

ΤΕΧΝΟΛΟΓΙΚΟ ΕΚΠΑΙΔΕΥΤΙΚΟ ΙΔΡΥΜΑ ΚΡΗΤΗΣ



Σχολή Εφαρμοσμένων Επιστημών
Τμήμα Μηχανικών Φυσικών Πόρων και Περιβάλλοντος Τ.Ε.

Πτυχιακή Εργασία

*“Έξυπνο Σύστημα Μέτρησης και Διαχείρισης
Ηλεκτρικής Ενέργειας σε Πραγματικό Χρόνο”*



Δημήτριος Μακρής

Επιβλέπων Καθηγητής: Δρ. Εμμανουήλ Σκουνάκης

Φεβρουάριος 2016

Στους γονείς μου, Γιώργο & Ελένη
Στην αδελφή μου Ματίνα

Abstract

This project is developed in order to make voltage and current measurements of a circuit and then calculate the energy consumption, the energy cost and the rate of CO₂ emissions into the atmosphere. At the same time, the remote access and control of consumption of the above data is feasible. The operation of the system is achieved by programming an Arduino Ethernet microcontroller connected with data acquisition sensors. The remote control is achieved via mobile devices which are connected to the internet. The ability to monitor the energy consumption in real time is designed to easily inform the users and achieve their awareness regarding the release of gases that contribute to global warming through their energy consumption as well as changing the consumers' attitudes of energy usage in the near future.

Περίληψη

Η παρούσα πτυχιακή εργασία έχει ως θέμα την υλοποίηση ενός συστήματος μέτρησης της τάσης και της έντασης του ρεύματος ενός κυκλώματος και στη συνέχεια υπολογισμού της ενεργειακής του κατανάλωσης καθώς και του ρυθμού εκπομπών CO₂ στην ατμόσφαιρα. Παράλληλα, προσφέρεται η δυνατότητα της απομακρυσμένης πρόσβασης και του ελέγχου της κατανάλωσης στα παραπάνω δεδομένα.

Η λειτουργία του συστήματος επιτυγχάνεται μέσω του προγραμματισμού ενός μικροελεγκτή Arduino Ethernet και της σύνδεσής του με αισθητήρες λήψης δεδομένων. Επίσης ο απομακρυσμένος έλεγχος γίνεται μέσω συσκευών κινητής τηλεφωνίας και του διαδικτύου.

Η δυνατότητα παρακολούθησης της ενεργειακής κατανάλωσης σε πραγματικό χρόνο έχει ως σκοπό την εύκολη ενημέρωση του χρήστη καθώς και την ευαισθητοποίησή του όσον αφορά στην έκλυση αερίων που συμβάλουν στο φαινόμενο του θερμοκηπίου μέσω της ενεργειακής του κατανάλωσης καθώς επίσης και της αλλαγής νοοτροπίας χρήσης των ενεργειακών πόρων από τους καταναλωτές στο άμεσο μέλλον.

Ευχαριστίες

Ευχαριστώ θερμά τον καθηγητή μου Δρ. Εμμανουήλ Σκουνάκη για την καθοδήγηση και τη βοήθειά του στο πλαίσιο υλοποίησης της παρούσας πτυχιακής εργασίας.

Επίσης τους συμφοιτητές και φίλους μου Μαρίνα Αγγελίδα, Μιχάλη Κωνσταντάκη, Αλέξανδρο Ελευθερίου, Έλενα Στριλιγκά, Γιώργο Τσιντζέλη και πολλούς άλλους που μέσω των συζητήσεων και της ανταλλαγής ιδεών όλα αυτά τα χρόνια συνέβαλαν στην ανάπτυξη του χαρακτήρα και του τρόπου σκέψης που έχω σήμερα.

Τέλος θέλω να απευθύνω ένα μεγάλο ευχαριστώ στην οικογένειά μου και ιδιαίτερα στους γονείς μου για την τεράστια υποστήριξη και εμπιστοσύνη που μου πρόσφεραν όλα αυτά τα χρόνια, δίνοντάς μου πάντοτε τα απαραίτητα εφόδια για να φτάσω τους στόχους μου.

Πίνακας Περιεχομένων

Abstract.....	3
Περίληψη.....	4
Ευχαριστίες.....	5
Πίνακας Περιεχομένων.....	6
Κατάλογος Εικόνων.....	8
Κατάλογος Πινάκων.....	9
Κεφάλαιο 1 Εισαγωγή	10
1.1 Ιστορική Αναδρομή	10
1.1.1 Η Ιστορία των Μετρητών Ηλεκτρικής Ενέργειας.....	10
1.1.2 Ηλεκτρολυτικοί Μετρητές.....	11
1.1.3 Μετρητές Εκκρεμούς.....	12
1.1.4 Μετρητές με Χρήση Κινητήρα.....	13
1.1.5 Μετρητές Επαγωγής.....	13
1.1.6 Περαιτέρω Βελτιώσεις και Νέες Λειτουργίες.....	14
Κεφάλαιο 2 Σύγχρονοι Μετρητές Ηλεκτρικής Ενέργειας	15
2.1 Ηλεκτρομηχανικοί Μετρητές.....	15
2.2 Ηλεκτρονικοί Μετρητές.....	16
2.3 Μετρητές Ενέργειας Μεμονωμένων Συσκευών.....	17
2.4 Έξυπνοι μετρητές.....	17
Κεφάλαιο 3 Το Προτεινόμενο Σύστημα – Τα Μηχανήματα.....	19
3.1 Ο Μικροελεγκτής Arduino.....	19
3.2 Η δομή του Μικροελεγκτή.....	19
3.3 Επίσημες πλακέτες Arduino.....	20
3.4 Οι ασπίδες (Shields) Arduino.....	21
3.5 Μικροελεγκτής Arduino Ethernet Rev.3 με PoE.....	22

3.6	Απαιτούμενα Υλικά Κατασκευής του Συστήματος.....	24
3.7	Αισθητήρας ρεύματος ACS712.....	25
3.8	Μετατροπέας USB2SERIAL.....	27
3.9	LCD Οθόνη.....	28
3.10	Ηλεκτρονόμος (Relay).....	29
Κεφάλαιο 4	Το Προτεινόμενο Σύστημα – Η Πλατφόρμα Διαχείρισης.....	32
4.1	Λογισμικό Arduino IDE.....	32
4.2	Σύνοψη της Μεθόδου Λήψης των Μετρήσεων και Υπολογισμών...	34
4.3	Σύνδεση «Internet of Things».....	36
4.4	Η Πλατφόρμα Thingspeak.com.....	37
4.5	Η Εφαρμογή Blynk.....	38
4.6	Ο Κώδικας του Έξυπνου Μετρητή Ηλεκτρικής Ενέργειας.....	41
Κεφάλαιο 5	Συμπεράσματα & Προτάσεις για το Μέλλον.....	49
5.1	Επαλήθευση Σωστής Λειτουργίας του Συστήματος.....	49
5.2	Ενσωμάτωση στο Πρωτόκολλο KNX.....	49
5.3	Συμπεράσματα και Προτάσεις για το Μέλλον.....	50
Αναφορές	52

Κατάλογος Εικόνων

Εικόνα 1.1: Ηλεκτρολυτικός Μετρητής Edison.....	11
Εικόνα 1.2: Ο Μετρητής Εκκρεμούς του Hermann Aron.....	12
Εικόνα 1.3: Ο Μετρητής Thomson (1889).....	13
Εικόνα 1.4: Επαγωγικός Μετρητής Shallenberger (1894).....	14
Εικόνα 2.1: Ηλεκτρομηχανικός Μετρητής Επαγωγής Τριών Φάσεων.....	15
Εικόνα 2.2: Ηλεκτρονικός Μετρητής.....	17
Εικόνα 2.3: Μετρητής Πρίζας.....	17
Εικόνα 2.4: Παράδειγμα Έξυπνου μετρητή.....	18
Εικόνα 3.1: Μικροελεγκτής Arduino Uno Rev.3.....	19
Εικόνα 3.2: Η πλακέτα Arduino LilyPad.....	20
Εικόνα 3.3: Ασπίδα Κινητήρα (Motor Shield v1.0).....	21
Εικόνα 3.4: Μικροελεγκτής Arduino Ethernet Rev.3 με Ενσωματωμένο PoE.....	23
Εικόνα 3.5: Σχηματικό Διάγραμμα του Μικροελεγκτή Arduino Ethernet Rev.3.....	24
Εικόνα 3.6: Αισθητήρας Ρεύματος ACS712 SparkFun Hall-Effect.....	25
Εικόνα 3.7: Γραφική Παράσταση Τάσης Εξόδου προς Ένταση του Ρεύματος.....	25
Εικόνα 3.8: Σχηματικό Διάγραμμα Αισθητήρα Ρεύματος ACS712 Sparkfun.....	26
Εικόνα 3.9: Παράδειγμα Σύνδεσης του ACS712 με τον Μικροελεγκτή Arduino.....	26
Εικόνα 3.10: Μετατροπέας USB2SERIAL (USB Mini).....	27
Εικόνα 3.11: Οθόνη LCD 16x2 Χαρακτήρων.....	28
Εικόνα 3.12: Διάγραμμα Σύνδεσης Arduino Ethernet με την LCD Οθόνη.....	29
Εικόνα 3.13: Arduino 2-Relay Module.....	29
Εικόνα 3.14: Σύνδεση του Arduino με Relay για τον Έλεγχο Κυκλώματος 230V.....	30
Εικόνα 3.15: Σχηματικό Διάγραμμα του Arduino 2-Relay Module.....	31
Εικόνα 4.1: Το Λογισμικό Arduino IDE.....	32
Εικόνα 4.2: Παράδειγμα Κώδικα για τον Έλεγχο ενός LED στο Arduino IDE.....	33
Εικόνα 4.3: Γραφική Παράσταση Read Value προς Πραγματική Τάση.....	35
Εικόνα 4.4: Internet of Things.....	37
Εικόνα 4.5: Λογότυπο ThingSpeak.....	37
Εικόνα 4.6: Το Περιβάλλον της Εφαρμογής Blynk.....	38
Εικόνα 4.7: Βιβλιοθήκη Blynk.....	39
Εικόνα 4.8: Κώδικας Blynk.....	40
Εικόνα 4.9: Μοναδικός Κωδικός «auth token».....	40
Εικόνα 5.1: Ηλεκτρονικό Πολύμετρο.....	49

Εικόνα 5.2: Λογότυπο Πρωτοκόλλου KNX.....	50
Εικόνα 5.3: Μη Επεμβατικός Αισθητήρας Ρεύματος AC.....	51

Κατάλογος Πινάκων

Πίνακας 1: Σύνοψη Χαρακτηριστικών του Arduino Ethernet Rev.3 με PoE.....	23
Πίνακας 2: Σύνδεση Οθόνης LCD στον Arduino.....	28

Κεφάλαιο 1: Εισαγωγή

1.1 Ιστορική Αναδρομή

1.1.1 Η Ιστορία του Μετρητή Ηλεκτρικής Ενέργειας

Το πρώτο μισό του δέκατου ένατου αιώνα έφερε μεγάλες ανακαλύψεις στον ηλεκτρομαγνητισμό. Το 1820, ο Γάλλος André-Marie Ampère (1775-1836) ανακάλυψε την ηλεκτροδυναμική αλληλεπίδραση μεταξύ των ρευμάτων. Το 1827, ο Γερμανός Georg Simon Ohm (1787-1854) ανακάλυψε τη σχέση μεταξύ τάσης και ρεύματος σε αγωγό. Το 1831, ο Βρετανός Michael Faraday (1791-1867) ανακάλυψε το νόμο της επαγωγής, στην οποία βασίζεται η λειτουργία των γεννητριών, κινητήρων και μετασχηματιστών.

Από το δεύτερο μισό του αιώνα, το έδαφος ήταν καλά προετοιμασμένο για πρακτικές εφαρμογές. Τις ανακαλύψεις ακολούθησαν εφευρέσεις και διπλώματα ευρεσιτεχνίας. Η λάμπα, το δυναμό, ο κινητήρας, ο μετασχηματιστής, ο μετρητής και ο στρόβιλος εφευρέθηκαν σε γρήγορη διαδοχή.

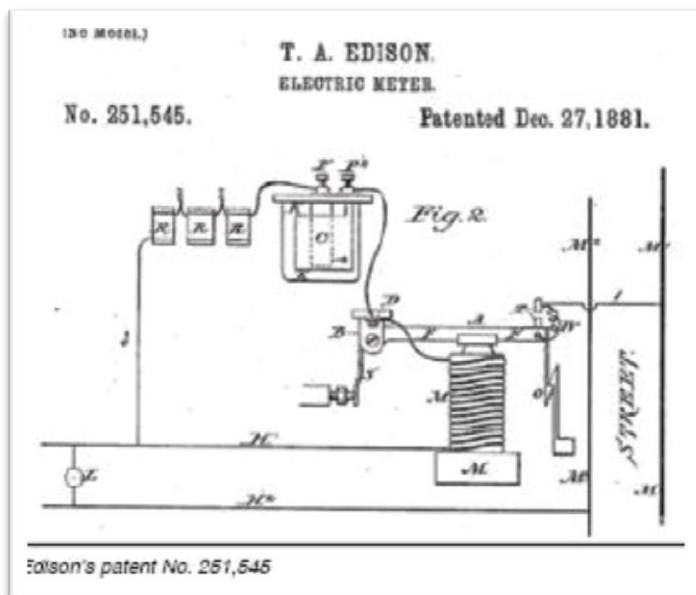
Ο Ούγγρος Ottó Titusz Bláthy, εφευρέτης του μετρητή ρεύματος επαγωγής και συνεφευρέτης του μετασχηματιστή, περιέγραψε την συναρπαστική περίοδο πίσω στο 1930 με αυτά τα λόγια: *«Στις μέρες μου ήταν εύκολο. Η επιστήμη ήταν σαν ένα τροπικό δάσος. Το μόνο που χρειαζόταν ήταν ένα καλό τσεκούρι, και όπου κι αν χτυπούσες θα έριχνες κάτω ένα τεράστιο δέντρο.»*

Με την εφεύρεση του δυναμό (Anyos Jedlik το 1861, Werner von Siemens το 1867) η ηλεκτρική ενέργεια μπορούσε πλέον να παραχθεί σε μεγάλες ποσότητες. Η πρώτη μαζική εφαρμογή της ηλεκτρικής ενέργειας ήταν ο φωτισμός. Όταν το νέο αυτό προϊόν, η ηλεκτρική ενέργεια, άρχισε να πωλείται, ήταν προφανές ότι το κόστος θα έπρεπε να προσδιοριστεί. Δεν ήταν σαφές ωστόσο, ποιες θα έπρεπε να είναι οι μονάδες που θα τιμολογείται και ποιες είναι οι πλέον κατάλληλες αρχές μέτρησης.

Ο πρώτος μετρητής *lamp-hour* εφευρέθηκε και κατοχυρώθηκε με δίπλωμα ευρεσιτεχνίας το 1872 από τον Samuel Gardiner στις Η.Π.Α. Ο μετρητής αυτός μετρούσε τη χρονική διάρκεια κατά την οποία παρέχονταν ενέργεια στο φορτίο, καθώς όλες οι λάμπες που ήταν συνδεδεμένες σε αυτόν τον μετρητή ελέγχονταν από έναν διακόπτη. Αργότερα, καθώς η υποδιαίρεση των κυκλωμάτων φωτισμού

έγινε εφικτή με την εισαγωγή του λαμπτήρα Edison ο παραπάνω μετρητής σταμάτησε πλέον να χρησιμοποιείται.

1.1.2 Ηλεκτρολυτικοί Μετρητές



Εικόνα 1.1: Ηλεκτρολυτικός Μετρητής Edison

Ο Thomas Alva Edison (1847-1931), ο οποίος παρουσίασε τα πρώτα ηλεκτρικά συστήματα διανομής για φωτισμό με τη χρήση συνεχούς ρεύματος, έκρινε ότι η ηλεκτρική ενέργεια πρέπει να πωλείται ακριβώς όπως το φυσικό αέριο – το οποίο επίσης χρησιμοποιούνταν ευρέως για φωτισμό εκείνη την περίοδο.

Ο μετρητής ηλεκτρισμού του που κατοχυρώθηκε με δίπλωμα ευρεσιτεχνίας στις ΗΠΑ το 1881 εκμεταλλευόταν την ηλεκτροχημική

επίδραση του ρεύματος. Περιείχε μια ηλεκτρολυτική κυψέλη, εντός της οποίας υπήρχε μια επακριβώς ζυγισμένη λωρίδα χαλκού που τοποθετούνταν στην αρχή της περιόδου χρέωσης. Το ρεύμα που διερχόταν μέσω του ηλεκτρολύτη, προκαλούσε εναπόθεση χαλκού.

Κατά το τέλος της περιόδου χρέωσης, η λωρίδα χαλκού ζυγίζονταν πάλι και η διαφορά αντιπροσώπευε το ποσό της ηλεκτρικής ενέργειας που είχε καταναλωθεί. Οι μετρητές αυτοί παρέμειναν σε χρήση μέχρι το τέλος του 19ου αιώνα. Υπήρχε όμως ένα μεγάλο μειονέκτημα. Η ερμηνεία του μετρητή ήταν δύσκολη για την αρμόδια υπηρεσία και αδύνατη για τον πελάτη.

Ο Edison αργότερα πρόσθεσε ένα μηχανισμό μέτρησης για να βοηθήσει στην ανάγνωση του μετρητή. Υπήρχαν και άλλοι ηλεκτρολυτικοί μετρητές, όπως ο γερμανικός μετρητής υδρογόνου *Siemens-Shuckert* και ο μετρητής υδραργύρου *Schott & Gen. Jena*.

Οι ηλεκτρολυτικοί μετρητές μπορούσαν να μετρήσουν μόνο *αμπερώρια* και δεν ήταν κατάλληλοι όταν η τάση παρουσίαζε διακυμάνσεις.

1.1.3 Μετρητές Εκκρεμούς

Μια άλλη πιθανή αρχή πάνω στην οποία ήταν πιθανό να βασιστεί ένας μετρητής ήταν η δημιουργία κάποιας κίνησης - ταλάντωσης ή περιστροφής - ανάλογη της ενέργειας, η οποία θα μπορούσε στη συνέχεια να αποτυπωθεί σε ένα μητρώο για να διαβαστεί.

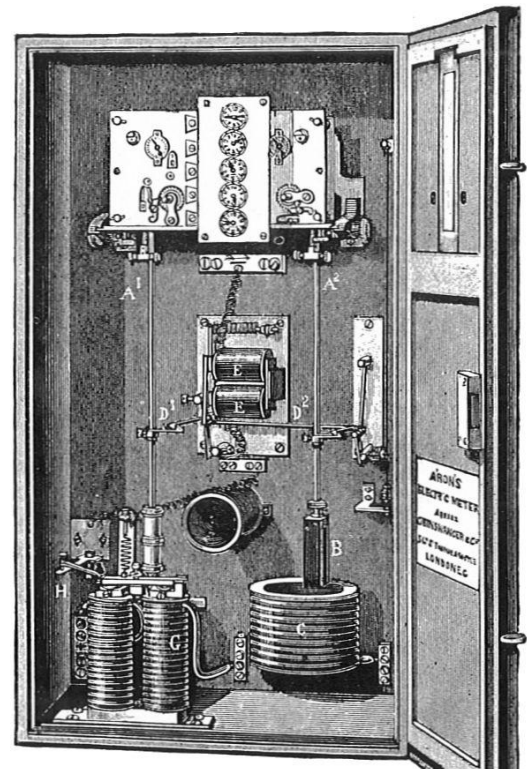
Η αρχή του μετρητή εκκρεμούς αρχικά περιγράφηκε από τους Αμερικανούς William Edward Ayrton και John Perry το 1881.

Το 1884, χωρίς να γνωρίζει για την εφεύρεσή των παραπάνω, ο Hermann Aron (1845-1902) κατασκεύασε ένα μετρητή εκκρεμούς στη Γερμανία.

Στην πιο προχωρημένη του μορφή, ο μετρητής αυτός είχε δύο εκκρεμή, με ένα πηνίο στον κάθε ένα από αυτά που συνδέονται με την τάση. Κάτω από τα εκκρεμή υπήρχαν δύο πηνία ρεύματος σε αντίθετες κατευθύνσεις. Ένα από τα εκκρεμή ως εκ τούτου λειτουργούσε πιο αργά και το άλλο πιο γρήγορα.

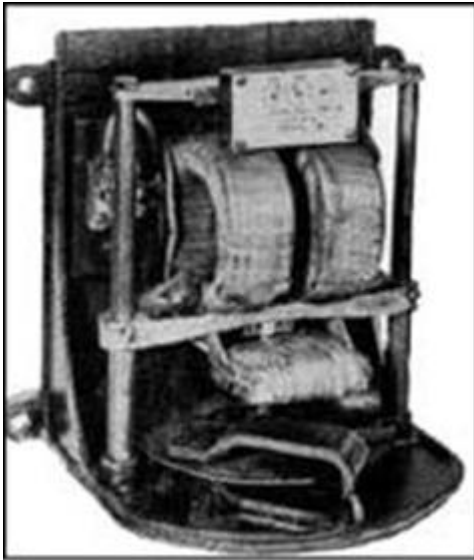
Η διαφορά μεταξύ των χρόνων ταλάντωσης οδήγησε στον μηχανισμό μέτρησης. Ο ρόλος των δύο εκκρεμών άλλαζε κάθε λεπτό, έτσι ώστε η αρχική διαφορά μεταξύ των χρόνων ταλάντωσης των εκκρεμών να αντισταθμιστεί. Ταυτόχρονα, το ρολόι μηδενιζόταν.

Οι μετρητές αυτοί ήταν ακριβοί, επειδή περιείχαν δύο ρολόγια και αντικαταστάθηκαν σταδιακά από μετρητές με χρήση κινητήρα. Οι μετρητές εκκρεμούς μετρούσαν αμπερώρια ή watt-ώρες, αλλά μπορούσαν να χρησιμοποιηθούν μόνο για το συνεχές ρεύμα.



Εικόνα 1.2: Ο Μετρητής Εκκρεμούς του Hermann Aron

1.1.4 Μετρητές με Χρήση Κινητήρα



Εικόνα 1.3: Ο Μετρητής Thomson (1889)

Μετά τους ηλεκτρολυτικούς μετρητές έκαναν την εμφάνισή τους οι μετρητές που χρησιμοποιούσαν έναν κινητήρα ώστε να επιτευχθεί η μέτρηση της ηλεκτρικής ενέργειας.

Σε αυτούς τους μετρητές η κινητήρια ροπή είναι ανάλογη με το φορτίο και εξισορροπείται από τη ροπή πέδησης, έτσι ώστε η ταχύτητα του ρότορα να είναι ανάλογη με το φορτίο όταν οι ροπές είναι σε ισορροπία.

Ο Αμερικανός Elihu Thomson (1853- 1937) ανέπτυξε το *Recording wattmeter* το 1889 για την εταιρεία General Electric.

Ήταν ένας κινητήρας του οποίου το στροφέιο διεγείρεται από την τάση διαμέσου ενός πηνίου και έναν αντιστάτη, χρησιμοποιώντας έναν μετατροπέα ηλεκτρικού ρεύματος.

Ο αντιστάτης ενεργοποιούνταν από το ρεύμα και η κινητήρια ροπή ήταν επομένως ανάλογη με το γινόμενο τάσης και ρεύματος.

Εδώ, ένας μόνιμος μαγνήτης και ένας δίσκος αλουμινίου στερεωμένος στο ρότορα δημιουργούσαν την ροπή πέδησης. Ο μετρητής αυτός χρησιμοποιήθηκε κυρίως για το συνεχές ρεύμα.

1.1.5 Μετρητές Επαγωγής

Το 1885, ο Ιταλός Galileo Ferraris ανακάλυψε ότι δύο εκτός φάσης AC πεδία μπορούσαν να προκαλέσουν την περιστροφή ενός συμπαγούς δίσκου ή κυλίνδρου. Ο Nikola Tesla και ο Shallenberger ανακάλυψαν ανεξάρτητα επίσης το ίδιο φαινόμενο το 1888. Ο τελευταίος ανέπτυξε έναν μετρητή αμπερώνων εναλλασσομένου ρεύματος. Η ροπή πέδησης παρέχονταν από ένα στρόβιλο .

Αυτός ο μετρητής δεν είχε τρόπο να συμπεριλαμβάνει την τάση κατά την μέτρηση και δεν μπορούσε να υπολογίζει την ισχύ, οπότε δεν ήταν κατάλληλος για χρήση σε κινητήρες. Αυτές οι ανακαλύψεις ήταν η βάση των μετρητών επαγωγής.

Το 1889 ο Ούγγρος *Otto Titusz Bláthy*, που δούλευε για την εταιρία Ganz στη Βουδαπέστη, κατοχύρωσε την πατέντα *Electric meter for alternating currents* (Germany No 52,793, USA No 423,210).



Εικόνα 1.4: Επαγωγικός Μετρητής Shallenberger (1894)

Το 1894 ο Oliver Blackburn Shallenberger ανέπτυξε έναν επαγωγικό μετρητή *watt-ώρων* για την εταιρία Westinghouse ο οποίος είχε τα πηνία του ρεύματος και της τάσης τοποθετημένα σε αντίθετες πλευρές του δίσκου και δύο μόνιμους μαγνήτες στον ίδιο δίσκο.

Ήταν ένας μεγάλος και βαρύς μετρητής βάρους περίπου 20kg.

1.1.6 Περαιτέρω Βελτιώσεις και Νέες Λειτουργίες

Στα επόμενα χρόνια, έγιναν πολλές βελτιώσεις στους μετρητές ηλεκτρικής ενέργειας, όπως μείωση βάρους και διαστάσεων, διεύρυνση του φάσματος του φορτίου, αντιστάθμιση των μεταβολών του συντελεστή ισχύος, τάσης και θερμοκρασίας, ελαχιστοποίηση της τριβής και επίτευξη μακροπρόθεσμης σταθερότητας μέσω της χρήσης καλύτερων μαγνητών πέδησης όπως επίσης και εξάλειψη του πετρελαίου από το έδρανο και το μητρώο.

Ως την αλλαγή του αιώνα, αναπτύχθηκαν μετρητές τριών φάσεων με τη χρήση δυο ή τριών συστημάτων μέτρησης συνδυασμένα σε ένα, δύο ή τρεις δίσκους.

Κεφάλαιο 2: Σύγχρονοι Μετρητές Ηλεκτρικής Ενέργειας

2.1 Ηλεκτρομηχανικοί μετρητές

Ένας από τους πιο κοινούς τύπους σύγχρονου μετρητή ηλεκτρικής ενέργειας είναι ο ηλεκτρομηχανικός μετρητής επαγωγής ο οποίος υπολογίζει την ενέργεια που καταναλώνεται σε watt-ώρες.

Ο μετρητής αυτός λειτουργεί με την καταμέτρηση των στροφών ενός μη - μαγνητικού, αλλά ηλεκτρικά αγώγιμου μεταλλικού δίσκου, ο οποίος είναι κατασκευασμένος ώστε να περιστρέφεται με ταχύτητα ανάλογη με την ισχύ που διέρχεται από το μετρητή.

Ο αριθμός των στροφών είναι συνεπώς ανάλογος της κατανάλωσης ενέργειας . Το πηνίο της τάσης καταναλώνει μια μικρή και σχετικά σταθερή ποσότητα ηλεκτρικής ενέργειας (συνήθως γύρω στα 2 watt) που δεν εγγράφεται στο μετρητή. Το ρεύμα του πηνίου καταναλώνει ομοίως ένα μικρό ποσό ενέργειας ανάλογο του τετραγώνου του ρεύματος που ρέει μέσα από αυτό , συνήθως 1-2 watt σε πλήρες φορτίο το οποίο εγγράφεται στο μετρητή.

Ο δίσκος ενεργοποιείται από δύο σειρές πηνίων τα οποία σχηματίζουν στην πραγματικότητα έναν κινητήρα επαγωγής δύο φάσεων .Το ένα πηνίο είναι συνδεδεμένο με τέτοιο τρόπο ώστε να παράγει μια μαγνητική ροή σε αναλογία με την τάση και το άλλο παράγει μια μαγνητική ροή ανάλογη του ρεύματος.

Το πεδίο του πηνίου τάσης καθυστερεί κατά 90 μοίρες, λόγω της επαγωγικής φύσης του πηνίου και διαβαθμίζεται χρησιμοποιώντας ένα πηνίο πέδησης στον δίσκο.

Το αποτέλεσμα είναι τέτοιο ώστε ασκείται μία δύναμη στο δίσκο ανάλογη του προϊόντος του στιγμιαίου ρεύματος, της τάσης και της γωνίας φάσης (συντελεστή ισχύος) μεταξύ τους.

Ένας μόνιμος μαγνήτης ασκεί μία αντίθετη δύναμη ανάλογη της ταχύτητας περιστροφής του δίσκου. Η ισορροπία μεταξύ αυτών των δύο αντίθετων δυνάμεων έχει ως αποτελέσματα ο δίσκος να περιστρέφεται με ταχύτητα ανάλογη της ισχύος ή του ρυθμού χρήσης ενέργειας. Ο δίσκος οδηγεί ένα μηχανισμό καταγραφής, ο



Εικόνα 2.1: Ηλεκτρομηχανικός Μετρητής Επαγωγής Τριών Φάσεων

οποίος μετράει τις στροφές προκειμένου να καταστεί δυνατή μια μέτρηση της συνολικής ενέργειας που χρησιμοποιείται.

Ο τύπος μετρητή που περιγράφεται παραπάνω χρησιμοποιείται σε μια παροχή εναλλασσομένου ρεύματος μονής φάσεως. Διαφορετικές διαμορφώσεις φάσεων χρησιμοποιούν διαφορετικά πηνία τάσης και ρεύματος.

Ο δίσκος υποστηρίζεται από έναν άξονα ο οποίος έχει έναν κοχλία που κινεί το μητρώο. Το μητρώο είναι μια σειρά από πίνακες που καταγράφουν την ποσότητα της ενέργειας που χρησιμοποιείται. Η ποσότητα της ενέργειας που αντιπροσωπεύεται από μία περιστροφή του δίσκου υποδηλώνεται με το σύμβολο Kh η οποία δίνεται σε μονάδες $watt - \acute{\omega}ρων$ ανά περιστροφή.

$$P = \frac{3600 \cdot Kh}{t}$$

Όπου t = ο χρόνος που χρειάζεται ο δίσκος για μια περιστροφή (sec) και P = ισχύς σε watts.

Οι περισσότεροι μετρητές ηλεκτρικής ενέργειας πρέπει να ελέγχονται χειροκίνητα , είτε από έναν εκπρόσωπο της εταιρείας ηλεκτρικής ενέργειας ή από τον πελάτη .

Σε περίπτωση που ο πελάτης διαβάζει το μετρητή, η ένδειξη μπορεί να παρέχεται στη ΔΕΗ μέσω τηλεφώνου, ταχυδρομείου ή μέσω του διαδικτύου .

Η εταιρεία ηλεκτρισμού θα απαιτήσει συνήθως μια επίσκεψη από έναν εκπρόσωπο της εταιρείας τουλάχιστον σε ετήσια βάση , προκειμένου να εξακριβώσει τις αναγνώσεις του πελάτη και να εξασφαλίσει την ασφάλεια του μετρητή.

2.2 Ηλεκτρονικοί Μετρητές

Οι ηλεκτρονικοί μετρητές εμφανίζουν την ενέργεια που χρησιμοποιείται σε μια οθόνη LCD ή LED, και μερικοί μπορούν επίσης να μεταδίδουν μετρήσεις σε απομακρυσμένες περιοχές.

Εκτός από τη μέτρηση της ενέργειας που χρησιμοποιείται, ηλεκτρονικοί μετρητές μπορούν επίσης να καταγράψουν άλλες παραμέτρους του φορτίου και της

προσφοράς όπως το στιγμιαίο και μέγιστο ποσοστό που απαιτείται για τη χρήση, τις τάσεις, τον συντελεστή ισχύος και την άεργο ισχύ.

Μπορούν επίσης να υποστηρίξουν μέτρηση ενέργειας ανά συγκεκριμένη στιγμή της ημέρας όπως για παράδειγμα την καταγραφή της ποσότητας ενέργειας που χρησιμοποιείται

κατά τη διάρκεια των ωρών αιχμής και τις ώρες εκτός αιχμής.



Εικόνα 2.2: Ηλεκτρονικός Μετρητής

2.3 Μετρητές Ενέργειας Μεμονωμένων Συσκευών



Εικόνα 2.3: Μετρητής Πρίζας

Οι μετρητές-πρίζες (*plug-loads*) μετρούν το ποσό της ενέργειας που χρησιμοποιείται από μεμονωμένες συσκευές.

Σήμερα υπάρχει μια ποικιλία των μοντέλων που διατίθενται στην αγορά, αλλά όλα λειτουργούν με την ίδια βασική αρχή.

Ο μετρητής είναι συνδεδεμένος σε πρίζα και η συσκευή που πρόκειται να μετρηθεί είναι συνδεδεμένη στον μετρητή. Αυτό το είδος μετρητή μπορεί να βοηθήσει στην εξοικονόμηση ενέργειας με τον εντοπισμό μεγάλων χρήσεων ενέργειας, ή τον εντοπισμό συσκευών που καταναλώνουν υπερβολική ισχύ σε κατάσταση αναμονής (*stand-by*).

2.4 Έξυπνοι μετρητές

Οι έξυπνοι μετρητές προσφέρουν πρόσθετες λειτουργίες, συμπεριλαμβανομένων της μέτρησης σε πραγματικό ή σχεδόν πραγματικό χρόνο, την κοινοποίηση διακοπής ρεύματος και την παρακολούθηση της ποιότητας της τάσης.

Μπορούν να επιτρέψουν στους οργανισμούς που καθορίζουν τις τιμές να εισάγουν διαφορετικές τιμές για την κατανάλωση με βάση την ώρα της ημέρας και την εποχή.

Αυτές οι διαφορές τιμών μπορούν να χρησιμοποιηθούν για να μειωθούν οι αιχμές της ζήτησης (μετατόπιση του φορτίου ή του φορτίου αιχμής), μειώνοντας την ανάγκη για πρόσθετες μονάδες παραγωγής ηλεκτρικής ενέργειας και ειδικότερα την πολύ ρυπογόνα και δαπανηρή λειτουργία των μονάδων ηλεκτροπαραγωγής φυσικού αερίου.

Η υπηρεσία που παρέχουν στους καταναλωτές έχει αποδειχθεί επίσης πως μειώνει την συνολική κατανάλωση ενέργειας.



Εικόνα 2.4: Παράδειγμα Έξυπνου Μετρητή

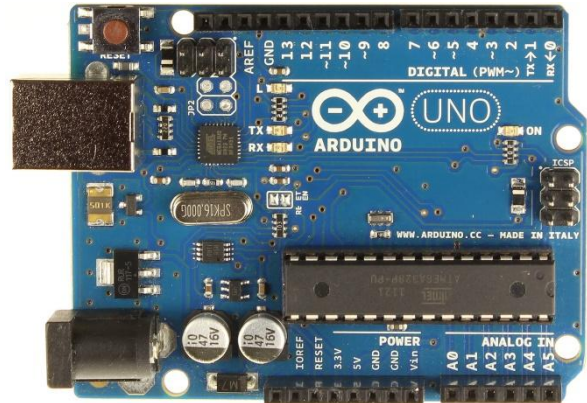
Ένας άλλος τύπος έξυπνου μετρητή χρησιμοποιεί ένα σύστημα παρακολούθησης φορτίου για να προσδιορίσει αυτόματα τον αριθμό και το είδος των συσκευών μιας κατοικίας, πόση ενέργεια χρησιμοποιεί η καθεμία και πότε.

Αυτός ο μετρητής χρησιμοποιείται από εταιρείες παραγωγής ηλεκτρικής ενέργειας για να κάνει έρευνες για τη χρήση ενέργειας.

Κεφάλαιο 3. Το Προτεινόμενο Σύστημα – Τα Μηχανήματα

3.1 Ο Μικροελεγκτής Arduino

Ο Arduino είναι μια απλή μητρική πλακέτα ανοικτού κώδικα, με ενσωματωμένο μικροελεγκτή και εισόδους/εξόδους η οποία μπορεί να προγραμματιστεί με τη γλώσσα Wiring που ουσιαστικά πρόκειται για τη γλώσσα προγραμματισμού C++ και ένα σύνολο από βιβλιοθήκες, υλοποιημένες



Εικόνα 3.1: Μικροελεγκτής Arduino Uno Rev.3

επίσης στην C++.

Ο 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 οι πληροφορίες για το υλικό είναι ελεύθερα διαθέσιμα για αυτούς που θέλουν να συναρμολογήσουν τον Arduino μόνοι τους.

3.2 Η Δομή του Μικροελεγκτή

Μία πλακέτα Arduino αποτελείται από ένα μικροελεγκτή Atmel AVR 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.

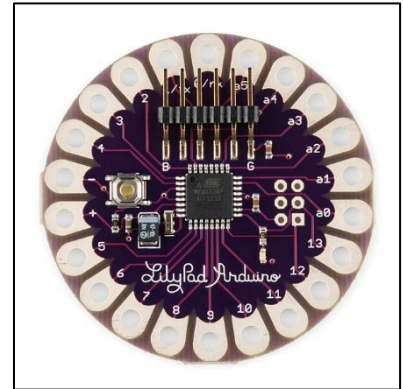
Ο μικροελεγκτής είναι από κατασκευής προγραμματισμένος με ένα bootloader, έτσι ώστε να μην χρειάζεται εξωτερικός προγραμματιστής.

Επίσης, ο Arduino και οι συμβατές με αυτόν μικροελεγκτές χρησιμοποιούν την τεχνολογία των shields, τυπωμένων boards επεκτάσεων κυκλωμάτων που συνδέονται στα παρεχόμενα από τον Arduino pin-headers.

Τα shields μπορούν να παρέχουν έλεγχο σε κινητήρες, GPS, Ethernet, LCD οθονών, κα. Ένας αριθμός από ασπίδες μπορεί επίσης να γίνει και DIY (Do It Yourself).

3.3 Επίσημες πλακέτες Arduino

Η πρωτότυπη πλακέτα του Arduino κατασκευάζεται από την Ιταλική εταιρία Smart Projects όμως κάποιες πλακέτες με την μάρκα του Arduino έχουν σχεδιαστεί από την Αμερικάνικη εταιρία SparkFun Electronics. Μέχρι τώρα έχουν χρησιμοποιηθεί εμπορικά οι εξής δεκαέξι εκδοχές του Arduino Hardware:



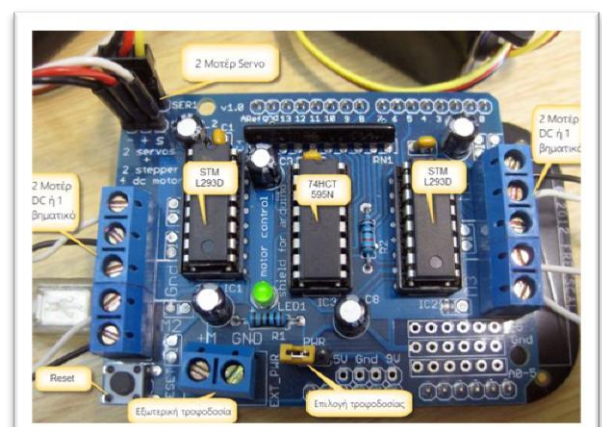
Εικόνα 3.2: Η πλακέτα Arduino LilyPad

1. Ο *Serial Arduino*, προγραμματισμένος με μία σειριακή DE-9 σύνδεση χρησιμοποιώντας τεχνολογία ATmega8
2. Ο *Arduino Extreme*, με ένα USB interface για προγραμματισμό χρησιμοποιώντας τεχνολογία ATmega8
3. Ο *Arduino Mini*, μία έκδοση μινιατούρας του Arduino χρησιμοποιώντας τεχνολογία surface-mounted ATmega168
4. Ο *Arduino Nano*, ένα ακόμα πιο μικρό, USB τροφοδοτούμενη έκδοχή του Arduino χρησιμοποιώντας τεχνολογία surface-mounted ATmega168 (ATmega328 για την νεότερη έκδοση)
5. Ο *LilyPad Arduino*, ένα μινιμαλιστικό σχέδιο για εφαρμογές ένδυσης και E-textiles χρησιμοποιώντας τεχνολογία surface-mounted ATmega328
6. Ο *Arduino NG*, με ένα USB interface για προγραμματισμό και χρησιμοποιώντας τεχνολογία ATmega8
7. Ο *Arduino NG plus*, με ένα USB interface για προγραμματισμό και χρησιμοποιώντας τεχνολογία atmega168
8. Ο *Arduino Bluetooth*, με Bluetooth interface για προγραμματισμό χρησιμοποιώντας τεχνολογία ATmega168
9. Ο *Arduino Diecimila*, με ένα USB interface και χρησιμοποιεί τεχνολογία ATmega168 σε ένα DIP28 πακέτο

10. Ο *Arduino Duemilanove* (“2009”), χρησιμοποιεί τεχνολογία ATmega168 (ATmega328 για την καινούργια έκδοση) και τροφοδοτείται μέσω ενέργειας USB/DC, αυτόματα εναλλασσόμενης
11. Ο *Arduino Mega*, χρησιμοποιώντας τεχνολογία surface-mounted ATmega1280 για περαιτέρω I/O και μνήμη[11]
12. Ο *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. Ο *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. Ο *Arduino Leonardo*, με ένα ATmega32U4 chip που εξαλείφει την ανάγκη για συνδεσιμότητα μέσω USB και μπορεί να χρησιμοποιηθεί ως ψηφιακό πληκτρολόγιο ή ποντίκι. Κυκλοφόρησε στο Maker Faire Bay Area το 2012
15. Ο *Arduino Esplora*, με εμφάνιση που παραπέμπει σε χειριστήριο κονσόλας βιντεοπαιχνιδιών με joystick και ενσωματωμένους αισθητήρες για ήχο, φως, θερμοκρασία και επιτάχυνση
16. Ο *Arduino Due* είναι ένα μικροχειριστήριο board βασισμένο στην τεχνολογία Atmel SAM3X8E ARM Cortex-M3 CPU. Είναι το πρώτο board της Arduino βασισμένη σε επεξεργαστή 32-bit ARM microcontroller

3.4 Οι Ασπίδες (Shields) Arduino

Τα Arduino αλλά και τα διάφορα boards συμβατά με Arduino χρησιμοποιούν την τεχνολογία των shields, δηλαδή τυπωμένων boards επεκτάσεων κυκλωμάτων που συνδέονται στα κανονικά παρεχόμενα Arduino pins. Τα shields μπορούν να παρέχουν έλεγχο σε διάφορες λειτουργίες



Εικόνα 3.3: Ασπίδα Κινητήρα (Motor Shield v1.0)

όπως κινητήρες, GPS, Ethernet, LCD εικόνας ή breadboarding (προτυποποίησης). Ένας αριθμός από ασπίδες μπορεί επίσης να γίνει αυτοσχέδια.

Παρακάτω παρατίθεται ένα παράδειγμα Arduino Shield:

Αν η κατασκευή κάποιου project απαιτούσε τον έλεγχο ενός μοτέρ τότε θα χρησιμοποιούσαμε μια Ασπίδα Κινητήρα όπως η DK Electronics Motor Shield v1.0 οποία έχει τα παρακάτω χαρακτηριστικά:

- 4 H-Bridges (4 Η-Γέφυρες) : L293D chipset το οποίο παρέχει 0.6A σε κάθε γέφυρα (1.2A μέγιστο) με θερμική προστασία. Μπορεί να διαχειριστεί μοτέρ από 4.5VDC έως 25VDC.
- Διαχειρίζεται μέχρι 4 κινητήρες συνεχούς ρεύματος (DC motors) με επιλογή ταχύτητας 8bit (0 – 255) και μέχρι 2 κινητήρες βηματικούς (Stepper motors).
- Είναι εξοπλισμένη με αντιστάσεις (Pull down resistors) που προστατεύουν τους κινητήρες απενεργοποιώντας τους κατά την εκκίνηση
- Κουμπί μηδενισμού (Reset button).
- Παρέχει συμβατότητα με Arduino Mega 1280 & 2560, Diecimila, Duemilanove και UNO.

Προκειμένου να λειτουργήσει η ασπίδα θα ήταν απαραίτητο να φορτώσουμε και μια βιβλιοθήκη με κατάλληλες συναρτήσεις που ελέγχουν τους κινητήρες.

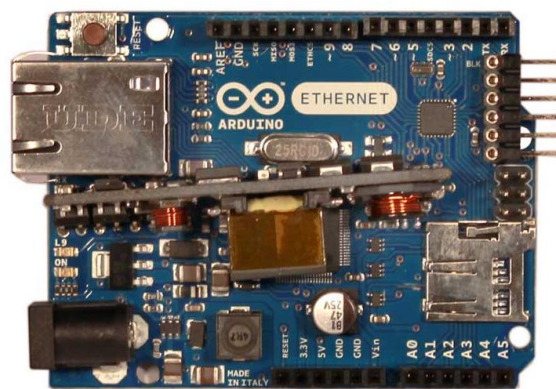
Η βιβλιοθήκη υπάρχει δωρεάν στο διαδίκτυο.

Στη συνέχεια από το μενού του Arduino IDE ακολουθώντας τα βήματα: Σχέδιο->Εισαγωγή βιβλιοθήκης->Import Library κάνουμε εισαγωγή της βιβλιοθήκης και είναι η ασπίδα κινητήρα είναι πλέον έτοιμη για χρήση.

3.5 Ο Μικροελεγκτής Arduino Ethernet Rev.3 με PoE

Η πλακέτα του Arduino που χρησιμοποιήσαμε τελικά είναι ή πλακέτα Arduino Ethernet Rev.3, η οποία είναι εξοπλισμένη με PoE. Η έννοια PoE (Power over Ethernet) περιγράφει ένα σύστημα το οποίο επιτρέπει την διάδοση ηλεκτρικής ενέργειας ταυτόχρονα με την διάδοση δεδομένων μέσω της σύνδεσης Ethernet.

Με αυτό τον τρόπο επιτυγχάνεται η παροχή ενέργειας αλλά και η σύνδεση δεδομένων μέσω ενός καλωδίου Ethernet. Ο Arduino Ethernet βασίζεται στον μικροεπεξεργαστή ATmega328. Έχει 14 ψηφιακά pins εισόδου/εξόδου, 6 αναλογικές εισόδους, έναν κρυσταλλικό ταλαντωτή 16 MHz, μια σύνδεση RJ45, μια υποδοχή ρεύματος, ένα ICSP header, και ένα κουμπί reset.

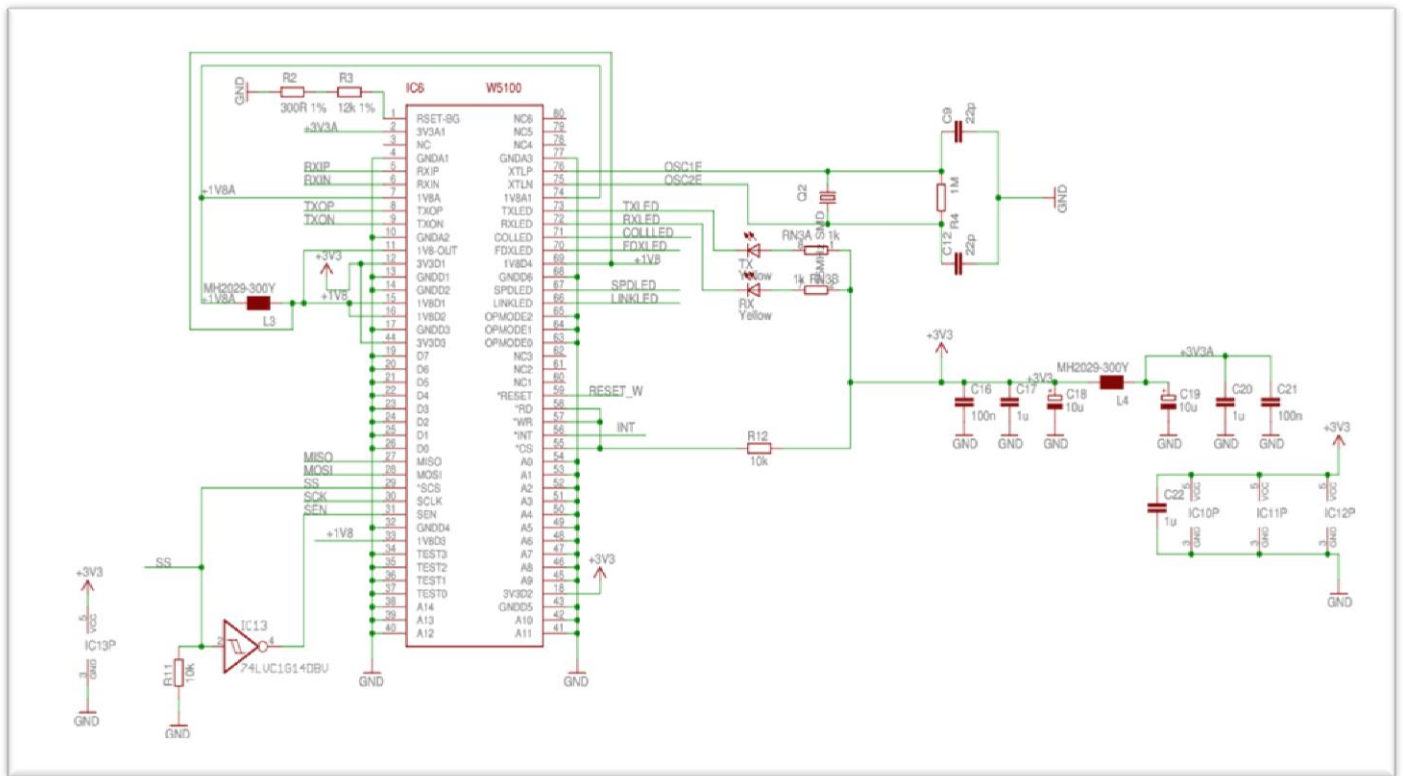


Εικόνα 3.4: Μικροελεγκτής Arduino Ethernet Rev.3 με Ενσωματωμένο PoE

Σύνοψη:

Μικροελεγκτής	ATmega328
Τάση λειτουργίας	5V
Τάση εισόδου (προτεινόμενη)	7-12V
Τάση εισόδου (όρια)	6-20V
Τάση εισόδου PoE (όρια)	36-57V
Digital I/O Pins	14 (of which 4 provide PWM output)
Arduino Pins reserved:	10 to 13 used for SPI
	4 used for SD card
	2 W5100 interrupt (when bridged)
Αναλογικά Pins εισόδου	6
DC ρεύμα ανά I/O Pin	40 mA
DC ρεύμα για pin 3.3V	50 mA
Flash Memory	32 KB (ATmega328) of which 0.5 KB used by bootloader
SRAM	2 KB (ATmega328)
EEPROM	1 KB (ATmega328)
Ταχύτητα ρολογιού	16 MHz
W5100 TCP/IP Embedded Ethernet Controller	
Power Over Ethernet ready Magnetic Jack	
Micro SD card, with active voltage translators	
Μήκος	68.6 mm
Πλάτος	53.3 mm
Βάρος	28 g

Πίνακας 1: Σύνοψη Χαρακτηριστικών του Arduino Ethernet Rev.3 με PoE



Εικόνα 3.5: Σχηματικό Διάγραμμα του Μικροελεγκτή Arduino Ethernet Rev.3

3.6 Απαιτούμενα Υλικά Κατασκευής του Συστήματος

1. Arduino Ethernet Rev.3 με PoE
2. Οθόνη LCD 16x2 Χαρακτήρων
3. Breadboard
4. Αισθητήρας ρεύματος ACS712
5. Ποτενσιόμετρο (10K)
6. Καλώδιο Ethernet
7. Αντιστάσεις
8. USB2SERIAL μετατροπέας
9. Πυκνωτής 10μF
10. Λαμπτήρες DC 12V

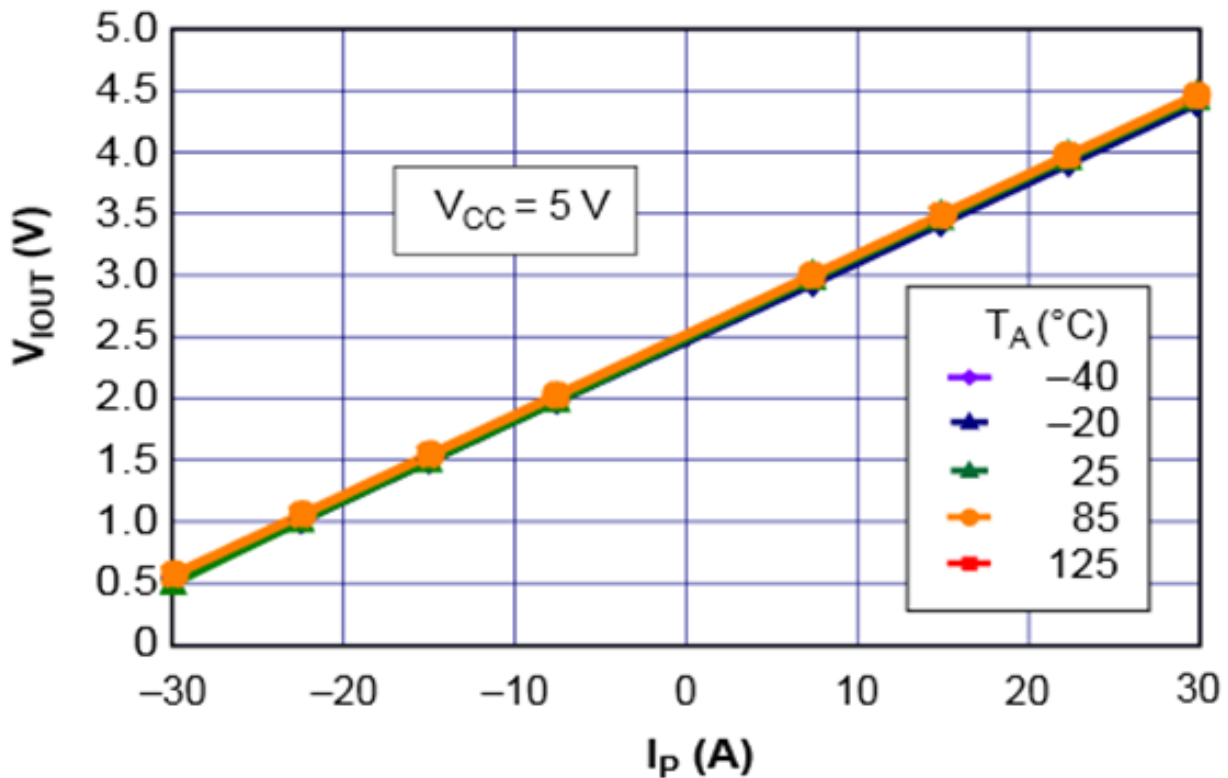
3.7 Αισθητήρας Ρεύματος ACS712



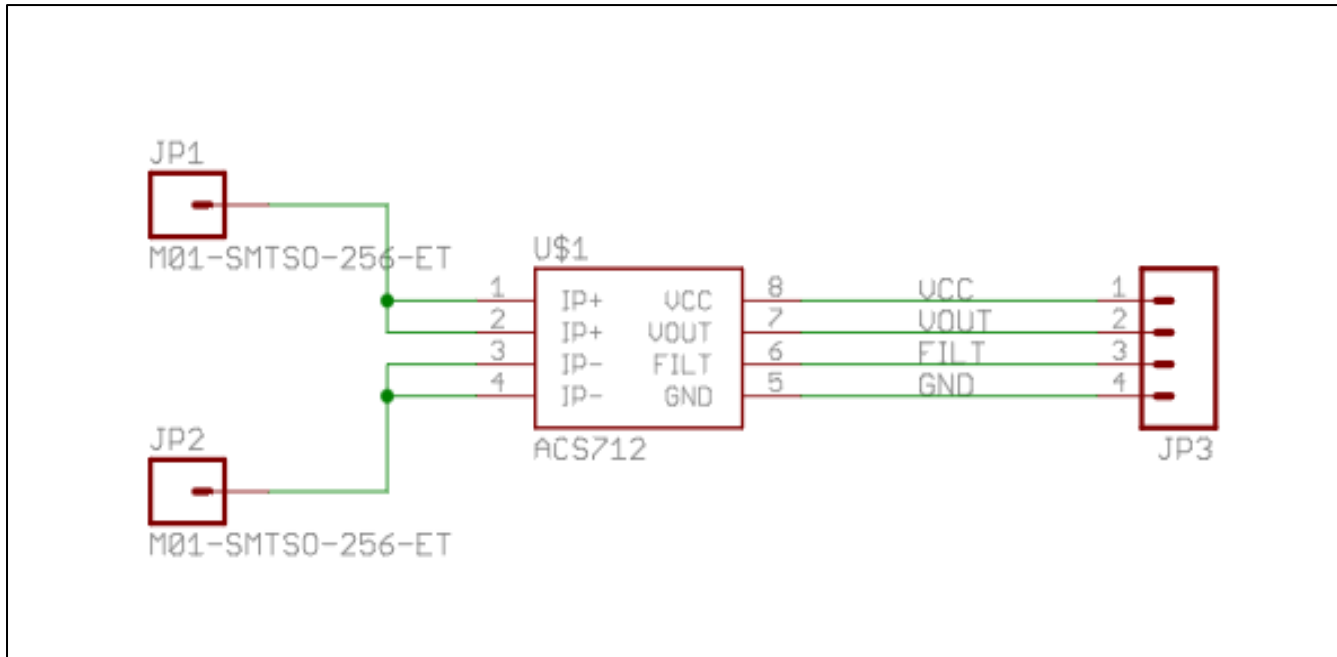
Εικόνα 3.6: Αισθητήρας Ρεύματος ACS712 SparkFun Hall-Effect

Ο αισθητήρας ACS712 SparkFun Hall-Effect παρέχει ακριβή μέτρηση του ρεύματος για AC αλλά και DC σήματα.

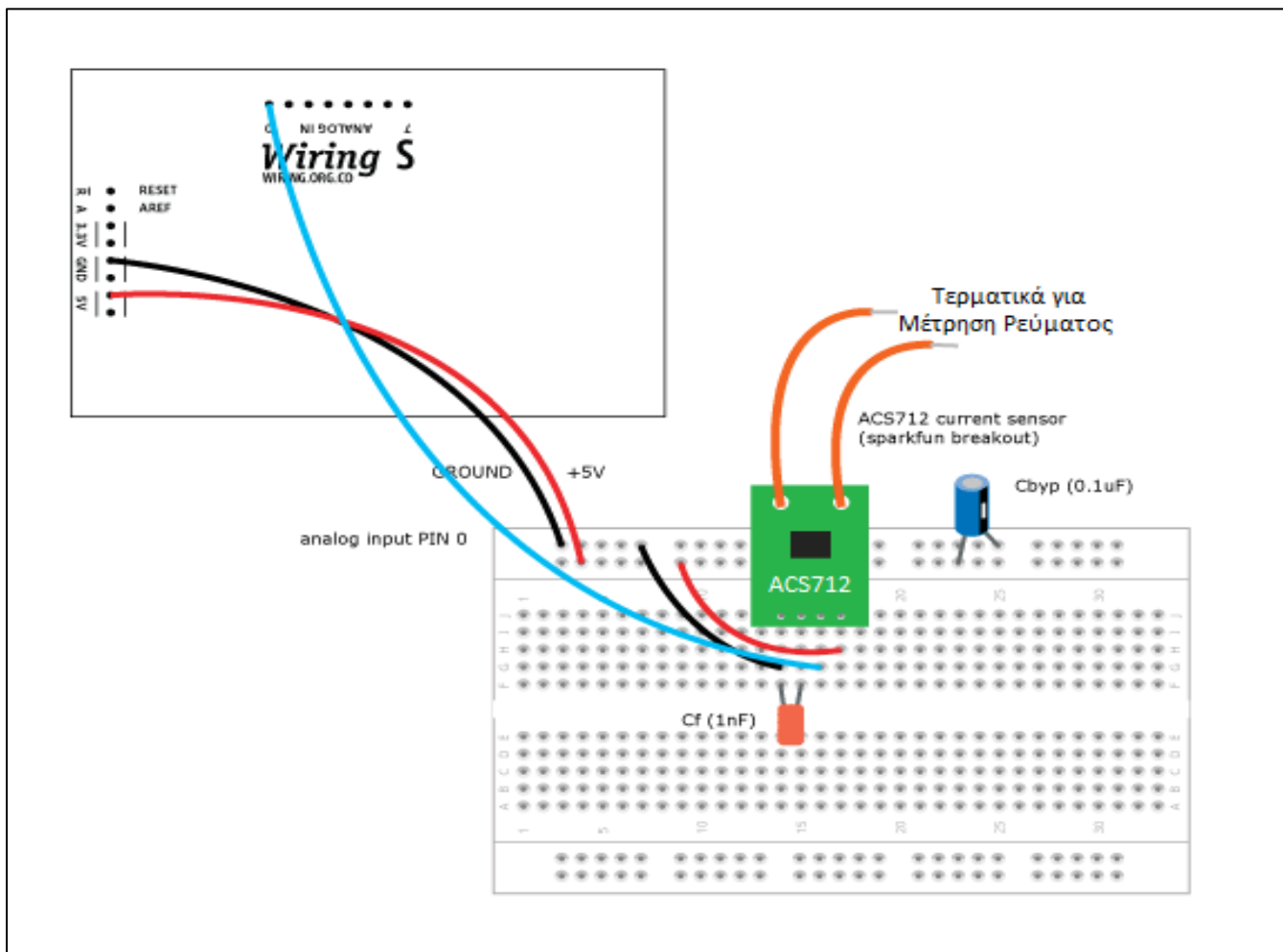
Η κατασκευή της συσκευής από χοντρό αγωγό χαλκού επιτρέπει την επιβίωση της συσκευής σε έως και 5 φορές μεγαλύτερες εντάσεις ρεύματος από τις προβλεπόμενες. Το ACS712 εξάγει ένα αναλογικό σήμα εξόδου τάσης που μεταβάλλεται γραμμικά με την τρέχουσα μέτρηση ρεύματος. Η συσκευή απαιτεί 5 Volt DC για την VCC.



Εικόνα 3.7: Γραφική Παράσταση Τάσης Εξόδου προς Ένταση του Ρεύματος



Εικόνα 3.8: Σχηματικό Διάγραμμα Αισθητήρα Ρεύματος ACS712 Sparkfun



Εικόνα 3.9: Παράδειγμα Σύνδεσης του Αισθητήρα ACS712 με τον Μικροελεγκτή Arduino

Παράδειγμα κώδικα:

```
int sensorValue;

void setup()
{
  Serial.begin(9600); // sets the serial port to 9600
}

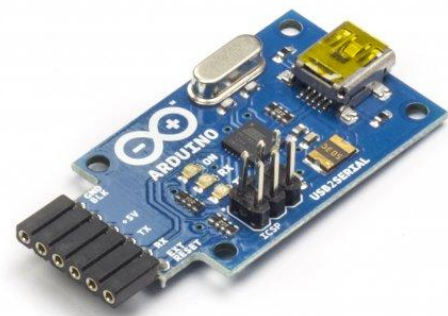
void loop()
{
  sensorValue = analogRead(0);           // read analog input pin 0
  Serial.println(sensorValue, DEC);      // prints the value read
  delay(100);                            // wait 100ms for next reading
}
```

3.8 Μετατροπέας USB2SERIAL

Λόγω του ότι η πλακέτα *Arduino Ethernet* που χρησιμοποιήσαμε για αυτό το εγχείρημα δεν διαθέτει εγκαταστημένη είσοδο USB, έπρεπε να συνδέσουμε έναν μετατροπέα που θα καθιστούσε εφικτό τον προγραμματισμό του μικροελεγκτή μέσω ενός ηλεκτρονικού υπολογιστή.

Ο μετατροπέας της επιλογής μας είναι ο *USB 2 Serial Converter (USB Mini)* και έχει την δυνατότητα μετατροπής μίας σύνδεσης USB σε 5 volts TX και RX έτσι ώστε να συνδεθεί απευθείας σε έναν *Arduino Ethernet* ή σε οποιαδήποτε άλλη πλακέτα μέσω μιας συμβατής υποδοχής FTDI καλωδίου.

Διαθέτει έναν επεξεργαστή Atmega8U2 προγραμματισμένο ως USB-to-serial μετατροπέα, πρόκειται για τον ίδιο μικροεπεξεργαστή που βρίσκεται και στον *Arduino Uno*. Η σειριακή μονάδα USB διαθέτει μία ISCP διεπαφή. Τέλος τα pinouts



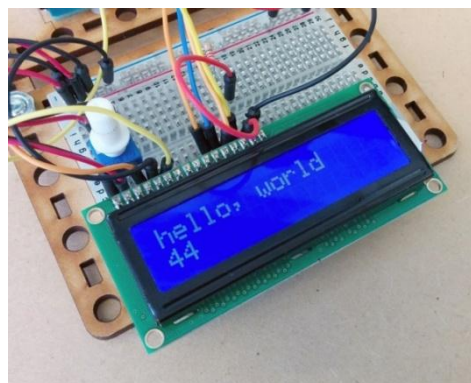
Εικόνα 3.10: Μετατροπέας USB2SERIAL (USB Mini)

του συνδέσμου είναι συμβατά με ένα απλό FTDI header (όπως επίσης και οι σειριακοί προσαρμογείς Adafruit και Sparkfun).

3.9 LCD Οθόνη

Η οθόνη που επιλέξαμε είναι LCD 16x2 χαρακτήρων και συμπεριλαμβάνεται στο Arduino Starter Kit, μπορεί ωστόσο να προμηθευτεί και ξεχωριστά.

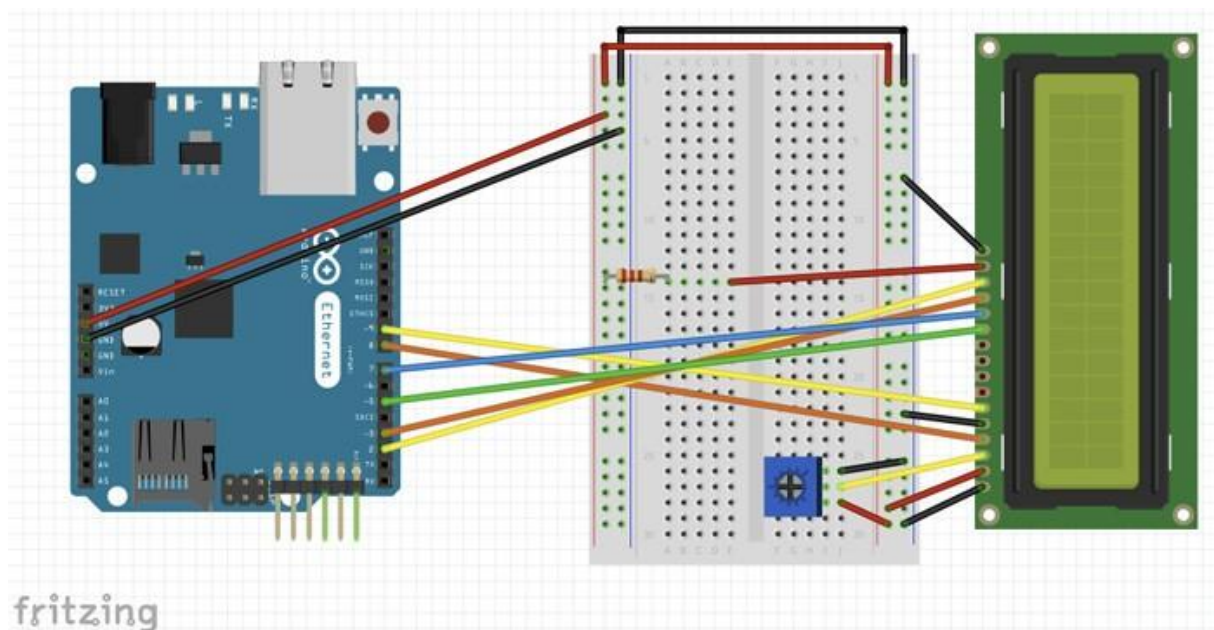
Αφού αποσυνδέσουμε το Arduino Ethernet Board από το ρεύμα, συνδέουμε την LCD σύμφωνα με το παρακάτω διάγραμμα:



Εικόνα 3.11: Οθόνη LCD 16x2 Χαρακτήρων

LCD Pin	Συνδεδεμένο με:
1	Arduino GND
2	Arduino 5V
3	Ποτενσιόμετρο 10 kΩ
4	Arduino pin 8
5	Arduino GND
6	Arduino pin 9
7	-
8	-
9	-
10	-
11	Arduino pin 5
12	Arduino pin 7
13	Arduino pin 3
14	Arduino pin 2
15	Αντίσταση 220 Ω
16	Arduino GND

Πίνακας 2: Σύνδεση Οθόνης LCD στον Arduino



Εικόνα 3.12: Διάγραμμα Σύνδεσης Arduino Ethernet με την LCD Οθόνη

3.10 Ηλεκτρονόμος (Relay)

Ο ηλεκτρονόμος ή αλλιώς «ρελέ» είναι ένας ηλεκτρικός διακόπτης που ανοίγει και κλείνει ένα ηλεκτρικό κύκλωμα κάτω από τον έλεγχο ενός άλλου ηλεκτρικού/ηλεκτρονικού κυκλώματος. Αυτό το 'άλλο' κύκλωμα στην προκειμένη περίπτωση είναι το Arduino του συστήματός μας.

Η αρχή λειτουργίας του ηλεκτρονόμου είναι η εξής:

Όταν ηλεκτρικό ρεύμα διαρρέει το πηνίο του ηλεκτρονόμου, το παραγόμενο μαγνητικό πεδίο έλκει έναν σπλισμό που είναι μηχανικά συνδεδεμένος σε μια κινούμενη επαφή. Έτσι, η κινούμενη επαφή είτε συνδέεται με μια σταθερή επαφή είτε αποσυνδέεται από αυτή.

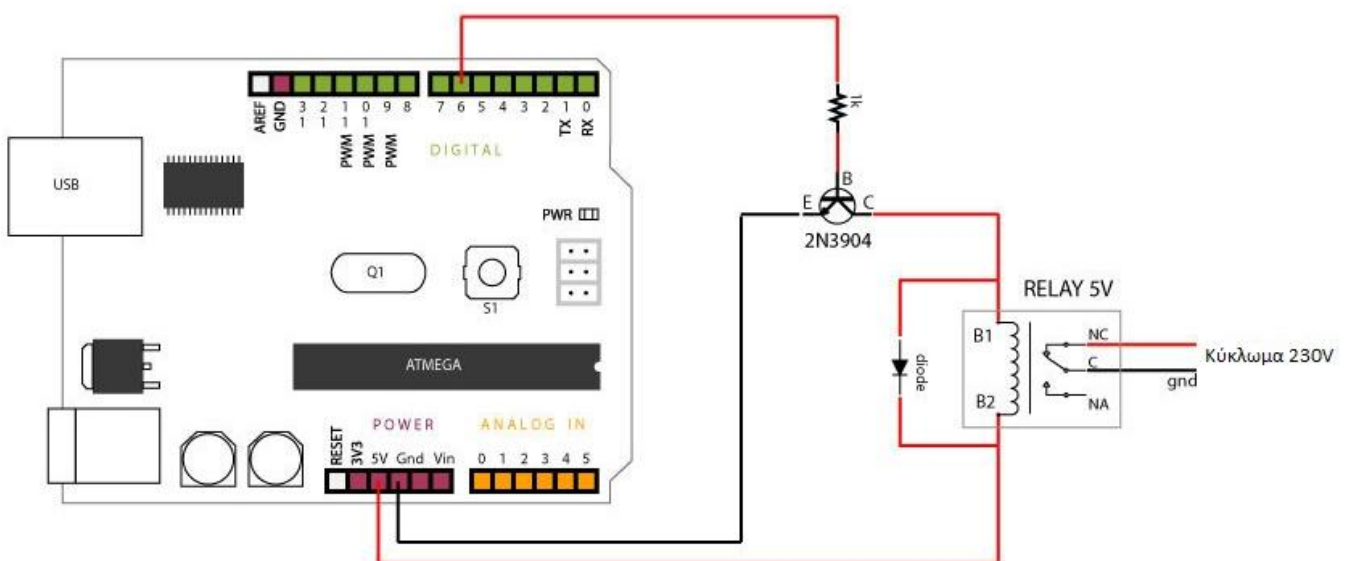


Εικόνα 3.13: Arduino 2-Relay Module

Μόλις το ηλεκτρικό ρεύμα στο πηνίο διακοπεί, ο σπλισμός επιστέφει στη θέση ηρεμίας του εξαιτίας μιας δύναμης επαναφοράς, που είναι ίση με το ήμισυ της μαγνητικής.

Μία επαφή ενός ρελέ μπορεί να είναι Κανονικά-Ανοικτή (Normally Open, NO), Κανονικά-Κλειστή (Normally Closed, NC) ή μεταγωγικός (change-over), ανάλογα με τον τύπο της.

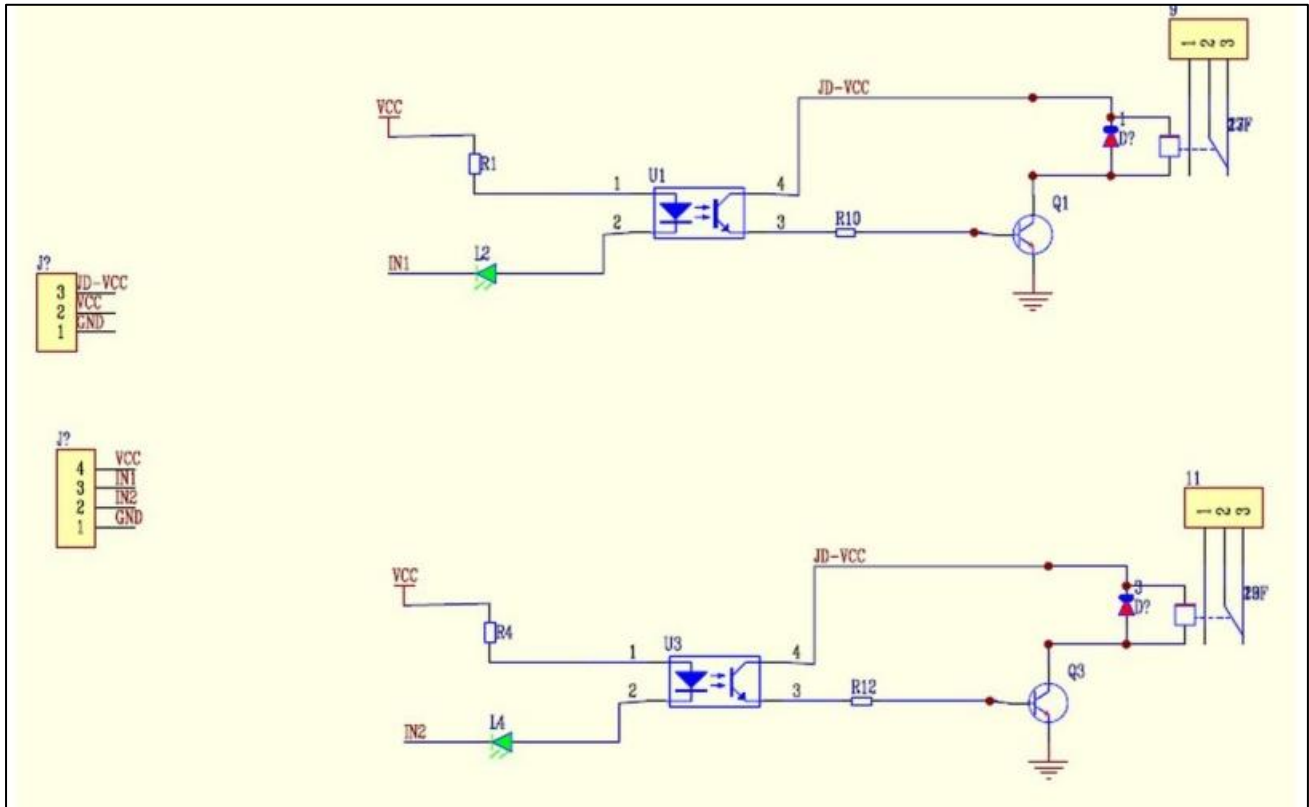
- Μια επαφή NO κλείνει το κύκλωμα όταν το ρελέ ενεργοποιείται
- Μια επαφή NC ανοίγει το κύκλωμα όταν το ρελέ ενεργοποιείται
- Μια επαφή Change-Over μπορεί να ελέγχει δύο κυκλώματα. Ισοδυναμεί με μια επαφή κανονικά-ανοικτή και μια επαφή κανονικά-κλειστή που έχουν ένα κοινό ακροδέκτη και το αποτέλεσμα της αλλάζει όταν το ρελέ ενεργοποιείται.



Εικόνα 3.14: Παράδειγμα Σύνδεσης του Arduino με Relay για τον Έλεγχο Κυκλώματος 230V

Η χρήση του ηλεκτρονόμου είναι πολύ χρήσιμη σε περιπτώσεις που η τάση του κυκλώματος που θέλουμε να ελέγξουμε είναι μεγαλύτερη από την επιτρεπτή τάση λειτουργίας του Arduino.

Η ύπαρξη του ηλεκτρονόμου λοιπόν μπορεί να μας επιτρέψει το έλεγχο οικιακών κυκλωμάτων τάσης 230V AC χωρίς κανένα πρόβλημα για τον μικροελεγκτή παρά το ότι η τάση λειτουργίας του είναι 5V DC.



Εικόνα 3.15: Σχηματικό Διάγραμμα του Arduino 2-Relay Module

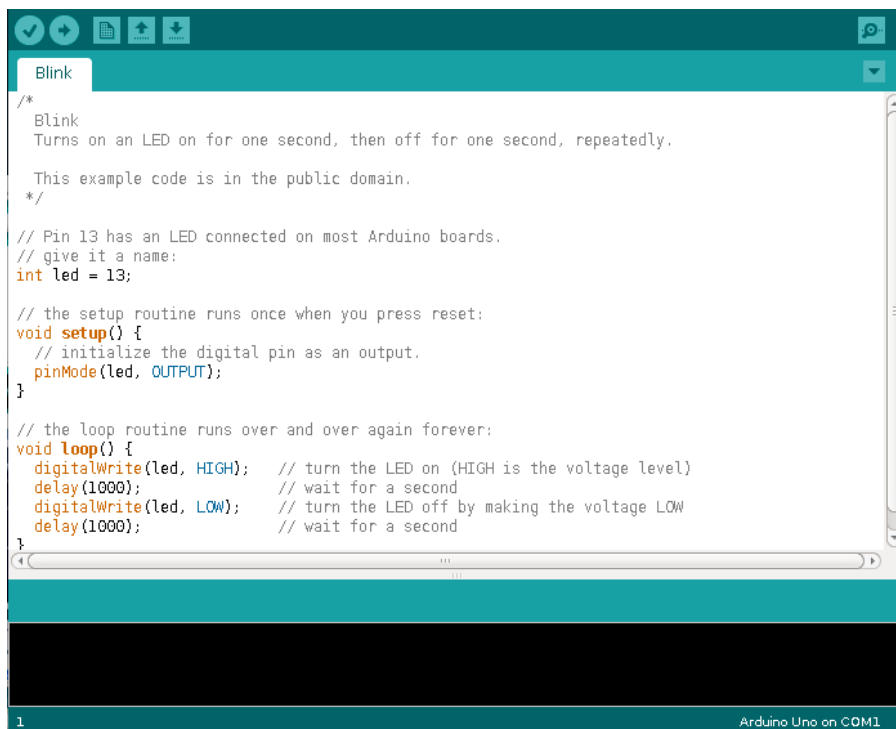
Κεφάλαιο 4. Το Προτεινόμενο Σύστημα – Η Πλατφόρμα Διαχείρισης

4.1 Λογισμικό – Arduino IDE

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.

Έχει σχεδιαστεί με τέτοιο τρόπο για να εισαγάγει τον προγραμματισμό σε αυτούς που δεν είναι εξοικειωμένοι με την ανάπτυξη λογισμικού.

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. Ένα



```
Blink
/*
 * Blink
 * Turns on an LED on for one second, then off for one second, repeatedly.
 *
 * This example code is in the public domain.
 */
// Pin 13 has an LED connected on most Arduino boards.
// give it a name:
int led = 13;

// the setup routine runs once when you press reset:
void setup() {
  // initialize the digital pin as an output.
  pinMode(led, OUTPUT);
}

// the loop routine runs over and over again forever:
void loop() {
  digitalWrite(led, HIGH); // turn the LED on (HIGH is the voltage level)
  delay(1000);             // wait for a second
  digitalWrite(led, LOW);  // turn the LED off by making the voltage LOW
  delay(1000);             // wait for a second
}
1 Arduino Uno on COM1
```

Εικόνα 4.1: Το Λογισμικό Arduino IDE

πρόγραμμα (κώδικας) που γράφτηκε για Arduino ονομάζεται σκίτσο (sketch).

Τα Arduino προγράμματα είναι γραμμένα σε C ή C++. Το Arduino IDE έρχεται με μια βιβλιοθήκη λογισμικού που ονομάζεται "Wiring" από το πρωτότυπο σχέδιο Wiring γεγονός που καθιστά πολλές κοινές λειτουργίες εισόδου/εξόδου πολύ πιο εύκολες.

Οι χρήστες πρέπει μόνο να ορίσουν δύο λειτουργίες για να κάνουν ένα πρόγραμμα κυκλικής εκτέλεσης:

- **Void setup():** Μία συνάρτηση που τρέχει μία φορά στην αρχή του προγράμματος η οποία αρχικοποιεί τις ρυθμίσεις
- **Void loop():** Μία συνάρτηση η οποία καλείται συνέχεια μέχρι η πλακέτα να απενεργοποιηθεί

Ένα τυπικό πρώτο πρόγραμμα για έναν μικροελεγκτή αναβοσβήνει απλά ένα LED. Στο περιβάλλον του Arduino ο χρήστης μπορεί να γράψει ένα πρόγραμμα σαν αυτό:

```
#define LED_PIN 13

void setup () {
  pinMode (LED_PIN, OUTPUT); // enable pin 13 for digital output
}

void loop () {
  digitalWrite (LED_PIN, HIGH); // turn on the LED
  delay (1000); // wait one second (1000 milliseconds)
  digitalWrite (LED_PIN, LOW); // turn off the LED
  delay (1000); // wait one second
}
```

Εικόνα 4.2: Παράδειγμα Κώδικα για τον Έλεγχο ενός LED στο Arduino IDE

Είναι ένα χαρακτηριστικό των περισσότερων πλακετών Arduino ότι έχουν ένα LED και μία αντίσταση φορτίου που συνδέονται μεταξύ του pin 13 (ή pin 9 για Arduino Ethernet) και του εδάφους, ένα βολικό χαρακτηριστικό για πολλά απλά τεστ.

Ο προηγούμενος κώδικας δεν θα αναγνωριστεί από ένα κανονικό μεταγλωττιστή C++ ως έγκυρο πρόγραμμα, έτσι ώστε όταν ο χρήστης κάνει κλικ στο κουμπί "Upload to I / O board" στο IDE, ένα αντίγραφο του κώδικα θα γραφτεί σε ένα προσωρινό αρχείο με ένα παραπάνω include στην κορυφή και μία πολύ απλή συνάρτηση main() στο τέλος, για να φτιάξει ένα έγκυρο C++ πρόγραμμα. Το IDE του Arduino χρησιμοποιεί το GNU toolchain και το AVR Libc για να μεταγλωττίζει προγράμματα και το avrdude για να φορτώνει προγράμματα στην πλακέτα.

Δεδομένου ότι η πλατφόρμα Arduino χρησιμοποιεί Atmel μικροελεγκτές, το

περιβάλλον ανάπτυξης της Atmel, το AVR Studio ή το νεότερη έκδοση του Atmel Studio, μπορεί επίσης να χρησιμοποιηθεί για την ανάπτυξη λογισμικού για το Arduino.

4.2 Σύνοψη της Μεθόδου Λήψης των Μετρήσεων και Υπολογισμού Δεδομένων

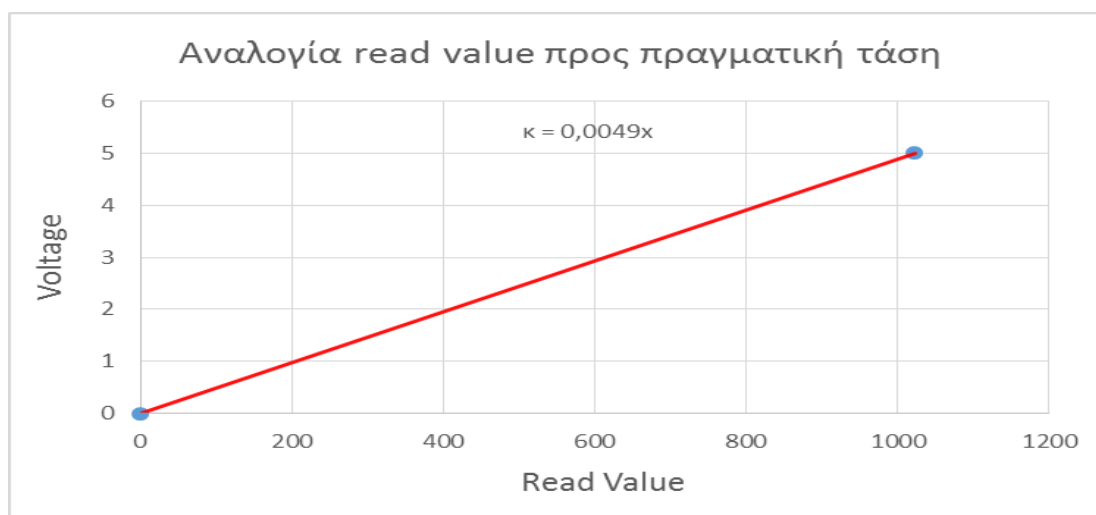
Για τον υπολογισμό της τιμής τάσης του συσσωρευτή δεν είναι απαραίτητη η χρήση κάποιου αισθητήρα καθώς το Arduino έχει ενσωματωμένες αναλογικές υποδοχές οι οποίες λαμβάνουν τιμές 0-1023 ανάλογα την πραγματική τιμή τάσης.

Οι τιμές αυτές αντιστοιχούν σε 0 Volts η τιμή 0 και σε 5 Volts η τιμή 1023 αντίστοιχα. Η σύνδεση προϋποθέτει την δημιουργία ενός κυκλώματος διαιρέτη τάσης (Voltage Divider) δηλαδή μίας απλής κυκλωματικής διάταξης η οποία αποτελείται από δύο αντιστάτες συνδεδεμένους εν σειρά, στα άκρα των οποίων εφαρμόζεται η τάση εισόδου. Ως τάση εξόδου λαμβάνεται η διαφορά δυναμικού ανάμεσα στους ακροδέκτες της μίας εκ των δύο αντιστάσεων.

Οι τιμές που μπορεί να πάρει η τάση εξόδου κυμαίνονται από το 0 έως την τάση εισόδου. Ο υπολογισμός της τάσης εξόδου βρίσκεται εύκολα με τον εξής τύπο:

$$V_{out} = \frac{R_2}{R_1 + R_2} \cdot V_{in}$$

Για να υπολογίσουμε την πραγματική τιμή τάσης αρκεί να φτιάξουμε την παρακάτω



Εικόνα 4.3: Γραφική Παράσταση Read Value προς Πραγματική Τάση

γραφική παράσταση στην οποία αφού ενώσουμε τα δύο γνωστά σημεία μεταξύ τους (0,0 και 1023,5) στη συνέχεια υπολογίζουμε την κλίση της ευθείας που δημιουργείται.

Ο υπολογισμός της κλίσης γίνεται ως εξής:

$$κ = \frac{y_2 - y_1}{x_2 - x_1} = \frac{(5 - 0)}{(1023 - 0)} = \frac{5}{1023} = 0,0049$$

Γνωρίζοντας πλέον την κλίση της ευθείας στην ουσία έχουμε υπολογίσει πόσα Volts αντιστοιχούν σε κάθε μία τιμή του αναλογικού pin δηλαδή ξέρουμε πως 1 τιμή αντιστοιχεί σε 0,0049V.

Άρα για να υπολογίζουμε την πραγματική τάση που επιθυμούμε χρησιμοποιούμε τον εξής τύπο:

Πραγματική τιμή τάσης = (0,0049V • 3 • Τιμή αναλογικού pin)

Ο λόγος που πολλαπλασιάζουμε επί 3 είναι η ύπαρξη του διαιρέτη τάσης ο οποίος «κόβει» στα 3 την πραγματική τάση.

Για τον υπολογισμό της έντασης του ρεύματος χρησιμοποιούμε όπως αναφέρεται και παραπάνω τον αισθητήρα ρεύματος *ACS712 SparkFun Hall Effect* τον οποίο έχουμε συνδέσει σε σειρά μεταξύ του συσσωρευτή μας και του φορτίου.

Λόγω της ύπαρξης έντονου «θορύβου» στην αναλογική έξοδο του αισθητήρα δεν λαμβάνουμε υπόψη μας μόνο μια τιμή τη φορά αλλά παίρνουμε 1000 μετρήσεις ανά δευτερόλεπτο και στη συνέχεια υπολογίζουμε την μέση τιμή των μετρήσεων αυτών.

Όπως ξέρουμε η ισχύς είναι προϊόν της τάσης επί την ένταση του ρεύματος άρα για να την υπολογίσουμε χρησιμοποιούμε τον παρακάτω τύπο:

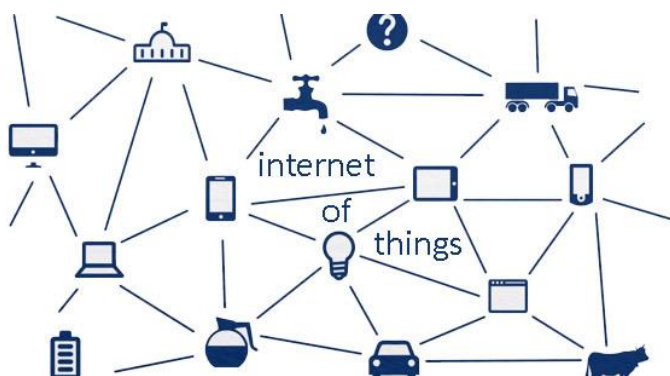
$$P = V \cdot I$$

Τέλος αφού υπολογιστεί η ισχύς μπορούμε εύκολα να υπολογίσουμε το κόστος της ηλεκτρικής ενέργειας ανά ώρα καθώς και το ρυθμό εκπομπών CO₂ που αντιστοιχεί στην κατανάλωσή μας.

Ο υπολογισμός του κόστους της ενέργειας ανα ώρα υπολογίζεται μέσω του πολλαπλασιασμού της ισχύος με την τιμή της watt-ώρας (Wh) σε ευρώ και ο υπολογισμός του ρυθμού εκπομπών CO₂ αντίστοιχα μέσω του πολλαπλασιασμού της ισχύος επί τη μάζα CO₂ που εκλύεται ανά watt-ώρα (Wh) κατά την παραγωγή της ηλεκτρικής ενέργειας.

4.3 Σύνδεση “Internet of Things”

Με τον όρο internet of things εννοούμε την διασύνδεση συσκευών και αισθητήρων με δυνατότητα πρόσβασης στο internet με σκοπό την πραγματοποίηση μετρήσεων και συλλογή δεδομένων τα οποία στην συνέχεια θα επεξεργαστούν και θα χρησιμοποιηθούν άμεσα για ενημέρωση ενός ανθρώπου μίας επιχείρησης ή ενός άλλου μηχανήματος για την εκτέλεση κάποιας συγκεκριμένης λειτουργίας.



Εικόνα 4.4: Έως το 2020 εκτιμάται πως θα υπάρχουν περισσότερες από 50 εκατομμύρια συνδεδεμένες συσκευές

Χρησιμοποιείται ευρέως στην ιατρική στις μεταφορές και στις επικοινωνίες. Στο σύντομο μέλλον όλο και περισσότερες συσκευές θα συνδέονται μεταξύ τους με χαμηλό κόστος και θα σε βοηθούν στην καθημερινότητά μας χωρίς πολλές φορές να το γνωρίζουμε.

Κάθε «πράγμα» είναι μοναδικά αναγνωρίσιμο μέσω ενσωματωμένων υπολογιστικών συστημάτων της, αλλά είναι σε θέση να λειτουργεί στο πλαίσιο της υφιστάμενης υποδομής του Διαδικτύου.

Οι ειδικοί εκτιμούν ότι το Ίντερνετ των πραγμάτων θα αποτελείται από περίπου 50 δισεκατομμύρια αντικείμενα από το 2020

4.4 Η Πλατφόρμα Thingspeak.com

Η σελίδα που χρησιμοποιήσαμε για το ανέβασμα των δεδομένων και στην συνέχεια την οπτικοποίηση τους είναι το Thingspeak.com.



Εικόνα 4.5: Λογότυπο ThingSpeak

Το ThingSpeak είναι μια εφαρμογή IoT ανοιχτού κώδικα και API η οποία επιτρέπει την αποθήκευση και την ανάκτηση δεδομένων από διάφορες συσκευές ή “things” μέσω του internet.

Μας δίνει την δυνατότητα να έχουμε πρόσβαση σε γραφήματα της ενεργειακής μας κατανάλωσης σε πραγματικό χρόνο, από οποιοδήποτε σημείο του κόσμου μέσω του διαδικτύου.

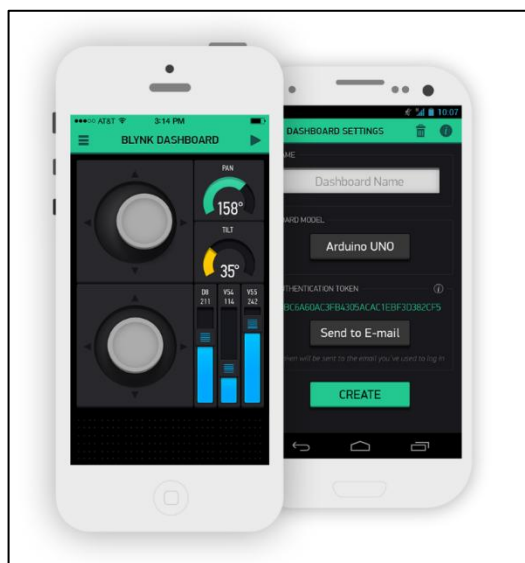
4.5 Η Εφαρμογή Blynk

Ένα έξυπνο σύστημα μέτρησης και διαχείρισης της ηλεκτρικής ενέργειας δεν θα μπορούσε να είναι ολοκληρωμένο αν δεν μας δινόταν η δυνατότητα του απομακρυσμένου ελέγχου της

κατανάλωσής μας. Η εφαρμογή που επιλέξαμε για το συγκεκριμένο σκοπό είναι το *Blynk*.

Το Blynk είναι μια πλατφόρμα συμβατή με εφαρμογές Android καθώς και IOS που προσφέρει την δυνατότητα ελέγχου του Arduino, Raspberry Pi καθώς και άλλως μικροελεγκτών μέσω του διαδικτύου.

Αποτελείται από ένα ψηφιακό ταμπλό, όπου ο χρήστης μπορεί να χτίσει ένα γραφικό περιβάλλον για το εγχείρημά του το οποίο μπορεί να περιλαμβάνει διάφορα widgets όπως κουμπιά, γραφικές παραστάσεις, sliders κ.λπ και στη συνέχεια να ορίσει την λειτουργία των widgets αυτών.

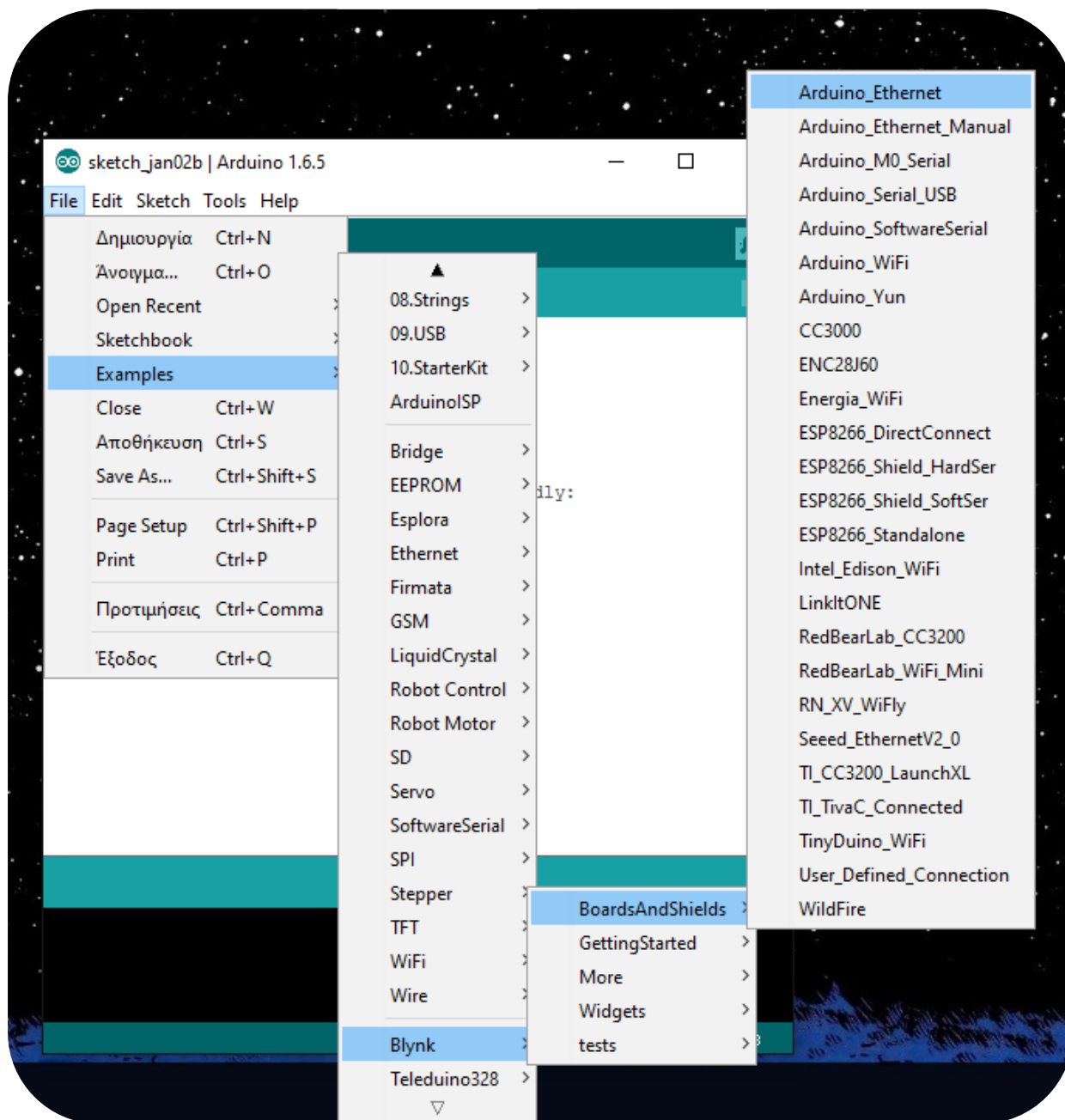


Εικόνα 4.6: Το Περιβάλλον της Εφαρμογής Blynk

Το Blynk δεν είναι συμβατό μόνο με συγκεκριμένες πλακέτες μικροελεγκτών ή shields αλλά παρέχει υποστήριξη για τα hardware της επιλογής του χρήστη. Αυτό σημαίνει πως είτε ο μικροελεγκτής μας είναι συνδεδεμένος στο internet μέσω Ethernet Shield ή Wi-fi module κ.ο.κ το blynk μπορεί να συνδεθεί.

Για να χρησιμοποιήσουμε το Blynk πρέπει πρώτα να εγκαταστήσουμε στην Android ή IOS συσκευή μας την εφαρμογή η οποία υπάρχει δωρεάν στα app stores. Στη συνέχεια πρέπει να κατεβάσουμε την βιβλιοθήκη για το Arduino IDE η οποία διατίθεται επίσης δωρεάν στο επίσημο site <http://www.blynk.cc>.

Αφού εγκαταστήσουμε την βιβλιοθήκη στο Arduino IDE τότε κάνουμε κλικ στο κεντρικό μενού: File>Examples>Blynk>BoardsAndShields και στη συνέχεια επιλέγουμε την πλακέτα που χρησιμοποιούμε, στην προκειμένη περίπτωση Arduino_Ethernet.



Εικόνα 4.7: Βιβλιοθήκη Blynk

Ο κώδικας που θα εμφανιστεί είναι ο εξής:

```
#define BLYNK_PRINT Serial // Comment this out to disable prints and save space
#include <SPI.h>
#include <Ethernet.h>
#include <BlynkSimpleEthernet.h>

// You should get Auth Token in the Blynk App.
// Go to the Project Settings (nut icon).
char auth[] = "5e719fe133bf44b5852a8d8819678814";

void setup()
{
  Serial.begin(9600);
  Blynk.begin(auth);
  // You can also specify server.
  // For more options, see BoardsAndShields/Arduino_Ethernet_Manual example
  //Blynk.begin(auth, "your_server.com", 8442);
  //Blynk.begin(auth, IPAddress(192,168,1,100), 8888);
}

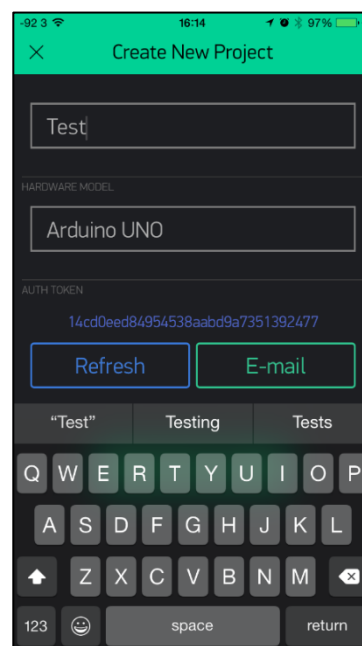
void loop()
{
  Blynk.run();
}
```

Εικόνα 4.8: Κώδικας Blynk

*Ο προσωπικός κωδικός auth token μας αποστέλλεται στη διεύθυνση email που ορίσαμε στην εφαρμογή και είναι μοναδικός για κάθε εγχείρημα.

Αφού λάβουμε το email αντιγράφουμε το auth token και το επικολλούμε στο Arduino Sketch στο σημείο του κώδικα του παραδείγματος που εμφανίζεται η εντολή: `char auth [] = ""`

Έπειτα, επιστρέφοντας στην εφαρμογή Blynk μπορούμε πλέον να δημιουργήσουμε ένα γραφικό περιβάλλον με τα widgets που χρειαζόμαστε και στη συνέχεια να επεξεργαστούμε τη λειτουργία τους ορίζοντας τα pins του Arduino που θα χρησιμοποιούνται ως είσοδος και έξοδος.



Εικόνα 4.9: Μοναδικός Κωδικός «auth token»

Όταν πλέον ολοκληρώσουμε τη διαδικασία τότε πατώντας το κουμπί *play* στο πάνω δεξιά μέρος της εφαρμογής πραγματοποιείται η σύνδεση μεταξύ της εφαρμογής και του Arduino και μπορούμε να ελέγξουμε την συσκευή μας μέσω του διαδικτύου.

4.6 Ο Κώδικας του Έξυπνου Μετρητή Ηλεκτρικής Ενέργειας

Παρακάτω παρατίθεται το sketch με το οποίο προγραμματίστηκε ο Arduino του εγχειρήματος:

```
int readValue;
float watt;
float energy_cost_perhour;
float average;
double Voltage;
float totcost;
float avgamps;
float gCO2_per_hour;

#include <LiquidCrystal.h> //Συμπεριλαμβάνει την library της LCD module για
ευκολότερη χρήση.

LiquidCrystal lcd(8, 9, 5, 7, 3, 2); //Προσδιορισμός pins στα οποία είναι συνδεδεμένη
η οθόνη LCD.

#include <SPI.h> //Συμπεριλαμβάνει την library της SPI.

#include <Ethernet.h>

// Local Network Settings

byte mac[] = { 0x2C, 0x33, 0x7A, 0xF9, 0x84, 0x3F }; // Διεύθυνση MAC
IPAddress ip(147, 95, 134, 160);
IPAddress myDns(194, 177, 198, 2);
IPAddress gateway(147, 95, 134, 254);
IPAddress subnet(255, 255, 255, 0);
```

```
// ThingSpeak Settings

char thingSpeakAddress[] = "api.thingspeak.com";
String writeAPIKey = "P7JL3I0NNDBGPYKQ";

const int updateThingSpeakInterval = 15 * 1000; // Time interval in milliseconds to
update ThingSpeak (number of seconds * 1000 = interval)

// Variable Setup
long lastConnectionTime = 0;
boolean lastConnected = false;
int failedCounter = 0;

// Εκκίνηση Arduino Ethernet Client
EthernetClient client;

//Συμπεριλαμβάνει την library του Ethernet.

#include <Dhcp.h>
#include <Dns.h>
#include <EthernetClient.h>
#include <EthernetServer.h>
#include <EthernetUdp.h>

#define BLYNK_PRINT Serial // Comment this out to disable prints and save space
#include <SPI.h>
#include <Ethernet.h>
#include <BlynkSimpleEthernet.h>

IPAddress server_ip (147, 95, 134, 161);

char auth[] = "5e719fe133bf44b5852a8d8819678814";
```

```
void setup() {  
  
  lcd.begin(16, 2);  
  Serial.begin(9600); // sets the serial port to 9600.  
  startEthernet(); // Start Ethernet on Arduino  
  
  Serial.begin(9600);  
  
  Blynk.begin(auth, "cloud.blynk.cc", 8442, ip, myDns, gateway, subnet, mac);  
}  
  
void loop() {  
  
  Blynk.run();  
  
  long milisec = millis(); // Υπολογισμός χρόνου σε milliseconds.  
  
  long time = milisec / 1000; // Μετατροπή των milliseconds σε seconds.  
  
  readValue = analogRead(A0);  
  
  Voltage = (0.0049 * 3.2 * readValue);  
  
  //ACS712 Αισθητήρας ρεύματος ,Παίρνει 1000 μετρήσεις απο τον αισθητήρα με  
  διαφορά 1milisec και βγάζει το μέσο όρο.  
  
  float average = 0;  
  
  for (int i = 0; i < 1000; i++) {  
  
    average = average + (0.0264 * analogRead(A1) - 13.54) / 1000;  
  
  }  
}
```

```
float watt = 0;

for (int i = 0; i < 60; i++) {
    watt = Voltage * average;
} // power=voltage*current

delay (1000);

//energy = energy+(watt * time) / 3600; //Τα Watt-sec μετατρέπονται σε Watt-Hr
διαιρώντας με 1hr(3600sec).

energy_cost_perhour = watt * 0.35; //Υπολογισμός κόστους ενέργειας που
καταναλώθηκε

gCO2_per_hour = 1.12 * watt; //Υπολογισμός της ποσότητας CO2 που εκλείεται αν
ώρα

// Εκτύπωση των τιμών στη σειριακή οθόνη.

Serial.print("VOLTS : ");
Serial.println(Voltage);
Serial.print("AMPERES :");
Serial.println(average);
Serial.print("WATTS :");
Serial.println(watt);
Serial.print("EUROS/HOUR :");

Serial.println(energy_cost_perhour);
Serial.print("GRAMS OF CO2 / HOUR :");
Serial.println(gCO2_per_hour);

//Εμφάνιση στην οθόνη LCD.
```

```
lcd.setCursor(1, 0);
lcd.print(Voltage);
lcd.print("V");

lcd.setCursor(9, 0);
lcd.print(average);
lcd.print("Amp");

lcd.setCursor(1, 1); // Τοποθέτηση κέρσορα στην 1η στήλη και 1η γραμμή

lcd.print(watt);
lcd.print("W ");

lcd.setCursor(8, 1);
lcd.print(energy_cost_perhour);
lcd.print("e/Hr");

//-----Φόρτωση δεδομένων στο Thingspeak.com-----

// Read value from Analog Input Pin 0 and 1
String analogValue0 = String(Voltage, DEC);
String analogValue1 = String(average, DEC);

// Εκτύπωση του Update Response στη σειριακή οθόνη
if (client.available())
{
  char c = client.read();

  Serial.print(c);
}
```

```
// Αποσύνδεση από το ThingSpeak

if (!client.connected() || lastConnected)
{
  Serial.println("...disconnected");

  Serial.println();

  client.stop();

  delay(200);
}

// Update ThingSpeak

if (!client.connected() && (millis() - lastConnectionTime >
updateThingSpeakInterval))
{
  updateThingSpeak ("field1=" + analogValue0 + "&field2=" + analogValue1 +
"&field3=" + watt + "&field4=" + energy_cost_perhour + "&field5=" +
gCO2_per_hour);
  delay(15000); }

// Ελέγχει αν το Arduino Ethernet χρειάζεται επανεκκίνηση

if (failedCounter > 3 ) {

  startEthernet();

}

lastConnected = client.connected();

}

void updateThingSpeak(String tsData)
```

```
{  
  if (client.connect(thingSpeakAddress, 80))  
  {  
    client.print("POST /update HTTP/1.1\n");  
    client.print("Host: api.thingspeak.com\n");  
    client.print("Connection: close\n");  
    client.print("X-THINGSPEAKAPIKEY: " + writeAPIKey + "\n");  
    client.print("Content-Type: application/x-www-form-urlencoded\n");  
    client.print("Content-Length: ");  
    client.print(tsData.length());  
    client.print("\n\n");  
    client.print(tsData);  
    lastConnectionTime = millis();  
  }  
  if (client.connected())  
  {  
    Serial.println("Connecting to ThingSpeak...");  
    Serial.println();  
    failedCounter = 0;  
  }  
  else  
  {  
    failedCounter++;  
    Serial.println("Connection to ThingSpeak failed (" + String(failedCounter, DEC) +  
    ")");  
    Serial.println();  
  }  
}
```

```
else
{
    failedCounter++;

    Serial.println("Connection to ThingSpeak Failed (" + String(failedCounter, DEC) +
    "");

    Serial.println();

    lastConnectionTime = millis();
}
}

void startEthernet()
{

    client.stop();

    Serial.println("Connecting Arduino to network...");

    Serial.println();

    delay(100);

    Ethernet.begin(mac, ip, myDns, gateway, subnet);

    Serial.println("Arduino connected to network using DHCP");

    Serial.println();

    delay(100);
}
```


Κεφάλαιο 5 – Συμπεράσματα και Προτάσεις για το Μέλλον

5.1 Επαλήθευση Ορθής Λειτουργίας του Συστήματος

Η επαλήθευση της ορθής λειτουργίας του συστήματος που δημιουργήσαμε έγινε μέσω της χρήσης ενός ηλεκτρονικού πολυμέτρου. [8]

Το πολύμετρο είναι ένα φορητό ηλεκτρονικό όργανο το οποίο παρέχει τη δυνατότητα μέτρησης μεταξύ άλλων της τάσης, της έντασης και της αντίστασης σ' ένα ηλεκτρικό κύκλωμα.

Είναι μια εξαιρετικά πρακτική συσκευή που μπορεί να χρησιμοποιηθεί για να ανιχνευτούν ηλεκτρικά προβλήματα σε οικιακές συσκευές, στα ηλεκτρονικά συστήματα ενός αυτοκινήτου κ.α.



Εικόνα 5.1: Ηλεκτρονικό Πολύμετρο

Για τη μέτρηση της τάσης συνδέουμε τους ακροδέκτες παράλληλα στο κύκλωμα ενώ για τη μέτρηση της έντασης του ρεύματος η σύνδεση γίνεται σε σειρά. Η μέτρηση της τάσης του κυκλώματος από το σύστημα του Arduino που κατασκευάσαμε έχει μια μικρή απόκλιση από την πραγματική τιμή τάσης λόγω του περιορισμού ευαισθησίας των αναλογικών pins του Arduino.

Η μέτρηση της έντασης του ρεύματος μέσω του αισθητήρα ACS712 συμπίπτει απόλυτα με την πραγματική ένταση.

5.2 Ενσωμάτωση στο πρωτόκολλο KNX

Η τεχνολογία KNX επιτρέπει τη συγκέντρωση σε ένα ολοκληρωμένο περιβάλλον ενός ευρέως φάσματος λειτουργιών ενός κτιρίου, όπως ο έλεγχος του φωτισμού, των ηλεκτρικών ρολών, της θέρμανσης, του κλιματισμού καθώς και της διαχείρισης φορτίων και καταναλώσεων.



Εικόνα 5.2: Λογότυπο Πρωτοκόλλου KNX

Αυτό επιτυγχάνεται μέσω της διασύνδεση των ηλεκτρολογικών λειτουργιών ενός κτιρίου σε ένα ενιαίο δίκτυο BUS και στη συνέχεια τον προγραμματισμό του μέσα από ένα εξειδικευμένο λογισμικό.

Το KNX έχει σχεδιαστεί έτσι ώστε να είναι ανεξάρτητη από οποιαδήποτε συγκεκριμένη πλατφόρμα υλικού. Ένα Δίκτυο Συσκευών KNX μπορεί να ελεγχθεί από οτιδήποτε, από έναν μικροελεγκτή 8-bit μέχρι έναν υπολογιστή, ανάλογα με τις ανάγκες της εκάστοτε εφαρμογής.

5.3 Συμπεράσματα και Προτάσεις για το Μέλλον

Ο κύριος στόχος της πτυχιακής εργασίας ήταν απο την αρχή η δημιουργία ενός συστήματος που θα μπορούσε πραγματικά να φανεί χρήσιμο στον χρήστη και να έχει πρακτική εφαρμογή στην καθημερινότητά του.

Παράλληλα, η ενημέρωση που παρέχεται απο το σύστημα όσον αφορά στο ρυθμό εκπομπής CO₂ μπορεί να συμβάλει στην ευαισθητοποίησή του χρήστη έτσι ώστε να διατηρεί την ενεργειακή του κατανάλωση στα απαραίτητα για τις ανάγκες του επίπεδα και να αποφεύγει την άσκοπη χρήση ηλεκτρικής ενέργειας.

Το σύστημα μέτρησης και διαχείρισης ηλεκτρικής ενέργειας που υλοποιήσαμε στο πλαίσιο της παρούσας πτυχιακής εργασίας κατασκευάστηκε πειραματικά, δηλαδή με τέτοιο τρόπο ώστε να μπορεί να χρησιμοποιηθεί μόνο για κυκλώματα χαμηλής τάσης για λόγους ασφαλείας. Η βασική αρχή λειτουργίας του όμως είναι ακριβώς η ίδια με ένα σύστημα που μπορεί να χρησιμοποιηθεί σε ένα κύκλωμα οικιακής τάσης 230 Volt AC.

Το σύστημα που κατασκευάσαμε μπορεί αρκετά εύκολα στο μέλλον να χρησιμοποιηθεί για τη μέτρηση της ενεργειακής κατανάλωσης ενός σπιτιού ή μιας επιχείρησης αφού υποστεί πρώτα κάποιες μετατροπές.



Εικόνα 5.3: Μη Επεμβατικός Αισθητήρας Ρεύματος AC

Η κυριότερη μετατροπή που απαιτείται για κάτι τέτοιο είναι η αντικατάσταση του αισθητήρα ρεύματος ACS712 που χρησιμοποιούμε στο σύστημά μας, με έναν μη επεμβατικό αισθητήρα που μπορεί να εφαρμοστεί σε οποιοδήποτε καλώδιο χωρίς να χρειάζεται το κόψιμο του καλωδίου και η σύνδεση του σε σειρά.

Αναφορές

- [1] Διαθέσιμο online (2016): <http://www.metering.com/the-history-of-the-electricity-meter>
- [2] Διαθέσιμο online (2016): https://en.wikipedia.org/wiki/Electricity_meter
- [3] Διαθέσιμο online (2016): <https://www.arduino.cc/en/Main/ArduinoBoardEthernet>
- [4] Διαθέσιμο online (2016): https://en.wikipedia.org/wiki/Power_over_Ethernet
- [5] Διαθέσιμο online (2016): <https://el.wikipedia.org/wiki/Arduino>
- [6] Διαθέσιμο online (2016): <http://bit.ly/1WGOJEv>
- [7] Διαθέσιμο online (2016): <http://techblog.gr/internet/internet-of-things-for-dummies-6413/>
- [8] Διαθέσιμο online (2016): <https://en.wikipedia.org/wiki/Multimeter>
- [9] Διαθέσιμο online (2016): <https://en.wikipedia.org/wiki/Relay>
- [10] Διαθέσιμο online (2016): <http://www.blynk.cc/>
- [11] Διαθέσιμο online (2016): <http://www.epictinker.com/Non-Invasive-AC-Current-Sensor-100A-Max-p/sct-013-000.htm>
- [12] Διαθέσιμο online (2016): <http://greentitans.gweb.io/lightbulb.jpg>
- [13] Διαθέσιμο online (2016): https://en.wikipedia.org/wiki/KNX_%28standard%29
- [14] Διαθέσιμο online (2016): <http://www.techniki-ekp.gr/%CF%80%CE%B9%CF%83%CF%84%CE%BF%CF%80%CE%BF%CE%B9%CE%AE%CF%83%CE%B5%CE%B9%CF%82/knx-%CE%AD%CE%BE%CF%85%CF%80%CE%BD%CE%B1-%CE%BA%CF%84%CE%AF%CF%81%CE%B9%CE%B1/knx/>
- [15] Διαθέσιμο online (2016): <https://www.dei.gr/el/i-dei/perivallon/perivallontiki-stratigiki/klimatiki-allagi>