεύμα των κινητήρων είναι αντίστροφο της ταχύτητας τους.

Δεδομένου ότι η ροπή είναι ανάλογη προς τις αμπερ-στροφές (ρεύμα επί των αριθμό στροφών του καλωδίου στο τύλιγμα), και το ρεύμα είναι αντίστροφο της ταχύτητας, η ροπή πρέπει επίσης να είναι αντίστροφη της ταχύτητας.

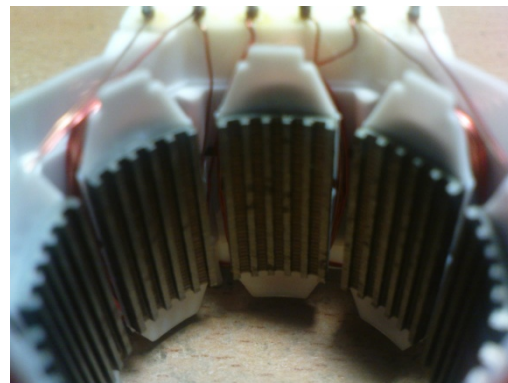
Σε έναν ιδανικό βηματικό κινητήρα, όσο η ταχύτητα προσεγγίζει το μηδέν τόσο η ροπή θα προσέγγιζε το άπειρο ενώ με άπειρη ταχύτητα η ροπή θα ήταν μηδέν. Επειδή το ρεύμα είναι ανάλογο προς τη ροπή, το ρεύμα θα ήταν επίσης άπειρο σε μηδενική ταχύτητα. (Βλέπε Σχήμα 4.)

Ηλεκτρικά, μια πραγματική μηχανή διαφέρει από ένα ιδανικό κινητήρα καταρχήν από την ύπαρξη μιας διάφορης του μηδενός αντίστασης τυλίγματος. Επίσης ο σίδηρος στη μηχανή υπόκειται στο μαγνητικό κορεσμό, καθώς επίσης και απώλεια ρεύματος στον στροβιλισμό και λόγω υστέρησης.

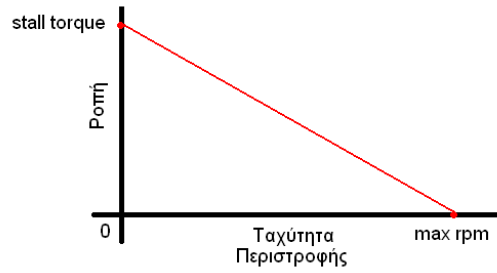
Ο μαγνητικός κορεσμός θέτει ένα όριο στο ρεύμα και στην αναλογικότητα της ροπής ενώ ο στροβιλισμός και η υστέρηση (απώλειες σιδήρου) μαζί με την αντίσταση του τυλίγματος (απώλειες χαλκού) προκαλούν έκλυση θερμότητας.



**Εικόνα 2.** Μέρη βηματικού κινητήρα

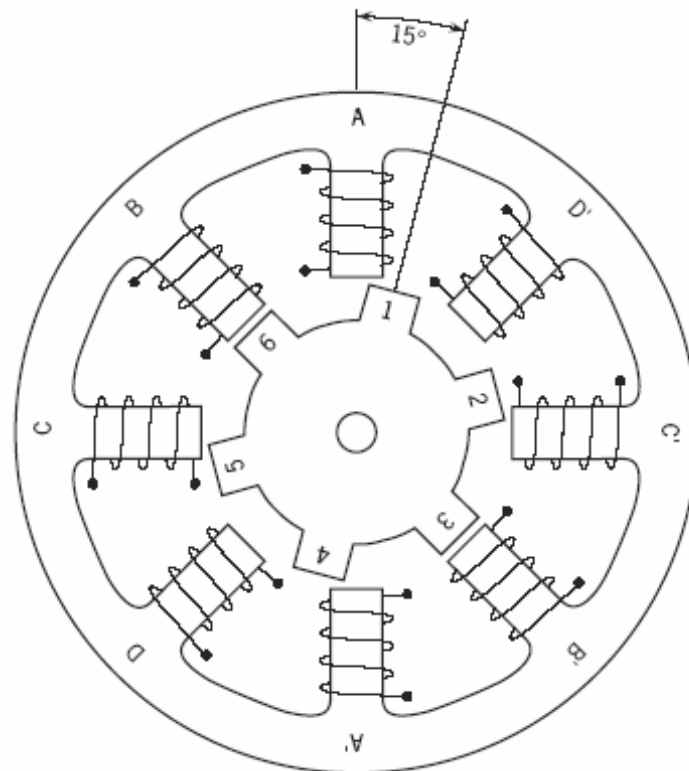


**Εικόνα 3.** Στάτης βηματικού κινητήρα



**Εικόνα 4.** Ταχύτητα-Ροπή

Ο κινητήρας αυτός αποτελείται από έναν ρότορα μαλακού σιδήρου με οδοντώσεις και έναν στάτορα με τέσσερα ζεύγη ηλεκτρομαγνητών: A και A', B και B', C και C', D και D'. Για να κινηθεί ο ρότορας εφαρμόζεται διαδοχικά σε κάθε ένα από τα παραπάνω ζεύγη μια τάση ηλεκτρικού ρεύματος. Όταν το ηλεκτρικό ρεύμα μεταφέρεται από το ένα ζεύγος ηλεκτρομαγνητών στο διπλανό του ο ρότορας μετατοπίζεται κατά 15 μοίρες λόγω των μαγνητικών δυνάμεων που εφαρμόζονται σε αυτόν. Η γωνία αυτή ονομάζεται **βήμα** του κινητήρα. (Βλέπε Σχήμα 5)



**Εικόνα 5.** Διατομή βηματικού κινητήρα

## Πλεονεκτήματα βηματικού κινητήρα

- Σε αντίθεση με τους κινητήρες συνεχούς ρεύματος, δεν χρειάζεται φρένα για να μένει ακίνητος ή για να επιβραδυνθεί.
- Στις μικρές ταχύτητες περιστροφής, αλλά και κατά την εκκίνησή του, παράγει μεγάλες τιμές ροπής.
- Είναι πολύ αξιόπιστος καθώς για τη λειτουργία του δεν απαιτούνται κινούμενες ηλεκτρικές επαφές, όπως στον κινητήρα συνεχούς ρεύματος και έτσι η διάρκεια ζωής του εξαρτάται μόνο από την αξιοπιστία του εδράνου κύλισης.
- Δεν απαιτείται χρήση αισθητήρων και κυκλωμάτων ανάδρασης για τον προσδιορισμό της θέσης του άξονα κίνησης.
- Ο βηματικός κινητήρας μπορεί να επιτύχει μεγάλο εύρος ταχυτήτων περιστροφής.
- Ο βηματικό κινητήρας μπορεί να επιτύχει πολύ χαμηλές ταχύτητες περιστροφής.

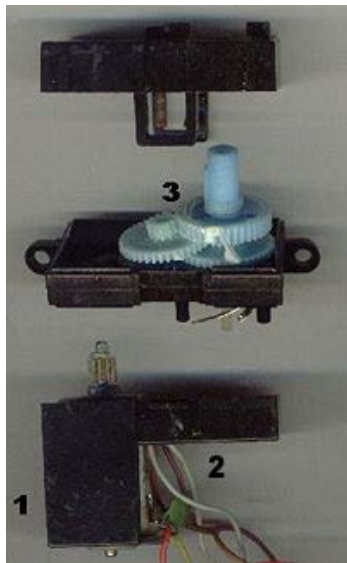
## Μειονεκτήματα βηματικού κινητήρα

- Θορυβώδης λειτουργία.
- Αδυναμία περιστροφής σε υψηλές ταχύτητες.
- Κατά τη μετακίνηση φορτίων μεγάλης μάζας μπορεί να μη σταματήσει ακαριαία ο κινητήρας, λόγω της αυξημένης αδράνειας.

## **1.4 Ηλεκτρικοί κινητήρες σέρβο (R/C motors)**

Το σέρβο είναι συσκευή που αποτελείται από έναν ηλεκτροκινητήρα [3] συνεχούς ρεύματος (1), ένα ηλεκτρονικό κύκλωμα που ελέγχει τη θέση του τελικού άξονα κίνησης (2) και ένα κιβώτιο υποβιβασμού της σχέσης μετάδοσης του κινητήρα (3). (Βλέπε σχήμα 6)

Ο τελικός άξονας κίνησης δεν εκτελεί πλήρεις περιστροφές, αλλά περιστρέφεται μεταξύ δύο ακραίων θέσεων A, B. Για τη λειτουργία του σέρβο απαιτείται η παροχή της κατάλληλης ηλεκτρικής τάσης αλλά και ενός σήματος ελέγχου που καθορίζει τη θέση περιστροφής του τελικού άξονα. (Βλέπε σχήμα 7)



**Εικόνα 6.** Τμήματα του σέρβο



**Εικόνα 7.** Κίνηση του σέρβο

Τα σέρβο χρησιμοποιούνται σε τηλεκατευθυνόμενα μοντέλα αλλά και σε πολλές ρομποτικές εφαρμογές.

### **Πλεονεκτήματα**

- Χαμηλό κόστος
- Μικρές διαστάσεις και εύχρηστο σχήμα:  
όλα τα μέρη ενός σέρβο περιβάλλονται από ένα συμπαγές περίβλημα από το οποίο εξέρχονται ο άξονας κίνησης
- Παράγουν υψηλές τιμές ροπής
- Δεν απαιτείται χρήση αισθητήρων και κυκλωμάτων ανάδρασης για τον προσδιορισμό της θέσης του άξονα κίνησης.

### **Μειονεκτήματα**

- Αδυναμία εκτέλεσης πλήρους και συνεχούς περιστροφής

## 1.5 Ηλεκτροκινητήρες υψίσυχνοι

Οι υψίσυχνοι κινητήρες χρησιμοποιούνται κυρίως για επεξεργασία υλικών, όπως: ξύλο, πλαστικό, γυαλί, μάρμαρο, καθώς και ελαφρών κραμάτων. Η μεγάλη γκάμα των κινητήρων, η τεχνογνωσία και η εμπειρία στον τομέα των κινητήρων αυτών, επιτρέπουν την ικανοποίηση των εφαρμογών-απαιτήσεων των πελατών με ιδιαίτερη αξιοπιστία.

Με εύρος ισχύος από **0.05-15 kW**, από **6.000-40.000 rpm (100-670 Hz)**, τα υλικά, η σχεδίαση και η συναρμολόγηση των κινητήρων, πρέπει να είναι προσεγμένα, καθώς οι υψηλές ταχύτητες πολλαπλασιάζουν και το πιο μικρό σφάλμα. Τα ρουλεμάν είναι κεραμικά, σχεδιασμένα και κεντραρισμένα, ώστε σε τόσο υψηλές ταχύτητες να μπορούν να στηρίξουν το φορτίο στο οποίο θα υποβληθούν.

Στις απλές εκδόσεις οι κινητήρες είναι σχεδιασμένοι για υψηλή απόδοση και ακρίβεια, καθώς και αντοχή σε μεγάλα αξονικά και ακτινικά φορτία.

Έτσι υπάρχουν κινητήρες με αυτόματη εναλλαγή εργαλείων για CNC μηχανές, και κινητήρες με κούλο άξονα, υδρόψυκτοι, για κατεργασία μαρμάρου και πέτρας.



Εικόνα 8. Υψίσυχνοι ηλεκτροκινητήρες

## **1.6 Επαγωγικοί κινητήρες και inverter**

Τα τελευταία χρόνια η τάση της βιομηχανίας είναι η χρήση όλο και λιγότερων κινητήρων συνεχούς ρεύματος (DC κινητήρες). Αυτό οφείλεται στο ότι οι dc κινητήρες είναι μεγάλοι , ακριβοί και χρειάζονται περιοδική συντήρηση (στα καρβουνάκια).

Ένας επαγωγικός κινητήρας λειτουργεί σαν ένας μετασχηματιστής. Όταν ο στάτορας συνδεθεί σε τριφασική παροχή , ένα μαγνητικό πεδίο αναπτύσσεται, το οποίο τον περιστρέφει στην συχνότητα της παροχής. Αυτό το πεδίο περνά από το κενό του αέρα στον στάτορα και τον ρότορα και προκαλεί ρεύμα το οποίο ρέει στα τυλίγματα του ρότορα. Αυτή η διαδικασία παράγει μια δύναμη στον ρότορα για όσο το ρεύμα αλληλεπιδρά με το μεταβαλλόμενο μαγνητικό πεδίο και το μοτέρ περιστρέφεται.

Οι επαγωγικοί κινητήρες είναι απλοί ,χαμηλού κόστους ,αξιόπιστοι και ευρέως χρησιμοποιούμενοι από τον κόσμο. Με σκοπό να ελέγξουμε την ταχύτητα ενός επαγωγικού κινητήρα, χρειάζεται ένας σύνθετος ελεγκτής τον οποίο ονομάζουμε ρυθμιστή στροφών ή inverter.

## **ΚΕΦΑΛΑΙΟ 2 - ΜΙΚΡΟΕΛΕΓΚΤΕΣ**

## **2.1 Ιστορία μικροεπεξεργαστών- μικροελεγκτών**

Η επιθυμία των κατασκευαστικών συστημάτων να [4] δημιουργήσουν συστήματα με περισσότερες δυνατότητες και με μικρότερο μέγεθος , οδήγησε στην ανάγκη για ενσωμάτωση όλων των λειτουργιών ενός υπολογιστή σε ένα ή μερικά ολοκληρωμένα κυκλώματα.

Η ανάπτυξη της τεχνολογίας που χρησιμοποιήθηκε για την κατασκευή των κυκλωμάτων αυτών είχε σαν αποτέλεσμα την ανάπτυξη του μικροεπεξεργαστή.

Οι μικροεπεξεργαστές είναι υπεύθυνοι για την έμπνευση και τη δημιουργία μερικών από τις μεγαλύτερες καινοτομίες στα συστήματα υπολογιστών.

Αυτές οι καινοτομίες περιλαμβάνουν τους embedded μικροελεγκτές , τους προσωπικούς υπολογιστές, τους σύγχρονους σταθμούς εργασίας, συσκευές χειρός και κινητές συσκευές (όπως τους επεξεργαστές των κινητών τηλεφώνων), servers (εξυπηρετητές) εφαρμογών και αρχείων , web servers για το internet, υπερυπολογιστές χαμηλού κόστους και ευρείας κλίμακας δίκτυα υπολογιστών. Οι μικροεπεξεργαστές είναι επεξεργαστές συνόλου εντολών (instruction set processors, ISP). Ένας ISP εκτελεί εντολές ενός προκαθορισμένου συνόλου εντολών. Η λειτουργικότητα του εξαρτάται ανάλογα από το σύνολο εντολών που είναι ικανός να εκτελέσει ο μικροεπεξεργαστής. Σε αυτό το σύνολο εντολών κωδικοποιούνται όλα τα προγράμματα που τρέχουν σε έναν μικροεπεξεργαστή. Αυτό το προκαθορισμένο σύνολο εντολών ονομάζεται επίσης αρχιτεκτονική συνόλου εντολών (instruction set architecture, ISA). Το ISA χρησιμεύει ως μια διασύνδεση ανάμεσα στο λογισμικό(software) και το υλικό(hardware), δηλαδή ανάμεσα στα προγράμματα και τους επεξεργαστές.



Με τον όρο μικροελεγκτής αναφερόμαστε σε έναν τύπο επεξεργαστή ο οποίος μπορεί να λειτουργήσει με ελάχιστα εξωτερικά εξαρτήματα λόγω των πολλών ενσωματωμένων υποσυστημάτων που διαθέτει. Ένας μικροελεγκτής είναι ένα ενσωματωμένο τσιπ(ολοκληρωμένο κύκλωμα) που αποτελεί συχνά μέρος ενός συστήματος. Όπως ένας τυπικός Η/Υ έτσι και ο μικροελεγκτής περιλαμβάνει CPU , RAM , ROM για αποθήκευση δεδομένων και λογισμικού αντίστοιχα, μνήμη flash για μόνιμη αποθήκευση, θύρες εισόδου-εξόδου, μετατροπέα αναλογικού σε ψηφιακό σήμα και το αντίστροφο και τέλος timers. Καθώς είναι σχεδιασμένος να εκτελεί μόνο μια συγκεκριμένη εργασία για τον έλεγχο ενός απλού συστήματος ,είναι πολύ μικρότερος και απλουστέρα σχεδιασμένος ώστε να μπορεί να περιλαμβάνει όλες τις λειτουργίες που απαιτούνται σε ένα μόνο ολοκληρωμένο κύκλωμα.

Χρησιμοποιείται ευρέως σε όλα τα ενσωματωμένα συστήματα έλεγχου χαμηλού και μεσαίου κόστους όπως για παράδειγμα αυτά που χρησιμοποιούνται σε αυτοματισμούς , ηλεκτρονικά προϊόντα (ψηφιακές μηχανές ,παιχνίδια,...), ηλεκτρικές συσκευές και κάθε είδους αυτοκινούμενα τροχοφόρα οχήματα. Στις μέρες μας η χρήση του μικροελεγκτή είναι καθολική για το λόγο ότι κάθε προϊόν το οποίο αλληλεπιδρά με έναν χρήστη περιλαμβάνει έναν μικροελεγκτή ,ο οποίος παίζει τον ρόλο του «εγκεφάλου» των ηλεκτρονικών κυκλωμάτων.

## 2.2 Εξέλιξη μικροεπεξεργαστών

Στις αρχές της δεκαετίας του 70 οι επεξεργαστές αποτελούνταν από δεκάδες πύλες και περίπλοκα κυκλώματα τα οποία με την παρατεταμένη χρήση τους εξέπεμπαν θερμότητα και έπιαναν πολύ χώρο. Καθώς το μέγεθος μειώθηκε οι επεξεργαστές ονομαστήκαν μικροεπεξεργαστές.

- Το 1971 παρουσιάζεται ο πρώτος μικροεπεξεργαστής Intel 4004 από τον Ted Hoff και τον συνεργάτη του Stan Mazor. Ο Intel 4004 ήταν ένας 4bit επεξεργαστής ο οποίος αποτελούνταν από 2300 τρανζίστορ με συχνότητα ρολογιού 108KHz ,εκτελούσε 60.000 πράξεις το δευτερόλεπτο και μπορούσε να δει 640 bytes μνήμης. Αρχικά εφαρμόστηκε για τη δημιουργία αριθμομηχανών ενώ το Νοέμβριο του 1971 η Intel ανακοίνωσε τον πρώτο μικροϋπολογιστή. Μέσα στην επόμενη χρονιά εμφανίστηκε ο διάδοχος του , ο 8008.
- Το 1974 εμφανίζεται ο 8bit μικροεπεξεργαστής της Intel ,8080 που ήταν αποτέλεσμα εξέλιξης του 8008. Αποτελούνταν από 6000 τρανζίστορ με συχνότητα λειτουργίας 2MHz. Την ίδια περίοδο η Motorola παρουσίασε τον 6800 που χρησιμοποιήθηκε σε υπολογιστές, σε όλα τα παιχνίδια pinball καθώς και σε βιομηχανικές κατασκευές ελέγχου. Είχε 4000 τρανζίστορ ,78 εντολές ,σήμα χρονισμού στα 1ή 2 MHz και 16bit πλάτος διαύλου διευθύνσεων.
- Το 1975 η Zilog φτιάχνει τον Z80 έναν 8bit μικροεπεξεργαστή βασισμένο στον 8080 του οποίου η γλώσσα μηχανής είναι υπερσύνολο αυτής του Intel 8080. Το σήμα χρονισμού ήταν στα 3.5MHz με 16bit πλάτος διευθύνσεων ενώ θα μπορούσε να δει 64Kbytes μνήμης. Κατά την δεκαετία του 80 είχε την μεγαλύτερη δημοτικότητα καθώς επικεντρώθηκε στο χαμηλό κόστος σε συνδυασμό με τη μικρή συσκευασία, τις χαμηλές απαιτήσεις και τον συνυπολογισμό των στοιχείων κυκλώματος που κανονικά θα έπρεπε να παρασχεθούν σε ένα χωριστό τσιπ.

- Το 1978 κάνουν τη εμφάνισή τους οι πρώτοι 16bit μικροεπεξεργαστές. Η Intel παρουσιάζει τον 8086/8088 του οποίου η συχνότητα λειτουργιάς έχει ανέβει στα 10 MHz και απαιτεί 29.000 τρανζίστορ . Αντίστοιχα η Motorola εμφάνισε τον 68000 με συχνότητα λειτουργιάς 8MHz και περιέχει 68.000 τρανζίστορ. Ο συνδυασμός της υψηλής ταχύτητας , του μεγάλου χώρου αποθήκευσης(16Mbytes) και του αρκετά χαμηλού κόστους ,τον έκανε να χρησιμοποιηθεί στους υπολογιστές Apple Lisa Macintosh.
- Το 1982 η Intel δημιουργεί τον 80286 με συχνότητα λειτουργιάς αρχικά στα 6 και μετά στα 12.5MHz και 134.000 τρανζίστορ. Περιλάμβανε δίαυλο δεδομένων 16bit ,δίαυλο διευθύνσεων 24 bit και μπορούσε να δει μέχρι 16Mbytes μνήμης. Είχε τη δυνατότητα να λειτουργεί στην κατάσταση protected mode (προστατευμένη κατάσταση λειτουργιάς). Μετά την επιτυχία της Motorola με τον 68000, οδηγήθηκε στον σχεδιασμό του MC68010 με αποτέλεσμα την προσθήκη υποστήριξης της εικονικής μνήμης.
- Το 1985 εμφανίζονται οι πρώτοι 32bit μικροεπεξεργαστές. Από τη μια ο Intel 80486 όπου περιέχει 275.000 τρανζίστορ και συχνότητα λειτουργιάς στα 33MHz, με μνήμη που έφτανε τα 4Gbytes και από την άλλη ο MC86020 της Motorola που είχε 200.000 τρανζίστορ και συχνότητα λειτουργιάς 16MHz. Έγιναν ιδιαίτερα δημοφιλή στη microcomputer Unix αγορά ενώ πολλές μικρές επιχειρήσεις παρήγαγαν τα συστήματα desktop.
- Το 1989 κάνει την εμφάνιση του ο 32bit μικροεπεξεργαστής Intel 80486 με 1.200.000 τρανζίστορ και 50 MHz συχνότητα λειτουργιάς.
- Το 1993 εμφανίζεται ο Intel Pentium της οικογενείας P5 ο οποίος περιέχει 3.100.000 τρανζίστορ και λειτουργούσε στα 60 και 66MHz. Την ίδια περίοδο η Digital παρουσιάζει τον πρώτο 64bit μικροεπεξεργαστή Alpha.
- Το 1997 η Intel εισάγει τον μικροεπεξεργαστή Pentium II με τεχνολογία MMX για την υποστήριξη πολυμέσων. Είχε 7.500.000 τρανζίστορ και η συχνότητα λειτουργιάς του ήταν στα 300MHz.

- Το 1999 η Intel ανακοινώνει τον Pentium III με 9.500.000 τρανζίστορ και η συχνότητα λειτουργίας του ήταν στα 450MHz.
- Το 2000 εμφανίστηκε ο Pentium IV όπου ήταν σχεδιασμένος σύμφωνα με την μικροαρχιτεκτονική NetBurst. Η συχνότητα λειτουργίας του έφτασε στο 1GHz αλλά χωρίς αυτό να παρέχει σημαντική αύξηση της ταχύτητας επεξεργασίας σε σχέση με τον Pentium III.

Από το σημείο αυτό και μέχρι σήμερα έχουμε φτάσει τους μικροεπεξεργαστές στο επίπεδο των πολλαπλών πυρήνων (2,4 ακόμα και 8 πυρήνες). Και η τεχνολογία συνεχίζει να αναπτύσσεται.



**Εικόνα 9.** Μικροεπεξεργαστές Intel της «οικογένειας» Pentium



**Εικόνα 10.** Επεξεργαστής Intel 4004 του 1971



**Εικόνα 11.** Επεξεργαστής Intel 8086 του 1978

## **2.3 Διαφορές μικροεπεξεργαστή-μικροελεγκτή**

Ο μικροεπεξεργαστής είναι ένα ολοκληρωμένο κύκλωμα (IC) που έχει μόνο τη CPU στο εσωτερικό του δηλαδή μόνο τις εξουσίες επεξεργασίας, όπως οι Pentium I , II, III, IV της Intel κλπ. Αυτοί οι μικροεπεξεργαστές δεν έχουν μνήμη RAM, ROM και αλλά περιφερικά στο τσιπ. Για να γίνει λειτουργικός ένας μικροεπεξεργαστής ο σχεδιαστής του συστήματος πρέπει να τα προσθέσει στο εξωτερικό του. Εφαρμογές του μικροεπεξεργαστή περιλαμβάνουν τα επιτραπέζια PC , τα laptop κλπ.

Αλλά αυτό δεν ισχύει στους μικροελεγκτές. Ο μικροελεγκτής έχει CPU, επιπροσθέτως με ένα σταθερό ποσό της μνήμης RAM, ROM και αλλά περιφερικά, όλα ενσωματωμένα chip. Μερικές φορές χαρακτηρίζεται σαν ένας μίνι υπολογιστής. Σήμερα διαφορετικοί κατασκευαστές είναι οι ATMEL , Microchip, Philips, Motorola και πολλοί άλλοι. Οι μικροελεγκτές είναι σχεδιασμένοι για να εκτελούν συγκεκριμένες διεργασίες. Αυτό σημαίνει εφαρμογές , όπου η σχέση εισροών και εκροών να είναι καθορισμένες. Για παράδειγμα πληκτρολόγια ,ποντίκια, τηλέφωνο ,πλυντήριο ρούχων, αυτοκίνητα, κινητά τηλέφωνα κλπ., οι εφαρμογές είναι πολύ συγκεκριμένες και χρειαζόμαστε μικρούς πόρους, όπως μνήμη RAM, ROM ,θύρες I/O κλπ με σκοπό να μπορούν αν ενσωματωθούν σε ένα μόνο chip. Αυτό με τη σειρά του μειώνει το μέγεθος και το κόστος.

Η ταχύτητα ρολογιού ενός μικροεπεξεργαστή είναι αρκετά υψηλή σε σύγκριση με του μικροελεγκτή. Ενώ οι μικροελεγκτές λειτουργούν από 30 έως 80+MHZ , οι σημερινοί μικροεπεξεργαστές λειτουργούν πάνω από 2GHz, δεδομένου ότι εκτελούν πολύπλοκα καθήκοντα. Η σύγκριση του μικροελεγκτή και του μικροεπεξεργαστή όσον αφορά το κόστος δεν είναι δικαιολογημένη. Αναμφίβολα ένας μικροελεγκτής είναι πολύ φθηνότερος από ένα μικροεπεξεργαστή.

Ωστόσο ένας μικροελεγκτής δεν μπορεί να χρησιμοποιηθεί στην θέση ενός μικροεπεξεργαστή καθώς και το αντίθετο αφού κάνει την εφαρμογή πολύ δαπανηρή. Ο μικροεπεξεργαστής δεν μπορεί να χρησιμοποιηθεί αυτόνομα. Χρειάζεται άλλα περιφερειακά όπως RAM, ROM ,Buffer, Θύρες I/O κλπ κάνοντας το σύστημα γύρω από έναν μικροεπεξεργαστή, να είναι πολύ ακριβό.

Με λίγα λόγια τα πλεονεκτήματα ενός μικροελεγκτή είναι :

- Χαμηλό κόστος
- Μικρότερο μέγεθος
- Χαμηλή κατανάλωση ισχύος
- Αυτονομία
- Επίτευξη ελέγχου μετρήσεων σε πραγματικό χρόνο
- Περισσότεροι διαθέσιμοι ακροδέκτες για ψηφιακές εισόδους και εξόδους.

## ***2.4 Κατασκευαστές μικροελεγκτών***

Οι περισσότερες εταιρίες παράγουν μεγάλη γκάμα μικροελεγκτών. Από πολύ μικρούς και φθηνούς για απλές εφαρμογές έως ιδιαίτερα προηγμένους για πολύ απαιτητικές εφαρμογές. Μερικοί από του γνωστότερους κατασκευαστές μικροελεγκτών είναι :

- Intel
- Atmel
- Epson
- Hitachi
- Microchip
- Nec
- Toshiba
- Analog devices
- Texas instruments

## 2.5 Κατηγορίες μικροελεγκτών

Λόγω της τάσης ενσωμάτωσης των μικροελεγκτών σε κάθε ηλεκτρική συσκευή αλλά και του ισχυρότατου ανταγωνισμού ,έχουμε καταλήξει στην παράγωγη ανταγωνιστικών μοντέλων μαζικής παράγωγης καθώς και στην παράγωγη μικροελεγκτών για πιο εξειδικευμένες εφαρμογές. Έτσι διακρίνονται οι εξής κατηγορίες:

- Μικροελεγκτές (καμιά φορά 4bit αλλά συνήθως 8bit) πολύ χαμηλού κόστους ,γενικής χρήσης, με πολύ μικρό αριθμό ακροδεκτών. Για να μην μπορεί να αντιγραφεί εύκολα το εσωτερικό λογισμικό τους, σχεδιάζονται με έμφαση στη χαμηλή κατανάλωση ισχύος και την αυτάρκεια.
- Μικροελεγκτές (συνήθως 8bit αλλά και 16 ή 32bit ) χαμηλού κόστους ,γενικής χρήσης, με μέτριο έως σχετικά μεγάλο αριθμό ακροδεκτών. Διαθέτουν μεγάλο αριθμό κοινών περιφερικών ,όπως θύρες UART, SPI ή CAN, μετατροπείς αναλογικού σε ψηφιακό και το αντίστροφο. Στους κατασκευαστές της Άπω Ανατολής (Ιαπωνία, Κορέα), συνηθίζεται η ενσωμάτωση ελεγκτών οθόνης υγρών κρυστάλλων ,πληκτρολογίου και εννίοτε παρέχουν δυνατότητα επέκτασης της μνήμης τους.
- Μικροελεγκτές (κυρίως 32bit) μέσου κόστους , γενικής χρήσης, με μεγάλο αριθμό ακροδεκτών. Χαρακτηρίζονται από έμφαση στην ταχύτητα εκτέλεσης εντολών , υψηλή αυτάρκεια περιφερειακών και μεγάλες δυνατότητες εσωτερικής ή εξωτερικής μνήμης προγράμματος(Flash) και RAM. Στο χώρο αυτό έχουν ισχυρή παρουσία οι αρχιτεκτονικές με υψηλή μεταφερσιμότητα λογισμικού (portability) από τον ένα στον άλλο κατασκευαστή.
- Μικροελεγκτές εξειδικευμένων εφαρμογών, οι οποίοι ενσωματώνουν συνήθως κάποιο εξειδικευμένο πρωτόκολλο επικοινωνίας το οποίο υλοποιείται πάντοτε σε hardware. Τέτοιοι μικροελεγκτές χρησιμοποιούνται σε τηλεπικοινωνιακές συσκευές όπως τα μόντεμ.

## **2.6 Γλώσσες προγραμματισμού μικροελεγκτών**

Η επιτυχία καθορίζεται σε μεγάλο βαθμό από τη διαθεσιμότητα [5] και την ευχρηστία των σχετικών εργαλείων ανάπτυξης, όπως μεταφραστές από γλώσσες υψηλού επιπέδου σε γλώσσα κατανοητή από τον μικροελεγκτή (Assembly), προγραμματιστές της εσωτερικής μνήμης και εργαλεία εκσφαλμάτωσης (Debugger). Στους μικροελεγκτές, τα εργαλεία αυτά δεν αποτελούνται ποτέ μόνο από το λογισμικό, καθώς δεν υπάρχει τυποποιημένος τρόπος επικοινωνίας με αυτούς.

Η C, η C++ και οι παραλλαγές τους είναι από τις πιο διαδοσμένες γλώσσες προγραμματισμού των μικροελεγκτών. Η Assembly μπορεί να χρησιμοποιείται σε τμήματα του λογισμικού όπου απαιτείται ταχύτητα ή μικρό μέγεθος χρησιμοποιούμενης μνήμης. Όμως οι μεγαλύτερες απαιτήσεις σε λειτουργικότητα και η ευκολία προγραμματισμού της C έναντι της Assembly, σε συνδυασμό με την επάρκεια μνήμης των σύγχρονων μικροελεγκτών, έχουν εκτοπίσει την Assembly από τις περισσότερες εφαρμογές.



# **ΚΕΦΑΛΑΙΟ 3 – ARDUINO**

### 3.1 Εισαγωγή στο Arduino

Το Arduino είναι μια υπολογιστική πλατφόρμα βασισμένη σε μια [6] απλή μητρική πλακέτα ανοικτού κώδικα, με ενσωματωμένο μικροελεγκτή και εισόδους/εξόδους, και η οποία μπορεί να προγραμματιστεί με τη γλώσσα Wiring (ουσιαστικά πρόκειται για τη γλώσσα προγραμματισμού C++ και ένα σύνολο από βιβλιοθήκες, υλοποιημένες επίσης στην C++ ).

Το Arduino μπορεί να χρησιμοποιηθεί για την ανάπτυξη ανεξάρτητων διαδραστικών αντικειμένων αλλά και να συνδεθεί με υπολογιστή μέσω προγραμμάτων σε Processing, Max/MSP, Pure Data, SuperCollider. Οι περισσότερες εκδόσεις του Arduino μπορούν να αγοραστούν προ-συναρμολογημένες· το διάγραμμα και πληροφορίες για το υλικό είναι ελεύθερα διαθέσιμα για αυτούς που θέλουν να συναρμολογήσουν το Arduino μόνοι τους.

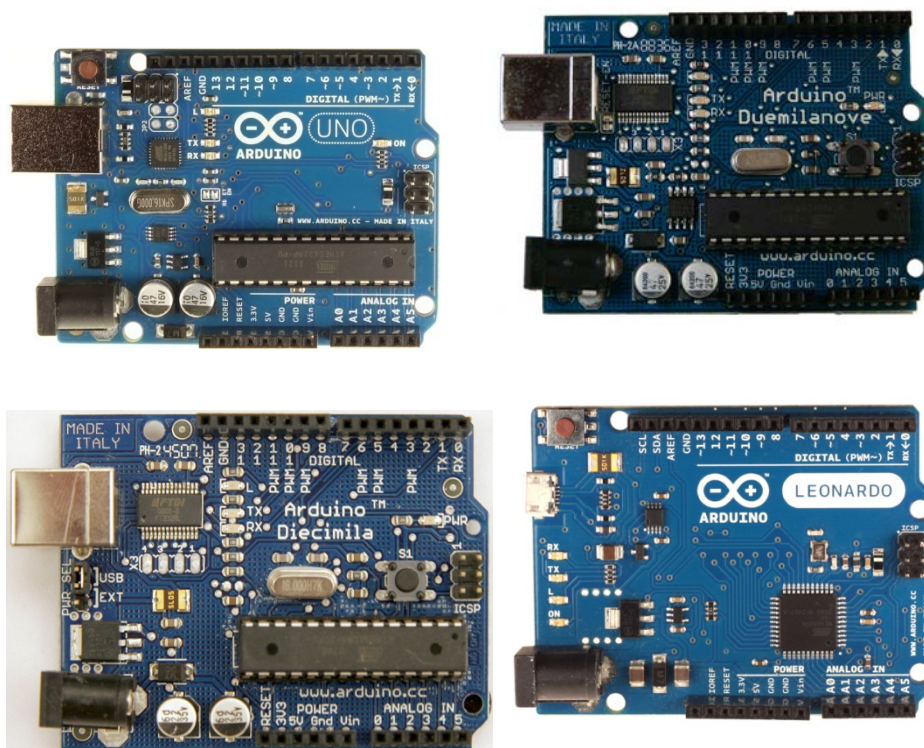
Στην ουσία, πρόκειται για ένα ηλεκτρονικό κύκλωμα που [7] βασίζεται στον μικροελεγκτή ATmega της Atmel και του οποίου όλα τα σχέδια, καθώς και το software που χρειάζεται για την λειτουργία του, διανέμονται ελεύθερα και δωρεάν ώστε να μπορεί να κατασκευαστεί από τον καθένα.



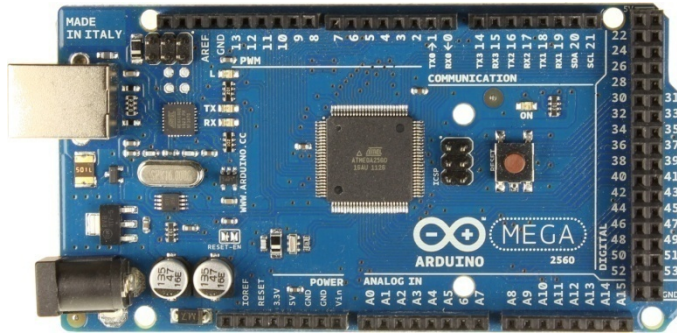
Εικόνα 12. Επεξεργαστής ATmega 32L

Αφού κατασκευαστεί, μπορεί να συμπεριφερθεί σαν ένας μικροσκοπικός υπολογιστής, αφού ο χρήστης μπορεί να συνδέσει επάνω του πολλαπλές μονάδες εισόδου/εξόδου και να προγραμματίσει τον μικροελεγκτή να δέχεται δεδομένα από τις μονάδες εισόδου, να τα επεξεργάζεται και να στέλνει κατάλληλες εντολές στις μονάδες εξόδου.

Ακριβώς επειδή απευθύνεται κυρίως σε αρχάριους των ηλεκτρονικών και επειδή, παρά τις αναλυτικότερες οδηγίες που υπάρχουν, δεν έχουν όλοι τις γνώσεις και τα μέσα να κατασκευάσουν μια ηλεκτρονική πλακέτα, κυκλοφορούν έτοιμες, προκατασκευασμένες πλακέτες Arduino που μπορούμε να προμηθευτούμε για περίπου €25. Με λίγα χρήματα παραπάνω μάλιστα, οι περισσότεροι προμηθευτές διαθέτουν Arduino Starter Kit, τα οποία, εκτός από το ίδιο το Arduino, περιέχουν διάφορα άλλα εξαρτήματα και εργαλεία που μπορεί να χρειαστούμε για τις πρώτες μας εφαρμογές (όπως το απαραίτητο καλώδιο USB για την σύνδεση με τον υπολογιστή, jumper wires, καλώδια, LED, διακόπτες, ποτενσιόμετρα, αντιστάσεις, διόδους, τρανζίστορ κ.λπ.).



**Εικόνα 13.** Arduino UnoR3, Duemilanove, Diecimila, Leonardo

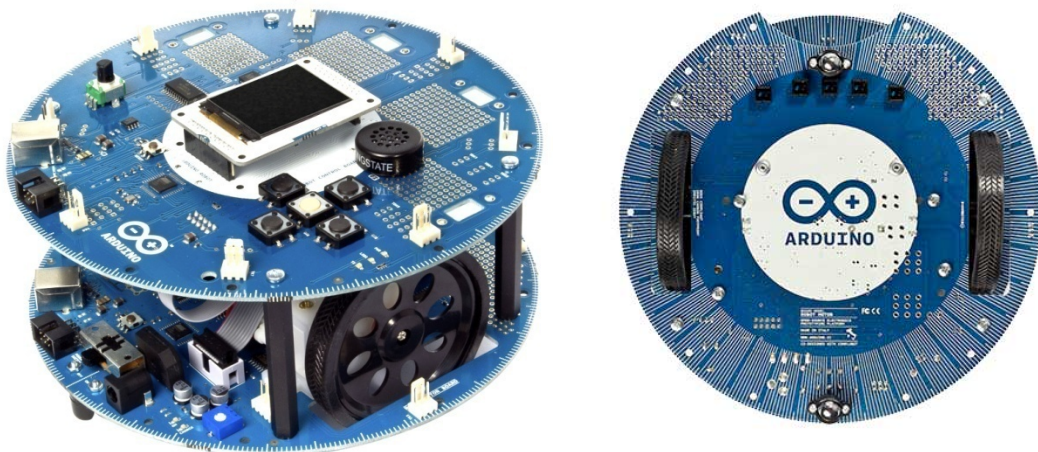


Εικόνα 14. Arduino Mega

### 3.2 Ιστορικό του Arduino

Το 2005, ένα σχέδιο κίνησε προκειμένου να φτιαχτεί μία συσκευή για τον έλεγχο προγραμμάτων διαδραστικών σχεδίων από μαθητές, η οποία θα ήταν πιο φθηνή από άλλα πρωτότυπα συστήματα διαθέσιμα εκείνη την περίοδο. Οι ιδρυτές Massimo Banzì και David Cuartielles και ονόμασαν το σχέδιο από τον Arduino της Inrea, όπου με δυο ακόμη φοιτητές που επιλέχτηκαν να γράψουν το λογισμικό για τη συσκευή, ξεκίνησαν να παράγουν πλακέτες σε ένα μικρό εργοστάσιο στην Ιβρέα, κωμόπολη της επαρχίας Τορίνο στην περιοχή Πεδεμόντιο της βορειοδυτικής Ιταλίας- την ίδια περιοχή στην οποία στεγαζόταν η εταιρία υπολογιστών Olivetti. Ο Ηλεκτρολόγος-Μηχανικός Gianluca Martino, κλήθηκε να κάνει μια αρχική παρτίδα των 200 μικροελεγκτών. Το πρώτο Arduino που φτιάχτηκε ονομάστηκε “Serial Arduino” και περιλάμβανε ένα ATmega8 με άμεση σύνδεση RS-232 με το μικροελεγκτή και όλα τα συστατικά του. Στη συνέχεια σχεδιάστηκε η έκδοση 2.0 και μια μονόπλευρη εκδοχή σαφέστερη για τους χομπίστες. Οι εκδόσεις που ακλούθησαν ήταν όλες FTDI USB μετατροπέα. Μετά το USB v1.0 και v2.0 κυκλοφόρησε το **Arduino Extreme** το οποίο αύξησε την ποσότητα των επιφανειακών εξαρτημάτων. Το **Arduino Nuova Generazione** μεταβαίνει σε έναν απλούστερο μετατροπέα USB και μετατρέπεται από το ATmega8 σε ATmega168. Οι βελτιώσεις συνεχίστηκαν με το **Diecimila** και έτσι :

- τον Οκτώβρη του 2008 ανακοινώθηκε το **Duemilanove** όπου [8] αρχικά βασίστηκε στο Atmel Atmega168, αλλά μετά στάλθηκε με το ATmega328.
- Τον Μάρτιο του 2009 ανακοινώθηκε το Arduino **Mega** το οποίο είναι βασισμένο στο Atmel ATmega1280
- Από τον Μάιο του 2011 πάνω από 300,000 Arduino ήταν σε χρήση σε όλο τον κόσμο
- Τον Ιούλιο του 2012 ανακοινώθηκε το Arduino **Leonardo** το οποίο είναι βασισμένο στο Atmel ATmega32u4
- Τον Οκτώβριο του 2012 ανακοινώθηκε το Arduino **Due** το οποίο είναι βασισμένο στο Atmel SAM3X8E, που είχε πυρήνα ARM Cortex-M3
- Τον Νοέμβριο του 2012 ανακοινώθηκε το Arduino **Micro** το οποίο είναι βασισμένο στο Atmel ATmega32u4
- Τον Μάιο του 2013 ανακοινώθηκε το Arduino **Robot** το οποίο είναι βασισμένο στο Atmel ATmega32u4 και ήταν το πρώτο επίσημο Arduino με ρόδες
- Τον Μάιο του 2013 ανακοινώθηκε το Arduino **Yun** το οποίο είναι βασισμένο στο ATmega32u4 και στο Atheros AR9331 και ήταν το πρώτο προϊόν WiFi που συνδύαζε το Arduino με το Linux.



Εικόνα 15. Arduino Robot

### **3.3 Δυνατότητες και πλεονεκτήματα του Arduino**

Υπάρχει πληθώρα άλλων μικροελεγκτών και αναπτυξιακών στο εμπόριο για να ασχοληθεί κάποιος εκεί έξω. Ο Basic Stamp της Parallax, ο BX-24 της Netmedia, το Handyboard του MIT και πολύ άλλη όμοιας λειτουργικότητας. Όλα αυτά τα εργαλεία που προαναφέραμε είναι απλά και για τον αρχάριο χρήστη καθώς "κρύβουν" τις δύσκολες λεπτομέρειες της αρχιτεκτονικής και επιτρέπουν τον άμεσο προγραμματισμό του μικροελεγκτή, προσφέροντας τα πάντα σε ένα και μόνο "πακέτο" έτοιμο για χρήση.

Ο Arduino διαφέρει από τους προηγούμενους γιατί απλοποιεί την διαδικασία να δουλεύει κάποιος με μικροελεγκτές, αλλά κάποια πλεονεκτήματα που προσφέρει σε σχέση με άλλους μικροελεγκτές για χρήση από δασκάλους, μαθητές και άλλους hobbiστες είναι τα παρακάτω:

- Φθηνός - Οι πλακέτες του Arduino είναι εξαιρετικά φθηνές σε σχέση με άλλες πλατφόρμες μικροελεγκτών. Ειδικά δε μπορεί με τα σχηματικά που κυκλοφορούν στο Internet να κατασκευάσει κάποιος την φθηνότερη εκδοχή ενός Arduino. Ωστόσο ακόμα και αν προμηθευτεί την έτοιμη (μονταρισμένη πλακέτα) αυτή θα κοστίσει το μέγιστο 50 Euro.
- Τρέχει σε διάφορα Λειτουργικά Συστήματα. Οι μηχανικοί λογισμικού, ανέπτυξαν το περιβάλλον προγραμματισμού του Arduino για Windows, Macintosh OSX και για λειτουργικά συστήματα Linux. Τα περισσότερα συστήματα ανάπτυξης μικροελεγκτών περιορίζονται στα Windows.
- Απλό, ξεκάθαρο προγραμματιστικό περιβάλλον. Το περιβάλλον προγραμματισμού ενός Arduino ενδείκνυται για αρχάριους, αλλά είναι ταυτόχρονα και ευέλικτο και για πιο προχωρημένους χρήστες.
- Ανοιχτού λογισμικού και λογισμικού που επεκτείνεται και παραμετροποιείται. Το software του Arduino διανέμεται με την μορφή εργαλείων ανοιχτού λογισμικού και είναι διαθέσιμο προς επέκταση για έμπειρους προγραμματιστές. Η γλώσσα προγραμματισμού του μπορεί να επεκταθεί διαμέσου των

βιβλιοθηκών την C++ και οι άνθρωποι που θέλουν να ασχοληθούν περισσότερο με τους μικροελεγκτές μπορούν να μεταβούν από τον Arduino στην AVR-C που είναι για προγραμματισμό των Atmel μικροελεγκτών και η γλώσσα στην οποία βασίστηκε το λογισμικό του Arduino. Ομοίως μπορεί κάποιος να προσθέσει κώδικα της AVR-C στο πρόγραμμα που έχει γράψει για τον Arduino του.

- Ανοιχτού Υλικού το οποίο μπορεί να επεκταθεί. Ο Arduino βασίζεται στους μικροελεγκτές της Atmel ATmega8 και ATmega168. Τα σχηματικά για τα αναπτυξιακά είναι κάτω από την άδεια της Creative Commons, επιτρέποντας σε έμπειρους σχεδιαστές να κατασκευάσουν το δικό τους αναπτυξιακό, εξελίσσοντας το ήδη υπάρχον χωρίς να έχουν νομικά προβλήματα. Η ακόμη καλύτερα όχι τόσο έμπειροι χρήστες μπορούν να επιδιώξουν την αντιγραφή και κατασκευή της πλακέτας σε ράστερ για να καταλάβουν την λειτουργία ενός Arduino.

### ***3.4 Εκδόσεις Arduino***

Δεκαεννιά εκδοχές του Arduino που έχουν χρησιμοποιηθεί εμπορικά μέχρι τώρα είναι:

1. Το **Serial Arduino**, είναι προγραμματισμένο με μία σειριακή DE-9 σύνδεση χρησιμοποιώντας τεχνολογία ATmega8
2. Το **Arduino Extreme**, είναι εφοδιασμένο με ένα USB interface για προγραμματισμό χρησιμοποιώντας τεχνολογία ATmega8
3. Το **Arduino Mini**, είναι μία έκδοση μινιατούρας του Arduino χρησιμοποιώντας τεχνολογία surface-mounted ATmega168
4. Το **Arduino Nano**, είναι ένα ακόμα πιο μικρό, μια USB τροφοδοτούμενη έκδοχή του Arduino χρησιμοποιώντας τεχνολογία surface-mounted ATmega168 (ATmega328 για την νεότερη έκδοση)
5. Το **LilyPad Arduino**, είναι ένα μινιμαλιστικό σχέδιο για εφαρμογές ένδυσης και E-textiles χρησιμοποιώντας τεχνολογία surface-mounted ATmega328
6. Το **Arduino NG**, με ένα USB interface για προγραμματισμό και χρησιμοποιώντας τεχνολογία ATmega8

7. Το **Arduino NG plus**, με ένα USB interface για προγραμματισμό και χρησιμοποιώντας τεχνολογία ATmega168
8. Το **Arduino Bluetooth**, με Bluetooth interface για προγραμματισμό χρησιμοποιώντας τεχνολογία ATmega168
9. Το **Arduino Diecimila**, με ένα USB interface και χρησιμοποιεί τεχνολογία ATmega168 σε ένα DIP28 πακέτο
10. Το **Arduino Duemilanove** (“2009”), χρησιμοποιεί τεχνολογία ATmega168 (ATmega328 για την καινούργια έκδοση) και τροφοδοτείτε μέσω ενέργειας USB/DC, αυτόματα εναλλασσόμενης
11. Το **Arduino Mega**, χρησιμοποιώντας τεχνολογία surface-mounted ATmega1280 για περαιτέρω I/O και μνήμη
12. Το **Arduino Uno**, χρησιμοποιώντας την ίδια τεχνολογία ATmega328 όπως το τελευταίο μοντέλο Duemilanove, αλλά ενώ το Duemilanove χρησιμοποιεί ένα FTDI chipset για το USB, το Uno χρησιμοποιεί τεχνολογία ATmega8U2 προγραμματισμένο ως σειριακός μετατροπέας
13. Το **Arduino Mega2560**, χρησιμοποιεί τεχνολογία surface-mounted ATmega2560 φέρνοντας την ολική μνήμη στα 256kB. Επίσης ενσωματώνει την νέα τεχνολογία ATmega8U2 (ATmega16U2 σε αναθεώρηση τύπου 3) USB chipset.
14. Το **Arduino Leonardo**, με ένα ATmega32U4 chip που εξαλείφει την ανάγκη για συνδεσιμότητα μέσω USB και μπορεί να χρησιμοποιηθεί ως ψηφιακό πληκτρολόγιο ή ποντίκι.
15. Το **Arduino Esplora**, με εμφάνιση που παραπέμπει σε χειριστήριο κονσόλας βιντεοπαιχνιδιών με joystick και ενσωματωμένους αισθητήρες για ήχο, φώς, θερμοκρασία και επιτάχυνση
16. Το **Arduino Due** είναι ένα μικροχειριστήριο βασισμένο στην τεχνολογία Atmel SAM3X8E ARM Cortex-M3 CPU. Είναι το πρώτο board της Arduino βασισμένο σε επεξεργαστή 32-bit ARM microcontroller.
17. Το **Arduino Micro** το οποίο είναι βασισμένο στο Atmel ATmega32u4
18. Το **Arduino Robot** το οποίο είναι βασισμένο στο Atmel ATmega32u4 και ήταν το πρώτο επίσημο Arduino με ρόδες



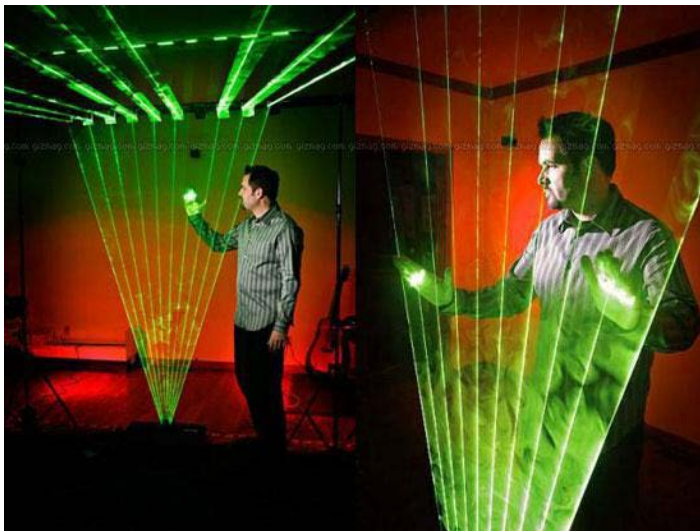
19. Το **Arduino Yun** το οποίο είναι βασισμένο στο ATmega32u4 και στο Atheros AR9331 και ήταν το πρώτο προϊόν WiFi που συνδύαζε το Arduino με το Linux.

### **3.5 Εφαρμογές Arduino**

#### ΕΦΑΡΜΟΓΕΣ ΣΤΗ ΜΟΥΣΙΚΗ

- **Laser Harp**

Το πιο γνωστό project με την χρήση Arduino είναι το Laser Harp (Laser άρπα). Το 2009 ο Stephen Hopley επηρεασμένος από μια συναυλία του Jarre αποφάσισε να κατασκευάσει την δική του Laser άρπα. Το Laser Harp είναι ένα μουσικό όργανο που αλληλεπιδρά με το φως. Αποτελείται από μια συσκευή η οποία παράγει μια σειρά κάθετων φωτεινών γραμμών (Laser) που ξεκινούν από το δάπεδο. Ο μουσικός διακόπτοντας τις δέσμες φωτός παράγει μια ποικιλία μουσικών ήχων. Ρόλο για την παραγωγή του ήχου δεν παίζει μόνο η διακοπή της δέσμης αλλά και το ύψος που θα τοποθετηθεί το εμπόδιο από το δάπεδο. Η άρπα δεν παράγει από μόνη της τους ήχους έτσι απαιτείται να συνδεθεί με ένα νέας τεχνολογίας synthesizer ώστε να λαμβάνει από το Arduino σειριακά τα MIDI (Musical Instrument Digital Interface) δεδομένα που παράγει.



**Εικόνα 16.** Laser harp

## ΕΦΑΡΜΟΓΕΣ ΣΤΟΝ ΜΟΝΤΕΛΙΣΜΟ

Μια άλλη διαδεδομένη εφαρμογή του Arduino κυρίως ανάμεσα στους φίλους του μοντελισμού είναι η κατασκευή τηλεκατευθυνόμενου οχήματος. Για την κατασκευή του απαιτείται ένας μικροελεγκτής που θα είναι υπεύθυνος για τον έλεγχο του οχήματος και ένα σύστημα για την ασύρματη επικοινωνία του Arduino με τη συσκευή που θα το τηλεκατευθύνει. Η δουλειά του μικροελεγκτή είναι να ελέγχει τους απαραίτητους κινητήρες και σερβοκινητήρες οι οποίοι τροφοδοτούνται από μια πηγή ρεύματος (συνήθως μπαταρία) για την κατεύθυνση του οχήματος. Η τηλεκατεύθυνση μπορεί να εφαρμοστεί σε οποιαδήποτε μορφή "σκάφους" (αυτοκίνητο, πλοίο, αεροπλάνο,...)



**Εικόνα 17.** Τηλεκατεθνομενο όχημα με υπερηχητικό αισθητήρα (supersonic sensor) της Parallax.

## ΕΦΑΡΜΟΓΕΣ ΣΤΟ ΣΠΙΤΙ

Το Arduino μπορεί επίσης να χρησιμοποιηθεί για να διευκολύνει τον τρόπο διαβίωσης των ανθρώπων μέσα στο σπίτι. Αυτό επιτυγχάνεται με διάφορες κατασκευές, όπως :

- **Συναγερμός :** ο ελεγκτής μπορεί να συνδυαστεί με μια ποικιλία αισθητήρων με σκοπό να λειτουργεί ως ένας συναγερμός σπιτιού. Αυτό μπορεί να γίνει ανιχνεύοντας την κίνηση, ελέγχοντας συγκεκριμένες πόρτες και παράθυρα ακόμη και την θερμοκρασία σε περίπτωση πυρκαγιάς. Ενώ με ένα κατάλληλο πρόγραμμα στον υπολογιστή έχει την δυνατότητα ακόμη και να ενημερώνει απομακρυσμένα τους υπεύθυνους του κτιρίου.

## ΕΦΑΡΜΟΓΕΣ ΣΤΗΝ ΡΟΜΠΟΤΙΚΗ

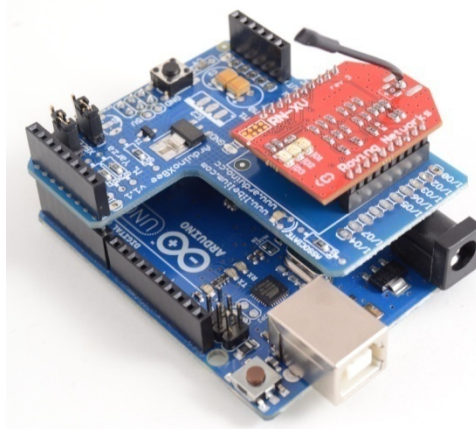
Η μεγάλη ποικιλία των συμβατών αισθητήρων για το Arduino που διατίθενται στο εμπόριο κάνουν τον σχεδιασμό ρομποτικών συστημάτων πολύ ενδιαφέρον, αφού με μια ασύρματη επικοινωνία του Arduino με έναν υπολογιστή, μπορεί να αυξήσει τις δυνατότητες των ρομπότ σε πολύ μεγάλο βαθμό.

### **3.6 Arduino Shields**

Τα Arduino και τα συμβατά Arduino boards χρησιμοποιούν την τεχνολογία των shields. Τα shield είναι ολοκληρωμένες πλακέτες που είναι σχεδιασμένες ώστε να κουμπώνουν πάνω στο Arduino προεκτείνοντας την λειτουργικότητά του. Είναι η hardware αντίστοιχη έννοια των plug-in, add-on και extension που υπάρχουν στο software. Μερικά από τα πιο δημοφιλή shields που κυκλοφορούν στο εμπόριο για το Arduino είναι:

- **Ethernet shield:** Δίνει στο Arduino την δυνατότητα να δικτυωθεί σε ένα LAN ή στο internet μέσω ενός τυπικού καλωδίου Ethernet.
- **WiFi shield:** Όμοιο με το Ethernet shield, χωρίς φυσικά το καλώδιο.
- Διάφορα **shield οθόνης:** Προσθέτουν οθόνη στο Arduino. Κυκλοφορούν από απλές οθόνες τύπου calculator μέχρι OLED touchscreen υψηλής ανάλυσης τύπου iPhone.
- **Wave shield:** Δίνει στο Arduino την δυνατότητα να παίζει ήχους/μουσική από κάρτες SD.
- **GPS shield:** Προσθέτει GPS δυνατότητες στο Arduino (εντοπισμό στίγματος).

- Διάφορα **motor shields**: Σας επιτρέπουν να οδηγήσετε εύκολα μοτέρ διάφορων τύπων (απλά DC, servo, stepper κ.λπ.) από το Arduino.
- **ProtoShield**: Μία προσχεδιασμένη πλακέτα πρωτοτυποποίησης, συμβατή στις διαστάσεις του Arduino και χωρίς εξαρτήματα για να φτιάξετε το δικό σας shield.



**Εικόνα 18.** Shield για Arduino Uno R3 με WiFi Module Roving RN-171



**Εικόνα 19.** Ethernet shield για Arduino

Τα shield είναι σχεδιασμένα ώστε αφού κουμπωθούν πάνω στο Arduino να προωθούν τις υποδοχές του, ώστε να μπορούμε να συνδέσουμε επιπλέον τα δικά μας εξαρτήματα ή να κουμπώσουμε και επόμενο shield. Φυσικά, το κάθε shield χρησιμοποιεί ορισμένους από τους πόρους συνδεσιμότητας του Arduino και έτσι δεν μπορούμε να συνδέσουμε απεριόριστα shield. Μάλιστα κάποια shield μπορεί να μην

είναι συμβατά μεταξύ τους γιατί χρησιμοποιούν τα ίδια pin του Arduino για επικοινωνία με αυτό.

Επίσης, επειδή κάποια shield δεν προωθούν τις συνδέσεις του Arduino (όπως π.χ. οι οθόνες οι οποίες δεν έχουν νόημα αν τις καλύψουμε από πάνω με ένα επόμενο shield), υπάρχουν ειδικά extender shield που κουμπώνουν στο Arduino και δίνουν την δυνατότητα σε δύο άλλα shield να κουμπώσουν πάνω τους, λειτουργώντας σαν πολύπριζα.

### ***3.7 Βασικά χαρακτηριστικά του ArduinoUno***



**Εικόνα 20.** Οι δυο όψεις του Arduino Uno r3

Πρόκειται για μια πλατφόρμα ανοιχτού λογισμικού, που μπορεί [9] να χρησιμοποιηθεί από οποιονδήποτε που ενδιαφέρεται να δημιουργήσει διακρατικές εφαρμογές.

Μία πλακέτα Arduino αποτελείται από ένα μικροελεγκτή Atmel AVR (ATmega328 και ATmega168 στις νεότερες εκδόσεις, ATmega8 στις παλαιότερες) και συμπληρωματικά εξαρτήματα για την διευκόλυνση του χρήστη στον προγραμματισμό και την ενσωμάτωση του σε άλλα κυκλώματα.

Όλες οι πλακέτες περιλαμβάνουν ένα γραμμικό ρυθμιστή τάσης 5V και έναν κρυσταλλικό ταλαντωτή 16MHz (ή κεραμικό αντηχητή σε κάποιες παραλλαγές). Ο μικροελεγκτής είναι από κατασκευής προγραμματισμένος με ένα bootloader, έτσι ώστε να μην χρειάζεται εξωτερικός προγραμματιστής.

Το Arduino Uno διαθέτει 14 ψηφιακά pins (Βλέπε Εικόνα 21) τα οποία μπορούν να οριστούν είτε σαν είσοδοι ,είτε σαν έξοδοι. Από τα 14 αυτά ψηφιακά pins κάποια εκτός από την λειτουργία I/O έχουν και άλλες επιπρόσθετες λειτουργίες :

- Digital pins 1, 2 : για σειριακή επικοινωνία UART
- Digital pins 3, 5, 6, 9, 10, 11: PWM (Pulse Width Modulation), για να έχουμε ψηφιακούς παλμούς μεταβλητού πλάτους.

Είναι επίσης σημαντικό, να αναφερθεί ότι και οι αναλογικές είσοδοι μπορούν να χρησιμοποιηθούν σαν ψηφιακά pins.



**Εικόνα 21.** Digital pins

Το Arduino Uno έχει 6 γραμμές για αναλογική είσοδο (Βλέπε Εικόνα 22), σε καθεμία από τις οποίες συνδέεται ένας μετατροπέας από αναλογικό σε ψηφιακό (Analog to Digital Converter - ADC). Ο κάθε μετατροπέας έχει 10 bit ανάλυσης, με αποτέλεσμα να επιστρέφει ακέραιους από 0 έως 1023, ενώ η βασική λειτουργία των αναλογικών pins είναι να διαβάζουν αναλογικές εισόδους π.χ. αισθητήρες. Μπορούν επίσης να χρησιμοποιηθούν σαν ψηφιακές γραμμές εισόδου/εξόδου γενικού σκοπού, όπως ακριβώς και τα pins 0-13.



Εικόνα 22. Analog input pins

### 3.8 Γλώσσα Arduino

Η γλώσσα του Arduino βασίζεται στη γλώσσα Wiring, μια παραλλαγή C/C++ για μικροελεγκτές αρχιτεκτονικής AVR όπως ο ATmega, και υποστηρίζει όλες τις βασικές δομές της C καθώς και μερικά χαρακτηριστικά της C++. Για compiler χρησιμοποιείται ο AVR gcc και ως βασική βιβλιοθήκη C χρησιμοποιείται η AVR libc.

Λόγω της καταγωγής της από την C, στην γλώσσα του Arduino [10] μπορείτε να χρησιμοποιήσετε ουσιαστικά τις ίδιες βασικές εντολές και συναρτήσεις, με την ίδια σύνταξη, τους ίδιους τύπων δεδομένων και τους ίδιους τελεστές όπως και στην C. Πέρα από αυτές όμως, υπάρχουν κάποιες ειδικές εντολές, συναρτήσεις και σταθερές που βοηθούν για την διαχείριση του ειδικού hardware του Arduino. Οι πιο σημαντικές από αυτές επεξηγούνται στον πίνακα που ακολουθεί:

Όρισμα	Είδος	Τύπος	Παράμετροι	Περιγραφή
LOW	Σταθερά	int	-	Έχει την τιμή 0 και είναι αντίστοιχη του λογικού false.
HIGH	Σταθερά	int	-	Έχει την τιμή 1 και είναι αντίστοιχη του λογικού true.
INPUT	Σταθερά	int	-	Έχει την τιμή 0 και είναι αντίστοιχη του λογικού

				false.
OUTPUT	Σταθερά	int	-	Έχει την τιμή 1 και είναι αντίστοιχη του λογικού true.
pinMode	Εντολή	-	( <i>pin, mode</i> )	Καθορίζει αν το συγκεκριμένο ψηφιακό <i>pin</i> θα είναι <i>pin</i> εισόδου ή <i>pin</i> εξόδου ανάλογα με την τιμή που δίνεται στην παράμετρο <i>mode</i> (INPUT ή OUTPUT αντίστοιχα).
digitalWrite	Εντολή	-	( <i>pin, pinstatus</i> )	Θέτει την κατάσταση <i>pinstatus</i> (HIGH ή LOW) στο συγκεκριμένο ψηφιακό <i>pin</i> .
digitalRead	Συνάρτηση	int	( <i>pin</i> )	Επιστρέφει την κατάσταση του συγκεκριμένου ψηφιακού <i>pin</i> (0 για LOW και 1 για HIGH) εφόσον αυτό είναι <i>pin</i> εισόδου.
analogReference	Εντολή	-	( <i>type</i> )	Δέχεται τις τιμές DEFAULT, INTERNAL ή EXTERNAL στην παράμετρο <i>type</i> για να καθορίσει την τάση αναφοράς ( $V_{ref}$ ) των αναλογικών εισόδων (5V, 1.1V ή η εξωτερική τάση με την οποία τροφοδοτείται το <i>pin</i> AREF αντίστοιχα)
analogRead	Συνάρτηση	int	( <i>pin</i> )	Επιστρέφει έναν ακέραιο από 0 έως 1023, ανάλογα με την τάση που τροφοδοτείται το συγκεκριμένο <i>pin</i> αναλογικής εισόδου στην κλίμακα 0 ως $V_{ref}$ .
analogWrite	Εντολή	-	( <i>pin, value</i> )	Θέτει το συγκεκριμένο ψηφιακό <i>pin</i> σε κατάσταση ψευδοαναλογικής εξόδου (PWM). Η παράμετρος <i>value</i> καθορίζει το πλάτος του παλμού σε σχέση με την περίοδο του παραγόμενου σήματος στην κλίμακα από 0 ως 255 (π.χ. με <i>value</i> 127, το πλάτος του παλμού είναι ίσο με μισή περίοδο).
millis	Συνάρτηση	unsigned long	()	Μετρητής που επιστρέφει το χρονικό διάστημα σε ms από την στιγμή που άρχισε η εκτέλεση του προγράμματος. Λάβετε υπόψη ότι λόγω του τύπου μεταβλητής (unsigned long δηλ. 32bit) θα γίνει overflow σε $2^{32}$ ms δηλαδή περίπου σε 50 μέρες, οπότε ο μετρητής θα ξεκινήσει πάλι από το μηδέν.
delay	Εντολή	-	( <i>time</i> )	Σταματά προσωρινά την ροή του προγράμματος για <i>time</i> ms. Η παράμετρος <i>time</i> είναι unsigned long (από 0 ως $2^{32}$ ). Σημειώστε ότι παρά την προσωρινή παύση, συναρτήσεις των οποίων η εκτέλεση ενεργοποιείται από interrupt θα εκτελεστούν κανονικά κατά την διάρκεια μιας delay.
attachInterrupt	Εντολή	-	( <i>interrupt, fu</i> )	Θέτει σε λειτουργία το συγκεκριμένο <i>interrupt</i> , ώστε



			<i>function, triggermode)</i>	να ενεργοποιεί την συνάρτηση <i>function</i> , κάθε φορά που ικανοποιείται η συνθήκη που ορίζεται από την παράμετρο <i>triggermode</i> : <ul style="list-style-type: none"> <li>• LOW (ενεργοποίηση όταν η κατάσταση του pin που αντιστοιχεί στο συγκεκριμένο interrupt γίνει LOW)</li> <li>• RISING (όταν από LOW γίνει HIGH)</li> <li>• FALLING (όταν από HIGH γίνει LOW)</li> <li>• CHANGE (όταν αλλάξει κατάσταση γενικά)</li> </ul>
detachInterrupt	Εντολή	-	<i>(interrupt)</i>	Απενεργοποιεί το συγκεκριμένο <i>interrupt</i> .
noInterrupts	Εντολή	-	()	Σταματά προσωρινά την λειτουργία όλων των <i>interrupt</i>
interrupts	Εντολή	-	()	Επαναφέρει την λειτουργία των <i>interrupt</i> που διακόπηκε προσωρινά από μια εντολή <i>noInterrupts</i> .
Serial.begin	Μέθοδος κλάσης	-	<i>(datarate)</i>	Θέτει τον ρυθμό μεταφοράς δεδομένων του σειριακού interface (σε baud)
Serial.println	Μέθοδος κλάσης	-	<i>(data)</i>	Διοχετεύει τα δεδομένα <i>data</i> για αποστολή μέσω του σειριακού interface. Η παράμετρος <i>data</i> μπορεί να είναι είτε αριθμός είτε αλφαριθμητικό.

Το ολοκληρωμένο περιβάλλον ανάπτυξης (IDE) του Arduino είναι μία εφαρμογή γραμμένη σε Java, που λειτουργεί σε πολλές πλατφόρμες, και προέρχεται από το IDE για τη γλώσσα προγραμματισμού Processing και το σχέδιο Wiring. Περιλαμβάνει ένα πρόγραμμα επεξεργασίας κώδικα με χαρακτηριστικά όπως είναι η επισήμανση σύνταξης και ο συνδυασμός αγκύλων και είναι επίσης σε θέση να μεταγλωττίζει και να φορτώνει προγράμματα στην πλακέτα με ένα μόνο κλικ. Δεν υπάρχει συνήθως καμία ανάγκη να επεξεργαστούμε αρχεία *make* ή να τρέξουμε προγράμματα σε ένα περιβάλλον γραμμής εντολών. Ένα πρόγραμμα ή κώδικας που γράφτηκε για Arduino ονομάζεται σκίτσο (*sketch*).

## **ΚΕΦΑΛΑΙΟ 4**

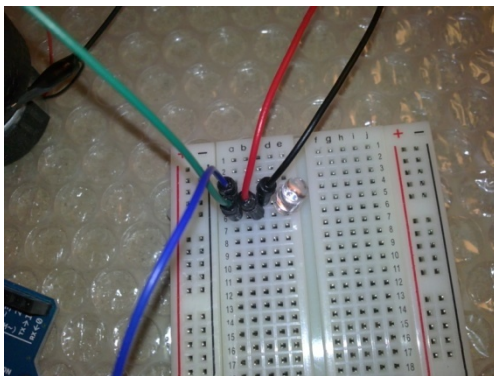
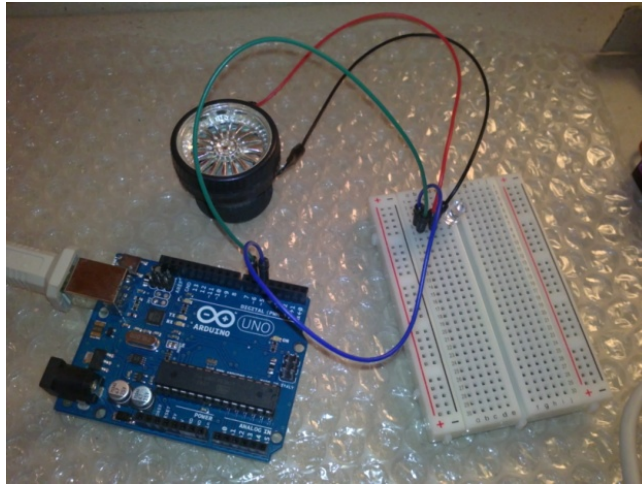
# **Υλοποίηση συστήματος ελέγχου**

Ξεκινώντας λοιπόν να υλοποιήσουμε στο σύστημα ελέγχου θα πρέπει να έχουμε τα εξής υλικά :

- 1 Arduino Uno R3
- 1 Breadboard
- 1 stepper motor
- 1 Led
- 1 διακόπτη
- 1 καλώδιο usb
- Jumper wires

#### Βήμα 1<sup>ο</sup>: Συνδεσμολογία

Συνδέουμε το stepper motor με το Arduino πάνω στην διάτρητη πλακέτα (breadbord) χρησιμοποιώντας τα καλώδια (jumper wires). Συγκεκριμένα το συνδέουμε στα ψηφιακά pins 5, 6. Όπως βλέπουμε και στις εικόνες 23, 24, 25, το σύστημα μας είναι έτοιμο και το μόνο που μένει είναι να το συνδέσουμε στον υπολογιστή και να φορτώσουμε το πρόγραμμα που επιθυμούμε.



Φορτώνουμε λοιπόν, το παρακάτω πρόγραμμα στο προγραμματιστικό περιβάλλον του Arduino :

```
sketch_jun29a | Arduino 1.0.5-r2
Αρχείο Πρεεργασία Σχέδιο Εργασία Βοήθεια
sketch_jun29a $
int motorPin1 = 5; // One motor wire connected to digital pin 5
int motorPin2 = 6; // One motor wire connected to digital pin 6

// The setup() method runs once, when the sketch starts

void setup() {
  // initialize the digital pins as an output:
  pinMode(motorPin1, OUTPUT);
  pinMode(motorPin2, OUTPUT);
}

// the loop() method runs over and over again,
// as long as the Arduino has power
void loop()
{
  rotateRight(200, 1000);
  rotateLeft(0, 2000);
  rotateLeft(200, 1000);
}

void rotateLeft(int speedOfRotate, int length){
  analogWrite(motorPin1, speedOfRotate); //rotates motor
  digitalWrite(motorPin2, LOW); // set the Pin motorPin2 LOW
  delay(length); //waits
  digitalWrite(motorPin1, LOW); // set the Pin motorPin1 LOW
}

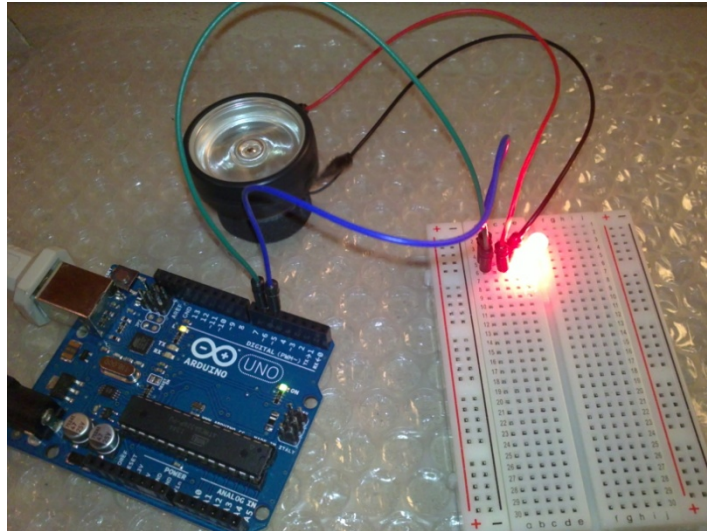
void rotateRight(int speedOfRotate, int length){
  analogWrite(motorPin2, speedOfRotate); //rotates motor
  digitalWrite(motorPin1, LOW); // set the Pin motorPin1 LOW
  delay(length); //waits
  digitalWrite(motorPin2, LOW); // set the Pin motorPin2 LOW
}

void rotateLeftFull(int length){
  digitalWrite(motorPin1, HIGH); //rotates motor
  digitalWrite(motorPin2, LOW); // set the Pin motorPin2 LOW
  delay(length); //waits
  digitalWrite(motorPin1, LOW); // set the Pin motorPin1 LOW
}

void rotateRightFull(int length){
  digitalWrite(motorPin2, HIGH); //rotates motor
  digitalWrite(motorPin1, LOW); // set the Pin motorPin1 LOW
  delay(length); //waits
  digitalWrite(motorPin2, LOW); // set the Pin motorPin2 LOW
}

Φόρτωση επιτυχής.
Μέγεθος δυαδικού σχεδίου: 1.422 bytes (από ένα μέγιστο 32.256 byte)
16
```

Μόλις ολοκληρωθεί η φόρτωση βλέπουμε ότι ο κινητήρας μας αρχίζει να κινείται σύμφωνα με το πρόγραμμά μας.



Δηλαδή δίνοντας εμείς :

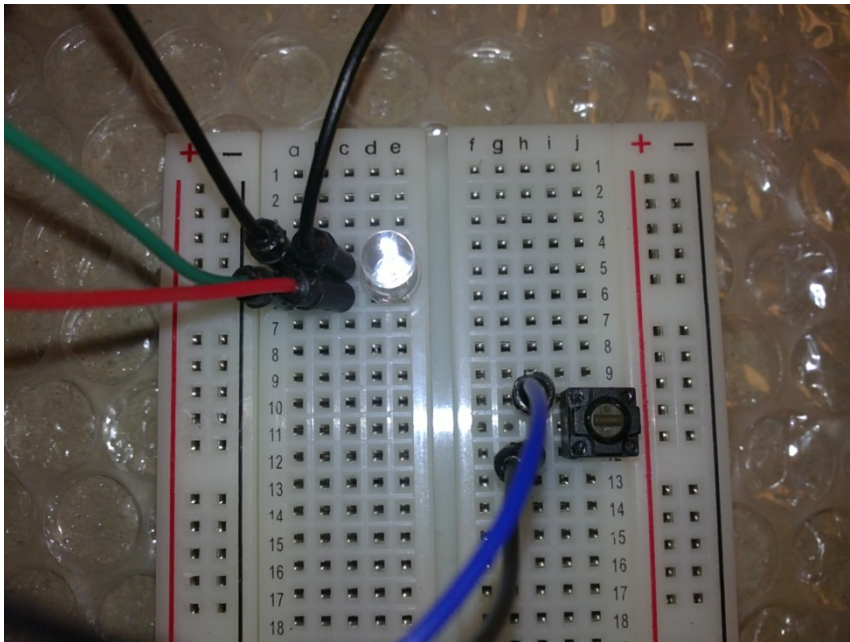
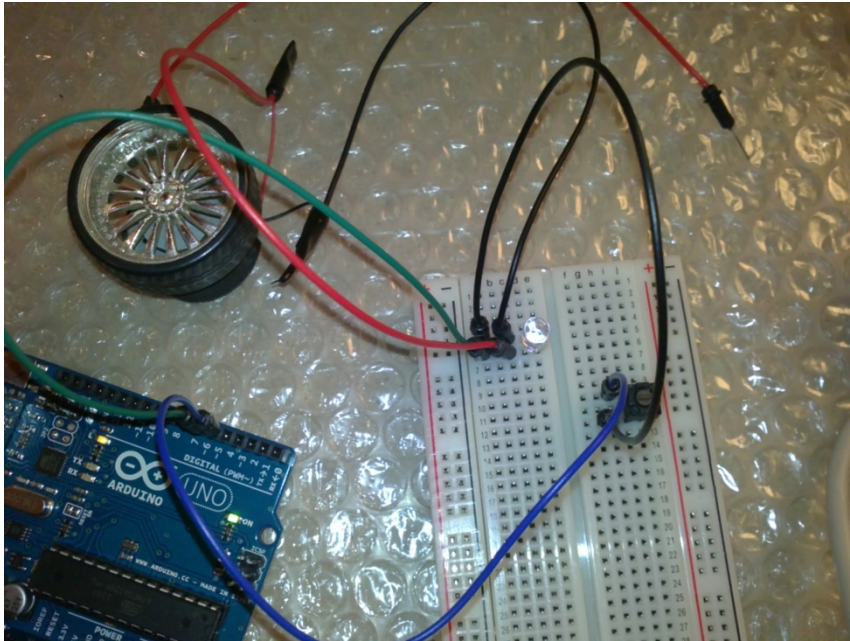
```
void loop()
{
  rotateRight(200, 1000);
  rotateLeft(0, 2000);
  rotateLeft(200, 1000);
}
```

Τότε ο κινητήρας μας ξεκινά να κινείται δεξιά, μετά να σταματά και τέλος να γυρίζει αριστερά. Μπορούμε να ρυθμίσουμε ακριβώς το πόσο γρήγορα θα γυρίζει, για πόση χρονική διάρκεια και προς ποια κατεύθυνση, από την εντολή

```
rotate_x(int speedOfrotate , int length);
```

και όπου x ,θα βάζουμε την κατεύθυνση. Δηλαδή Left ή Right. Όπου speedOfrotate θα βάζουμε τιμές από 0-255 ,αφού οι έξοδοι μας είναι ψηφιακές και όπου length θα βάζουμε τα δευτερόλεπτα που θέλουμε να γυρίζει ο κινητήρας, επί 1000.

Επειδή όμως το σύστημα μας δεν θα σταματήσει όσο θα υπάρχει ρεύμα, θα βάλουμε ένα διακόπτη και θα επιλεγούμε εμείς πότε θα ξεκινά και μόλις αφήνουμε τον διακόπτη θα σταμάτα. Δηλαδή :



Αφού τελειώσαμε με την συνδεσμολογία , απλοποιούμε λίγο το πρόγραμμα για γίνει πιο κατανοητή η λειτουργία του κουμπιού. Αλλάζουμε μόνο το κομμάτι του προγράμματος με την εντολή :

```
void loop()
```

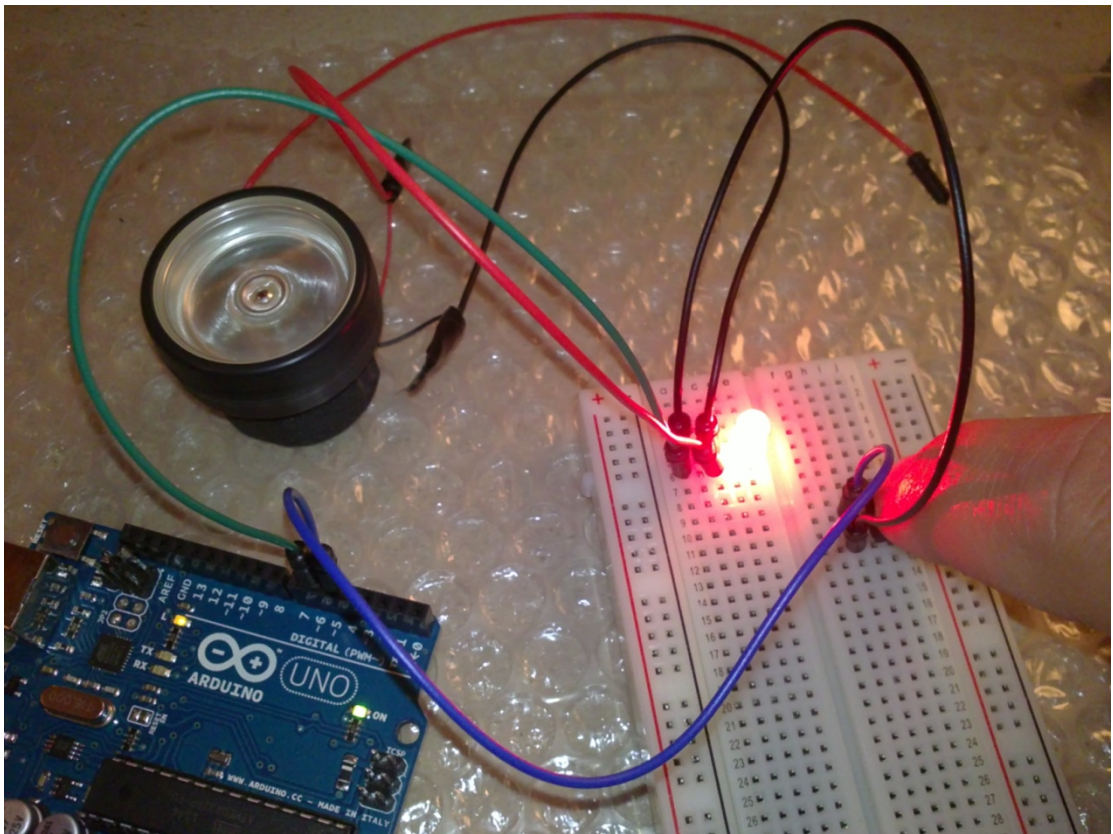
```
{
```

```
rotateRight(200, 1000);  
  
rotateLeft(0, 2000);  
  
rotateLeft(200, 1000);  
  
}
```

Και το κάνουμε πιο απλό :

```
void loop()  
{  
  rotateRight(255, 1000);  
}
```

Έτσι λοιπόν ,φορτώνοντας τις αλλαγές στο πρόγραμμα, ο ηλεκτροκινητήρας μας θα λειτουργεί μόνο με το πάτημα του κουμπιού.



**ΑΝΑΦΟΡΕΣ**



- [1] <http://el.wikipedia.org/>
- [2] <http://www.mechatronics.gr/>
- [3] <http://www.robolab.tuc.gr/> (Μαθήματα Ρομποτικής-Σάββας Πιπερίδης )
- [4] Πτυχιακή εργασία: «Η διδασκαλία εκπαιδευτικής ρομποτικής με τη χρήση μικροελεγκτών (π.χ. ARDUINO, PIC)”, Βασιλική Παυλή, ΤΕΙ ΛΑΡΙΣΑΣ, 2013
- [5] Πτυχιακή εργασία: «Η διδασκαλία εκπαιδευτικής ρομποτικής με τη χρήση μικροελεγκτών (π.χ. ARDUINO, PIC)”, Βασιλική Παυλή, ΤΕΙ ΛΑΡΙΣΑΣ, 2013
- [6] <http://el.wikipedia.org/wiki/Arduino>
- [7] <http://deltahacker.gr/2009/08/01/arduino-intro/>
- [8] [http://el.wikipedia.org/wiki/Arduino#cite\\_note-2](http://el.wikipedia.org/wiki/Arduino#cite_note-2)
- [9] Διπλωματική εργασία: «Οικιακός αυτοματισμός με χρήση μικροελεγκτή», Μαρτίνης Στέλιος, Πανεπιστήμιο Αιγαίου, 2013
- [10] <http://deltahacker.gr/2009/08/01/arduino-intro/>