



ΠΑΝΕΠΙΣΤΗΜΙΟ
ΠΑΤΡΩΝ
UNIVERSITY OF PATRAS

ΣΧΟΛΗ ΟΙΚΟΝΟΜΙΚΩΝ ΕΠΙΣΤΗΜΩΝ ΚΑΙ ΔΙΟΙΚΗΣΗΣ ΕΠΙΧΕΙΡΗΣΕΩΝ
ΤΜΗΜΑ ΔΙΟΙΚΗΤΙΚΗΣ ΕΠΙΣΤΗΜΗΣ ΚΑΙ ΤΕΧΝΟΛΟΓΙΑΣ

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

Η ΧΡΗΣΗ ΤΟΥ ARDUINO ΣΤΗ ΔΗΜΙΟΥΡΓΙΑ ΕΦΑΡΜΟΓΩΝ



ΟΝΟΜΑΤΕΠΩΝΥΜΟ: ΠΑΠΑΓΙΑΝΝΟΠΟΥΛΟΣ ΑΓΓΕΛΟΣ

ΕΠΟΠΤΕΥΩΝ ΚΑΘΗΓΗΤΗΣ: ΔΗΜΗΤΡΙΟΣ ΠΑΠΑΔΟΠΟΥΛΟΣ
ΠΑΤΡΑ – ΜΑΪΟΣ 2019

ΠΡΟΛΟΓΟΣ

Θα ήθελα να ευχαριστήσω τον επιβλέποντα καθηγητή μου κ. Παπαδόπουλο Δημήτρη, του Τ.Ε.Ι. Δυτικής Ελλάδας του Τμήματος Διοίκησης Επιχειρήσεων Πάτρας, για την υποστηρικτική καθοδήγηση και τις ανατροφοδοτήσεις για την ολοκλήρωση της πτυχιακής μου εργασίας. Η παρούσα εργασία με τίτλο «Η ΧΡΗΣΗ ΤΟΥ ARDUINO ΣΤΗ ΔΗΜΙΟΥΡΓΙΑ ΕΦΑΡΜΟΓΩΝ» με την χρήση του Arduino UNO και με κάποιες προεκτάσεις του, εκπονήθηκε στα πλαίσια του προγράμματος σπουδών του Τμήματος Διοίκησης Επιχειρήσεων Πάτρας. Σκοπός της εργασίας μου ήταν να αναδειχτεί κάποιο μέρος της τεχνολογία που μπορεί να εισβάλει στην ζωή κάποιου ατόμου, ηλικιωμένου ή με ειδικές ανάγκες και να μπορεί να αυτό - εξυπηρετείται, ενημερώντας τον για βασικές καταστάσεις ενός σπιτιού.

Κλείνοντας θα ήθελα να ευχαριστήσω την οικογένειά μου διότι μου έδωσε την δυνατότητα να σπουδάσω σε μία άλλη πόλη καθώς και την στήριξη που μου παρείχε όλα αυτά τα χρόνια ώστε να μπορέσω να ολοκληρώσω επιτυχώς τις σπουδές μου.

ΠΕΡΙΛΗΨΗ

Η εργασία αυτή είναι ένα αποτέλεσμα κάτω υπό την καθοδήγηση του διδάσκοντα, αναπτύχθηκε ο διάλογος και η σωστή συνεργασία με συνέπεια να φτάσουμε στο επιθυμητό αποτέλεσμα. Στόχος μας είναι η κατασκευή ενός έξυπνου σπιτιού με απλά υλικά, που να έχει τη δυνατότητα να ελέγχετε σε μεγάλες αποστάσεις. Αρχικά, θα αναλύσουμε την ιστορία των ρομπότ – αυτοματισμών πότε και από πού ξεκίνησαν, τις χρήσεις τους και πώς έχουν εξελιχθεί σήμερα. Ακολούθως, θα εξηγήσουμε τι είναι το Arduino και από τι αποτελείται. Έπειτα, θα περιγράψουμε τι είναι το έξυπνο σπίτι και ποιες είναι οι λειτουργίες του. Έπειτα, θα αναφέρουμε τους στόχους αυτής της εργασίας και τις δεξιότητες που αναπτύσσονται μέσα από αυτή. Στη συνέχεια, θα περιγράψουμε αναλυτικά τις βασικές μονάδες της κατασκευής και τα στάδια υλοποίησής της, τον προγραμματισμό και την σύνδεση στο διαδίκτυο

Λέξεις κλειδιά: Ρομποτικά συστήματα, αυτοματισμοί, μικροελεγκτής, Arduino, αισθητήρας

ABSTRACT

This paper is a result down under the guidance of the teacher, dialogue and good cooperation have developed, resulting in the desired result. Our goal is to build an intelligent home with simple materials that can be controlled over long distances. Initially, we will analyze the history of robots - automation when and where they started, their uses and how they have evolved today. Next, we will explain what Arduino is and what it is. Then, we will describe what is the smart home and what its functions are. Then, we will mention the aims of this work and the skills developed through it. We will then describe in detail the basic units of the construction and the stages of its implementation, programming and internet connection

Keywords: Robotic systems, automation, microcontroller, Arduino, sensor

Πίνακας Περιεχομένων

1. Εισαγωγή	5
1.2 Ιστορική αναδρομή των ρομπότ	6
1.3 Ο Αυτοματισμός - Ρομποτική στην κπαίδευση.....	10
1.4 Το ρομπότ	11
2 Ολοκληρωμένα μικροσυστήματα επεξεργασίας – Arduino	12
2.1. Η ιστορία του Arduino.....	12
2.2. Τι είναι το Arduino	12
2.3. Μοντέλα Arduino	13
2.4. Arduino Sensors	14
3. Πλατφόρμα ανάπτυξης λογισμικού Arduino IDE	16
3.1 Δομή γλώσσας Wiring	16
3.2 Λογισμικό IDE.....	22
Κεφάλαιο 4	25
4.1 Συστήματα Αυτομάτου ΕΛΕΓΧΟΥ.....	25
4.2 Η εξέλιξη στον ψηφιακό έλεγχο	26
4.3 Συστήματα και Μεθόδοι (Smart Home Technology)	27
Κεφάλαιο 5	30
5.1 Σχεδιασμός και εφαρμογή ενός έξυπνου οικιακού συστήματος	30
5.2 Επισκόπηση του συστήματος	31
κεφαλαιο 6	33
6. Περιγραφή Υλικών	33
6.1 Αισθητήρας θερμοκρασίας LM35 (Temperature Sensor Using LM35).....	34
6.2 Αισθητήρας θερμοκρασίας και υγρασίας (DHT11 HUMIDITY SENSOR)	35
6.3 Αισθητήρας απόστασης (Ultrasonic Sensor HC-SR04)	37
6.4 ΑισθητήραςΒροχής (Rain Sensor Module).....	38
6.5 ChannelRelayModule 4.....	40
6.6 Μαγνητικός διακόπτης επαφής (door sensor)	43
6.7 Ηχείο (Buzzer).....	44
6.8 Μονάδα Ethernet (ENC28J60)	45
7. Κατασκευαστικό Μέρος	47
7.1 Η κατασκευή του σπιτιού	47
7.2 Κατασκευή Ηλεκτρολογικού Υλικού.....	48
7.3 Τοποθέτηση – Συνδεση κυκλωμάτων στο σπίτι	50
8 Κεφάλαιο	52

8.1 Κώδικας Εφαρμογής	52
8.2 Βιβλιοθήκη pitches.h.....	58
ΣυμπέρασμαΤΑ	60
Βιβλιογραφία	62

1. ΕΙΣΑΓΩΓΗ

Η Ρομποτική είναι ο κλάδος της τεχνολογίας που ασχολείται με τη σχεδίαση, την ανάπτυξη και τη μελέτη ρομπότ. Η επιστήμη της Ρομποτικής αποτελείται από τον συνδυασμό διαφόρων επιστημών, όπως της πληροφορικής, των μαθηματικών, της φυσικής και της μηχανικής. Η λέξη ρομπότ (Robot) προέρχεται από το Σλαβικό «robota» που σημαίνει εργασία. Τα πρώτα ρομπότ που αναφέρονται στη λογοτεχνία είναι ο Τάλως από την ελληνική μυθολογία και οι 20 τρίποδες λέβητες του θεού Ηφαίστου θεωρούμενοι «θαύμα ιδέσθαι», αλλά και άλλες περιπτώσεις. Τα ρομπότ είναι αυτόματες μηχανές με προγραμματισμένη συμπεριφορά, η χρήση των οποίων αποσκοπεί στην αντικατάσταση του ανθρώπου στην εκτέλεση ενός έργου, τόσο σε φυσικό επίπεδο όσο και σε επίπεδο λήψης αποφάσεων. Τα τελευταία χρόνια άρχισε να αναπτύσσεται όλο και μεγαλύτερος αριθμός τεχνολογικών λύσεων για τους ηλικιωμένους και τα άτομα με ειδικές ανάγκες λόγω της γήρανσης του πληθυσμού και της αύξησης της ευαισθητοποίησης σχετικά με τα κοινά προβλήματα των ατόμων με ειδικές ανάγκες. Σε έξυπνες κατοικίες, μια ομάδα έξυπνων συσκευών και αισθητήρων σε δίκτυο μπορούν να βοηθήσουν τους ηλικιωμένους και τους ανάπηρους να είναι πιο ανεξάρτητοι αφήνοντας στα μέλη της οικογένειας, τους συγγενείς και τους γιατρούς να κρατούν τις καρτέλες από μακριά. Κατά συνέπεια, σε αυτή τη μελέτη, παρουσιάζεται μια δειγματική εφαρμογή της έξυπνης οικιακής τεχνολογίας. Στην παρουσιαζόμενη εφαρμογή, μια ομάδα αισθητήρων χρησιμοποιείται για την κατασκευή ενός αυτόνομου έξυπνου σπιτιού. Παρόλο που η εφαρμογή που παρουσιάζεται είναι απλώς ένα απλό παράδειγμα για το πώς μπορούν να χρησιμοποιηθούν έξυπνα σπίτια, έχει τη δυνατότητα να επηρεάσει όλες τις περιοχές που βελτιώνουν την καθημερινή ζωή των ηλικιωμένων και όλων των ειδών ατόμων.

1.2 ΙΣΤΟΡΙΚΗ ΑΝΑΔΡΟΜΗ ΤΩΝ ΡΟΜΠΟΤ

- 3000 – 2000 π.Χ. Μινωικά χρόνια - Ο Φύλακας «Ρομπότ». Ο Τάλως, το πρώτο μυθικό ρομπότ, κατασκευάστηκε από τον θεό Ήφαιστο και ήταν δώρο για τον βασιλιά της Κρήτης Μίνωα. Το όνομά του σήμαινε ήλιος και είχε σαν καθήκον να προστατεύει το νησί της Κρήτης αλλά και να τηρούνται οι νόμοι μεταξύ των ανθρώπων. Ο χάλκινος αυτός ήρωας συμβολίζει την τεχνολογική εξέλιξη στον τομέα της μεταλλουργίας στα μινωικά χρόνια.
- 3000 π.Χ. Εφευρέθηκε άβακας στην Ανατολή.
- 10ος π.Χ. Ένας Κινέζικος μύθος λέει πως ο Yan Shi κατασκεύασε για τον βασιλιά του Mu της δυναστείας του Zhou ένα ανθρωποειδές ρομπότ που τον άφησε άφωνο για τη φυσικότητα των κινήσεών του.
- 8ος π.Χ. Ο Όμηρος χρησιμοποιεί για πρώτη φορά τη λέξη «αυτόματο» στην Ιλιάδα, όταν γράφει, πως αυτόματα άνοιξαν οι πύλες του ουρανού. Επίσης, στην Οδύσσεια αναφέρει ότι τα πλοία του βασιλιά των Φαιάκων Αλκίνοου δεν είχαν ούτε κυβερνήτες ούτε πηδάλια.
- 460 π.Χ. περίπου. Ο Δημόκριτος γεννήθηκε στα Άβδηρα της Θράκης. Χαρακτηρίζεται σαν ένας από τους σημαντικότερους και πολυγραφότερους φιλοσόφους του αρχαίου ελληνικού κόσμου, αν και έχει διασωθεί ελάχιστο από το έργο του. Ασχολήθηκε σχεδόν με όλους τους τομείς της ανθρώπινης γνώσης. Η πιο γνωστή του θεωρία είναι ότι η ύλη αποτελείται από αόρατα, άφθαρτα και αδιάσπαστα στοιχεία, τα άτομα. Πίστευε πως βασικός στόχος της ζωής είναι η ευθυμία.
- 420 π.Χ. Ο Αρχύτας από τον Τάραντα κατασκεύασε ένα ξύλινο περιστέρι που πετούσε με τη βοήθεια πεπιεσμένου αέρα ή ατμού και είχε τη δυνατότητα να διανύσει 200 μέτρα.
- 322 π.Χ. Ο Αριστοτέλης στα «Πολιτικά» [c1253 b 34]γράφει: «...αν κάθε όργανο μπορούσε, κατά διαταγή, να εργαστεί μόνο του, αν η σαΐτα του αργαλειού μπορούσε να υφάνει μόνη της, αν το δοξάρι μπορούσε να κινήσει μόνο επάνω στη λύρα, οι επιστάτες δε θα είχαν ανάγκη από εργάτες, ούτε οι αφέντες από δούλους...».
- 285-222 π.Χ. Ο Κτησίβιος έζησε αυτά τα χρόνια και θεωρείται ο πατέρας του πεπιεσμένου αέρα.
- 200 π.Χ. Κινέζοι τεχνίτες κατασκεύασαν μια ολόκληρη μηχανική ορχήστρα.
- 150-100 π.Χ. Ο μηχανισμός των Αντικυθήρων είναι ο αρχαιότερος αυτοματισμός που σώζεται μέχρι και στις μέρες μας στο Αρχαιολογικό Μουσείο Αθηνών και είχε τη δυνατότητα να προβλέπει τις θέσεις των πλανητών.
- 10-70 μ.Χ. Τη χρονική περίοδο αυτή έζησε ο Ήρων ο Αλεξανδρεύς, ένας από τους μεγαλύτερους εφευρέτες της αρχαιότητας που δούλεψε βελτιώνοντας την προσπάθεια του Κτησίβου και στηριζόμενος στο έργο της ατμομηχανής δημιουργώντας το προγραμματιζόμενο αυτοκινούμενο τρίκυκλο ρομπότ «Ο Ηρακλής σκοτώνει τον δράκο» που λειτουργούσε με νερό.
Επίσης, κατασκεύασε έναν μηχανισμό για να ανοίγουν και να κλείνουν αυτόματα οι πύλες του ναού της Εφεσίας Αρτέμιδος με τη βοήθεια θερμαινόμενου αέρα.
- 1136-1206 μ.Χ. Την περίοδο αυτή ο Άραβας Al-Jazari κατασκεύασε το πρώτο ανθρωποειδές ρομπότ που ήταν ένας προγραμματιζόμενος τυμπανιστής. Δηλαδή ήταν μια μικρή βάρκα με τέσσερα ανθρωπάκια – μουσικούς που έπαιζαν τύμπανα, άρπα και φλάουτο. για να διασκεδάζει τη βασιλική οικογένεια. Ενώ το 1206 δημοσίευσε ένα βιβλίο για την αυτοματοποίηση και τις μηχανικές συσκευές με αρκετά σύνθετες πληροφορίες για την εποχή του.

- 1495 μ.Χ. Ο Ιταλός Leonardo da Vinci σχεδίασε (και ενδέχεται να κατασκεύασε) ένα ανθρωποειδές ρομπότ με πανοπλία που μπορούσε να κάνει αρκετές κινήσεις και είναι το παλαιότερο ανθρωποειδές σχέδιο ρομπότ που σώζεται έως τις μέρες μας.
- 1525 μ.Χ. Ο Hans Bullman δημιούργησε τα πρώτα πραγματικά ανδροειδή με ανθρώπινη μορφή.
- 1540 μ.Χ. Ο Ganiello Torriano κατασκεύασε ένα ρομπότ που έπαιζε βιολί με το όνομα «Lute Player Lady».
- 1543 μ.Χ. Ο John Dee κατασκεύασε ένα μηχανικό σκαθάρι χρησιμοποιώντας τις μαθηματικές του γνώσεις, που έμοιαζε τόσο αληθινό που χαρακτηρίστηκε «μαγικό».
- 1564 μ.Χ. Ο Pare Ambroise δημοσίευσε ένα σχέδιο για τεχνητό χέρι που χρησιμοποιεί μηχανικά τους μυς, με τη βοήθεια οργανικού σκελετού.
- 1620 μ.Χ. Ο William Oughtred κατασκεύασε έναν λογαριθμικό κανόνα.
- 1623 μ.Χ. Ο Wihelm Schickard δημιουργεί μια αριθμομηχανή με τέσσερις λειτουργίες και το όνομα «rechenuhr».
- 1642 μ.Χ. Ο Blaise Pascal κατασκεύασε τη δική του αριθμομηχανή που στην πορεία έγινε μαζικής παραγωγής.
- 1679 μ.Χ. Ο Gottfried Wilhelm von Leibniz ανέπτυξε και τελειοποίησε το δυαδικό σύστημα της αριθμητικής.
- 1725 μ.Χ. Ο Lorenz Rosenegge εγκατέστησε ένα μηχανικό θέατρο στο Helbrunn chateau.
- 1738 μ.Χ. Ο Jacques de Vaucanson εφευρίσκει τρία αυτόματα ρολόγια.
- 1738 μ.Χ. Ο Jacques de Vaucanson σχεδιάζει μια μηχανική πάπια που κουνάει τον λαιμό και τα φτερά της και καταπίνει φαγητό.
- 1752 μ.Χ. Ο Benjamin Franklin διεξάγει το περίφημο πείραμα με τον χαρταετό και αποδεικνύει ότι ο φωτισμός είναι μορφή ηλεκτρικής ενέργειας οδηγώντας την έρευνά του σε συμπεράσματα που έχουν χρήση έως και σήμερα.
- 1760 μ.Χ. Ο Friedrich von Knauss δημιούργησε μια αληθινή γραφομηχανή.
- 1771 μ.Χ. Ο Richard Arkwright δημιούργησε το πρώτο εργοστάσιο.
- 1772 μ.Χ. Ο Pierre Jaquet- Droz έφτιαξε το ονομαζόμενο «writer» ένα ρομπότ με παιδική μορφή που γράφει αφήνοντας αποστάσεις και βάζοντας σημεία
- στίξης ενώ όλα ελέγχονταν από μια προγραμματιζόμενη υπολογιστική συσκευή με εντυπωσιακή πολυπλοκότητα.
- 1804 μ.Χ. Ο Joseph – Marie Jacquard δημιούργησε ένα αυτοματοποιημένο σύστημα για αργαλειούς χρησιμοποιώντας ένα σύστημα ελέγχου κάρτας.
- 1810 μ.Χ. Ο Friedrich Kauffman κατασκευάζει μια μηχανική τρομπέτα που ελέγχεται και ενισχύεται από τύμπανο.
- 1824 μ.Χ. Ο Hisashige Tanaka ξεκινάει την τέχνη της μηχανικής κούκλας.
STEMC11501 STEMC11512 STEMC11532 Σελίδα 5
- 1835 μ.Χ. Ο Charles Babbage σχεδιάζει μια πλήρης αναλυτική υπολογιστική μηχανή με μνήμη, επεξεργαστή και πρωτοποριακές για την εποχή λειτουργίες χωρίς βέβαια αυτή να κατασκευαστεί.
- 1839 μ.Χ. Ο Sir William Grove κατασκευάζει την πρώτη κυψέλη καυσίμων.
- 1843 μ.Χ. Ο Augusta Ada King δημοσίευσε τις σημειώσεις του πάνω στην υπολογιστική μηχανή του Babbage που θεωρείται το πρώτο παράδειγμα προγράμματος ηλεκτρονικού υπολογιστή.
- 1847 μ.Χ. Ο George Boole δημοσιεύει το «Mathematical Analysis of Logic» που βασίζεται σε αυτό, το αναπτύσσει και δημιουργεί έναν ειδικό τομέα στην Άλγεβρα που βασίζεται στους υπολογιστές και ονομάζεται «Boolean Algebra».

- 1868 μ.Χ. Ο Zadoc P. Dederick εφηύρε ένα ρομπότ που λειτουργούσε με ατμό και μετέφερε κάρτα.
- 1873 μ.Χ. Ο William Thomson δημιουργεί έναν αναλογικό υπολογιστή με συγκεκριμένες δυνατότητες.
- 1882 μ.Χ. Ο Σέρβος Nicola Tesla, ένας από τους μεγαλύτερους εφευρέτες όλων των εποχών, παρουσίασε το πρώτο τηλεκατευθυνόμενο πλοίο που το ονόμασε τηλεαυτόματον. Όταν παρουσίασε την εφεύρεσή του, το κοινό της Νέας Υόρκης πίστεψε ότι έκανε κάποιο κόλπο. Τα τηλεχειριστήρια μπήκαν στις ζωές μας δεκαετίες αργότερα.
- 1887 μ.Χ. Ο Thomas Edison κατασκευάζει μια κούκλα που έχει τις δυνατότητες ομιλίας.
- 1889 μ.Χ. Ο Herman Hollerith εφευρίσκει μια μηχανή συλλογής πληροφοριών για να χρησιμοποιηθεί παράλληλα με την απογραφή του 1890.
- 1892 μ.Χ. Ο Seward Babbitt and Henry Aikken εφευρίσκει έναν γερανό με γάντζο σχεδιασμένο να μετακινεί μεταλλικές ράβδους.
- 1895 μ.Χ. Ο Nicola Tesla σχεδιάζει το πρώτο εργοστάσιο παραγωγής ενέργειας στους καταρράκτες του Νιαγάρα.
- 1896 μ.Χ. Ο Herman Hollerith ξεκινάει μια εταιρεία «Μηχανή Συλλογής Πληροφοριών», η οποία τελικά γίνεται η IBM.
- 1897 μ.Χ. Ο Joseph John Thomson δημιουργεί μια σειρά από πειράματα τα οποία τον βοήθάνε να κατανοήσει την ύπαρξη και τα χαρακτηριστικά του ηλεκτρονίου.
- 1898 μ.Χ. Ο Nicola Tesla κατοχυρώνει την σχεδίαση μιας τηλεκατευθυνόμενης ασύρματης βάρκας.
- 1921 μ.Χ. Στο θεατρικό έργο του Carel Kapek, R.U.R, βρίσκεται η πρώτη καταγεγραμμένη χρήση της λέξης ρομπότ. Η υπόθεση του έργου μιλάει για έναν επιστήμονα που εφευρίσκει τα ρομπότ για να βοηθήσουν τους ανθρώπους στις επαναλαμβανόμενες απλές εργασίες. Μόλις όμως αυτά χρησιμοποιήθηκαν σε πολέμους, επιτέθηκαν στους ανθρώπους και κατέλαβαν τον κόσμο. Ο όρος βασίζεται στην τσέχικη λέξη robota, η οποία σημαίνει δουλειά, δηλαδή δουλοπρεπής εργασία.
- 1927 μ.Χ. Ο Fritz Lang χρησιμοποιεί τη λέξη robot στην ταινία του «Metropolis» και δίνει δημοσιότητα στον όρο. Μια ηθοποιός υποδύεται το ρομπότ «Μαρία», ένα από τα πιο αναγνωρίσιμα κινηματογραφικά ρομπότ.
- 1927 μ.Χ. Ο Roy Wensley κατασκευάζει το «Herbert Televox» για την εταιρεία «WestingHouse and Electric and Manufacturing Co», το πρώτο μιας σειράς ρομπότ τα οποία δημιούργησε η εταιρεία.
- 1928 μ.Χ. Ο Makoto Nishimura δημιουργεί ένα μεγάλο ανθρωποειδές ρομπότ με το όνομα «Gokutensoku» ικανό να κινείται, να γράφει και να έχει εκφράσεις προσώπου.
- 1932 μ.Χ. Κατασκευάζεται το πρώτο παιχνίδι ρομπότ, η Λίλιπουτ. Ήταν ένα μεταλλικό κουρδιστό παιχνίδι που περπατούσε και είχε ύψος 15 εκατοστών.
- 1937 μ.Χ. Ο Alan Turing δημοσιεύει μια εργασία στα μαθηματικά υψίστης σημασίας για τη μελλοντική εξέλιξη υπολογιστών.
- 1938 μ.Χ. Ο Joseph Barnett κατασκευάζει το πιο πετυχημένο ρομπότ της εταιρείας «Westing House and Electric and Manufacturing Co» με όνομα Electro Moto-Man με ύψος 2.10 μ. και βάρος 120 κιλά, με είκοσι έξι κινήσεις, ένα λεξιλόγιο επτακοσίων λέξεων, του οποίου ο χειρισμός γινόταν με φωνητικές εντολές ενώ πήρε μέρος το 1939 στον διαγωνισμό World'S Fair.
- 1942 μ.Χ. Ο Ισαάκ Ασίμοφ, ο πολυδιαβασμένος συγγραφέας επιστημονικής φαντασίας, διατυπώνει για πρώτη φορά τους τρεις νόμους της ρομποτικής στο σύντομο διήγημα «Φαύλος κύκλος». 1. Ένα ρομπότ δεν μπορεί να τραυματίσει

άνθρωπο, ούτε με την αδράνειά του να επιτρέψει βλάβη σε άνθρωπο. 2. Ένα ρομπότ πρέπει να υπακούει τις εντολές που του δίνουν οι άνθρωποι, εκτός αν αυτές οι εντολές έρχονται σε αντίθεση με τον πρώτο νόμο. 3. Ένα ρομπότ πρέπει να προστατεύει την ύπαρξή του, εφόσον αυτό έρχεται σε αντίθεση με τον Πρώτο ή τον Δεύτερο νόμο.

- 1948 μ.Χ. Ο Γκρέν Γουόλτερ κατασκευάζει στην Αγγλία, το πρώτο ρομπότ, την Elsie, που κινούνταν με βάση ερεθίσματα που λάμβανε από αισθητήρα φωτός. (Ρομποτικός πληθυσμός 1948: 1)
- 1956 μ.Χ. Στις ΗΠΑ, δημιουργείται η πρώτη εταιρεία κατασκευής ρομπότ, η Unimation.
- 1961 μ.Χ. Ιδρύεται το Εθνικό Κέντρο Έρευνας Φυσικών Επιστημών «Δημόκριτος». Αρχικός του στόχος ήταν η διεξαγωγή έρευνας στην πυρηνική φυσική. Σήμερα το Ε.Κ.ΕΦ.Ε. «Δημόκριτος» καταπιάνεται με όλες τις θετικές επιστήμες και εστιάζει τόσο στην έρευνα όσο και στην εκπαίδευση.
- 1962 μ.Χ. Η Αμερικάνικη Αυτοβιομηχανία General Motors χρησιμοποιεί τον πρώτο ρομποτικό βραχίονα, τον Unimate για την κατασκευή αυτοκινήτων. Εκτελούσε επικίνδυνες και επαναλαμβανόμενες εργασίες.
- 1968 μ.Χ. Στην ταινία του Stanley Kubrick «Οδύσσεια του Διαστήματος» HAL 9000 αναπτύσσει από μόνος του σκέψη και μένει στην ιστορία με τη φράση του: «Συγγνώμη Ντέιβ, φοβάμαι ότι δεν μπορώ να το κάνω αυτό». Ο HAL αντανakλά την ανησυχία της εποχής για την αυξανόμενη δύναμη των μηχανών πάνω στους ανθρώπους.
- 1970 μ.Χ. Ο Shakey κάνει την αρχή με τα κινούμενα ρομπότ που πραγματοποιούν λογικές διεργασίες. Αποστολή του ήταν να ανοίγει και να κλείνει πόρτες και διακόπτες!
- 1973 μ.Χ. Στην Ιαπωνία κατασκευάζεται ο Wabot-I, το παλαιότερο σε φυσική κλίμακα ανθρωπόμορφο ρομπότ. Μπορούσε να επικοινωνεί στα ιαπωνικά, να μετράει αποστάσεις και να μετακινεί αντικείμενα.
- 1975 μ.Χ. Στις ΗΠΑ κατασκευάζεται ο ρομποτικός βιομηχανικός βραχίονας, P.U.M.A., ένας από τους πιο εξελιγμένους βραχίονες με πολλά μοντέλα μετέπειτα.
- 1977 μ.Χ. Στην ταινία «Ο πόλεμος των άστρων» εμφανίζονται δύο ρομπότ με δραστικά διαφορετική μορφή, ειδικά κατασκευασμένα για να επιτελούν διαφορετικούς ρόλους. Ο ανθρωπόμορφος C3PO και ο R2D2.
- 1982 μ.Χ. Η ταινία “Blade Runner” παρουσιάζει τους ανθρώπους να έχουν δημιουργήσει ρομπότ που δύσκολα μπορεί κάποιος να τα ξεχωρίσει από τις ανθρώπινες οντότητες.
- 1993 μ.Χ. Ο Ντάντε 2, ένα 8ποδο ρομπότ, κατάφερε να μπει στον κρατήρα ενός ηφαιστείου και να μας δώσει πληροφορίες σε συνθήκες που κανένας άνθρωπος δε θα μπορούσε να επιβιώσει.
- 1996 μ.Χ. Ιαπωνική Honda κατασκευάζει τον P2, ένα ανθρωποειδές ρομπότ που περπατά, ανεβαίνει σκάλες και μεταφέρει βάρη.
- 1997 μ.Χ. Το πρώτο τουρνουά ρομποτικού ποδοσφαίρου διεξάγεται στην Ιαπωνία. Στόχος των διοργανωτών, μια ρομποτική ποδοσφαιρική ομάδα να κερδίσει την καλύτερη ανθρώπινη ομάδα μέχρι το 2050.
- 1999 μ.Χ. Η Sony κατασκευάζει το σκυλί Aibo, το πρώτο ρομποτικό σκυλί ικανό να αλληλεπιδρά με τον άνθρωπο.
- 2000 μ.Χ. Σταθμός για την επιστήμη της Ρομποτικής είναι το ρομπότ Asimo (Advanced Step in Innovative Mobility). Μπορεί να τρέχει, να επικοινωνεί με τους ανθρώπους, να αναγνωρίζει εκφράσεις και περιβάλλοντα και να αντιδρά με αυτά. Στην ιατρική δημιουργείται το ρομποτικό χειρουργικό σύστημα Da Vinci. Τα

περισσότερα ρομπότ στον κόσμο εργάζονται στη βιομηχανία. Ο ρομποτικός πληθυσμός είναι περίπου 1.000.000.

- 2002 μ.Χ. Roomba, το ρομπότ – σκούπα και τα διαφορετικά μετέπειτα μοντέλα του, έχει κάνει χιλιάδες πωλήσεις.
- 2003 μ.Χ. Η NASA στέλνει στον Άρη 2 τα δίδυμα ρομπότ εξερευνητές Spirit και Opportunity, τους Mars exploration Rovers, για να μελετήσουν την ιστορία του νερού στον πλανήτη. Ρομποτικός πληθυσμός 2003: 2.100.000
- 2007 μ.Χ. Με πρωτοβουλία του Εργαστηρίου Τεχνολογίας Γνώσεων και Λογισμικού δημιουργείται η Roboskel, η δραστηριότητα ρομποτικής του Ινστιτούτου Πληροφορικής και Τηλεπικοινωνιών Ε.Κ.Ε.Φ.Ε. «Δημόκριτος».
- 2010 μ.Χ. Κατασκευάστηκε στη Γερμανία το ρομπότ Indigo, το πρώτο ρομπότ της ομάδας Roboskel του Δημόκριτου. Την ίδια χρονιά λειτούργησε σαν ξεναγός στην Αρχαία Αγορά των Αθηνών, στον «Ελληνικό Κόσμο», το πολιτιστικό κέντρο του Ιδρύματος Μείζονος Ελληνισμού. Έκτοτε χρησιμοποιείται για ερευνητικούς σκοπούς.
- 2010 μ.Χ. Ο Robonaut, το πρώτο ανθρωποειδές ρομπότ που ταξιδεύει στο διάστημα.
- 2012 μ.Χ. Η Google παρουσίασε το πρώτο αυτοκίνητο – ρομπότ που μπορεί να κινείται χωρίς οδηγό. Το 2017 αναμένεται να κυκλοφορήσει στην παγκόσμια αγορά.
- 2013 μ.Χ. Η δραστηριότητα Roboskel αποκτά το δεύτερο ρομπότ, τον Sek. Έχει τη δυνατότητα να αλλάζει εύκολα εξοπλισμό και να δοκιμάζεται σε ποικίλα περιβάλλοντα και εφαρμογές. Φωτογραφία: Ρομπότ Sek από «Δημόκριτο».
- 2013 μ.Χ. Κατασκευάζεται το μικρότερο ρομπότ του κόσμου, ο Robobee, ένα ιπτάμενο ρομπότ που ζυγίζει λιγότερο από ένα γραμμάριο και έχει άνοιγμα φτερών 3 μόλις εκατοστών. Αναμένεται στο μέλλον να χρησιμοποιηθεί σε επιχειρήσεις διάσωσης. Ρομποτικός πληθυσμός: 18.000.000
- 2014 μ.Χ. Το ρομπότ Ερατώ υποδέχεται τους επισκέπτες του νοσοκομείου «Υγεία» στην Αθήνα.

1.3 Ο ΑΥΤΟΜΑΤΙΣΜΟΣ - ΡΟΜΠΟΤΙΚΗ ΣΤΗΝ ΚΠΑΙΔΕΥΣΗ

Γενικά, το ρομπότ – αυτοματισμοί μπορεί να χρησιμοποιηθεί στην εκπαίδευση - στην Πρωτοβάθμια, Δευτεροβάθμια ακόμα και στην Τριτοβάθμια - αλλά και εκτός σχολικού χώρου ως ένα αποτελεσματικό εργαλείο για την ανάπτυξη γνωστικών δομών από τα παιδιά. Οι αυτοματισμοί, σε σχέση με άλλες μηχανικές κατασκευές, έχουν μία ιδιαίτερη όψη και είναι πολύ πιο εύκολο για τους μαθητές να τα οικειοποιηθούν. Απεικονίζουν μια πρακτική εφαρμογή της Φυσικής, των υπολογιστών και της μηχανικής και ενισχύουν το ενδιαφέρον στις θετικές επιστήμες, γι' αυτό εντάσσονται και στην STEM εκπαίδευση. Ενισχύουν την ικανότητα των παιδιών στην επίλυση προβλημάτων και προάγουν τη λογική και κριτική σκέψη. Επίσης, αυξάνουν την αυτοπεποίθηση των παιδιών, τα οποία δουλεύουν για την επίτευξη συγκεκριμένου στόχου. Αρχικά η ρομποτική στην τάξη βοηθάει τον δάσκαλο να συνδυάσει πολλούς τομείς μαθημάτων (Sirsath , Dhole , Mohire , Naik , & Ratnaparkhi, 2018).

Η εκπαίδευση με τη ρομποτική πραγματοποιείται με την ενεργή συμμετοχή των μαθητών ή φοιτητών, οι οποίοι δουλεύουν σε ομάδες χρησιμοποιώντας ένα εκπαιδευτικό πακέτο που περιέχει επεξεργαστή (μυαλό), αισθητήρες (αισθήσεις) ως εισόδους της κατασκευής, κινητήρες ως εξόδους και δομικά στοιχεία για την ολοκλήρωση της κατασκευής. Η ενασχόληση με τη ρομποτική ενέχει δύο ειδών δραστηριότητες, μια κατασκευαστική και μια προγραμματιστική. Οι διδασκόμενοι κατασκευάζουν ρομποτικές κατασκευές και στη συνέχεια τις προγραμματίζουν, ώστε να δώσουν λύσεις σε αυθεντικά προβλήματα που θέτει ο εκπαιδευτικός με κριτήριο τις εμπειρίες τους, τα ενδιαφέροντα και τις ανάγκες τους.

1.4 ΤΟ ΡΟΜΠΟΤ

Ένα ρομπότ, είναι πολύ περισσότερο από ένα μηχανικό σύστημα. Σ' ένα ρομπότ, επιδιώκουμε να αναπαράγουμε ανθρώπινες και γενικότερα ζωικές λειτουργίες και δυνατότητες, τον τρόπο που οι ζωικές υπάρξεις αυτού του πλανήτη κινούνται, συμπεριφέρονται και δρουν στον εξωτερικό κόσμο. Πολλές φορές αρκεί η αναπαραγωγή ενός ανθρώπινου ή ζωικού αυτοματισμού, για να χαρακτηρίσουμε ένα μηχανικό σύστημα ρομπότ. Για παράδειγμα, εύκολα αποδεχόμαστε σαν ρομπότ ένα σύστημα που μπορεί να κινείται, όπως ένας άνθρωπος ή ένα ζώο. Ένα ακόμα βασικό χαρακτηριστικό ενός ρομπότ είναι η αυτόνομη λειτουργία. Αυτόνομη λειτουργία δεν σημαίνει απλά πως το ρομπότ θα πρέπει να μπορεί να εκτελεί τη λειτουργία που έχει σχεδιαστεί να κάνει, χωρίς τη δική μας επίβλεψη ή παρέμβαση. Θα πρέπει ακόμα να μπορεί να account / cope / accommodate. Ένα ρομπότ σχεδιάζεται, για να εκτελεί μία λειτουργία σε πραγματικές συνθήκες που δεν μένουν αμετάβλητες, αλλά μπορεί να αλλάζουν, χωρίς από πριν να μπορούμε να προβλέψουμε πώς αυτές μπορεί να μεταβληθούν. Θα πρέπει λοιπόν, το ρομπότ να μπορεί να λειτουργεί σ' ένα δυναμικό περιβάλλον, όπου οποιαδήποτε στιγμή, οι συνθήκες σε αυτό το περιβάλλον μπορεί να αλλάξουν και χωρίς τη δική μας παρέμβαση, το ρομπότ θα πρέπει να account / cope / accommodate γι' αυτές τις μεταβολές. Θα πρέπει δηλαδή, το ρομπότ να μπορεί να προσαρμόζει, δηλαδή να τροποποιεί δυναμικά τη λειτουργία και τη συμπεριφορά του, στις μεταβολές του εξωτερικού περιβάλλοντος. Ένα ρομπότ δηλαδή, δεν είναι ένα σύστημα που λειτουργεί μηχανικά, ανεξάρτητα από τι συμβαίνει στο εξωτερικό περιβάλλον, αλλά που μπορεί να αντιλαμβάνεται αυτές τις μεταβολές και ανάλογα να μεταβάλλει τη συμπεριφορά του. Έτσι, ερχόμαστε σ' ένα βασικό χαρακτηριστικό του ανθρώπου και των ζωικών υπάρξεων του πλανήτη μας που ένα σύστημα – μηχανικό αντικείμενο, απαραίτητα θα πρέπει να περιλαμβάνει, προκειμένου να χαρακτηριστεί σαν ρομπότ. Αυτό που είναι το βασικότερο χαρακτηριστικό του ανθρώπου και των ζώων, αλλά και ενός μηχανικού συστήματος, για να χαρακτηριστεί ως ρομπότ, είναι η ευφυΐα.

Σύμφωνα με τον Henry Plotkin, στο Nature of Knowledge, ευφυΐα είναι η δυνατότητα της προσαρμογής. Η δυνατότητα, δυναμικά, να μεταβάλλουμε τη συμπεριφορά μας, σύμφωνα με τις μεταβολές που συμβαίνουν στο εξωτερικό περιβάλλον ή στον βαθμό που μπορούμε, να προσαρμόζουμε το εξωτερικό περιβάλλον στους δικούς μας αντικειμενικούς σκοπούς. Για να αποδώσουμε ευφυΐα σε μία μηχανή – ένα μηχανικό σύστημα, δηλαδή τη δυνατότητα να προσαρμόζει τη λειτουργία της στο εξωτερικό περιβάλλον, συμβάλλει αυτό το πρόγραμμα.

2 ΟΛΟΚΛΗΡΩΜΕΝΑ ΜΙΚΡΟΣΥΣΤΗΜΑΤΑ ΕΠΕΞΕΡΓΑΣΙΑΣ – ARDUINO

2.1. Η ΙΣΤΟΡΙΑ ΤΟΥ ARDUINO

Στην Ivrea της Ιταλίας το 2005 ξεκίνησε ένα σχέδιο προκειμένου να κατασκευαστεί μια συσκευή για τον έλεγχο προγραμμάτων και διαδραστικών σχεδίων από μαθητές, με χαμηλότερο κόστος από άλλα πρωτότυπα συστήματα που ήταν διαθέσιμα εκείνη την περίοδο. Εμπνευστής του σχεδίου αυτού ήταν ο καθηγητής Massimo Banzi, ο οποίος θέλησε να καταστήσει ευκολότερη τη μάθηση των ηλεκτρονικών για τους μαθητές του. Για το λόγο αυτό ζήτησε βοήθεια από τον David Cuatzielles, μηχανικό από το πανεπιστήμιο του Malmo. Οι ιδρυτές Massimo Banzi και David Cuatzielles ξεκίνησαν να παράγουν πλακέτες σε ένα μικρό εργοστάσιο στην Ivrea, στην ίδια περιοχή στην οποία στεγαζόταν η εταιρία υπολογιστών Olivetti. Ονόμασαν το έργο τους Arduin of Ivrea «Arduino» που μεταφράζεται ελεύθερα ως «γενναίος φίλος», με σκοπό να δημιουργήσουν έναν μικροελεγκτή προσιτό ως προς τη χρήση του. Την ανάπτυξη του λογισμικού για τον μικροελεγκτή την ανέθεσαν σε δύο φοιτητές του πανεπιστημίου Malmo. Η πρώτη παρτίδα που παράχθηκε αποτελούνταν από 200 μικροελεγκτές, υπό την εποπτεία του ηλεκτρολόγου μηχανικού Gianluca Martino. Οι μικροελεγκτές αυτοί ονομάστηκαν Serial Arduino και περιελάμβαναν μία ATmega8 με άμεση σύνδεση RS-232 με το μικροελεγκτή και όλα τα επιμέρους περιφερειακά του. Έκτοτε έχουν δημιουργηθεί δεκάδες πλακέτες Arduino παγκοσμίως, χάρη στο ευέλικτο και εύκολο στη χρήση hardware και software. Το Arduino μπορεί να χρησιμοποιηθεί από τον οποιοδήποτε με στοιχειώδεις γνώσεις ηλεκτρονικών που ενδιαφέρεται να δημιουργήσει διαδραστικά αντικείμενα ή περιβάλλοντα, με μόνο περιορισμό τη φαντασία του.

2.2. ΤΙ ΕΙΝΑΙ ΤΟ ARDUINO

Το Arduino (Εικόνα 1) αποτελεί μια υπολογιστική πλατφόρμα βασιζόμενη σε μητρική πλακέτα ανοικτού κώδικα, που περιέχει έναν προγραμματιζόμενο μικροελεγκτή (MCU) και εισόδους/εξόδους (I/O) για σύνδεση με το φυσικό κόσμο. Ο μικροελεγκτής του προγραμματίζεται χρησιμοποιώντας τη γλώσσα προγραμματισμού wiring. Είναι βασισμένη στη c/c++, και περιλαμβάνει ένα σύνολο από βιβλιοθήκες, υλοποιημένες επίσης στη c++.



Εικόνα 1, Το arduino και το λογότυπό του

Μέσα από το περιβάλλον ανάπτυξης κώδικα IDE (ελεύθερο και δωρεάν, Εικόνα 2) η γλώσσα wiring μπορεί να συνταχθεί και να υλοποιηθεί σε οποιοδήποτε λειτουργικό σύστημα. Είναι ανοικτού κώδικα λογισμικό και μας επιτρέπει να αναπτύξουμε



ένα υπολογιστικό σύστημα, το οποίο θα ελέγχει συσκευές του φυσικού κόσμου. Το Arduino μπορεί να χρησιμοποιηθεί για να αναπτύξουμε διαδραστικά αυτοματοποιημένες οντότητες, ικανές να δεχθούν ως εισόδους μια πληθώρα αισθητηρίων οργάνων, αλλά και διαφόρων συσκευών εξόδου, που είναι ικανές να ελέγχουν άλλες συσκευές του φυσικού κόσμου. Η υπολογιστική πλατφόρμα του Arduino μας παρέχει τη δυνατότητα να υλοποιούμε project αυτόνομα σε επίπεδο hardware ή αλληλοεξαρτώμενα σε επίπεδο software, που επικοινωνούν με άλλα ολοκληρωμένα μικροσυστήματα επεξεργασίας και υπολογιστές.

Εικόνα 2 , Περιβάλλον ανάπτυξης κώδικα IDE

Ειδικότερα, η υπολογιστική πλατφόρμα Arduino αποτελείται από μια πλακέτα με πυρήνα, έναν μικροελεγκτή Atmel AVR, όπως ο ATmega328 ή ο ATmega2560 (Εικόνα 3) ανάλογα με την έκδοσή του. Διαθέτει από 8 έως 16 σειριακές θύρες και έως 50 ψηφιακές (αναλόγως με τον τύπο και την έκδοση της πλακέτας) με τις οποίες αλληλεπιδρά με διάφορες συσκευές.

Μικροελεγκτής



Miniature Computer

- Processor, Storage and RAM all in one tiny package!
- Atmel Microcontroller (MCU), typically ATmega328

ATmega328 Information

Component	Specification
Clock Speed	16MHz
Flash Memory	32K
EEPROM	1K
SRAM	2K
Analog -> Digital	6Ch 10bit
Communication	SPI
Digital	14 I/O
PWM	6 Channel (Digital)

Εικόνα 3, Ο ATmega328

2.3. ΜΟΝΤΕΛΑ ARDUINO

Τα μοντέλα Arduino (Εικόνα 6) που κυκλοφορούν στην αγορά μέχρι σήμερα είναι:

- Arduino UNO
- Arduino LilyPad Usb
- Arduino Ethernet

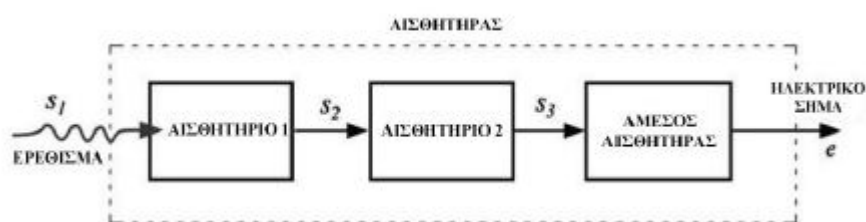


- Arduino Leonard
- Arduino Mega 2560
- Arduino Mega ADK
- Arduino Mega ADK REV3
- Arduino Frio
- Arduino Nano
- Arduino Pro
- Arduino Pro Mini
- Arduino Micro
- Arduino Due
- Arduino Esplora

Εικόνα 6, Διάφοροι τύποι arduino

2.4. ARDUINO SENSORS

Αισθητήρια ονομάζουμε τις ηλεκτρονικές διατάξεις οι οποίες είναι σε θέση να μετατρέπουν ενέργεια άλλης μορφής σε ηλεκτρική κατάλληλη για να μετρηθεί από μετρητικές διατάξεις. Το αισθητήριο έρχεται να γεφυρώσει ένα χάσμα μεταξύ του φυσικού κόσμου και των οργάνων που πρέπει να δημιουργήσει ο άνθρωπος, ώστε να έχει μια ολοκληρωμένη εικόνα των μεταβολών του. Με άλλα λόγια αποτελούν ηλεκτρονικές συσκευές οι οποίες παρακολουθούν τις μεταβολές του φυσικού κόσμου και παράγουν μια έξοδο ηλεκτρικού σήματος, ανάλογη των μεταβολών αυτών. Τα αισθητήρια δε λειτουργούν αυτόνομα, αλλά είναι πάντα μέρος ενός ευρύτερου συστήματος, που μπορεί να ενσωματώνει άλλους ανιχνευτές, επεξεργαστές σήματος, καταγραφείς δεδομένων κ. α.



Εικόνα 7, Δομή αισθητήρα

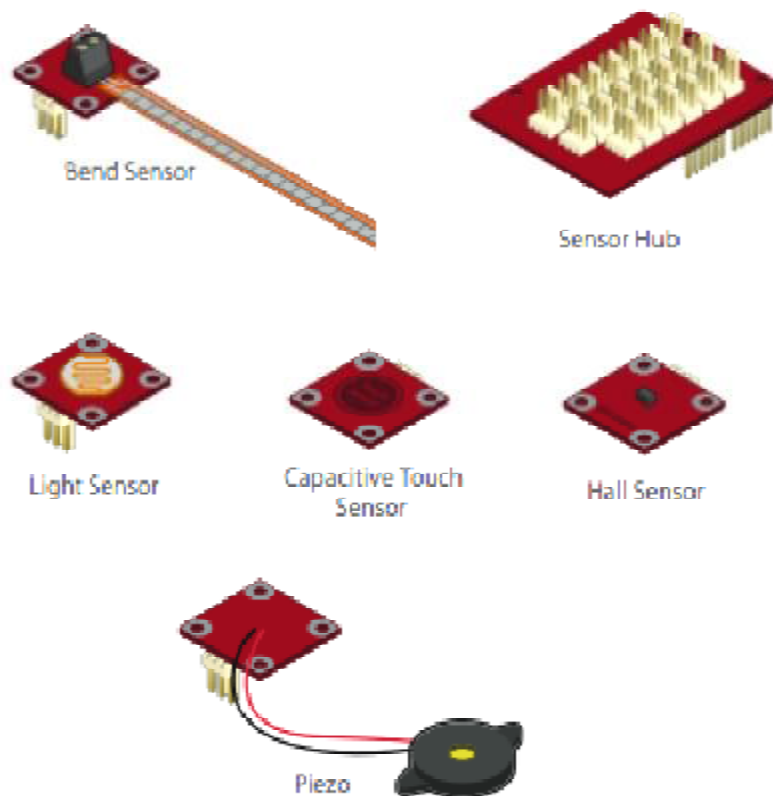
Κάποια από τα πιο γνωστά αισθητήρια arduino που κυκλοφορούν στην αγορά παρουσιάζονται παρακάτω:

- Push buttons Sensors
- Touch pads Sensors
- Photoresistors
- Variable resistors
- Ultrasound -Proximity range finder Sensors
- Thermistors (temperature) Sensors

- Electro Magnetic Sensors
- Gas Sensors
- Hall Effect Sensors
- Magnetometer
- PIR Motion sensor
- Digital Humidity and Temperature Sensor
- Sensor Hub
- Bend Sensor
- Capacitive Touch Sensor
- Light Sensor
- Piezo Sensor

Οι αισθητήρες συνδέονται στις ψηφιακές και αναλογικές εισόδους του arduino, το οποίο στη συνέχεια, κάνοντας χρήση των κατάλληλων βιβλιοθηκών, διαβάζει τις τιμές τάσης στην είσοδό του και στη συνέχεια μετατρέπει την εισερχόμενη τάση σε κατάλληλες τιμές, τις οποίες μπορούμε να αξιοποιήσουμε.

Οι αισθητήρες που χρησιμοποιούνται από την υπολογιστική πλατφόρμα arduino, είναι κατά κανόνα φθηνοί και αξιόπιστοι, με μικρή κατανάλωση ρεύματος και μικροί σε μέγεθος. Λόγω της πληθώρας των αισθητηρίων που κυκλοφορούν στην αγορά (Εικόνα 8), μπορούμε εύκολα να αναζητήσουμε στο διαδίκτυο και σε διάφορες κοινότητες που ασχολούνται με το arduino, αφθονία βιβλιοθηκών.



3. ΠΛΑΤΦΟΡΜΑ ΑΝΑΠΤΥΞΗΣ ΛΟΓΙΣΜΙΚΟΥ ARDUINO IDE

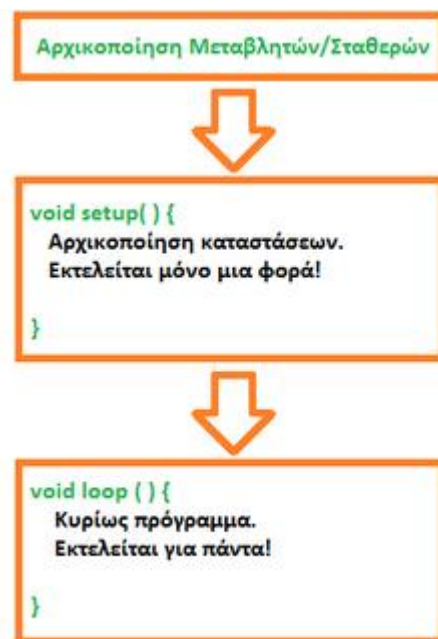
3.1 ΔΟΜΗ ΓΛΩΣΣΑΣ WIRING

Όπως αναφέρθηκε και σε προηγούμενη παράγραφο, 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, στη γλώσσα του Arduino μπορούμε να χρησιμοποιήσουμε τις ίδιες βασικές εντολές και συναρτήσεις, με την ίδια σύνταξη, τους ίδιους τύπων δεδομένων και τους ίδιους τελεστές όπως και στη γλώσσα C. Πέρα από αυτές όμως, υπάρχουν κάποιες ειδικές εντολές, συναρτήσεις και σταθερές που βοηθούν για την διαχείριση του ειδικού hardware του Arduino.

Για την σύνταξη ενός προγράμματος ξεκινάμε χωρίζοντας πάντα το πρόγραμμα σε τρία μέρη: τη δομή, τις μεταβλητές/σταθερές και τις συναρτήσεις.

Πρώτα εξετάζουμε το κομμάτι της δομής. Η δομή ενός προγράμματος Arduino (Σχήμα 1) χωρίζεται σε τρία μέρη με την ακόλουθη σειρά: τη δήλωση μεταβλητών, το κομμάτι κώδικα που περιέχει την αρχικοποίηση καταστάσεων και μεταβλητών καθώς και τον κώδικα που θέλουμε να τρέξει μόνο μια φορά στο Arduino και το κομμάτι του κώδικα loop() που περιέχει το κυρίως πρόγραμμα μας και θα τρέχει συνέχεια (μέχρι να βγάλουμε το Arduino από το ρεύμα).



Σχήμα 1, Βασική δομή προγράμματος σε γλώσσα Wiring

Οι μεταβλητές παίζουν πολύ σημαντικό ρόλο στην σύνταξη του κώδικα καθώς αλλάζοντας τις τιμές τους κατά την διάρκεια εκτέλεσης του προγράμματος μπορούμε να επιτύχουμε διάφορες λειτουργίες. Μπορούν να πάρουν διάφορες τιμές, όπως νούμερα, χαρακτήρες, και

τα δύο ή να έχουν λογική τιμή True ή False (αληθής - ψευδής). Ανάλογα με την τιμή αυτή τις αρχικοποιούμε ή τις δηλώνουμε αντίστοιχα στο πρώτο τμήμα της δομής του προγράμματος μας.

Κάθε όνομα που δίνουμε σε μια μεταβλητή ή σταθερά θα πρέπει να υπάρχει μόνο μια φορά μέσα στον κώδικά μας. Στις σταθερές ισχύει ότι και παραπάνω, αλλά πριν τον τύπο βάζουμε το χαρακτηριστικό 'const' (const “τύπος” “όνομα σταθεράς”). Μια σταθερά χρησιμοποιείται σαν συντόμευση μέσα στο πρόγραμμα, αποφεύγοντας να γράφουμε τιμές τις οποίες μπορεί να ξεχάσουμε παρακάτω. Μια μεταβλητή μπορεί να μην έχει αρχική τιμή, και η τιμή της να υπολογίζεται μετά από εκτέλεση κάποιας εντολής μέσα στο πρόγραμμα (Deerali , Mohd. , & Shreerang , 2013).

Οι πιο σημαντικές από τις εντολές ενός προγράμματος επεξηγούνται στους πίνακες που ακολουθούν:

Βασικές δομές/εντολές	Λειτουργία
setup()	Η συνάρτηση setup() καλείται στην αρχή κάθε σχεδίου. Εκεί αρχικοποιούνται οι μεταβλητές, οι λειτουργίες των pins, οι βιβλιοθήκες κ.τ.λ. Εκτελείται μόνο μια φορά ή κάθε φορά που γίνεται reset.
loop()	Η λειτουργία loop() εκτελείται μετά τη setup(), και όπως δηλώνει και το όνομά της, εκτελείται συνέχεια και ουσιαστικά εκεί βρίσκεται το πρόγραμμα που θέλουμε να εκτελεστεί.
if...else	Δομή επιλογής.
switch case	Δομή επιλογής ανάλογα με την τιμή μιας μεταβλητής. Σε κάθε περίπτωση χρησιμοποιούμε την εντολή break για να τερματίσει η δομή.
for	Δομή επανάληψης με γνωστό αριθμό επαναλήψεων.
while	Δομή επανάληψης· εκτελείται ο βρόχος συνέχεια έως ότου η συνθήκη που ακολουθεί τη while να γίνει ψευδής (false).
do...while	Δομή επανάληψης, ο βρόχος εκτελείται τουλάχιστον μία φορά γιατί η συνθήκη ελέγχεται στο τέλος του βρόχου.
break	Χρησιμοποιείται για την έξοδο από ένα βρόχο (for, while, do..while) ή από μια switch...case παρακάμπτοντας την κανονική εκτέλεση των εντολών.

continue	Χρησιμοποιείται για την παράκαμψη μιας επανάληψης ενός βρόχου και μεταπήδηση στην επόμενη επανάληψη.
return	Τερματίζει μια συνάρτηση ή επιστρέφει μια τιμή σε μια συνάρτηση που έχει κληθεί (τερματίζοντας την).
goto	Μεταφέρει τη ροή του προγράμματος σε άλλο σημείο χρησιμοποιώντας ετικέτες.
;	Χρησιμοποιείται στο τέλος μιας εντολής.
{ }	Περικλείουν ένα σύνολο εντολών. Ένα άγκιστρο όταν ανοίγει θα πρέπει κάπου να κλείνει.
//	Εισάγεται από τον προγραμματιστή για ευκολία ανάγνωσης του σχεδίου.
#define	Με τη #define δηλώνονται στην αρχή του σχεδίου οι σταθερές που δεν θα αλλάξουν τιμή σε όλη τη διάρκεια της εκτέλεσης του σχεδίου.
#include	Χρησιμοποιείται για να δηλώνονται στο σχέδιο διάφορες βιβλιοθήκες.
delay(x)	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.
millis	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.
delayMicroseconds(x)	Καθυστερεί την εκτέλεση του σχεδίου για x microseconds.

Πίνακας 1, Βασικές δομές-Εντολές

Μεταβλητές/Τύποι δεδομένων	Λειτουργία
HIGH/LOW	Όταν διαβάζεται ή γράφεται η τιμή ενός digital pin, υπάρχουν δύο δυνατές τιμές που μπορεί να πάρει: η HIGH και η LOW.
true/false	Λογικές σταθερές για να εκφραστεί η «αλήθεια» ή το «ψέμα».
INPUT	Έχει την τιμή 0 και είναι αντίστοιχη του λογικού false.
OUTPUT	Έχει την τιμή 1 και είναι αντίστοιχη του λογικού true.
void	Χρησιμοποιείται για τη δήλωση συναρτήσεων. Οι συναρτήσεις αυτές δεν επιστρέφουν καμιά τιμή.
boolean	Μία Boolean μεταβλητή κρατά μια από τις τιμές true ή false. Κάθε Boolean μεταβλητή καταλαμβάνει 1 byte μνήμης.
char	Μία μεταβλητή char κρατά έναν χαρακτήρα. Οι χαρακτήρες αποθηκεύονται σαν αριθμοί με πρόσημο από -128 έως 127 (σύμφωνα με τον κώδικα ASCII)
unsigned char	Τύπος char χωρίς πρόσημο (κωδικοποιεί αριθμούς από 0 έως 255).
byte	Μία μεταβλητή byte αποθηκεύει έναν αριθμό χωρίς πρόσημο 8bits από το 0 έως 255 (ίδιο με unsigned char).
int	Μία μεταβλητή int αποθηκεύει ακέραιους αριθμούς. Καταλαμβάνει 2 bytes στη μνήμη. Το εύρος των ακεραίων είναι από -32,768 έως 32,767.
unsigned int	Μια τέτοια μεταβλητή αποθηκεύει ακεραίους αριθμούς χωρίς πρόσημο. Καταλαμβάνει 2 bytes στη μνήμη. Το εύρος των ακεραίων είναι από 0 έως 65,535.
word	Μία μεταβλητή word αποθηκεύει έναν 16bit αριθμό χωρίς πρόσημο από 0 έως 65,535
long	Χρησιμοποιείται για την αποθήκευση αριθμών των 32bits
float	Μια μεταβλητή τύπου float αυτή αποθηκεύει αριθμούς με υποδιαστολή. Έχει 6-7 ψηφία ακρίβειας δεκαδικών.
double	Αποθηκεύει δεκαδικούς αριθμούς. Χρησιμοποιεί 32bits (4 bytes). Διαφέρει

	με τη float στην ακρίβεια των δεκαδικών.
string - char array	Χρησιμοποιείται για την αποθήκευση strings. Τα strings αποθηκεύονται σαν πίνακες (array). Ένας χαρακτήρας τοποθετείται σε μονά εισαγωγικά ('...'), ενώ πολλοί χαρακτήρες σε διπλά εισαγωγικά ("..").
String – object	Η κλάση String μας επιτρέπει να διαχειριζόμαστε τα strings με διαφορετικό και πιο εύκολο τρόπο από την char.
array	Πρόκειται για μια συλλογή από μεταβλητές στις οποίες έχουμε πρόσβαση χρησιμοποιώντας απλά ένα δείκτη.
static	Χρησιμοποιείται για να δηλωθούν μεταβλητές οι οποίες θα είναι ορατές μόνο από μια συνάρτηση. Μια static μεταβλητή διατηρεί τα δεδομένα της μεταξύ των κλήσεων της ίδιας συνάρτησης.
volatile	Χρησιμοποιείται για να κατευθύνει τον compiler να φορτώσει τα δεδομένα της συγκεκριμένης μεταβλητής από τη RAM και όχι από τους καταχωρητές που αποθηκεύεται το πρόγραμμα και οι άλλες μεταβλητές. Χρησιμοποιείται κυρίως για να διαχειριζόμαστε σωστά τα interrupts.
const	Χρησιμοποιείται για να δηλώσουμε μια μεταβλητή σαν σταθερά

Πίνακας 2, Μεταβλητές και τύποι δεδομένων

Ψηφιακές συναρτήσεις I/O	Λειτουργία
pinMode()	Καθορίζει αν το συγκεκριμένο ψηφιακό <i>pin</i> θα είναι pin εισόδου ή pin εξόδου ανάλογα με την τιμή που δίνεται στην παράμετρο <i>mode</i> (INPUT ή OUTPUT αντίστοιχα).
digitalWrite()	Θέτει την κατάσταση <i>pinstatus</i> (HIGH ή LOW) στο συγκεκριμένο ψηφιακό <i>pin</i> .
digitalRead()	Επιστρέφει την κατάσταση του συγκεκριμένου ψηφιακού <i>pin</i> (0 για LOW και 1 για HIGH) εφόσον αυτό είναι pin εισόδου.

Πίνακας 3, Ψηφιακές συναρτήσεις εισόδου/εξόδου

Αναλογικές I/O	Λειτουργία
analogReference(type)	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 pin AREF αντίστοιχα)
analogRead()	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()	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

Πίνακας 4, Αναλογικές συναρτήσεις εισόδου/εξόδου

Συναρτήσεις σειριακής επικοινωνίας	Λειτουργία
serial.available()	Λαμβάνει τον αριθμό των bytes (χαρακτήρων) που είναι διαθέσιμοι για διάβασμα στη σειριακή θύρα και τους αποθηκεύει στον προσωρινό καταχωρητή σειριακής θύρας, ο οποίος έχει μέγεθος 64 bytes.
serial.read()	Διαβάζει δεδομένα από τη σειριακή θύρα.
serial.print()	Γράφει δεδομένα στη σειριακή θύρα σαν ASCII χαρακτήρες.
serial.println()	Διοχετεύει τα δεδομένα <i>data</i> για αποστολή μέσω του σειριακού interface. Η παράμετρος <i>data</i> μπορεί να είναι είτε αριθμός είτε αλφαριθμητικό.
Serial.begin	Θέτει τον ρυθμό μεταφοράς δεδομένων του σειριακού interface (σε baud)

Πίνακας 5, Βασικές συναρτήσεις σειριακής επικοινωνίας

Εντολές Διακοπών	Λειτουργία
attachInterrupt	Θέτει σε λειτουργία το συγκεκριμένο <i>interrupt</i> , ώστε 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"> • LOW (ενεργοποίηση όταν η κατάσταση του pin που αντιστοιχεί στο συγκεκριμένο interrupt γίνει LOW) • RISING (όταν από LOW γίνει HIGH) • FALLING (όταν από HIGH γίνει LOW) • CHANGE (όταν αλλάξει κατάσταση γενικά)
detachInterrupt	Απενεργοποιεί το συγκεκριμένο <i>interrupt</i> .
noInterrupts	Σταματά προσωρινά την λειτουργία όλων των interrupt
interrupts	Επαναφέρει την λειτουργία των interrupt που διακόπηκε προσωρινά από μια εντολή noInterrupts.

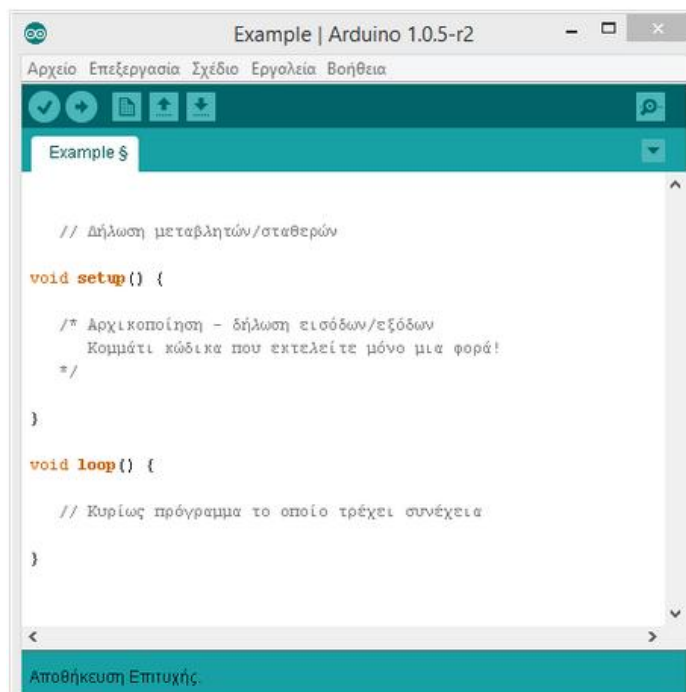
Πίνακας 6, Εντολές διακοπών

3.2 ΛΟΓΙΣΜΙΚΟ IDE

Το Arduino IDE (freeware λογισμικό, Εικόνα 9) αποτελεί ένα ολοκληρωμένο περιβάλλον ανάπτυξης (integrated development environment) και είναι απλούστερο και πιο φιλικό προς το χρήστη, σε αντίθεση με παρόμοια περιβάλλοντα ανάπτυξης όπως το Eclipse , το Xcode και το Visual Studio.

Περιέχει έναν επεξεργαστή πηγαίου κώδικα («editor»), μεταγλωττιστή («compiler»), εργαλεία αυτόματης παραγωγής κώδικα, αποσφαλματωτή («debugging»), συνδέτη, σύστημα ελέγχου εκδόσεων και εργαλεία κατασκευής γραφικών διασυνδέσεων χρήστη για τις υπό ανάπτυξη εφαρμογές.

Μέσα στο μενού υπάρχουν οι επιλογές για την σειριακή θύρα («serial port») με την οποία θα επικοινωνεί το Arduino με τον Η/Υ καθώς και η καρτέλα επιλογής της υπολογιστικής πλατφόρμας Arduino που χρησιμοποιείται κάθε φορά («board»).









Εικόνα 9, Λογισμικό IDE

Με άλλα λόγια συνδέουμε το hardware μέρος του arduino με το λογισμικό IDE για να φορτώσουμε το πρόγραμμα που έχουμε συντάξει και να πραγματοποιηθεί η μεταξύ τους επικοινωνία.

Για την αποκατάσταση της επικοινωνίας αυτής θα πρέπει το Arduino να συνδεθεί σε μια από τις θύρες USB του υπολογιστή και λόγω του ελεγκτή Serial-over-USB, θα πρέπει να αναγνωριστεί από το λειτουργικό σύστημα ως εικονική σειριακή θύρα.

Για την σύνδεση απαιτείται ένα καλώδιο USB από Type A σε Type B, όπως αυτό των εκτυπωτών. Για την αναγνώριση από το λειτουργικό σύστημα είναι απαραίτητη η εγκατάσταση τον οδηγό του FTDI chip (δηλαδή του ελεγκτή Serial-over-USB) ο οποίος υπάρχει στον φάκελο drivers του Arduino IDE.

Μετά από αυτό στο κεντρικό παράθυρο του Arduino IDE θα εμφανιστεί, στο μενού Tools → Serial Port , η εικονική σειριακή θύρα (συνήθως COM#) έτοιμη να δεχτεί τους κώδικες. Ο κώδικας που έχει γραφεί για το Arduino ονομάζεται sketch. Στον ακόλουθο πίνακα παρουσιάζονται τα εργαλεία του περιβάλλοντος ανάπτυξης, υπό μορφή κουμπιών.

Εργαλείο	Περιγραφή
 Verify	Ελέγχει για συντακτικά λάθη στον κώδικα.
 Upload	Μεταγλωττίζει τον κώδικα και τον φορτώνει στο Arduino. Αν δεν είναι συντακτικά σωστός δεν μπορεί να γίνει η φόρτωση.
 New	Δημιουργεί ένα νέο sketch.
 Open	Παραθέτει ένα menu με όλα τα sketch. Ενεργοποιώντας ένα από αυτά, θα ανοίξει αυτόματα στο τρέχον παράθυρο.
 Save	Αποθηκεύει ένα sketch.
 Serial Monitor	Ανοίγει την σειριακή οθόνη και μέσω αυτής παρακολουθείται η ανταλλαγή δεδομένων που γίνεται στην σειριακή θύρα.

Πίνακας 6, Εργαλεία του περιβάλλοντος IDE

Αξίζει να τονιστεί πως με την ολοκλήρωση του προγράμματος sketch, για να αναγνωρισθεί από τον μεταγλωττιστή C++ ως έγκυρο πρόγραμμα, ο χρήστης θα πρέπει να κάνει κλικ στο κουμπί "Upload to I/O board" του IDE και ένα αντίγραφο του κώδικα να γραφτεί σε ένα προσωρινό αρχείο με ένα παραπάνω include στην κορυφή και μία πολύ απλή συνάρτηση main() στο τέλος, ώστε τελικά να φτιαχτεί ένα έγκυρο C++ πρόγραμμα.

Συγκεντρωτικά το Arduino IDE παρέχει:

- Ένα πρακτικό περιβάλλον για την συγγραφή των προγραμμάτων (τα οποία ονομάζονται sketch στην ορολογία του Arduino) .
- Μερικές έτοιμες βιβλιοθήκες για προέκταση της γλώσσας και για να χειριζόμαστε εύκολα μέσα από τον κώδικά μας τα εξαρτήματα που συνδέουμε στο Arduino.
- Τον compiler για την μεταγλώττιση των sketch

- Ένα serial monitor που παρακολουθεί τις επικοινωνίες της σειριακής (USB), αναλαμβάνει να στείλει αλφαριθμητικά της επιλογής μας στο Arduino μέσω αυτής, και είναι ιδιαίτερα χρήσιμο για το debugging των sketch.
- Την επιλογή να ανεβάσουμε το μεταγλωττισμένο sketch στο Arduino.

ΚΕΦΑΛΑΙΟ 4

4.1 ΣΥΣΤΗΜΑΤΑ ΑΥΤΟΜΑΤΟΥ ΕΛΕΓΧΟΥ

Συστήματα Αυτομάτου Ελέγχου ονομάζουμε κάθε διάταξη φυσικών στοιχείων τα οποία είναι διασυνδεδεμένα με τέτοιο τρόπο ώστε να καθοδηγούν, ελέγχουν ή ρυθμίζουν τον εαυτό τους ή άλλα συστήματα ώστε να συμπεριφέρονται (λειτουργούν) με ένα προδιαγεγραμμένο τρόπο.

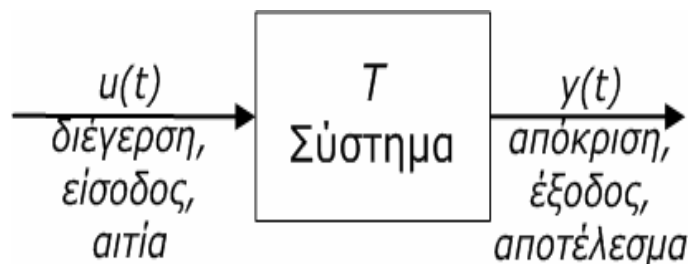
Παραδείγματα:

- Οικιακές συσκευές όπως ψυγείο, πλυντήριο, ηλεκτρικός θερμοσίφοντας
- Βιομηχανικά συγκροτήματα κατασκευής χαρτιού, ζάχαρης, λιπασμάτων αυτοκινήτων
- Φυτικός και ζωικός κόσμος: Το πληρέστερο παράδειγμα συστήματος αυτομάτου ελέγχου είναι ο άνθρωπος
- Οδήγηση αυτοκινήτου

Η Αναλυτική Δομή Συστημάτων Αυτομάτου Ελέγχου

Τα Σύστημα αποτελούνται από ένα σύνολο στοιχείων κατάλληλα συνδεδεμένων μεταξύ τους για να επιτελέσουν κάποιο έργο

- Είσοδο ονομάζουμε τη διέγερση, εντολή ή αιτία η οποία εφαρμόζεται στο σύστημα ώστε να επιτελεστεί το ζητούμενο έργο
- Έξοδο ονομάζουμε το πραγματικό αποτέλεσμα ή απόκριση η οποία λαμβάνεται από την εφαρμογή της εισόδου στο σύστημα.



Κατηγορίες Συστημάτων Αυτομάτου Ελέγχου

Τα συστήματα αυτομάτου ελέγχου διακρίνονται σε δύο κατηγορίες: Ανοικτά και Κλειστά συστήματα

- **Ü** Ανοικτό είναι ένα σύστημα όταν η είσοδος $u(t)$ δεν είναι συνάρτηση της εξόδου $y(t)$
- **Ü** Κλειστό είναι ένα σύστημα όταν η είσοδος $u(t)$ είναι συνάρτηση της εξόδου $y(t)$

Τα **ανοικτά** συστήματα έχουν δύο σημαντικά χαρακτηριστικά:

1. Η απόδοσή τους (δηλαδή το κατά πόσο επιτυγχάνουν την επιθυμητή έξοδο) εξαρτάται από το καλιμπράρισμα που τους έχει γίνει. Καλιμπράρισμα ονομάζουμε τη ρύθμιση της σχέσης εισόδου-εξόδου έτσι ώστε να επιτυγχάνεται η απαιτούμενη ακρίβεια όσον αφορά το επιθυμητό αποτέλεσμα (π.χ. Η ρύθμιση της διάρκειας ψησίματος στην τοστιέρα ώστε να επιτευχθεί το σωστό ψήσιμο του ψωμιού)
2. Δεν παρουσιάζουν προβλήματα ευστάθειας (π.χ. Στην περίπτωση της τοστιέρας δεν υπάρχει περίπτωση αδιάκοπης λειτουργίας επειδή δεν έχει επιτευχθεί το επιθυμητό αποτέλεσμα). Παραδείγματα ανοικτών συστημάτων αυτομάτου ελέγχου: Πλυντήριο: Ο

ρυθμιστής είναι το πρόγραμμα που θέτει το πλυντήριο σε μια σειρά από λειτουργίες όπως αλλαγή νερού, στύψιμο ρούχων κλπ. Η έξοδος του συστήματος (δηλ. ο βαθμός καθαρισμού των ρούχων δεν λαμβάνεται υπόψη στο καθορισμό της επόμενης λειτουργίας).

4.2 Η ΕΞΕΛΙΞΗ ΣΤΟΝ ΨΗΦΙΑΚΟ ΕΛΕΓΧΟ

Στα περισσότερα σύγχρονα συστήματα μηχανικής, είναι απαραίτητο να ελέγχεται η εξέλιξη με τον χρόνο μιας ή περισσοτέρων από τις μεταβλητές του συστήματος. Οι ελεγκτές πρέπει να εξασφαλίζουν ικανοποιητική συμπεριφορά παροδικής και σταθερής κατάστασης για αυτά τα συστήματα μηχανικής. Για την εξασφάλιση ικανοποιητικών επιδόσεων παρουσία διαταραχών και αβεβαιότητα μοντέλου, οι περισσότεροι ελεγκτές που χρησιμοποιούνται σήμερα χρησιμοποιούν κάποια μορφή αρνητικής ανατροφοδότηση. Απαιτείται ένας αισθητήρας για τη μέτρηση της ελεγχόμενης μεταβλητής και τη σύγκρισή της συμπεριφορά σε ένα σήμα αναφοράς. Η ενέργεια ελέγχου βασίζεται σε ένα ορισμένο σήμα σφάλματος ως τη διαφορά μεταξύ της τιμής αναφοράς και των πραγματικών τιμών.

Ο ελεγκτής που χειρίζεται το σήμα σφάλματος για να καθορίσει το επιθυμητό η δράση ελέγχου ήταν κλασικά ένα αναλογικό σύστημα, το οποίο περιλαμβάνει ηλεκτρικές, υγρών, πνευματικών ή μηχανικών εξαρτημάτων. Όλα αυτά τα συστήματα έχουν αναλογικούς εισόδους και εξόδους (δηλ. τα σήματα εισόδου και εξόδου τους καθορίζονται σε ένα συνεχές χρονικό διάστημα και έχουν τιμές που ορίζονται σε ένα συνεχές εύρος των τιμών). Τις τελευταίες δεκαετίες, υπήρχαν αναλογικοί ελεγκτές οι οποίοι αντικαθιστούνται από ψηφιακούς ελεγκτές των οποίων οι εισόδους και οι εξόδους ορίζονται σε διακριτά χρονικά περιστατικά (Charith Perera, 2008).

Οι ψηφιακοί ελεγκτές έχουν τη μορφή ψηφιακών κυκλωμάτων. Διαισθητικά, κάποιος θα πίστευε ότι οι ελεγκτές που παρακολουθούν συνεχώς την έξοδο ενός συστήματος θα ήταν ανώτερη από αυτά που βασίζουν τον έλεγχο τους στο δείγμα από τις τιμές παραγωγής. Φαίνεται ότι οι μεταβλητές ελέγχου (έξοδοι ελεγκτή) θα επιτύγχανε καλύτερο έλεγχο από εκείνους που αλλάζουν περιοδικά. Αυτό είναι στην πραγματικότητα αλήθεια! Αν όλοι οι άλλοι παράγοντες ήταν πανομοιότυποι για ψηφιακά και αναλογικό έλεγχο, ο αναλογικός έλεγχος θα ήταν ανώτερος από τον ψηφιακό έλεγχο. Τι τότε, είναι ο λόγος πίσω από την αλλαγή από αναλογικό σε ψηφιακό που έχει συμβεί κατά τη διάρκεια του τελευταίες δεκαετίες;

Γιατί ψηφιακό έλεγχο;

Ο ψηφιακός έλεγχος προσφέρει ξεχωριστά πλεονεκτήματα έναντι του αναλογικού ελέγχου που εξηγούν τη δημοτικότητά του. Εδώ είναι μερικά από τα πολλά πλεονεκτήματα:

Ακρίβεια. Τα ψηφιακά σήματα αντιπροσωπεύονται με όρους μηδενικών και με αυτά με συνήθως 12 bit ή περισσότερα για να αντιπροσωπεύει έναν μόνο αριθμό. Αυτό περιλαμβάνει ένα πολύ μικρό σφάλμα σε σύγκριση με τα αναλογικά σήματα, όπου η μετατόπιση θορύβου και παροχής ενέργειας είναι πάντα παρόντες.

Σφάλματα εφαρμογής. Η ψηφιακή επεξεργασία των σημάτων ελέγχου περιλαμβάνει προσθήκη και πολλαπλασιασμό με αποθηκευμένες αριθμητικές τιμές. Τα σφάλματα που προκύπτουν από την ψηφιακή αναπαράσταση και την αριθμητική είναι αμελητέα. Αντίθετα, το η επεξεργασία των αναλογικών σημάτων γίνεται με τη χρήση εξαρτημάτων όπως αντιστάσεις και πυκνωτές με πραγματικές τιμές που διαφέρουν σημαντικά από την ονομαστική τιμές σχεδιασμού.

Ευκαμψία. Ένας αναλογικός ελεγκτής είναι δύσκολο να τροποποιηθεί ή να επανασχεδιαστεί μία φορά εφαρμοστεί στο υλικό. Ένας ψηφιακός ελεγκτής εφαρμόζεται σε firmware ή λογισμικό και η τροποποίησή του είναι δυνατή χωρίς πλήρη αντικατάσταση του του αρχικού ελεγκτή. Επιπλέον, η δομή του ψηφιακού ελεγκτή χρειάζεται δεν ακολουθεί μία από τις απλές μορφές που χρησιμοποιούνται συνήθως στον αναλογικό έλεγχο.

Οι πιο σύνθετες δομές ελεγκτών περιλαμβάνουν μερικές επιπλέον αριθμητικές λειτουργίες και είναι εύκολα υλοποιήσιμα. Ταχύτητα. Η ταχύτητα του υλικού του υπολογιστή έχει αυξηθεί εκθετικά από το Δεκαετία του '80. Αυτή η αύξηση της ταχύτητας επεξεργασίας κατέστησε δυνατή τη δειγματοληψία και σήματα ελέγχου της διαδικασίας σε πολύ υψηλές ταχύτητες. Επειδή το διάστημα μεταξύ δειγμάτων, η περίοδος δειγματοληψίας, μπορεί να γίνει πολύ μικρό, ψηφιακοί ελεγκτές επίτευξη επιδόσεων που είναι ουσιαστικά ίδιες με εκείνες που βασίζονται σε συνεχείς παρακολούθηση της ελεγχόμενης μεταβλητής (Charith Perera, 2008).

Κόστος. Αν και οι τιμές των περισσότερων αγαθών και υπηρεσιών έχουν σταθερά αυξηθεί, το κόστος των ψηφιακών κυκλωμάτων συνεχίζει να μειώνεται. Προκαταβολές σε πολύ μεγάλη κλίμακα (VLSI) έχουν καταστήσει δυνατή την παραγωγή καλύτερων, ταχύτερα και πιο αξιόπιστα ολοκληρωμένα κυκλώματα και να τα προσφέρουμε στον καταναλωτή σε χαμηλότερη τιμή. Αυτό έχει κάνει τη χρήση ψηφιακών ελεγκτών πιο οικονομικά ακόμη και για μικρές εφαρμογές χαμηλού κόστους.

4.3 ΣΥΣΤΗΜΑΤΑ ΚΑΙ ΜΕΘΟΔΟΙ (SMART HOME TECHNOLOGY)

Η τεχνολογία Smart Home ξεκίνησε για περισσότερο από μια δεκαετία για να εισαγάγει την έννοια των συσκευών δικτύωσης και εξοπλισμού στο σπίτι. Σύμφωνα με την Ένωση Smart Homes, ο καλύτερος ορισμός της έξυπνης οικιακής τεχνολογίας είναι: η ενσωμάτωση της τεχνολογίας και των υπηρεσιών μέσω της οικιακής δικτύωσης για μια καλύτερη ποιότητα ζωής. Πολλά εργαλεία που χρησιμοποιούνται σε συστήματα υπολογιστών μπορούν επίσης να ενσωματωθούν στα Smart Home Systems. Στην παρούσα εργασία παρουσιάζουμε κάποιες Τεχνολογίες και τα εργαλεία που μπορούν να ενσωματωθούν ή να εφαρμοστούν σε συστήματα Smart Home.



Μια ανασκόπηση του Διαδικτύου των πραγμάτων για έξυπνο σπίτι:

Με την αναμενόμενη αύξηση του παγκόσμιου πληθυσμού, η ζήτηση ενέργειας αυξάνεται συνεχώς. Τα υπάρχοντα ηλεκτρικά δίκτυα δημιουργήθηκαν πριν από δεκαετίες και παρά το γεγονός ότι αναβαθμίζονται τακτικά, η ικανότητά τους να εκπληρώνουν τις μελλοντικές απαιτήσεις είναι αβέβαιη. Τα υφιστάμενα αποθέματα ορυκτών καυσίμων είναι περιορισμένα και επιβάλλουν επιβλαβείς εκπομπές, καθιστώντας αναπόφευκτες κοινωνικές και περιβαλλοντικές επιπτώσεις. Το αποτέλεσμα της τρέχουσας κατάστασης είναι η μετάβαση των παραδοσιακών συγκεντρωτικών πλεγμάτων σε ένα καταναλωμένο σύστημα παραγωγής υβριδικής ενέργειας που βασίζεται σε ανανεώσιμες πηγές ενέργειας, όπως τα αιολικά και ηλιακά συστήματα (Lund et al., 2015), η βιομάζα, κυψέλες καυσίμου και η παλιρροιακή ισχύς.

Το έξυπνο δίκτυο είναι μια ιδέα που ενσωματώνει τεχνολογίες πληροφορίας και επικοινωνίας (ΤΠΕ) με συστήματα ηλεκτρικού δικτύου, προκειμένου να επιτευχθεί αποτελεσματική και ευφυής παραγωγή και κατανάλωση ενέργειας (Iyer and Agrawal, 2010). Χαρακτηρίζεται από μια αμφίδρομη ροή τόσο της ηλεκτρικής ενέργειας όσο και της πληροφορίας .

Οι προσεγγίσεις στο έξυπνο δίκτυο περιλαμβάνουν νέες λύσεις που θα εκμεταλλεύονταν αποτελεσματικά το υπάρχον δίκτυο ηλεκτρικής ενέργειας, προκειμένου να μειώσουν ή να εξαλείψουν τα ρεύματα, τις τάσεις και τις υπερφόρτωση. Τα συστήματα θα μπορούσαν να ωφεληθούν, καθώς η ζήτηση φορτίου σε κρίσιμες καταστάσεις θα μειωνόταν. Εάν η ζήτηση είναι μεγαλύτερη από τη συνολική παραγωγή, τα συστήματα αυτά θα μπορούσαν να προλαμβάνουν την πτώση του δικτύου ή τις μεγάλες διακοπές και να αυξάνουν την αξιοπιστία, την ποιότητα, την ασφάλεια και την ασφάλεια του ηλεκτρικού δικτύου. Οι λύσεις έξυπνου δικτύου μπορούν να εφαρμοστούν σε όλα τα τμήματα του δικτύου: παραγωγή, μετάδοση και διανομή (Παρασκευόπουλος, 2001).

Το έξυπνο σπίτι αναφέρεται στο χρήση των ΤΠΕ στον εγχώριο έλεγχο, που κυμαίνονται από τις συσκευές ελέγχου μέχρι την αυτόματη προσαρμογή των χαρακτηριστικών του σπιτιού (παράθυρα, φωτισμός κ.λπ.). Ένα βασικό στοιχείο του έξυπνου σπιτιού είναι η χρήση αλγορίθμων ευφυούς προγραμματισμού ισχύος, οι οποίοι θα παράσχουν στους κατοίκους τη δυνατότητα να κάνουν βέλτιστες επιλογές σχετικά με τον τρόπο κατανάλωσης ηλεκτρισμού, προκειμένου να μειωθεί η κατανάλωση ενέργειας. Ένας άλλος όρος που χρησιμοποιείται συνήθως είναι το έξυπνο σπίτι ή ο αυτοματισμός του σπιτιού. Ο συνδυασμός των τεχνολογιών της πληροφορίας και των προηγμένων συστημάτων επικοινωνίας και ανίχνευσης δημιουργεί μια ποικιλία νέων δυναμικών εφαρμογών. Οι νέες εξειδικευμένες έννοιες, όπως η διάχυτη ή καθολική υπολογιστική (Greenfield, 2006), όπου ο υπολογισμός γίνεται να εμφανίζεται παντού και οπουδήποτε, διαθέτει τεράστιο δυναμικό για εφαρμογή σε έξυπνο δίκτυο (Parikh et al., 2010). Έξυπνες συσκευές ή αντικείμενα, ικανά για επικοινωνία και υπολογισμό, που κυμαίνονται από απλές οικιακές συσκευές αισθητήρων και έξυπνα τηλέφωνα, υπάρχουν κάθε μέρα γύρω μας.

Το ετερογενές δίκτυο που αποτελείται από τέτοιου είδους αντικείμενα εμπίπτει στην ομπρέλα μιας έννοιας με μια ταχέως αναπτυσσόμενη pop-ularity, που αναφέρεται ως Internet of Things (IoT). Το IoT αντιπροσωπεύει ένα παγκόσμιο δίκτυο μοναδικά διευθυντικών αντικειμένων που συνδέονται μεταξύ τους. Σύμφωνα με τους Gubbi. (2013), το IoT είναι η διασύνδεση των αισθητήριων και ενεργοποιητικών συσκευών που παρέχουν τη δυνατότητα ανταλλαγής πληροφοριών μεταξύ των πλατφορμών μέσω ενός ενοποιημένου πλαισίου, εξαλείφοντας μια κοινή εικόνα λειτουργίας για την παροχή καινοτόμων εφαρμογών.

Αυτό επιτυγχάνεται με την απρόσκοπτη αίσθηση παντού, την ανάλυση δεδομένων και την αναπαράσταση πληροφοριών με το Cloud computing ως το ενιαίο πλαίσιο. Επομένως, το Διαδίκτυο των πραγμάτων στοχεύει στη βελτίωση της άνεσης και της αποτελεσματικότητάς του, επιτρέποντας τη συνεργασία μεταξύ έξυπνων αντικειμένων.

Το πρότυπο IoT αποτελείται συνήθως από πολλά ασύρματα δίκτυα αισθητήρων (WSN) και αναγνώριση ραδιοσυχνοτήτων (RFID). Το ασύρματο δίκτυο αισθητήρων αποτελεί ένα παράδειγμα που εξεδόθη από την ερευνητική κοινότητα τις τελευταίες δύο δεκαετίες.

Ένα WSN αποτελείται από έξυπνη αντίληψη που μπορεί να επικοινωνήσει μέσω άμεσης ραδιοεπικοινωνίας. Οι συσκευές RFID δεν είναι τόσο εξειδικευμένες. Αποτελούνται κυρίως από δύο μέρη: ένα ενσωματωμένο κύκλωμα με κάποιες υπολογιστικές δυνατότητες και μια επικοινωνία κεραίας (Muhammad Raisul Alam, 2012).

ΚΕΦΑΛΑΙΟ 5

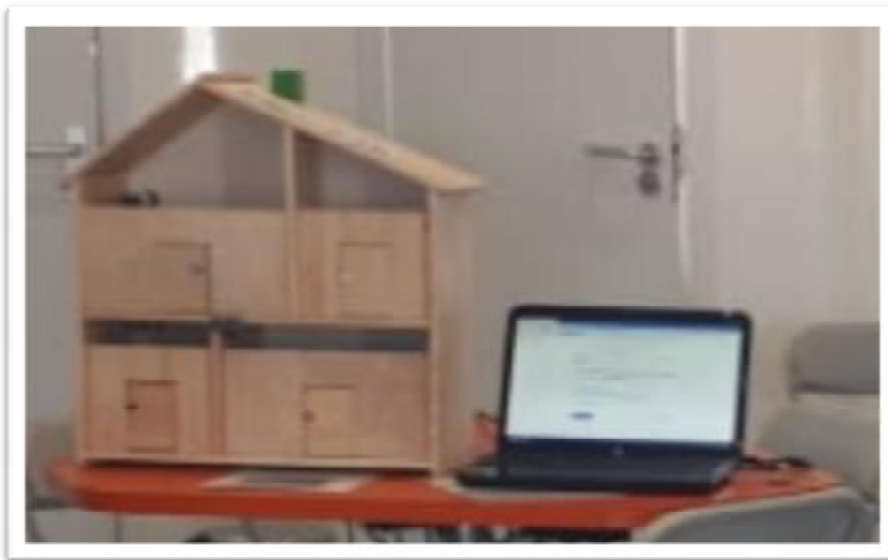
5.1 ΣΧΕΔΙΑΣΜΟΣ ΚΑΙ ΕΦΑΡΜΟΓΗ ΕΝΟΣ ΕΞΥΠΝΟΥ ΟΙΚΙΑΚΟΥ ΣΥΣΤΗΜΑΤΟΣ

Σε αυτή την εργασία, εφαρμόσαμε τις ακόλουθες τρεις λειτουργίες για να δημιουργήσουμε ένα έξυπνο περιβάλλοντος στο προτεινόμενο έξυπνο οικιακό σύστημα:

- (1) αυτοματοποιημένο σύστημα ελέγχου οικιακών συσκευών
- (2) ασφάλεια στο σπίτι.

Η αρχιτεκτονική του προτεινόμενου έξυπνου οικιακού συστήματος αποτελείται από μονάδα επεξεργασία πληροφοριών (μικροελεγκτής Arduino UNO), μια μονάδα λήψης αποφάσεων (προσωπικός υπολογιστής, υπολογιστής), μια έξυπνη διεπαφή παρακολούθησης και μια μονάδα οικιακών συσκευών.

Μια επισκόπηση του συστήματος για το έξυπνο οικιακό σύστημα απεικονίζεται στο σχήμα



Στο μπροστινό μέρος υπάρχει ένας αισθητήρας που χρησιμοποιείται για την ανίχνευση σημάτων κίνησης που παράγονται από τις κινήσεις των χεριών και των ποδιών (οποιαδήποτε εξωτερική κίνηση). Η μονάδα κυκλώματος ενσωματώνει και αισθητήρες θερμοκρασίας και υγρασίας για τη λειτουργία της ασφάλειας στο σπίτι. Η μονάδα επεξεργασίας είναι υπεύθυνη για τη σύνδεση ενσύρματου δέκτη μέσω Ethernet καλωδίου συλλέγοντας τα σήματα κίνησης, και τις θερμοκρασίες σε πραγματικό χρόνο. Ο έλεγχος οικιακών συσκευών λαμβάνει τις εντολές απόφασης on / off στα φώτα, στο σαλόνι, κουζίνα, μπάνιο και διάδρομο.

Ενώ το κόστος ζωής αυξάνεται, υπάρχει ολοένα και μεγαλύτερο ενδιαφέρον για τη συμμετοχή της τεχνολογίας. Έχοντας υπόψη, το πρόγραμμα Smart Home επιτρέπει στο χρήστη να δημιουργήσει και να διατηρήσει ένα σπίτι που είναι αρκετά έξυπνο ώστε να διατηρεί τα ενεργειακά επίπεδα μειωμένα ενώ παρέχει πιο αυτοματοποιημένες εφαρμογές. Ένα έξυπνο σπίτι θα επωφεληθεί από το περιβάλλον του και θα επιτρέψει τον απρόσκοπτο

έλεγχου του εάν ο χρήστης είναι παρών ή μακριά. Με ένα σπίτι που έχει αυτό το πλεονέκτημα, μπορείτε να ξέρετε ότι το σπίτι σας παρουσιάζει τις καλύτερες επιδόσεις. Με την εφαρμογή αυτού του συστήματος, είναι δυνατόν να διερευνηθεί μια ποικιλία διαφορετικών προκλήσεων στον τομέα της μηχανικής, όπως προγραμματισμός λογισμικού, σχεδιασμός PCB, πρωτόκολλα TCP / IP, λογική σχεδίαση διακομιστή Web και άλλες πτυχές (M. Sami Fadali, 2013).

5.2 ΕΠΙΣΚΟΠΗΣΗ ΤΟΥ ΣΥΣΤΗΜΑΤΟΣ.

Ένα μικρό και αποδοτικό έξυπνο οικιακό σύστημα παρουσιάζεται σε αυτή την εργασία. Αυτό το σύστημα έχει δύο βασικές ενότητες: τη μονάδα διασύνδεσης υλικού και τη μονάδα επικοινωνίας λογισμικού. Στην καρδιά αυτού του συστήματος είναι ο μικροελεγκτής Arduino UNO, ο οποίος είναι επίσης ικανός να λειτουργεί ως μικροελεγκτής web και τη διασύνδεση για όλες τις μονάδες υλικού. Όλες οι επικοινωνίες και τα χειριστήρια σε αυτό το σύστημα περνούν από τον μικροελεγκτή. Το έξυπνο οικιακό σύστημα προσφέρει δυνατότητα παρακολούθησης του περιβάλλοντος χρησιμοποιώντας τους αισθητήρες θερμοκρασίας, υγρασίας. Προσφέρει επίσης λειτουργίες μεταγωγής για τον έλεγχο φωτισμού, ανεμιστήρων / κλιματιστικών και άλλων οικιακών συσκευών που είναι συνδεδεμένες στο σύστημα ρελέ. Ένα άλλο χαρακτηριστικό αυτού του συστήματος είναι η ανίχνευση εισβολής που προσφέρει με τον αισθητήρα κίνησης και όλα αυτά μπορούν να ελεγχθούν από την εφαρμογή Android Smartphone ή την εφαρμογή web.

Σήμερα, ο αυτοματισμός στο σπίτι γίνεται απαραίτητος για τη βελτίωση της κατάστασης της ζωής μας. Η ευκολία και η ευκολία χρήσης οικιακών συσκευών είναι αυτό που προσφέρει ο αυτοματισμός στο σπίτι. Ο αυτοματισμός στο σπίτι προσφέρει ένα φουτουριστικό τρόπο ζωής, στο οποίο ένα άτομο αποκτά τον έλεγχο ολόκληρου του σπιτιού του χρησιμοποιώντας ένα έξυπνο τηλέφωνο, από την ενεργοποίηση μιας τηλεόρασης σε μια πόρτα κλειδώματος / ξεκλειδώματος. προσφέρει επίσης αποτελεσματική χρήση της ενέργειας. Αλλά για να αποκτήσετε αυτό το εγκατεστημένο σύστημα θα κοστίσει πολλά χρήματα και αυτός είναι ο κύριος λόγος για τον οποίο ο αυτοματισμός στο σπίτι δεν έχει λάβει μεγάλη ζήτηση, προσθέτοντας σε αυτό επίσης την πολυπλοκότητα της εγκατάστασης και της ρύθμισης. Επομένως, είναι απαραίτητο να είναι οικονομικά αποδοτικό και εύκολο στη διαμόρφωση, αν αυτό παρέχεται στους ανθρώπους τότε θα είναι πρόθυμοι να το αποκτήσουν στα σπίτια, στα γραφεία και στα σχολεία τους.

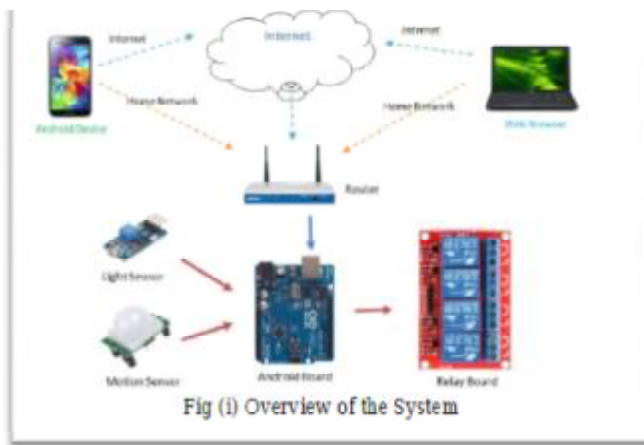


Fig (i) Overview of the System

Με άλλα λόγια, απαιτείται τροποποίηση του συστήματος για τον οικιακό αυτοματισμό, προκειμένου να μειωθεί η τιμή εφαρμογής του σε οικίες. Επίσης, ο αυτοματισμός στο σπίτι προσφέρει ευκολία στο μυαλό και το σώμα σε άτομα με ειδικές ανάγκες ή / και ηλικιωμένους στα σπίτια τους με ένα μόνο κλικ για να κάνουν ό, τι θέλουν όπως αναφέρθηκε παραπάνω. Επισκόπηση του συστήματος φαίνεται στο σχήμα (i)

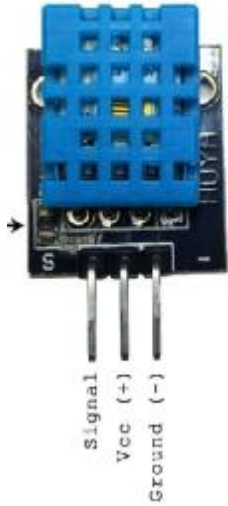
Το προτεινόμενο σύστημά μας είναι ένα αυτοματοποιημένο σπίτι βασισμένο σε arduino, το οποίο γίνεται με Arduino συνδεδεμένο σε ιντερνετ και ελεγχόμενο μέσω app Android ή δικτύου κοινωνικών μέσων. Αυτό το σύστημα ασχολείται με την ασφάλεια στο σπίτι και τις έξυπνες οικιακές τεχνολογίες, οι οποίες θα είναι οικονομικά αποδοτικές.

Το διάγραμμα μπλοκ του προτεινόμενου συστήματος φαίνεται στο Σχ. (Ii) Το Arduino μπορεί να αντιληφθεί το περιβάλλον λαμβάνοντας το σήμα εισόδου από διάφορους αισθητήρες και μπορεί να επηρεάσει το περιβάλλον του μέσω ενεργοποιητών. Ένας αναλογικός αισθητήρας θερμοκρασίας είναι ένα τσιπ που σας λέει ποια είναι η θερμοκρασία περιβάλλοντος.

Το DHT11 είναι ένας βασικός, εξαιρετικά χαμηλού κόστους ψηφιακός αισθητήρας θερμοκρασίας και υγρασίας. Χρησιμοποιεί έναν αισθητήρα υγρασίας και τον αθερμιστήρα για να μετρήσει τον περιβάλλοντα αέρα και εκπέμπει ένα ψηφιακό σήμα (δεν χρειάζονται αναλογικοί ακροδέκτες εισόδου). Είναι αρκετά απλό στη χρήση, αλλά απαιτεί προσεκτικό χρονισμό για να αρπάξει τα δεδομένα. Το μόνο πραγματικό μειονέκτημα αυτού του αισθητήρα είναι ότι μπορείτε να πάρετε μόνο νέα δεδομένα από αυτό μια φορά κάθε 2 δευτερόλεπτα, οπότε όταν χρησιμοποιείτε τη βιβλιοθήκη, οι μετρήσεις των αισθητήρων μπορούν να είναι μέχρι 2 δευτερόλεπτα. Οι αισθητήρες παθητικής υπέρυθρης ακτινοβολίας επιτρέπουν σε κάποιον να αισθανθεί την κίνηση, σχεδόν πάντα και χρησιμοποιείται για να ανιχνεύσει εάν ένας άνθρωπος έχει μετακινηθεί σε απόσταση από την περιοχή των αισθητήρων. Ο αισθητήρας είναι μια πυροηλεκτρική συσκευή που ανιχνεύει την κίνηση, μετρώντας τις αλλαγές στο επίπεδο υπέρυθρων που εκπέμπονται από τα γύρω αντικείμενα. Είναι μικρά, φθηνά, χαμηλής ισχύος, εύχρηστα και δεν φθείρονται. Για το λόγο αυτό βρίσκονται συνήθως σε συσκευές και συσκευές που χρησιμοποιούνται σε σπίτια ή επιχειρήσεις. Ο αισθητήρας μπορεί να λειτουργεί σε θερμοκρασίες από -20 έως 50 ° C και καταναλώνει λιγότερα από 150 mA στα 5 V. Για να επιτρέπεται η σύνδεση των βυσμάτων ηλεκτρικού ρεύματος και η εναλλαγή του ηλεκτρικού φορτίου στο σπίτι, χρησιμοποιούνται διακόπτες ρελέ. Οι διακόπτες ρελέ έχουν τη δυνατότητα να φέρουν μέγιστο φορτίο 10A στα 240V. Αυτό αρκεί για τη μεταφορά οποιασδήποτε οικιακής συσκευής, καθώς αυτές οι συσκευές δεν αντλούν πολύ ρεύμα. Η θωράκιση ethernet παρέχει συνδεσιμότητα στο διαδίκτυο για τον ενσωματωμένο μικροεπεξεργαστή ιστού που επιτρέπει την πρόσβαση στο διαδίκτυο και τα στοιχεία ελέγχου από μια εφαρμογή ιστού.

6.2 Αισθητήρας θερμοκρασίας και υγρασίας (DHT11 HUMIDITY SENSOR)

Λίγα λόγια



Αυτός ο αισθητήρας θερμοκρασίας και υγρασίας DHT11 διαθέτει αισθητήρα θερμοκρασίας και υγρασίας με βαθμονομημένη έξοδο ψηφιακού σήματος. Χρησιμοποιώντας την αποκλειστική λήψη ψηφιακού σήματος με τεχνολογία ανίχνευσης θερμοκρασίας & υγρασίας, εξασφαλίζει υψηλή αξιοπιστία και εξαιρετική μακροπρόθεσμη σταθερότητα. Αυτός ο αισθητήρας περιλαμβάνει μια μέτρηση υγρασίας αντιστατικού τύπου και ένα στοιχείο μέτρησης θερμοκρασίας NTC όπου συνδέεται με έναν μικροελεγκτή υψηλής απόδοσης 8 bit, ο οποίος προσφέρει εξαιρετική ποιότητα, γρήγορη απόκριση, αντι-παρεμβολές, την ικανότητα της σχέση κόστους-αποτελεσματικότητας.

Η σχετική υγρασία είναι η ποσότητα υδρατμών στον αέρα έναντι του σημείου κορεσμού του υδρατμού στον αέρα. Στο σημείο κορεσμού, ο υδρατμός αρχίζει να συμπυκνώνεται και να συσσωρεύεται σε επιφάνειες που σχηματίζουν δροσιά. Το σημείο κορεσμού αλλάζει με τη θερμοκρασία του αέρα. Ο ψυχρός αέρας μπορεί να κρατήσει λιγότερους υδρατμούς πριν γίνει κορεσμένος και ο θερμός αέρας μπορεί να συγκρατήσει περισσότερους υδρατμούς προτού κορεστεί.

Ο τύπος για τον υπολογισμό της σχετικής υγρασίας είναι:

$$RH = \left(\frac{\rho_w}{\rho_s} \right) \times 100\%$$

RH : Relative Humidity

ρ_w : Density of water vapor

ρ_s : Density of water vapor at saturation

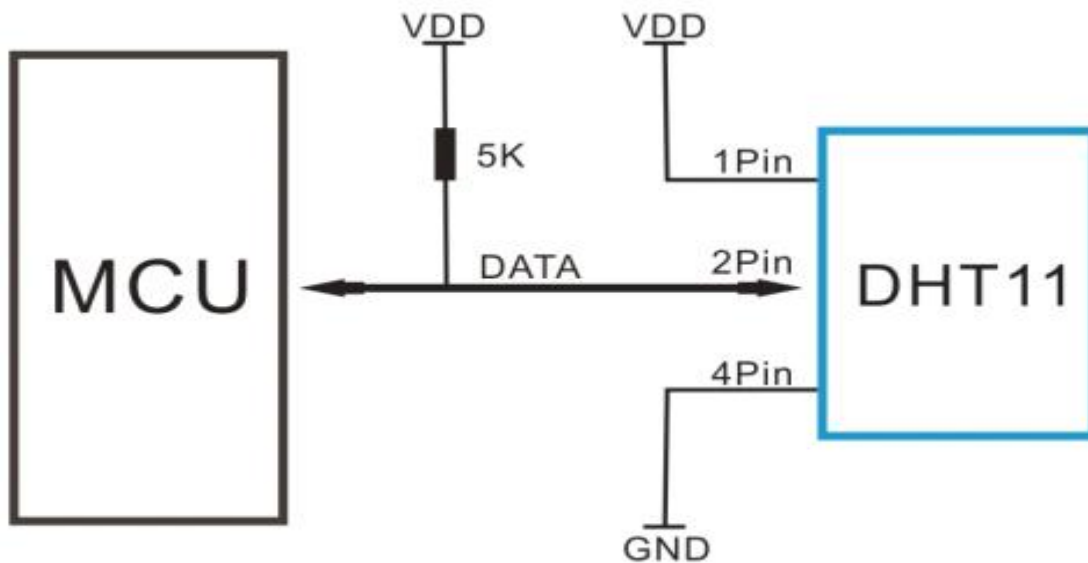
- Relative humidity: εκφράζεται ως ποσοστό.
- Density of water vapor: Πυκνότητα υδρατμών
- Density of water vapor at saturation: Πυκνότητα υδρατμών στον κορεσμό

Είναι ιδανικό για απομακρυσμένους μετεωρολογικούς σταθμούς, συστήματα περιβαλλοντικού ελέγχου, συστήματα παρακολούθησης αγροκτήματος, σπιτιού ή κήπου.

Χαρακτηριστικά

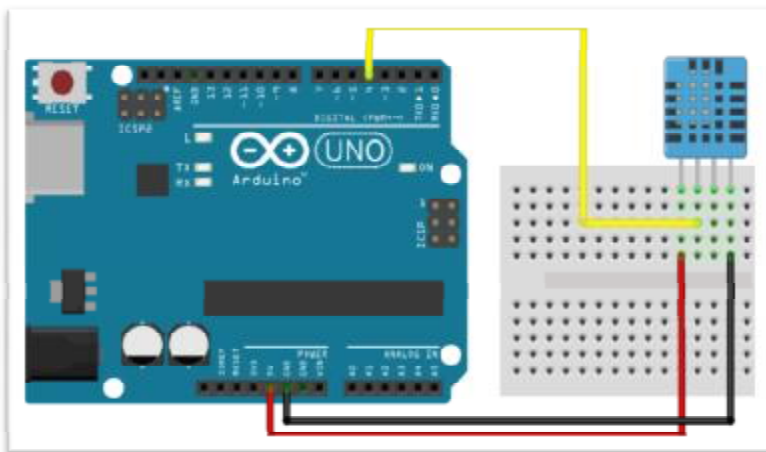
- Εύρος υγρασίας: 20-90% RH
- Ακρίβεια υγρασίας: $\pm 5\%$ RH
- Εύρος θερμοκρασίας: 0-50 ° C
- Θερμοκρασία Ακρίβεια: $\pm 2\%$ ° C
- Τάση λειτουργίας: 3V έως 5.5V

Η Επικοινωνία του αισθητήρα με την Μονάδα



Όταν η μονάδα MCU στέλνει ένα σήμα έναρξης, το DHT11 αλλάζει από τη λειτουργία κατανάλωσης χαμηλής κατανάλωσης σε κατάσταση λειτουργίας, περιμένοντας την ολοκλήρωση του σήματος έναρξης από το MCU. Μόλις ολοκληρωθεί, το DHT11 στέλνει ασήμα απόκρισης των δεδομένων 40-bit που περιλαμβάνουν τις πληροφορίες σχετικής υγρασίας και θερμοκρασίας σε MCU. Οι χρήστες μπορούν να επιλέξουν να συλλέξουν (διαβάσουν) ορισμένα δεδομένα. Χωρίς το σήμα έναρξης από το MCU, DHT11 δεν θα δώσει το σήμα απόκρισης στη MCU. Μόλις συλλεχθούν τα δεδομένα, το DHT11 θα αλλάξει στη λειτουργία χαμηλής κατανάλωσης ενέργειας έως ότου λάβει πάλι ένα σήμα έναρξης από τη μονάδα MCU.

Σύνδεση στον Μικροελεγκτή (Arduino)



6.3 ΑΙΣΘΗΤΗΡΑΣ ΑΠΟΣΤΑΣΗΣ (ULTRASONIC SENSOR HC-SR04)

Λίγα λόγια



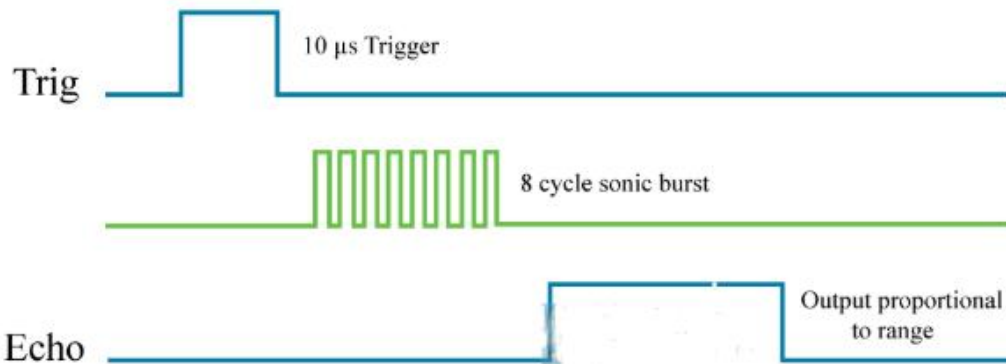
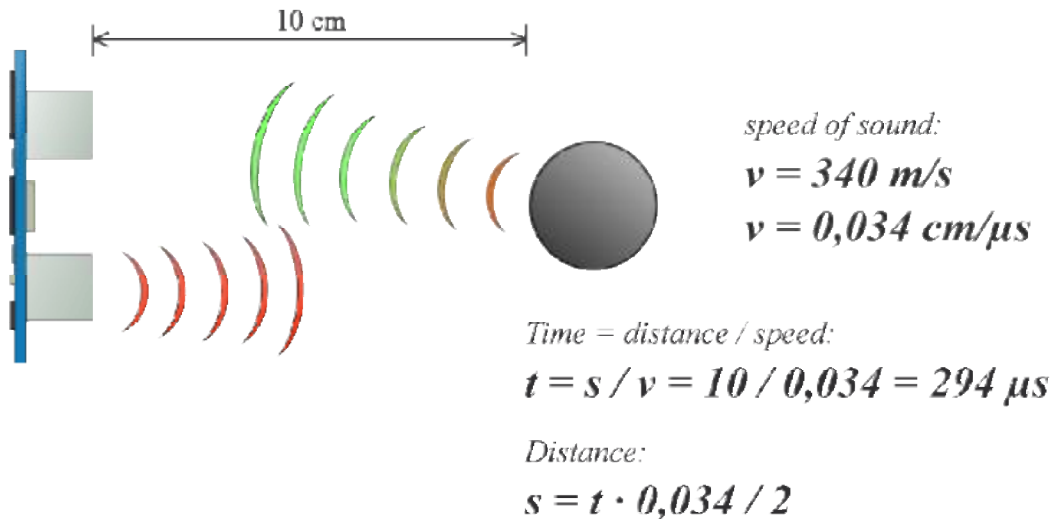
Εκπέμπει ένας υπέρηχο στα 40 000 Hz που ταξιδεύει στον αέρα και αν υπάρχει αντικείμενο ή εμπόδιο στη διαδρομή του, θα αναπηδήσει πίσω στην ενότητα λαμβάνοντας υπόψη τον χρόνο ταξιδιού και την ταχύτητα του ήχου, μπορείτε να υπολογίσετε την απόσταση.

Η υπερηχητική μονάδα HC-SR04 έχει 4 ακίδες, Ground, VCC, Trig και Echo. Οι ακροδέκτες Ground και VCC της μονάδας πρέπει να συνδέονται με τις γείωσης και τους ακροδέκτες των 5 volts στο Arduino Board αντιστοίχως και τους ακροδέκτες trig και echo με οποιαδήποτε ακροδέκτη Digital I / O Board του Arduino.

Λειτουργία

Για να δημιουργήσετε το υπερηχογράφημα, πρέπει να ρυθμίσετε το Trig σε υψηλή κατάσταση για 10 μ s. Αυτό θα στείλει ένα ηχητικό έκρηξη 8 κύκλων το οποίο θα ταξιδέψει στον ήχο της ταχύτητας και θα ληφθεί στον ακροδέκτη Echo. Ο ακροδέκτης Echo θα εξάγει τον χρόνο σε μικροδευτερόλεπτα που διανύθηκε το κύμα ήχου.

Για παράδειγμα, εάν το αντικείμενο βρίσκεται σε απόσταση 10 cm από τον αισθητήρα και η ταχύτητα του ήχου είναι 340 m / s ή 0,034 cm / μ s, το ηχητικό κύμα θα πρέπει να ταξιδέψει περίπου 294 μ δευτερόλεπτα. Αλλά αυτό που θα πάρετε από τον ακροδέκτη Echo θα είναι διπλάσιος του αριθμού επειδή το ηχητικό κύμα χρειάζεται να ταξιδέψει προς τα εμπρός και να αναπηδήσει προς τα πίσω. Επομένως, για να έχουμε την απόσταση σε cm, πρέπει να πολλαπλασιάσουμε την τιμή λαμβανόμενου χρόνου ταξιδιού από τον ακροδέκτη ηχώ κατά 0,034 και να την διαιρέσουμε κατά 2.



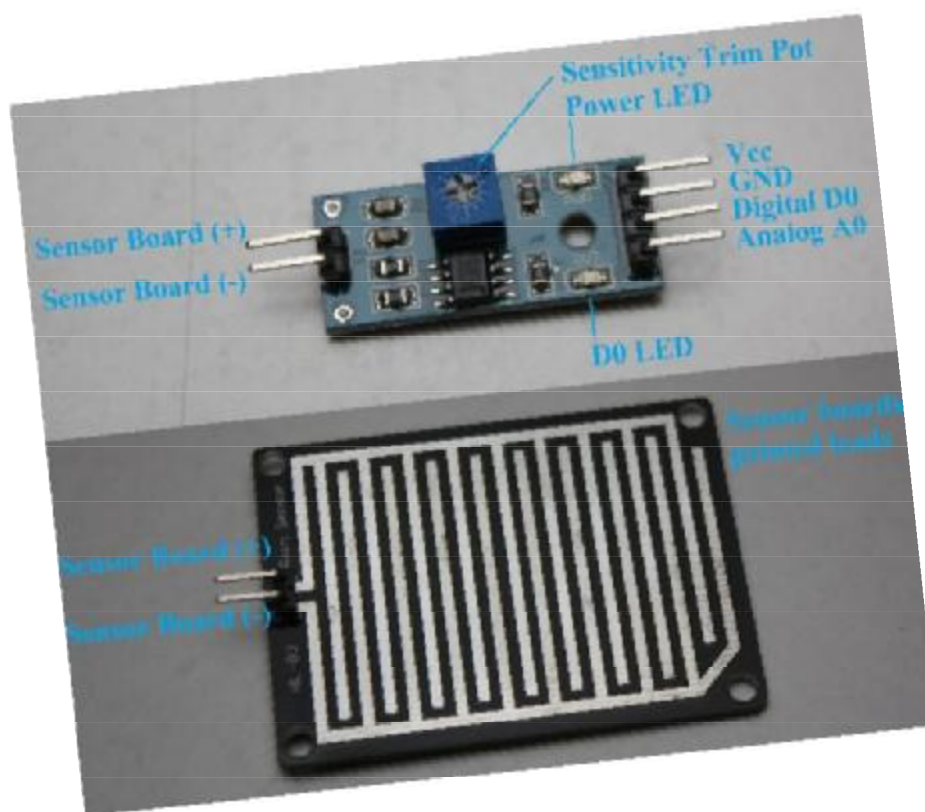
6.4 ΑΙΣΘΗΤΗΡΑΣ ΒΡΟΧΗΣ (RAIN SENSOR MODULE)

Λίγα λόγια για τον αισθητήρα Βροχής

Αυτή η μονάδα σας επιτρέπει να μετράτε την υγρασία μέσω των ακίδων αναλογικής εξόδου και παρέχει ψηφιακή έξοδο όταν ξεπεραστεί ένα κατώφλι υγρασίας. Η μονάδα βασίζεται στον ενισχυτή LM393 op amp.

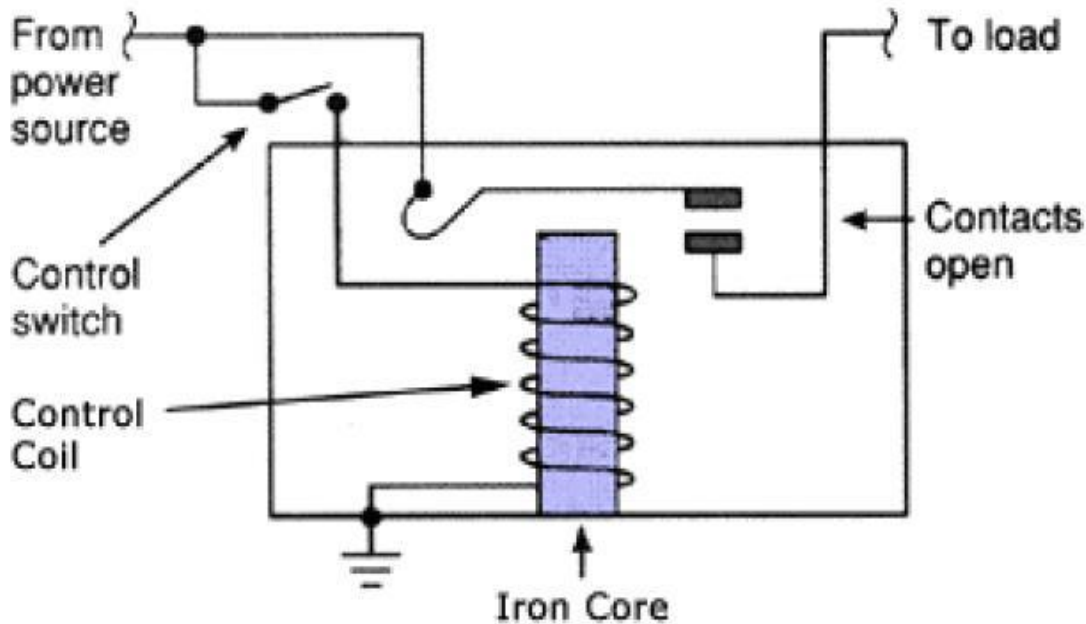
Περιλαμβάνει την ηλεκτρονική μονάδα και έναν πίνακα τυπωμένου κυκλώματος που "συλλέγει" τις σταγόνες βροχής. Καθώς συλλέγονται οι σταγόνες βροχής στην πλακέτα κυκλωμάτων, δημιουργούνται μονοπάτια παράλληλης αντίστασης που μετριοούνται μέσω του ενισχυτή op amp.

Όσο χαμηλότερη είναι η αντίσταση (ή το περισσότερο νερό), τόσο χαμηλότερη είναι η έξοδος τάσης. Αντίθετα, όσο λιγότερα είναι τα νερά, τόσο μεγαλύτερη είναι η τάση εξόδου στον αναλογικό πόλο. Μια εντελώς στεγνή σανίδα, για παράδειγμα, θα προκαλέσει έξοδο 5 βολτ



Pin, Control, or Indicator	Description
VCC	+5 Volts Power Source
GND	Ground or negative power source
DO	Digital Output. Goes low when moisture exceeds set threshold.
A0	Analog Output - Zero to five volts. The lower the voltage, the greater the moisture.
Power LED	Indicates that power is applied
Output LED	Illuminates when moisture has exceeded threshold set by sensitivity adjustment.
Sensitivity Adjustment	Clockwise is more sensitive. Counterclockwise is less sensitive.

Σύνδεση στον Arduino



Περιγραφή:

Υπάρχουν 5 μέρη σε κάθε ρελέ:

1. Ηλεκτρομαγνήτης - Αποτελείται από ένα πυρήνα σιδήρου περιτυλιγμένο από πηνίο καλωδίων. Όταν περάσει η ηλεκτρική ενέργεια, γίνεται μαγνητική. Ως εκ τούτου, ονομάζεται ηλεκτρομαγνήτης.
2. Βραχίονας - Η κινητή μαγνητική ταινία είναι γνωστή ως σπλισμός. Όταν ρέει ρεύμα διαμέσου αυτών, ενεργοποιείται το πηνίο παράγοντας έτσι ένα μαγνητικό πεδίο. Ο σπλισμός μπορεί να μετακινηθεί με συνεχές ρεύμα (DC) και εναλλασσόμενο ρεύμα (AC).
3. Άνοιγμα - Όταν δεν ρέουν ρεύματα μέσω του πηνίου στον ηλεκτρομαγνήτη, το ελατήριο τραβάει το σπλισμό μακριά έτσι ώστε το κύκλωμα να μην μπορεί να ολοκληρωθεί.
4. Σειτ ηλεκτρικών επαφών - Υπάρχουν δύο σημεία επαφής:
 - 4.1. Normally open- συνδεδεμένο όταν είναι ενεργοποιημένο το ρελέ και αποσυνδεθεί όταν είναι ανενεργό.
 - 4.2. Normally close- δεν είναι συνδεδεμένο όταν ο ηλεκτρονόμος είναι ενεργοποιημένος και είναι συνδεδεμένος όταν είναι ανενεργός.
5. Μορφοποιημένο πλαίσιο - Τα ρελέ καλύπτονται με πλαστικό για προστασία.

Αναλυτική Λειτουργία:

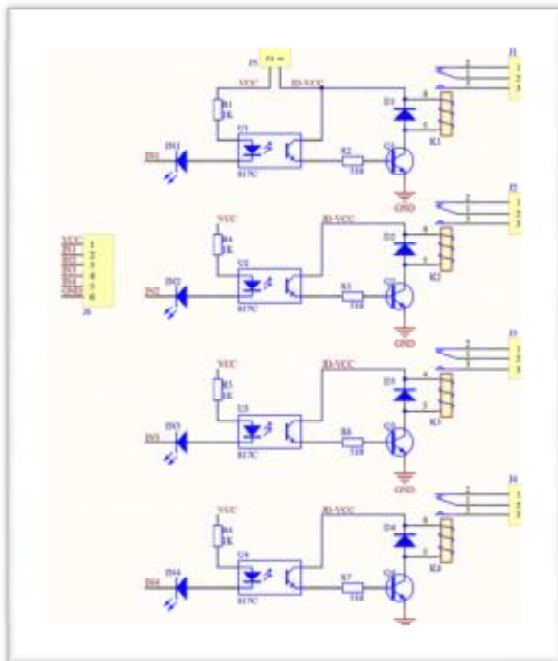
Το διάγραμμα δείχνει ένα διάγραμμα εσωτερικού τμήματος ενός ρελέ. Ένας πυρήνας σιδήρου περιβάλλεται από ένα πηνίο ελέγχου. Όπως φαίνεται, η πηγή ισχύος δίνεται στον ηλεκτρομαγνήτη μέσω ενός διακόπτη ελέγχου και μέσω επαφών στο φορτίο. Όταν αρχίσει να ρέει ρεύμα μέσω του πηνίου ελέγχου, ο ηλεκτρομαγνήτης αρχίζει να ενεργοποιεί και επομένως εντείνει το μαγνητικό πεδίο. Έτσι ο ανώτερος βραχίονας επαφής αρχίζει να έλκεται από τον κάτω σταθερό βραχίονα και έτσι κλείνει τις επαφές προκαλώντας βραχυκύκλωμα για την ισχύ στο φορτίο. Από την άλλη πλευρά, εάν ο ηλεκτρονόμος είχε ήδη απενεργοποιηθεί όταν οι επαφές ήταν κλειστές, τότε η επαφή κινείται αντίθετα και κάνει ανοιχτό κύκλωμα.

Μόλις το ρεύμα του πηνίου είναι απενεργοποιημένο, ο κινητός σπλισμός θα επιστρέψει με μια δύναμη πίσω στην αρχική του θέση. Αυτή η δύναμη θα είναι σχεδόν ίση με τη μισή δύναμη της μαγνητικής δύναμης. Αυτή η δύναμη παρέχεται κυρίως από δύο παράγοντες. Είναι η πηγή και η βαρύτητα.

Τα ρελέ κατασκευάζονται κυρίως για δύο βασικές λειτουργίες. Το ένα είναι χαμηλής τάσης και το άλλο είναι υψηλής τάσης. Για εφαρμογές χαμηλής τάσης, θα δοθεί μεγαλύτερη προτίμηση για τη μείωση του θορύβου ολόκληρου του κυκλώματος. Για εφαρμογές υψηλής τάσης, έχουν σχεδιαστεί κυρίως για τη μείωση ενός φαινομένου που ονομάζεται τόξο.

Χαρακτηριστικά:

- Ρελέ υψηλού ρεύματος, AC250V 10A, DC30V 10A
- 4 LED που υποδεικνύουν πότε είναι ενεργοποιημένα τα ρελέ
- Λειτουργεί με σήματα στάθμης λογικής από συσκευές 3.3V ή 5V
- Ο ρελέ Songle υψηλής ποιότητας χρησιμοποιείται με διπλό διπλό βύθισμα, κοινό τερματικό, κανονικά ανοικτό τερματικό και κανονικά κλειστό τερματικό
- Μέγεθος: 75mm (μήκος) * 55mm (πλάτος) * 19.3mm (Υψος)
- Υπάρχουν τέσσερις οπές βιδών σε κάθε γωνία της πλακέτας, εύκολη για εγκατάσταση και σταθεροποίηση. Η διάμετρος της οπής είναι 3,1mm



6.6 ΜΑΓΝΗΤΙΚΟΣ ΔΙΑΚΟΠΤΗΣ ΕΠΑΦΗΣ (DOOR SENSOR)

Αυτός ο αισθητήρας είναι ουσιαστικά ένας διακόπτης, εγκλωβισμένος σε πλαστικό κέλυφος ABS. Το άλλο μισό είναι ένας μαγνήτης. Όταν ο μαγνήτης είναι λιγότερο από 13 χιλιοστά μακριά, ο διακόπτης κλείνει. Συχνά χρησιμοποιείται για να ανιχνεύσει πότε μια πόρτα ή ένα συρτάρι είναι ανοιχτό, γι' αυτό και έχουν γλωσσίδια και βίδες στήριξης. - μια αφρώδη ταινία από ένα κατάστημα υλικού για την τοποθέτηση αυτών, που λειτουργεί καλά χωρίς να χρειάζεται βίδες.

ΤΕΧΝΙΚΕΣ ΛΕΠΤΟΜΕΡΕΙΕΣ

Περιβλήματος ABS

Ρεύμα: 100 mA max

Τάση: 200 VDC max

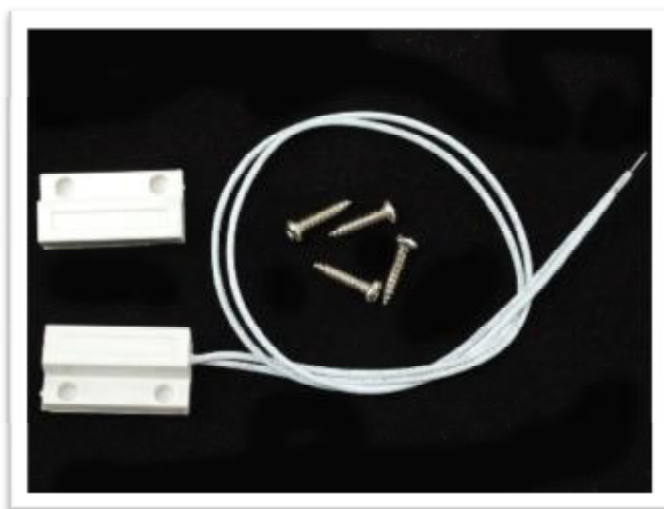
Απόσταση διακόπτη - μαγνήτη : 15mm max

Διαστάσεις:

Μέγεθος πλαισίου (κάθε πλευρά): 29mm x 14mm x 9mm / 1.1 "x 0.6" x 0.4 "

Μήκος καλωδίου: 305mm ± 12mm / 12 " ± 0.5"

Βάρος (ανά πλευρά): 5.4g



6.8 ΜΟΝΑΔΑ ETHERNET (ENC28J60)

Εισαγωγή:

Η μονάδα Ethernet ENC28J60 χρησιμοποιεί τον μικροεπεξεργαστή IC Microchip ENC28J60 Stand-Alone Ethernet Controller, ο οποίος διαθέτει πλήθος λειτουργιών για την αντιμετώπιση των περισσότερων απαιτήσεων πρωτοκόλλου δικτύου. Ο πίνακας συνδέεται απευθείας με τους περισσότερους μικροελεγκτές με μια τυπική διασύνδεση SPI με ταχύτητα μεταφοράς μέχρι 20MHz.

Αυτή η μονάδα διασύνδεσης Ethernet είναι ο απλούστερος τρόπος για να προσθέσετε συνδεσιμότητα LAN στα προϊόντα και τα έργα που βασίζονται σε μικροελεγκτές.

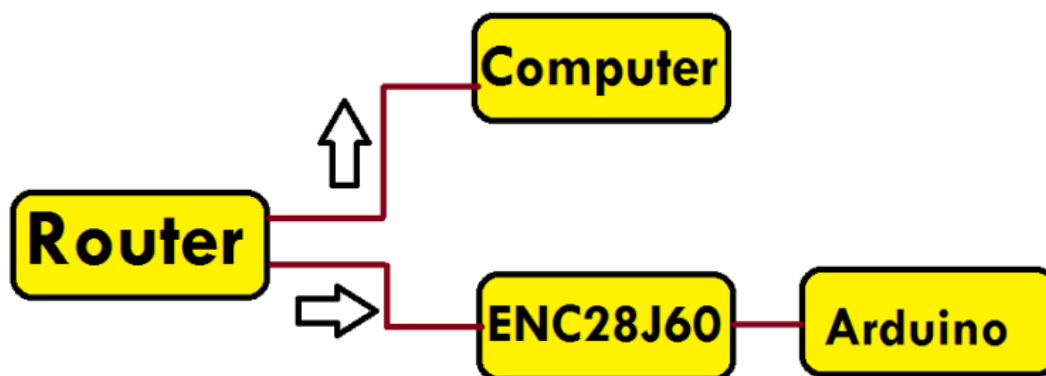
Χαρακτηριστικά:

1. Μονάδα Ethernet LAN για Arduino / AVR / LPC / STM3
2. ENC28J60 τσιπ Ethernet
3. Μπορεί να τοποθετηθεί εύκολα με τη μονάδα MCU
4. Διασύνδεση δικτύου: HR911105A
5. Τάση τροφοδοσίας: 3.3 V (DIO Tolerant DIO)
6. Ταλαντωτής κρυστάλλου 25Mhz
7. Μέγεθος (Π x Υ x Β): Περίπου. 2,3 x 1,3 x 0,7 ιντσών / 58 x 34 x 17 mm



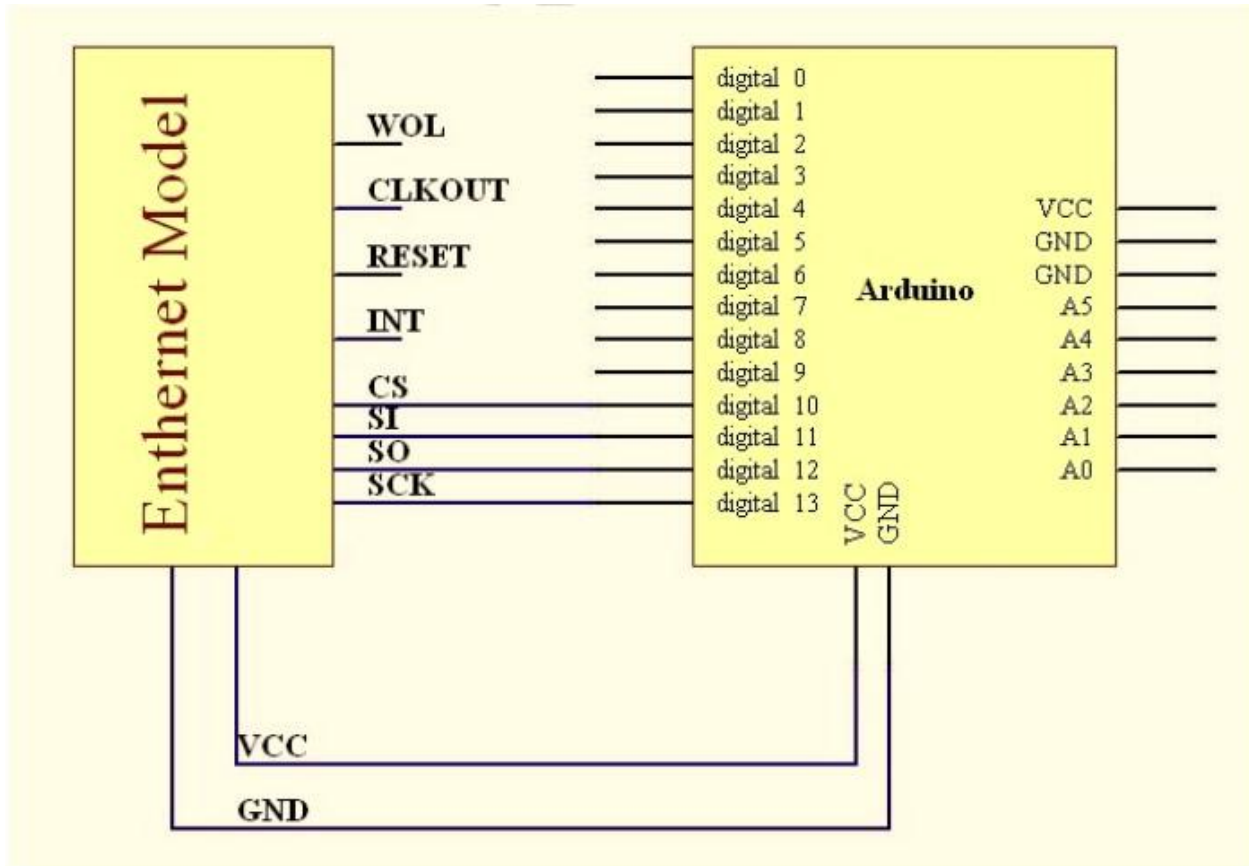
Διάγραμμα μπλοκ & Συνδέσεις:

Αυτό είναι το απλό δομικό διάγραμμα που εξηγεί πώς μια μονάδα Arduino & ENC28J60 μπορεί να συνδεθεί με τον δρομολογητή και τον υπολογιστή.



Ο δρομολογητής θα πρέπει να συνδεθεί με τοπικό δίκτυο και να έχει πολλαπλή θύρα εξόδου εισόδου. Από το δρομολογητή συνδέεται ένα καλώδιο ethernet στο Σύστημα Υπολογιστών και άλλο καλώδιο Ethernet θα πρέπει να συνδεθεί στο Module ENC28J60. Ομοίως, η ενότητα ENC28J60 συνδέεται με το Arduino. Η σύνδεση μεταξύ Arduino & ENC28J60 παρέχεται παρακάτω.

Αναλυτική Συνδεσμολογία Arduino με Ethernet Model



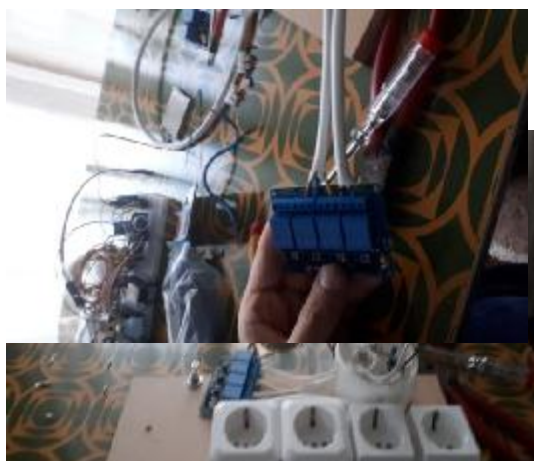
7. ΚΑΤΑΣΚΕΥΑΣΤΙΚΟ ΜΕΡΟΣ

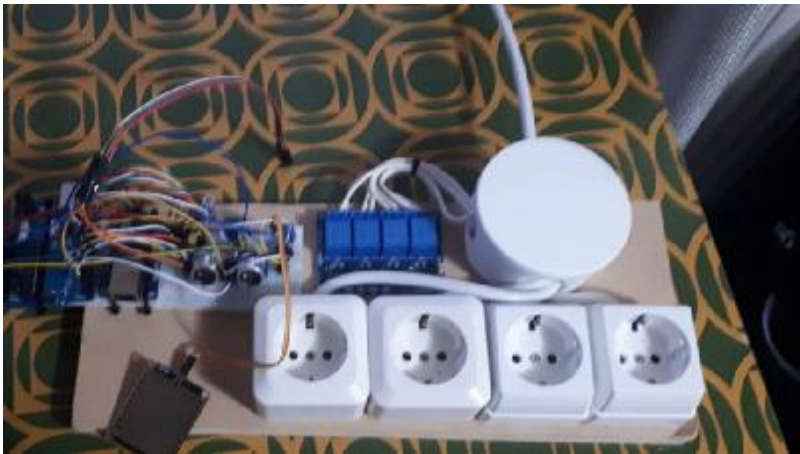
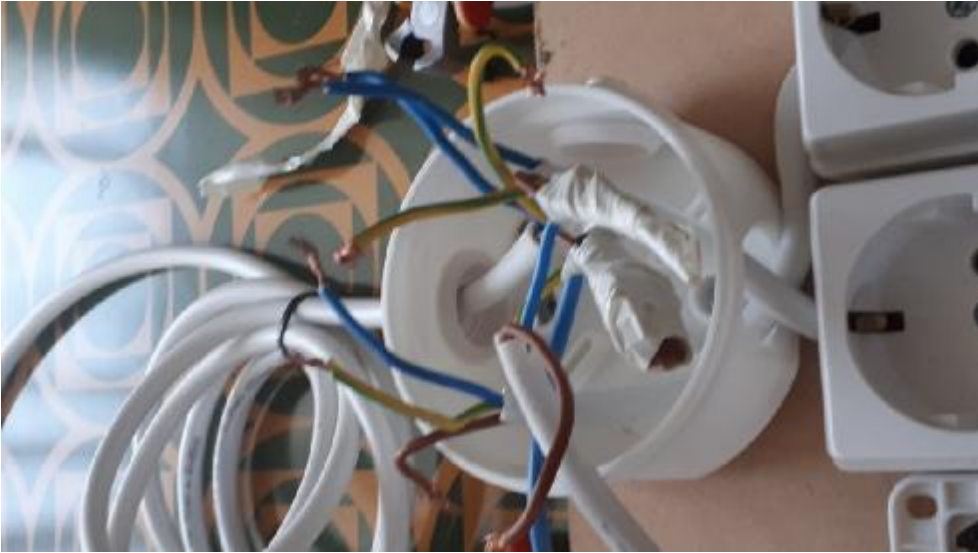
Η υλοποίηση έγινε σε πραγματικό σπίτι. Σε αυτήν την ενότητα θα δείξουμε την συναρμολόγηση ενός σπιτιού το οποίο θα φιλοξενήσει τόσο τον μικροελεγκτή μας (Arduino Uno) όσο και τις ηλεκτρονικές μας διατάξεις (αισθητήρες) που αναφέραμε στην προηγούμενη ενότητα.

7.1 Η ΚΑΤΣΚΕΥΗ ΤΟΥ ΣΠΙΤΙΟΥ



7.2 ΚΑΤΑΣΚΕΥΗ ΗΛΕΚΤΡΟΛΟΓΙΚΟΥ ΥΛΙΚΟΥ





7.3 ΤΟΠΟΘΕΤΗΣΗ – ΣΥΝΔΕΣΗ ΚΥΚΛΩΜΑΤΩΝ ΣΤΟ ΣΠΙΤΙ





8 ΚΕΦΑΛΑΙΟ

8.1 ΚΩΔΙΚΑΣ ΕΦΑΡΜΟΓΗΣ

```
#include "pitches.h"
//libraries
//#include <NewPing.h>
#include "etherShield.h"
#include "ETHER_28J60.h"
#include <OneWire.h>
#include "dht.h"
//#define ONE_WIRE_BUS 2
#define DHT11_PIN 4
//OneWire oneWire(ONE_WIRE_BUS);
dht DHT;
#include "Ultrasonic.h"
Ultrasonic ultrasonic(A3,A2);
int distance;
const int sensorMin = 0; // sensor minimum
const int sensorMax = 1024; // sensor maximum

//const int TriggerPin = 3;
//const int EchoPin = 2;
//NewPing sonar(TriggerPin, EchoPin, 100);

static uint8_t mac[6] = {0x54, 0x55, 0x58, 0x10, 0x01, 0x24}; // this just needs to be
unique for your network,
static uint8_t ip[4] = {192, 168, 1, 15}; // the IP address for your board.
Check your home hub
static uint16_t port = 80; // Use port 80 - the standard for
HTTP

ETHER_28J60 ethernet;
double tempin;
double tempout;
boolean light;
boolean light2;
boolean light3;
boolean light4;
//boolean PIR;
//int alarm;
//boolean check;
int sensorReading;
int range ;
int val;
int tempPin =1;
float cel;
int state;
const int windows = 7;
```

```

void setup()
{

  ethernet.setup(mac, ip, port);

  // define pins
  pinMode(A0, INPUT);//rain
  pinMode(5, OUTPUT);//led1
  pinMode(6, OUTPUT);//led2
  pinMode(8, OUTPUT);//led3
  pinMode(9, OUTPUT);//led4

  pinMode(3, INPUT);//TriggerPin
  pinMode(DHT11_PIN, INPUT);
  pinMode(7, INPUT);
  pinMode(windows, INPUT_PULLUP);

}

void loop()
{
  distance = ultrasonic.Ranging(CM); //ultrasonic
  state = digitalRead(windows);//windows

  //int cm = sonar.ping_cm();
  val = analogRead(tempPin);//lm35 A1
  float mv = ( val/1024.0)*5000;
  float cel = mv/15;
  float farh = (cel*9)/(5 + 32);

  int chk = DHT.read11(DHT11_PIN);//dht11
  char* params;

  while (params = ethernet.serviceRequest()) { //when you type in browser
http://192.168.1.15
    if (strcmp(params, "") == 0)
    {
      tempin = DHT.temperature;// sensors.requestTemperatures();
      tempout = DHT.humidity;// sensors.requestTemperatures();

      //ethernet.print("Instruction how make your Control home: <a
href='https://www.facebook.com/MicroLab.Aigialeia/'>link</a><width='200'
height='200'>");
      // digitalWrite(5, !light);
      // digitalWrite(6, !light2);
      // digitalWrite(8, !light3);
      // digitalWrite(9, !light4);

```

```

ethernet.print("<center>");
ethernet.print("<font color='teal'>");
ethernet.print("<h1>Control Home</h1>");
ethernet.print("</font>");
ethernet.print("<br>");

ethernet.print("<h2>Temp in: ");
ethernet.print(tempin);
ethernet.print(" C");
ethernet.print("</h2>");

ethernet.print("<h2>Humidity in: ");
ethernet.print(tempout);
ethernet.print(" % ");
ethernet.print("</h2>");

ethernet.print("<h2>Temp out: ");
ethernet.print(CEL);

ethernet.print(" C");
ethernet.print("</h2>");

//check = true;
ethernet.print("<h2>Alarm: ");
if ((distance > 1) && (distance < 20))
{
    ethernet.print("Movenment in: ");
    ethernet.print(distance);
    ethernet.print(" cm");
}
else
{
    ethernet.print("No movenment");
}
ethernet.print("</h2>");

ethernet.print("<h2>Rain: ");
sensorReading = analogRead(A0);// map the sensor range (three options):
range = map(sensorReading, sensorMin, sensorMax, 0, 3);// range value:
switch (range){
case 0: // Sensor getting wet
Serial.println("Flood");
ethernet.print("Flood");
break;
case 1: // Sensor getting wet
// Serial.println("Rain Warning");
ethernet.print("Rain Warning");
break;
case 2: // Sensor dry - To shut this up delete the " Serial.println("Not Raining"); "

```


below.

```
// Serial.println("Not Raining");  
ethernet.print("Not Raining");  
break; }  
delay(1); // delay between reads
```

```
//windows
```

```
ethernet.print("<h2>Window: ");  
if (state == HIGH)  
{  
  ethernet.print("On");  
  tone(3, NOTE_F5);  
}  
else  
{  
  ethernet.print("Off");  
  noTone(3);  
}  
ethernet.print("</h2>");  
ethernet.print("Lamp: ");  
ethernet.print("<a href='light'><button>Lamp</button></a>");
```

```
ethernet.print("<br>");  
ethernet.print("<br>");
```

```
ethernet.print("<meta http-equiv=\"refresh\" content=\"0\">");
```

```
}
```

```
// TEMP IN *****
```

```
if (strcmp(params, "tempin") == 0){  
  ethernet.print(tempin);          // print temperature in on ethernet  
}
```

```
// Humidity *****
```

```
if (strcmp(params, "Humidity in") == 0){  
  ethernet.print(tempout);        // print temperature out on ethernet  
}
```

```
// TEMP OUT *****
```

```
if (strcmp(params, "Temp out") == 0){  
  ethernet.print(CEL);            // print temperature out on ethernet  
}
```

```

// ALARM *****
if (strcmp(params, "cm") == 0){

    if ((distance > 1) && (distance < 20))
    {

        ethernet.print("Movenment");
        }
    else
    {
        ethernet.print("No movenment");
    }
    //ethernet.print(cm);

}

// RAIN *****
switch (range)
// switch (strcmp(params, "range") == 0)
{
case 0: // Sensor getting wet
Serial.println("Flood");
break;
case 1: // Sensor getting wet
Serial.println("Rain Warning");
break;
case 2: // Sensor dry - To shut this up delete the " Serial.println("Not Raining"); "
below.
Serial.println("Not Raining");
break;
}

// WINDOW *****
if (strcmp(params, "window") == 0){
if(state==HIGH){
    ethernet.print("close");
}
else{
    ethernet.print("open");
}
}

//LIGHT INTERNET*****
if(strcmp(params, "light") == 0)
{
    //light1
    ethernet.print("BedRoom Light 1 ");
}

```

```

ethernet.print("<a href='BedRoom_Light_On1'><button>on</button></a>");
ethernet.print("<a href='BedRoom_Light_off1'><button>off</button></a>");
//light2
ethernet.print("<br>");ethernet.print("<br>");
ethernet.print("BedRoom Light 2 ");
ethernet.print("<a href='BedRoom_Light_On2'><button>on</button></a>");
ethernet.print("<a href='BedRoom_Light_off2'><button>off</button></a>");
//light3
ethernet.print("<br>");ethernet.print("<br>");
ethernet.print("LivingRoom Light ");
ethernet.print("<a href='LivingRoom_Light_On'><button>on</button></a>");
ethernet.print("<a href='LivingRoom_Light_Off'><button>off</button></a>");

//light4
ethernet.print("<br>");ethernet.print("<br>");
ethernet.print("Entrance Light ");
ethernet.print("<a href='Entrance_On'><button>on</button></a>");
ethernet.print("<a href='Entrance_Off'><button>off</button></a>");

}
//light1

if(strcmp(params, "BedRoom_Light_On1") == 0)
{ ethernet.print("Change:Bedroom Light ON ");
digitalWrite(6, light2);}

if(strcmp(params, "BedRoom_Light_off1") == 0)
{ ethernet.print("Change:Bedroom Light OFF ");
digitalWrite(6, !light2);}

//light2
if(strcmp(params, "BedRoom_Light_On2") == 0)
{ ethernet.print("Change:Bedroom Light ON ");
digitalWrite(8, light);}

if(strcmp(params, "BedRoom_Light_off2") == 0)
{ ethernet.print("Change:Bedroom Light OFF ");
digitalWrite(8, !light);}

//light3

if(strcmp(params, "LivingRoom_Light_On") == 0)
{ ethernet.print("Change:LivingRoom Light ON ");
digitalWrite(5, light4);}

if(strcmp(params, "LivingRoom_Light_Off") == 0)
{ ethernet.print("Change:LivingRoom Light OFF ");
digitalWrite(5, !light4);}

//light4

```

```

    if(strcmp(params, "Entrance_On") == 0)
    { ethernet.print("Change:Entrance ON ");
      digitalWrite(9, light3);}

```

```

    if(strcmp(params, "Entrance_Off") == 0)
    { ethernet.print("Change:Entrance OFF ");
      digitalWrite(9, !light3);}

```

```

// ETHERNET RESPOND *****
ethernet.respond();
delay(20);
}

}

```

8.2 BIBAIOOHKH PITCHES.H

```

/*****
* Public Constants
*****/

#define NOTE_B0 31
#define NOTE_C1 33
#define NOTE_CS1 35
#define NOTE_D1 37
#define NOTE_DS1 39
#define NOTE_E1 41
#define NOTE_F1 44
#define NOTE_FS1 46
#define NOTE_G1 49
#define NOTE_GS1 52
#define NOTE_A1 55
#define NOTE_AS1 58
#define NOTE_B1 62
#define NOTE_C2 65
#define NOTE_CS2 69
#define NOTE_D2 73
#define NOTE_DS2 78
#define NOTE_E2 82
#define NOTE_F2 87
#define NOTE_FS2 93
#define NOTE_G2 98
#define NOTE_GS2 104
#define NOTE_A2 110
#define NOTE_AS2 117
#define NOTE_B2 123
#define NOTE_C3 131

```

```
#define NOTE_CS3 139
#define NOTE_D3 147
#define NOTE_DS3 156
#define NOTE_E3 165
#define NOTE_F3 175
#define NOTE_FS3 185
#define NOTE_G3 196
#define NOTE_GS3 208
#define NOTE_A3 220
#define NOTE_AS3 233
#define NOTE_B3 247
#define NOTE_C4 262
#define NOTE_CS4 277
#define NOTE_D4 294
#define NOTE_DS4 311
#define NOTE_E4 330
#define NOTE_F4 349
#define NOTE_FS4 370
#define NOTE_G4 392
#define NOTE_GS4 415
#define NOTE_A4 440
#define NOTE_AS4 466
#define NOTE_B4 494
#define NOTE_C5 523
#define NOTE_CS5 554
#define NOTE_D5 587
#define NOTE_DS5 622
#define NOTE_E5 659
#define NOTE_F5 698
#define NOTE_FS5 740
#define NOTE_G5 784
#define NOTE_GS5 831
#define NOTE_A5 880
#define NOTE_AS5 932
#define NOTE_B5 988
#define NOTE_C6 1047
#define NOTE_CS6 1109
#define NOTE_D6 1175
#define NOTE_DS6 1245
#define NOTE_E6 1319
#define NOTE_F6 1397
#define NOTE_FS6 1480
#define NOTE_G6 1568
#define NOTE_GS6 1661
#define NOTE_A6 1760
#define NOTE_AS6 1865
#define NOTE_B6 1976
#define NOTE_CS7 2217
```

ΣΥΜΠΕΡΑΣΜΑΤΑ

Σε αυτήν την εργασία ολοκληρώθηκε ένα σύστημα παρακολούθησης σπιτιού αποτελεσματικά χαμηλού κόστους και φιλικό προς το χρήστη. Όλο το σπίτι παραμένει κάτω από τον έλεγχο του χρήστη όλη την ώρα. Η προτεινόμενη αρχιτεκτονική χρησιμοποιεί έναν απλό διακομιστή Web για την επικοινωνία μεταξύ του απομακρυσμένου χρήστη και των οικιακών συσκευών μέσω ηλεκτρονικού υπολογιστή και Smartphone. Στο μέλλον θα υπάρξουν κάποιες συσκευές που θα είναι πιο αξιόπιστες, γρηγορότερες και φθηνότερες. Ωστόσο, προσπαθήσαμε να κάνουμε έναν καλό έλεγχο και ένα σύστημα ασφαλείας.

Τα εξαρτήματα που χρησιμοποιήσαμε μπορούν να αλλάξουν και να χρησιμοποιηθούν και με μία νεότερη πλακέτα (μικροελεγκτή), αλλά θα πρέπει να έχουν το σωστό λογισμικό και το σωστό πρόγραμμα οδήγησης. Όπως αναφέρθηκε προηγουμένως, αυτή η εργασία δεν είναι ένα πλήρες έργο. Πρόκειται μόνο για μια βασική δομή ενός ολοκληρωμένου συστήματος. Έχουμε κάνει όλες τις βασικές ανάγκες ενός τυπικού σπιτιού. Υπάρχουν πολλά άλλα πεδία για αυτό το έργο. Περισσότερες συσκευές μπορούν να προστεθούν σε αυτό το σύστημα με μια πιο ισχυρή μονάδα Arduino Mega. Όλες οι διαθέσιμες έξυπνες συσκευές μπορούν να συνδεθούν με αυτό το σύστημα.

Επιπλέον, αυτό το έργο θα μπορούσε να έχει μια βάση δεδομένων cloud για να αποθηκεύσει όλα τα δεδομένα. Μπορούν να αποθηκευτούν αναγνώσεις από όλους τους αισθητήρες με ημερομηνία και ώρα. Επίσης, θα μπορούσαμε να προσθέσουμε μια κάμερα παρακολούθησης έξω από το σπίτι για επιπλέον ασφάλεια.

Ωστόσο όλα τα καθήκοντα αυτού του έργου γίνονται με επιτυχία. Ήμασταν σε θέση να εκπληρώσουμε τους στόχους μας όπως προτείνονται σε αυτό το σύστημα. Σχεδόν όλες οι επιστημονικές και τελευταίες τεχνολογίες έχουν τόσο καλές όσο και κακές πλευρές. Αυτό δεν σημαίνει ότι πρέπει να αποφύγουμε την τεχνολογία. Αυτό το είδος εργασίας μας εμπνέει να κάνουμε κάτι καλύτερο για τη χώρα μας. Η έξυπνη τεχνολογία είναι μια ευλογία για τη χώρα μας.

ΒΙΒΛΙΟΓΡΑΦΙΑ

- Charith Perera, A. Z. (2008). Context Aware Computing for The Internet of Things: A Survey. *IEEE COMMUNICATIONS SURVEYS & TUTORIAL*.
- Deepali , J., Mohd. , M., & Shreerang , N. (2013, March). HomeAutomation and Security System Using Android ADK. *International Journal of Electronics Communication and Computer Technology(IJECCT)*, 3(2).
- M. Sami Fadali, A. V. (2013). *Digital Control Engineering, Analysis and Design*. 225 Wyman Street, Waltham, MA 02451, USA: Academic Press is an imprint of Elsevier.
- Muhammad Raisul Alam, M. M. (2012). A review of smart homes - Past, present, and future. *IEEE Transactions on Systems, Man and Cybernetics Part C: Applications and Reviews*, 4(6), 14.
- Sirsath , N. S., Dhole , P. S., Mohire , N. P., Naik , S. C., & Ratnaparkhi, N. S. (2018). *Home Automation using Cloud Network and Mobile Devices*. (D. o. Engineering, Επιμ.) India University of Pune: Vidyanagari, Parvati, Pune-411009, India University of Pune,.
- Παρασκευόπουλος, Π. (2001). Θεωρία. Στο Π. Παρασκευόπουλος, *Έλεγχος Συστημάτων με Υπολογιστές* (Τόμ. Τόμος Α: Θεωρία, σσ. 220-256). Αθήνα: Παρασκευόπουλος, Π.Ν.