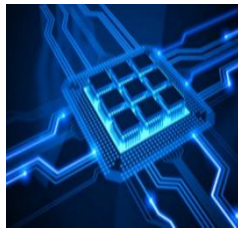


Τεχνολογικό Εκπαιδευτικό Ίδρυμα Κρήτης

Σχολή Τεχνολογικών Εφαρμογών Τ.Ε.Ι ΚΡΗΤΗΣ
Τμήμα Ηλεκτρολόγων Μηχανικών



Πτυχιακή εργασία

Τίτλος

Έλεγχος τάσης φορτίου μέσω ιστοσελίδας αναρτημένη από πλατφόρμα Arduino DUE

Σπουδαστής: Ντάκος Δημήτρης

Επιβλέπων καθηγητής: Παπαδάκης Νικόλαος

ΗΡΑΚΛΕΙΟ

2016

Περιεχόμενα

Ευχαριστίες	4
Abstract	5
Σύνοψη	5
ΚΕΦΑΛΑΙΟ 1 ^ο	6
1. ΕΙΣΑΓΩΓΗ	6
1.1 Περίληψη της πτυχιακής	6
1.2 Κίνητρο για την Διεξαγωγή της εργασίας.....	6
1.3 Δομή εργασίας.....	6
1.4 Μεθοδολογία υλοποίησης	6
ΚΕΦΑΛΑΙΟ 2 ^ο	7
2. ΘΕΩΡΙΕΣ	7
2.1 Το διαδίκτυο η ιντερνέτ.....	7
2.2 Παγκόσμιος ιστός (world wide web)	7
2.3 Ιστοσελίδα	7
2.4 Τι είναι η HTML (Hyper Text Markup Language)	7
2.5 Τι είναι CSS (Cascading style sheets-Διαδοχικά Φύλλα Στυλ)	7
2.6 Web hosting (Φιλοξενία ιστοσελίδων).....	7
2.7 Τι είναι ο server (Εξυπηρετητής)	8
2.8 Γλώσσες προγραμματισμού	8
2.8.1 Βιβλιοθήκες (υπολογιστών)	8
2.8.2 Μεταβλητές	8
2.8.3 Μικροελεγκτές	9
2.9 Τι είναι το λογισμικό (software).....	12
2.10 Βασικός έλεγχος ταυτότητας HTTP.....	12
2.11 Τι είναι JavaScript (JS).....	12
2.12 Τι είναι το Arduino.....	12
2.13 Υλικό (Hardware).....	13
2.14 Ασπίδες (Shields)	13
2.15 Ηλεκτρονόμος(relay).....	14
2.16 Λογισμικό ανάπτυξης (IDE)	14
2.17 Τεχνικά χαρακτηριστικά της πλατφόρμας	15
2.18 Θύρες Προγραμματισμού	17
Κεφάλαιο 3 ^ο	19



3. ΔΡΑΣΗ ΓΙΑ ΤΗΝ ΔΙΕΚΠΕΡΑΙΩΣΗ ΤΗΣ ΕΓΓΡΑΣΙΑΣ.....	19
3.1 Εγκατάσταση του Λογισμικού	19
3.1.1 Βιβλιοθήκη SPI	21
3.1.2 Βιβλιοθήκη Ethernet	22
3.2 Σύνδεση στο δίκτυο (web)	23
3.4 Προστασία της σελίδας	23
3.5 Αρχείο CSS	25
3.6 Έλεγχος του Arduino μέσω κουμπιών	28
3.7 Σελίδα HTML	29
3.8 Συνδεσμολογία με 4 relay και σχηματικό 4 relay	30
3.9 Τεχνολογίες αιχμής	32
Κεφάλαιο 4 ^ο	35
4.ΣΥΜΠΕΡΑΣΜΑΤΑ.....	35
4.1 Αποτελέσματα.....	35
4.2 Δυσκολίες – Προβλήματα	35
4.3 Ανάπτυξη και Αποσφαλμάτωση	36
4.4 Αδυναμίες.....	37
4.5 Διαφορές και προτεινόμενες εκδόσεις του Arduino.....	37
4.6 Χρήσης της Εφαρμογής	38
4.7 Μελλοντικές Επέκτασης	38
4.9 Βιβλιογραφία – Πηγες Πληροφοριών	40



Πίνακες

Πίνακας 1 - Τεχνικά χαρακτηριστικά τις πλατφόρμας	15
Πίνακας 2 - κύρια χαρακτηριστικά μερικών διαθέσιμων πλακετών	16
Πίνακας 3 - Βασικές λειτουργίες του IDE	20

Εικόνες

Εικόνα 1 – Δομή βασικών τμημάτων ενός μικροϋπολογιστή	9
Εικόνα 2 – Θύρες Προγραμματισμού Arduino Due	17
Εικόνα 3 – Arduino pin out diagram	18
Εικόνα 4 – Arduino pins SPI	21
Εικόνα 5 – Arduino Ethernet Shield schematic	22
Εικόνα 6 – SPI Bus connection	22
Εικόνα 7 – Κατασκευή κεφαλίδας εξουσιοδότησης	24
Εικόνα 8 – Συνδεσμολογία με 4 relay	31
Εικόνα 9 – Σχηματικό του 4 relay	32
Εικόνα 10 – Personal Energy Orb	33
Εικόνα 11 – Τυπικός κύκλος ανάπτυξης λογισμικού για το Arduino	36
Εικόνα 12 – Ethernet Shield με Micro-SD Card Reader	38



Ευχαριστίες

Αρχικά θα ήθελα να ευχαριστήσω την οικογένεια μου Γιώργο και Σταματία και τον αδελφό μου Αλέξανδρο για την στήριξη που μου προσέφεραν όλα αυτά τα χρονιά και που πάντα βρίσκονται δίπλα μου σε κάθε μου βήμα.

Επίσης ένα μεγάλο ευχαριστώ στον κύριο Παπαδάκη Νικόλαο για την ανάθεση του θέματος, και το αμείωτο ενδιαφέρον του. Ένα μεγάλο ευχαριστώ οφείλω για την απεριόριστη βοήθεια στο Τμήματος Ηλεκτρολογιών Μηχανικών για τις γνώσεις που μου παρείχανε.



Abstract

The purpose of the issue is to control small and large loads Arduino through an HTML page that you post on our platform. The Arduino sets a page where we have local and universal access. That means you can turn ON/OFF any appliance at your home from your iPhone, android, iPad, PC or laptop from anywhere in the world. As long as you have access to a web browser.

Σύνοψη

Σκοπός του θέματος είναι ο έλεγχος μικρών και μεγάλων φορτίων μέσω Arduino από μια html σελίδα η οποία αναρτάτε από την πλατφόρμα Arduino Due . Το Arduino θα ανεβάζει μια σελίδα από την οποία έχουμε τοπική ή παγκόσμια πρόσβαση. Αυτό σημαίνει ότι μπορείς να ανοιγοκλείνεις μια συσκευή στο σπίτι σου από το iPhone, Android, iPad, PC η laptop από οποιοδήποτε σημείο του κόσμου. Αρκεί να έχεις πρόσβαση σε ένα web browser.



ΚΕΦΑΛΑΙΟ 1⁰

1. ΕΙΣΑΓΩΓΗ

1.1 Περίληψη της πτυχιακής

Σκοπός της πτυχιακής αυτής ήταν να δούμε και να εργαστούμε στην πλατφόρμα Arduino Due έστω να μπορέσουμε να ελέγχουμε φορτία συσκευές μέσω του διαδικτύου στο οποίο απαιτείτε γνώση HTML κώδικα η C++.

1.2 Κίνητρο για την Διεξαγωγή της εργασίας

Το κίνητρο μου για την εργασία αυτή, είναι από προσωπικό ενδιαφέρον για την πλατφόρμα και για τις εφαρμογές έξυπνου σπιτιού. Είναι κάτι που θα ήθελα να ασχοληθώ, το οποίο είναι μέρος, κατασκευή και μελέτη έξυπνων σπιτιών μέσω υπολογιστικών συστημάτων.

1.3 Δομή εργασίας

Η πτυχιακή εργασία εμπεριέχει τέσσερα κεφάλαια τα οποία αναπτύσσονται σε πεδία τα οποία είναι τα ακόλουθα:

Στο πρώτο κεφάλαιο Εισαγωγή δίνεται μια περίληψη της πτυχιακής εργασίας και τα κίνητρα για την διεξαγωγή της.

Στο δεύτερο κεφάλαιο αναπτύσσεται η σχετική μεθοδολογία υλοποίησης, όπου παρουσιάζονται θεωρίες, μέθοδοι ,σχετικά μοντέλα που ακολουθήθηκαν για την εργασία.

Στο τρίτο κεφάλαιο έχουμε να κάνουμε με το σχέδιο δράσης για την εκπόνηση τις εργασίας και μια αναζήτηση σε τεχνολογίες αιχμής.

Στο τέταρτο κεφαλαίο και τελικό κεφάλαιο παρουσιάζονται τα αποτελέσματα της παρούσας πτυχιακής εργασίας.

1.4 Μεθοδολογία υλοποίησης

Αρχικά μελετήθηκαν το υλικά κομμάτια τις πλατφόρμας, έστω να μελετηθούν οι ανοχές τους. Δημιουργία του κώδικα για τη ανάρτηση του HTML μέσω του Arduino ανέβασμα τις σελίδας και δημιουργία sec τοποθέτησή του σε web hosting (φιλοξενία ιστοσελίδας).Εφαρμογή δικλίδας ασφαλείας (κωδικός χρήστη).



ΚΕΦΑΛΑΙΟ 2⁰

2. ΘΕΩΡΙΕΣ

2.1 Το διαδίκτυο η ιντερνέτ

Το διαδίκτυο ή το ιντερνέτ είναι ένα διασυνδεδεμένο δίκτυο υπολογιστών, για την εξυπηρητήσει εκατομμύρια χριστών σε ολόκληρο τον κόσμο όπου μπορούν να εναλλάσσουν πακέτα μεταξύ τους μέσω διαφόρων πρωτοκόλλων (τυποποιημένοι κανόνες επικοινωνίας) τα οποία υλοποιούνται σε επίπεδο υλικού και λογισμικού το κοινό αυτό δίκτυο καλείται διαδίκτυο.

2.2 Παγκόσμιος ιστός (world wide web)

Ο παγκόσμιος ιστός είναι η πιο δημοφιλής υπηρεσία του διαδικτύου, που μας επιτρέπει με ιδιαίτερα εύκολο τρόπο να έχουμε πρόσβαση σε μια τεράστια συλλογή ψηφιακών εγγράφων, τις ιστοσελίδες.

2.3 Ιστοσελίδα

Ιστοσελίδα η (Web page) είναι ένα είδος εγγράφου του παγκοσμίου ιστού (www) που περιλαμβάνει πληροφορίες με την μορφή κειμένου, υπερκείμενου εικόνας βίντεο και ήχου, διαχωρίζονται σε στατικές και δυναμικές:

1. Οι στατικές αλλάζουν την διάταξη τους και το περιεχόμενο μόνο εάν ο προγραμματιστής αναβαθμίσει την σελίδα. Μια απλή HTML σελίδα είναι παράδειγμα στατικού περιεχομένου.
2. Οι δυναμικές αλλάζουν και προσαρμόζονται σύμφωνα με την καταχώρηση ή με τις αλλαγές που δίνονται από τον τελικό χρήστη μέσω web browser.

2.4 Τι είναι η HTML (Hyper Text Markup Language)

Η HTML είναι η κύρια γλώσσα σήμανσης για τις ιστοσελίδες, και τα στοιχεία της είναι τα βασικά δομικά στοιχεία των ιστοσελίδων.

Μπορούμε να γράψουμε html με οποιοδήποτε συντάκτη κείμενων (WordPad, Notepad, Microsoft Word, WordPerfect, Open Office) σε μορφή κώδικα.

2.5 Τι είναι CSS (Cascading style sheets-Διαδοχικά Φύλλα Στυλ)

Η css είναι μια γλώσσα υπολογιστή που χρησιμοποιείται για τον έλεγχο της εμφάνισης ενός εγγράφου που γράφτηκε στις γλώσσες html και xhtml, δηλαδή για τον έλεγχο της εμφάνισης μιας ιστοσελίδας και γενικότερα ενός ιστό τόπου.

2.6 Web hosting (Φιλοξενία ιστοσελίδων)

Η φιλοξενία ιστοσελίδων είναι μια διαδικτυακή υπηρεσία που επιτρέπει σε ιδιώτες και εταιρείες να διαθέτουν μια ιστοσελίδα συνεχώς αναρτημένη στο διαδίκτυο,



χωρίς να χρειάζεται να επιβαρύνεται με το κόστος του ανάλογου εξοπλισμού (εξυπηρετητές) ή την ανάγκη εξυπηρέτησης μεγάλου αριθμού εξωτερικών συνδέσεων και εύρος σύνδεσης (bandwidth). Αυτό το αναλαμβάνουν οι εταιρίες (Web hosts) που προσφέρουν χώρο στον διακομιστή τους καθώς και μέρος της σύνδεσης τους στο δίκτυο.

Ο ορός web hosting αναφέρεται στην διαδικασία με την οποία αυτός που θα ανεβάσει μια ιστοσελίδα ενοικιάζει χώρο σε διακομιστές για να τοποθετήσει τα αρχεία του. Τα αρχεία αυτά, είναι και αυτά που θα στοιχειοθετούν την σελίδα του, οπου προσφέρονται μέσω αδιάλειπτης παροχής στους χρήστες –επισκέπτες.

Ο ιδιοκτήτης του ιστότοπου μπορεί να ανεβάζει τα αρχεία του μέσω προγράμματος (ftp client) στο διακομιστή που φιλοξενείται, να διαχειρίζεται τα αρχεία. Μερικές σελίδες για φιλοξενία είναι το Plesk, το Cpanel, το Dorbox, το Webmin κ.α.

2.7 Τι είναι ο server (Εξυπηρετητής)

Εξυπηρετητής η διακομιστής μπορεί να είναι υλικό η λογισμικό που αναλαμβάνει την παροχή υπηρεσιών, αιτήσεις άλλων προγραμμάτων, γνωστούς ως πελάτες (clients) που μπορούν να τρέχουν στον ίδιο υπολογιστή η σε άλλον μέσω δικτύου. Όταν το πρόγραμμα εξυπηρετεί όλο το 24ωρο της ημέρας μέσω ενός υπολογιστή τότε μπορούμε να λεμέ τον υπολογιστή αυτόν και (server) διακομιστή, αφού είναι η βασική λειτουργία του .

2.8 Γλώσσες προγραμματισμού

Γλώσσες προγραμματισμού είναι η τεχνητές γλώσσες που χρησιμοποιούνται για τον έλεγχο μιας μηχανής, η πιο συχνή χρήση είναι σε υπολογιστές. Οι γλώσσες προγραμματισμού όπως και η ανθρώπινοι ορίζοντες από ένα σύνολο συντακτικών και εννοιολογικών κανόνων, που ορίζουν την ορθότητα της δομής του νοήματος των προτάσεων της γλώσσας.

Οι γλώσσες μας διευκολύνουν στην διαχείριση και οργάνωση πληροφοριών, αλλά και για ακριβή διατύπωση αλγορίθμων.

2.8.1 Βιβλιοθήκες (υπολογιστών)

Οι βιβλιοθήκες είναι βοηθητικά κομμάτια κώδικα η δεδομένων τα οποία παρέχουν υποστήριξη λογισμικού. Η έννοια της βιβλιοθήκης είναι αναπόσπαστο τμήμα του δομημένου προγραμματισμού και αναπτύχθηκε παράλληλα με αυτόν.

2.8.2 Μεταβλητές

Μεταβλητή ονομάζουμε ένα γλωσσικό αντικείμενο που έχει τη δυνατότητα να λάβει διάφορες τιμές, μια κάθε φορά. Η τιμές αυτές είναι περιορισμένες συνήθως σε κάποιο τύπο δεδομένων. Μερικοί τύποι δεδομένων για το Arduino είναι:

1. **byte**: αποθηκεύει μια αριθμητική τιμή 8-bit χωρίς δεκαδικά ψηφία, παίρνουν τιμές από 0 μέχρι 255.
2. **int**: ακραίοι, παίρνουν τιμές από -32,768 μέχρι 32767.



3. **long**: μεγάλου μεγέθους ακέραιοι, παίρνουν τιμές από -2,147,483,648 μέχρι 2,147,483,647

4. **float**: πραγματικοί αριθμοί, παίρνουν τιμές από 3.4×10^{-38} μέχρι 3.4×10^{38}

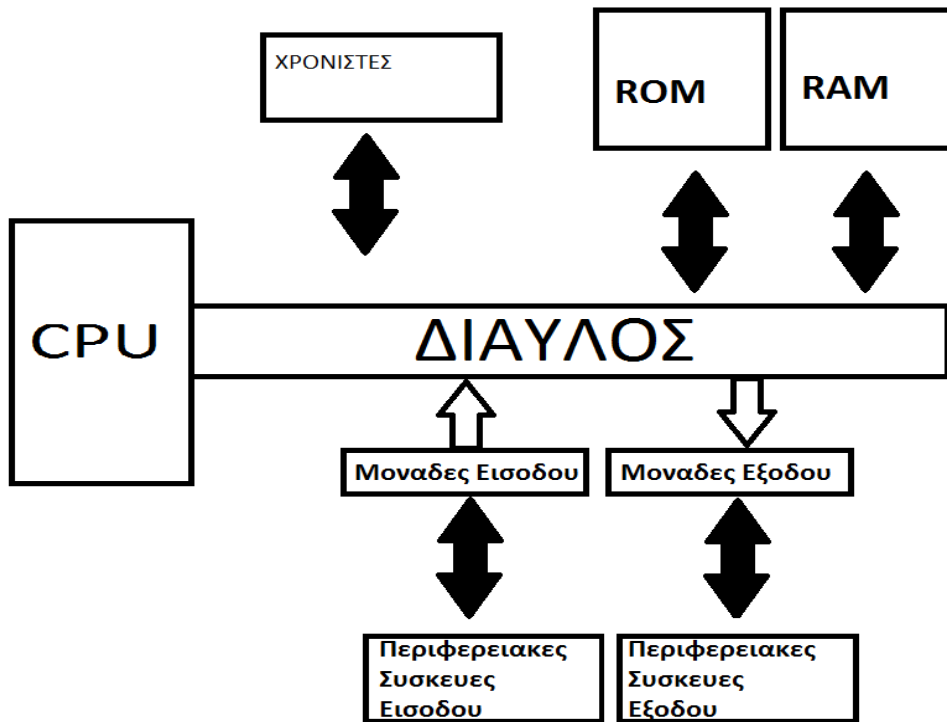
Τις μεταβλητές μπορούμε να τις δηλώσουμε στην αρχή του προγράμματός μας:

int myvariable;

Μπορούμε επίσης να δώσουμε αρχική τιμή στη μεταβλητή ταυτόχρονα με τη δήλωσή της:

int myvariable = 47;

2.8.3 Μικροελεγκτές



Εικόνα 1- Δομή βασικών τμημάτων ενός μικροϋπολογιστή

Οι μικροελεγκτές είναι ολοκληρωμένα υπολογιστικά συστήματα σχεδιασμένα για εφαρμογές οι οποίες απαιτούν εκτεινόμενη διασύνδεση με τον έξω κόσμο σε ένα συμπαγές και χαμηλού κόστους πακέτο.

Ενσωματώνουν σε ένα ολοκληρωμένο κύκλωμα χαρακτηριστικά (μνήμη, χρονιστές, κυκλώματα διασύνδεσης περιφερικών) που είναι επιπρόσθετα στους συνηθισμένους μικροεπεξεργαστές που απευθύνονται στα pc.

Λόγω μεγέθους και δυνατοτήτων χρησιμοποιούνται ευρέως σε εφαρμογές ενσωματωμένων συστημάτων, μηχανικών συστημάτων, ρομποτικής.

Τα ενσωματωμένα συστήματα είναι ένας συνδυασμός υλικού (hardware) και λογισμικού (software) τα οποία συνδυαστικά αποτελούν τμήμα μιας μεγαλύτερης διάταξης.

Παράδειγμα ενσωματωμένου συστήματος είναι ο μικροεπεξεργαστής (εγκέφαλος) διαχείρισης του κινητήρα στα σύγχρονα αυτοκίνητα.

Τα ενσωματωμένα συστήματα είναι σχεδιασμένα να λειτουργούν χωρίς ανθρώπινη παρέμβαση, και συνήθως ανταποκρινόμενα σε πραγματικό χρόνο.

Μερικές εφαρμογές ακόμα για ενσωματωμένα συστήματα μπορούν να χρησιμοποιηθούν:

- Καταναλωτικά προϊόντα (Τηλεόρασης, στερεοφωνικά, ηλεκτρονικά παιχνίδια μουσικά όργανα...)
- Εξοπλισμός γραφείου (Φωτοτυπικά, εκτυπωτές, φαξ ...)
- Μεταφορές (Σηματοδότηση δρόμων, μέσα μεταφοράς, συστήματα παρακολούθησης ατυχημάτων...)
- Αυτοκίνητα (Διαχείριση κινητήρα, αεροσακοί, ABS, ESP...)
- Τεχνολογία κτηρίων (Ασανσέρ, σύστημα κλιματισμού, φωτισμού, ασφάλειας...)
- Γεωργίας
- Αμυντικές και Διαστημικές εφαρμογές ...

Με λίγα λόγια παντού.

Αναλυτικά, τα πλεονεκτήματα των μικροελεγκτών είναι:

1. Αυτονομία, μέσω της ενσωμάτωσης σύνθετων περιφερειακών υποσυστημάτων όπως μνήμες και θύρες επικοινωνίας. Έτσι πολλοί μικροελεγκτές δεν χρειάζονται κανένα άλλο ολοκληρωμένο κύκλωμα για να λειτουργήσουν.
2. Η ενσωμάτωση περιφερειακών σημαίνει ευκολότερη υλοποίηση εφαρμογών λόγω των απλούστερων διασυνδέσεων. Επίσης, οδηγεί σε χαμηλότερη κατανάλωση ισχύος, μεγιστοποιώντας τη φορητότητα και ελαχιστοποιεί το κόστος της συσκευής στην οποία ενσωματώνεται ο μικροελεγκτής.
3. Χαμηλό κόστος.
4. Μεγαλύτερη αξιοπιστία, και πάλι λόγω των λιγότερων διασυνδέσεων.
5. Μειωμένες εκπομπές ηλεκτρομαγνητικών παρεμβολών και μειωμένη ευαισθησία σε αντίστοιχες παρεμβολές από άλλες ηλεκτρικές και ηλεκτρονικές συσκευές. Το πλεονέκτημα αυτό προκύπτει από το μικρότερο αριθμό και μήκος εξωτερικών διασυνδέσεων καθώς και τις χαμηλότερες ταχύτητες λειτουργίας.
6. Περισσότεροι διαθέσιμοι ακροδέκτες για ψηφιακές εισόδους-εξόδους (για δεδομένο μέγεθος ολοκληρωμένου κυκλώματος), λόγω της μη δέσμευσής τους για τη σύνδεση εξωτερικών περιφερειακών.
7. Μικρό μέγεθος συνολικού υπολογιστικού συστήματος.

Λόγω του ισχυρότατου ανταγωνισμού αλλά και της τάσης ενσωμάτωσης των μικροελεγκτών σε κάθε ηλεκτρική και ηλεκτρονική συσκευή, η βιομηχανία μικροελεγκτών έχει καταλήξει στην παραγωγή ανταγωνιστικών μοντέλων μαζικής παραγωγής καθώς και μικροελεγκτών για πιο εξειδικευμένες εφαρμογές. Έτσι διακρίνονται οι εξής κυρίως κατηγορίες:

- Μικροελεγκτές (καμιά φορά 4-bit αλλά συνήθως 8-bit) πολύ χαμηλού κόστους, γενικής χρήσης, με πολύ μικρό αριθμό ακροδεκτών (ακόμη και λιγότερους από 8). Σχεδιάζονται με έμφαση στη χαμηλή κατανάλωση ισχύος και την αυτάρκεια, ώστε να χρειάζονται ελάχιστα ή και καθόλου εξωτερικά εξαρτήματα και να μη μπορεί να αντιγραφεί εύκολα το εσωτερικό λογισμικό τους. Απουσιάζει η δυνατότητα επέκτασης της μνήμης τους. Μερικά μοντέλα είναι ευρέως γνωστά στους ερασιτέχνες ηλεκτρονικούς, όπως πχ οι περισσότεροι



μικροελεγκτές των σειρών PIC (Microchip), AVR (Atmel) και 8051 (Intel, Atmel, Dallas κα).

- Μικροελεγκτές (συνήθως 8-bit αλλά και 16 ή 32-bit) χαμηλού κόστους, γενικής χρήσης, με μέτριο έως σχετικά μεγάλο αριθμό ακροδεκτών. Διαθέτουν μεγάλο αριθμό κοινών περιφερειακών, όπως θύρες UART, I²C, SPI ή CAN, μετατροπείς αναλογικού σε ψηφιακό και ψηφιακού σε αναλογικό. Στους κατασκευαστές της Άπω Ανατολής (Ιαπωνία, Κορέα), συνηθίζεται η ενσωμάτωση ελεγκτών οθόνης υγρών κρυστάλλων και πληκτρολογίου. Μερικές φορές παρέχουν δυνατότητα εξωτερικής επέκτασης της μνήμης τους.
- Μικροελεγκτές (κυρίως 32-bit) μέσου κόστους, γενικής χρήσης, με μεγάλο αριθμό ακροδεκτών. Χαρακτηρίζονται από έμφαση στην ταχύτητα εκτέλεσης εντολών, υψηλή αυτάρκεια περιφερειακών και μεγάλες δυνατότητες εσωτερικής ή εξωτερικής μνήμης προγράμματος (FLASH) και RAM. Στο χώρο αυτό έχουν ισχυρή παρουσία οι αρχιτεκτονικές με υψηλή μεταφερσιμότητα λογισμικού (portability) από τον ένα στον άλλο κατασκευαστή. Πχ μεταξύ των μικροελεγκτών τύπου ARM ή MIPS, το σύνολο των βασικών εντολών που αναγνωρίζει η ALU είναι ακριβώς το ίδιο, μειώνοντας έτσι τις μεγάλες αλλαγές στο λογισμικό, όταν στο μέλλον ο πελάτης υιοθετήσει ένα μικροελεγκτή άλλου κατασκευαστή (αρκεί, φυσικά, να υποστηρίξει κι αυτός το σύνολο εντολών ARM ή MIPS, αντίστοιχα).
- Μικροελεγκτές εξειδικευμένων εφαρμογών, οι οποίοι ενσωματώνουν συνήθως κάποιο εξειδικευμένο πρωτόκολλο επικοινωνίας το οποίο υλοποιείται πάντοτε σε hardware. Τέτοιοι μικροελεγκτές χρησιμοποιούνται σε τηλεπικοινωνιακές συσκευές όπως τα μόντεμ.
- Η μεγάλη μερίδα πωλήσεων των μικροελεγκτών εξακολουθεί να αφορά αυτούς των 8-bit, καθώς είναι η κατηγορία με το χαμηλότερο κόστος και το μικρότερο μέγεθος λογισμικού για το ίδιο αποτέλεσμα, ιδίως επειδή οι σύγχρονες οικογένειες μικροελεγκτών 8-bit έχουν πολύ βελτιωμένες επιδόσεις σε σχέση με το παρελθόν.

Η επιτυχία μιας οικογένειας μικροελεγκτών καθορίζεται σε μεγάλο βαθμό από τη διαθεσιμότητα και την ευχρηστία των σχετικών εργαλείων ανάπτυξης, όπως μεταφραστές από γλώσσες υψηλού επιπέδου σε γλώσσα κατανοητή από τον μικροελεγκτή (assembly), προγραμματιστές της εσωτερικής μνήμης και εργαλεία εκσφαλμάτωσης (debuggers). Στους μικροελεγκτές, τα εργαλεία αυτά δεν αποτελούνται ποτέ μόνο λογισμικό, καθώς δεν υπάρχει τυποποιημένος τρόπος επικοινωνίας με αυτούς. Στον τομέα των εργαλείων ανάπτυξης, δραστηριοποιούνται όχι μόνο οι ίδιοι οι κατασκευαστές μικροελεγκτών αλλά και εξειδικευμένες εταιρείες.

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

Μερικοί από τους γνωστότερους κατασκευαστές μικροελεγκτών είναι οι

1. ARM (δεν κατασκευάζει αλλά παραχωρεί δικαιώματα χρήσης του πυρήνα)
2. Atmel



3. Epson
4. Freescale Semiconductor (πρώην Motorola)
5. Hitachi
6. Maxim (μετά την εξαγορά της Dallas)
7. Microchip

2.9 Τι είναι το λογισμικό (software)

Με τον όρο λογισμικό (software) ορίζουμε μια συλλογή από προγράμματα υπολογιστών, διαδικασίες και οδηγίες χρήσης που εκτελούν ορισμένες εργασίες σε ένα υπολογιστικό σύστημα.

Το λογισμικό περιλαμβάνει προγράμματα που έχουν αναπτυχθεί σε μια γλώσσα προγραμματισμού όπως για παράδειγμα η C, η C++, η Java. Το λογισμικό μπορεί να περιλαμβάνει τα προγράμματα, τα websites και αλλά πράγματα που είναι γραμμένα στις παραπάνω γλώσσες. Το λογισμικό είναι κωδικοποιημένο με συγκεκριμένο τρόπο, με τη βοήθεια ενός δυαδικού ψηφιακού συστήματος, ώστε να είναι κατανοητό από το υλικό.

Ο όρος software χρησιμοποιείται συχνά για να περιγράψει κάτι το οποίο δεν είναι υλικό.

2.10 Βασικός έλεγχος ταυτότητας HTTP

Για να περιορίσουμε την πρόσβαση σε αρχεία σε ένα διακομιστή Web με βασικού έλεγχου ταυτότητας, ο χρήστης πρέπει να εισάγει πιστοποιήσεις ID (Identification) δηλαδή να παρέχει ένα όνομα χρήστη και ένα κωδικό πρόσβασης. Η μη πιστοποιημένες συνδέσεις περιέχουν μια επικεφαλίδα HTTP WWW-Authenticate όπου εμπεριέχει τον βασικό έλεγχο της ταυτότητας του χρήστη. Όταν ο χρήστης θελήσει να στείλει τα διαπιστευτήρια ελέγχου του στον διακομιστή μπορεί να χρησιμοποιήσει την κεφαλίδα εξουσιοδότησης. Παρακάτω θα δούμε και την κατασκευή της κεφαλίδας.

2.11 Τι είναι JavaScript (JS)

Η JavaScript είναι μια γλώσσα προγραμματισμού διερμηνευμένη για ηλεκτρονικούς υπολογιστές η αρχική χρήση της γλώσσας αποτέλεσε μέρος της υλοποίησης των web browser, ώστε να μπορεί από την πλευρά του πελάτη (client) να επικοινωνούν με τον χρήστη, να ανταλλάσσουν δεδομένα σύγχρονα και να αλλάζουν δυναμικά το περιεχόμενο του εγγράφου που εμφανίζεται. Η JavaScript χρησιμοποιείται και σε εφαρμογές εκτός ιστοσελίδων τέτοια παραδείγματα είναι τα έγγραφα PDF και οι μικρές εφαρμογές της επιφάνειας εργασίας (desktop widgets).

2.12 Τι είναι το Arduino

Το Arduino είναι μια κατά βάση υπολογιστική πλατφόρμα που βασιζόμενη σε μια μητρική πλακέτα ανοικτού κώδικα, με ενσωματωμένο ή αποσπώμενο μικρό ελεγκτή και φυσικά διαθέσιμες εισόδους και εξόδους, και μπορεί να προγραμματιστεί με τη γλώσσα wiring (πρόκειται για μια γλώσσα προγραμματισμού C++ και ένα σύνολο από βιβλιοθήκες, υλοποιημένες σε C++).



Το Arduino χρησιμοποιείται για την ανάπτυξη ανεξάρτητων διαδραστικών αντικείμενων αλλά και να συνδεθεί με υπολογιστή μέσω προγραμμάτων σε pure data, supercollider, max/MSP, Processing.

Τα διαγράμματα και οι πληροφορίες για το υλικό είναι ελεύθερα διαθέσιμα για αυτούς που θέλουν να συναρμολογήσουν το Arduino.

Το πρόγραμμα Arduino έλαβε τιμητική στην κατηγορία Digital Communities στο Electronica 2006.

2.13 Υλικό (Hardware)

Τα κύρια κομμάτια του Arduino είναι ένας μικρό ελεγκτής Atmel AVR (ATmega328 και AT mega168 στις νεότερες εκδόσεις, AT mega8 στις παλιότερες) όλες οι πλακέτες περιλαμβάνουν ένα γραμμικό ρυθμιστή τάσης 5V και ένα κρυσταλλικό ταλαντωτή 16MHz.

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

Σε εννοιολογικό επίπεδο, στην χρήση του Arduino software stack, όλα τα boards προγραμματίζονται με μία RS-232 σειριακή σύνδεση, αλλά ο τρόπος που επιτυγχάνεται αυτό διαφέρει σε κάθε hardware εκδοχή. Οι σειριακές πλάκες Arduino περιέχουν ένα απλό level shifter κύκλωμα για να μετατρέπει μεταξύ σήματος επιπέδου RS-232 και TTL. Τα τωρινά Arduino προγραμματίζονται μέσω USB, αυτό καθίσταται δυνατό μέσω της εφαρμογής προσαρμοστικών chip USB-to-Serial όπως το FTDI FT232. Κάποιες παραλλαγές, όπως το Arduino mini οπότε χρησιμοποιεί ένα αναιρούμενο USB-to-Serial καλώδιο ή Board, Bluetooth, ή άλλες μεθόδους. (Όταν χρησιμοποιείτε με παραδοσιακά εργαλεία microcontroller αντί για το Arduino IDE, πρότυπος προγραμματισμός AVR ISP χρησιμοποιείτε).

Το Arduino έχει τα περισσότερα microcontroller I/O pins για χρήση από αλλά κυκλώματα. Παρέχουν 14ψηφιακά I/O pins έξι από τα οποία μπορούν να παράγουν παλμικά σήματα, και έξι αναλογικά δεδομένα.

Διάφορες εμπορικές ασπίδες plug-in(shields) είναι διαθέσιμες.

2.14 Ασπίδες (Shields)

Τα Arduino χρησιμοποιούν την τεχνολογία των Shields, είναι ολοκληρωμένες πλακέτες που είναι σχεδιασμένες ώστε να κουμπώνουν πάνω στο Arduino προεκτείνοντας την λειτουργικότητα τους. Είναι η hardware αντίστοιχη έννοια των plugin, addon και extension που υπάρχουν στο Software.

Ethernet Shield: Δίνει στο Arduino την δυνατότητα να δικτυωθεί σε ένα LAN ή στο Internet μέσω καλωδίου Ethernet.

Wife Shield: Όμοιο με το Ethernet Shield, χωρίς φυσικά το καλώδιο. Μερικά από αυτά τα Shields παρέχουν έλεγχο σε GPS, LCD οθόνη, κινητήρες κ.α.



2.15 Ηλεκτρονόμος(relay)

Ένα άλλο εξάρτημα το οποίο μας βοήθησε είναι και ο ηλεκτρονόμος που δεν είναι άλλο από ένα διακόπτης που ανοίγει και κλείνει ένα ηλεκτρικό κύκλωμα, ένας ηλεκτρομαγνήτης ελέγχει την ενεργοποίηση ή απενεργοποίηση μίας ή περισσοτέρων επαφών. Επειδή ένας ηλεκτρονόμος έχει την ικανότητα να ελέγχει ένα κύκλωμα εξόδου υψηλότερης ισχύς από το κύκλωμα εισόδου μπορεί κάλλιστα να θεωρηθεί και μια μομφή ηλεκτρικού ενισχυτή. Η επαφές που χρησιμοποιούνται στους ηλεκτρονόμους είναι ανοιχτές, κλειστές ή μεταγωγικές, ανάλογα με το τύπο τους .

2.16 Λογισμικό ανάπτυξης (IDE)

Το ολοκληρωμένο περιβάλλον ανάπτυξης (IDE) του Arduino είναι μια εφαρμογή όπου είναι γραμμένη σε java, που λειτουργεί σε πολλές πλατφόρμες, και προέρχεται από το IDE για τη γλώσσα προγραμματισμού Processing και το σχέδιο Wiring. Έχει σχεδιαστεί για να εισάγει τον προγραμματισμό και σε αυτούς που δεν είναι τόσο έμπειροι με την ανάπτυξη λογισμικού. Περιλαμβάνει ένα πρόγραμμα επεξεργασίας κώδικα με χαρακτηριστικά όπως είναι η επισήμανση σύνταξης και ο συνδυασμός των αγκύλων και είναι επίσης σε θέση να μεταγλωττίζει και να φορτώνει προγράμματα στην πλακέτα με ένα μόνο κλικ. Δεν υπάρχει συνήθως καμία ανάγκη να επεξεργαστείτε αρχεία ή να τρέξετε προγράμματα σε ένα περιβάλλον γραμμής εντολών. Ένα πρόγραμμα ή 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:

-setup():μία συνάρτηση που τρέχει μία φορά στην αρχή του προγράμματος η οποία αρχικοποιεί τις ρυθμίσεις.

-loop(): εδώ βάζουμε στον κώδικα μας όλες τις εντολές που είναι μέσα στην loop(), τρέχουν μέχρι να φτάσει στο τέλος της, έπειτα πάει στην αρχή και επαναλαμβάνει ξανά. Αυτό γίνεται μέχρι να διακάψουμε την τροφοδοσία ή να κάνουμε επανεκκίνηση της πλατφόρμας.



2.17 Τεχνικά χαρακτηριστικά της πλατφόρμας

Η πλακέτα βασίζεται σε 32-bit πυρήνα ARM μικροελεγκτής . Με 54 ψηφιακές εισόδους/εξοδούς, από τις οποίες 12 μπορούν να δεχτούν και αναλογικές εισόδους

AVR Arduino microcontroller

Microcontroller	AT91SAM3X8E
Operating Voltage	3.3V
Input Voltage (recommended)	7-12V
Input Voltage (limits)	6-16V
Digital I/O Pins	54
Analog Input Pins	12
Analog Output Pins	2 (DAC)
Total DC Output Current on all I/O lines	130 mA
DC Current for 3.3V Pin	800 mA
DC Current for 5V Pin	800 mA
Flash Memory	512 KB
SRAM	96 KB
Clock Speed	84 MHz
Length	101.52 mm
Width	53.3 mm
Weight	36 g

Πίνακας 1- Τεχνικά χαρακτηριστικά τις πλατφόρμας

Ένα κομμάτι των εισόδων είναι αναλογικές, είναι εφοδιασμένες με έναν αναλογικό – ψηφιακό μετατροπέα, επίσης κάποιες έξοδοι λειτουργούν και ως αναλογικές, με χρήση διαμόρφωσης πλάτους παλμού (Pulse Width Modulation – PWM) μερικές από της πλακέτες arduino παρέχουν επιπλέον και αναλογικές εξόδους.

Υπάρχουν διαφοροποιήσεις από μοντέλο σε μοντέλο που τις συναντάμε στα κομμάτια της χωρητικότητας των μνήμων RAM, EEPROM και Flash, καθώς και στην ταχύτητα (CPU clock) και στον αριθμό εισόδων - εξόδων. Η επιλογές στα μοντέλα καθορίζονται από τις ανάγκες τις εκάστοτε εφαρμογής .

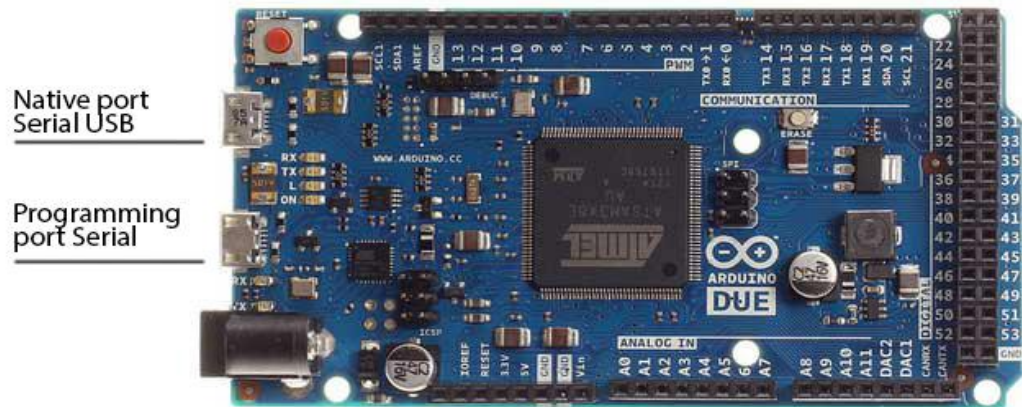


Επεξεργαστής	Flash	EEPROM	RAM	Είς./εξ.	Έξ.PWM	Αναλ.είσοδοι
ATmega2560	256KB	4KB	8KB	54	14	16
ATmega168	16KB	0.5KB	1KB	14	6	6
AT91SAM3X8E	512KB	0KB	96KB	54	12	12
ATmega168/328P	16/32KB	0.5KB	1/2KB	14	6	6
ATmega328P	32KB	1KB	2KB	14	6	8
ATmega32u4	32KB	1KB	2KB	14	6	12
ATmega168V/328V	16KB	0.5KB	1KB	14	6	6
ATmega1280	128KB	4KB	8KB	54	14	16
ATmega2560	256KB	4KB	8KB	54	14	16
ATmega168/328	16/32KB	0.5KB	1/2KB	14	6	8
ATmega328P	32KB	1KB	2KB	14	6	6

Πίνακας 2- κύρια χαρακτηριστικά μερικών διαθέσιμων πλακετών



2.18 Θύρες Προγραμματισμού

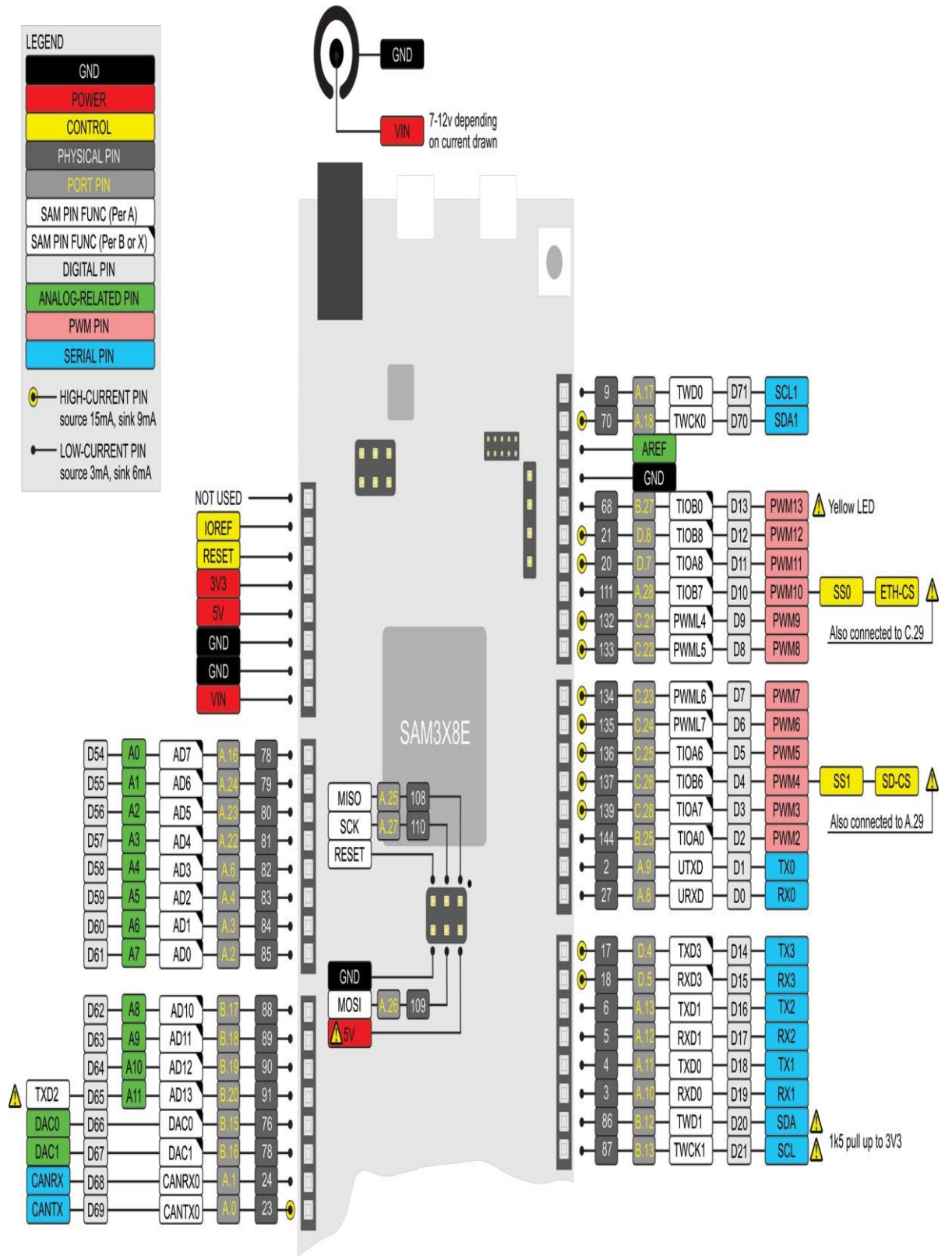


Εικόνα 2- Θύρες Προγραμματισμού Arduino Due

Το Arduino due παρέχει δυο θύρες προγραμματισμού της πλατφόρμας την θύρα Native και την θύρα Programming από της οποίες χρησιμοποιείτε κατά κόρον η θύρα Programming:

Για να ενεργοποιήσουμε την θύρα native, επιλέγουμε Arduino Due(Native Port) το λογισμικό που θα δούμε παρακάτω. Η Native συνδέεται απευθείας με τον επεξεργαστή Atmel SAM3X8E ενεργοποιώντας και απενεργοποιώντας την θύρα Native συνδέεται στα 1.200bps προκαλώντας μια ομαλή διαγραφή στην flash μνήμη της πλατφόρμας.

Για την ενεργοποίηση της θύρας Programming, επιλέγουμε Arduino Due(Programming Port), η θύρα προγραμματισμού χρησιμοποιεί το 16U2 ως USB σειριακά συνδέοντας το με το πρώτο UART του SAM3X(RX0 και τουTX0). Η 16U2 συνδέεται με το reset και το Delete του επεξεργαστή SAM3X.



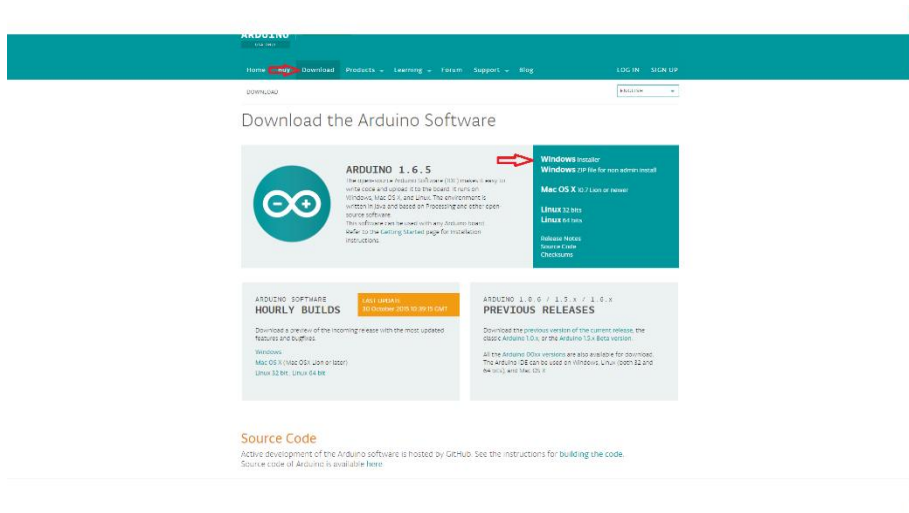
Εικόνα 3 – Arduino pin out diagram



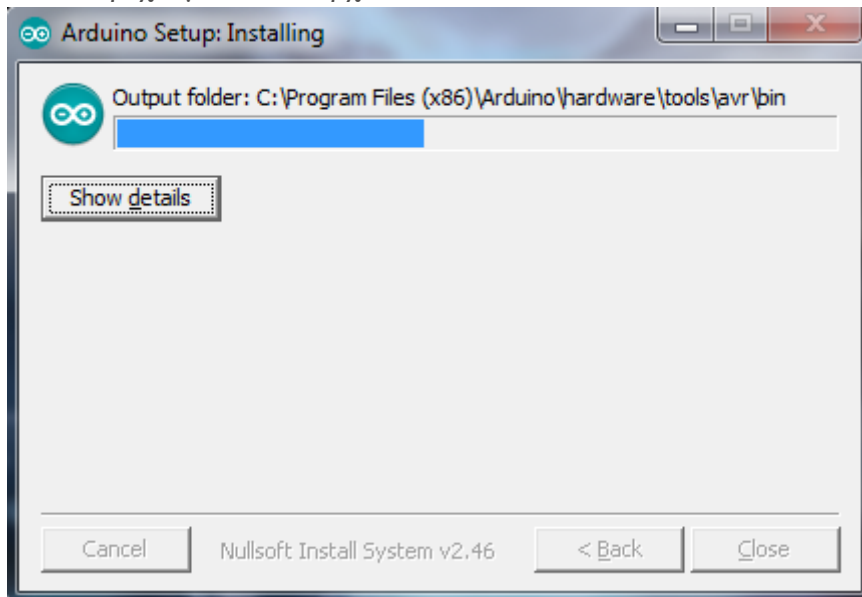
Κεφάλαιο 3^ο

3. ΔΡΑΣΗ ΓΙΑ ΤΗΝ ΔΙΕΚΠΕΡΑΙΩΣΗ ΤΗΣ ΕΓΓΡΑΣΙΑΣ

3.1 Εγκατάσταση του Λογισμικού Κατεβάζουμε το λογισμικό από: <https://www.arduino.cc/>



έπειτα τρέχουμε το **exe** αρχείο.



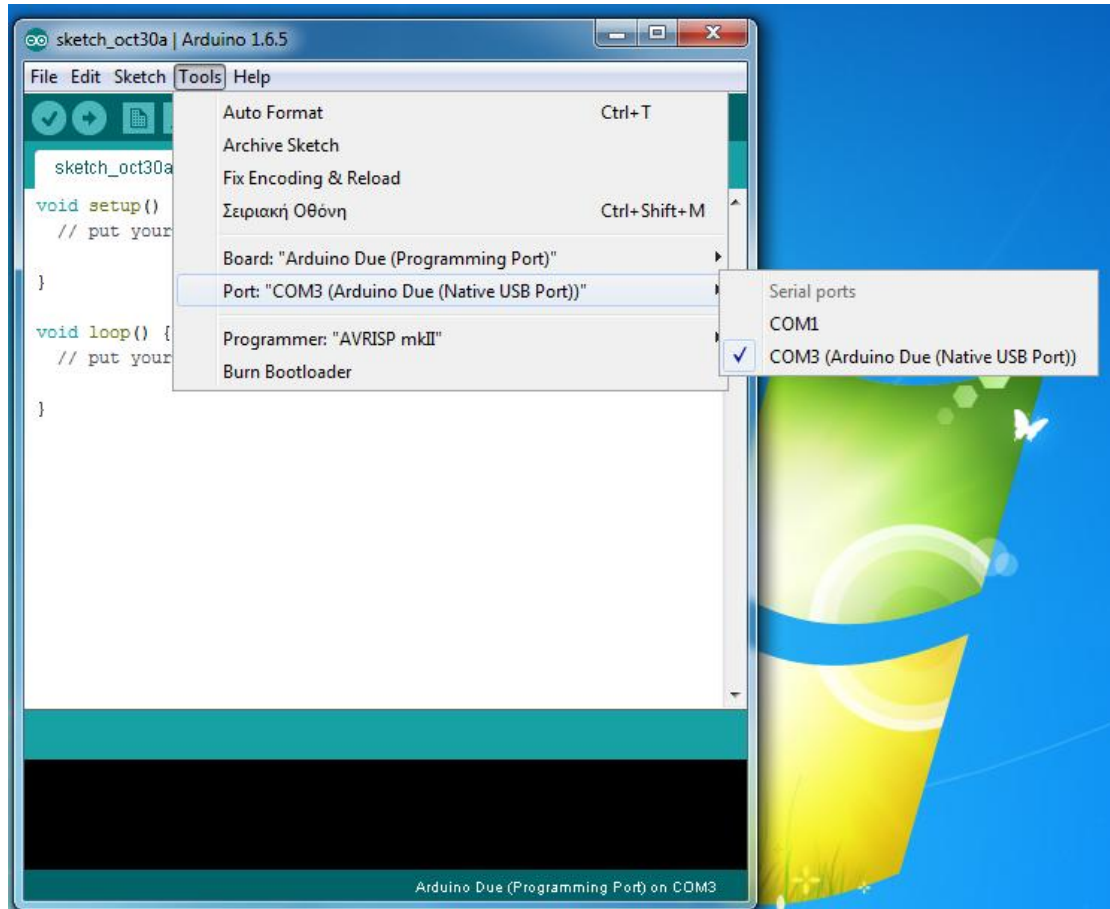
Όταν εγκαταστήσουμε το λογισμικό θα πρέπει να κάνουμε τα εξής ακόλουθα βήματα:






- 1.Tools -> Boards Manager και να ενημερώσουμε το λογισμικό για την συσκευή που θα τοποθετήσουμε αν δεν είναι ήδη στην λίστα του.
2. Tools ->port και να αλλά-



Ντάκος Δημήτρης : Έλεγχος τάσης φορτίου μέσω ιστοσελίδας ανακτημένη από πλατφόρμα Arduino DUE.

ξουμε το port στην σειριακή θύρα που έχει οριστεί από το υπολογιστή μας π.χ COM3

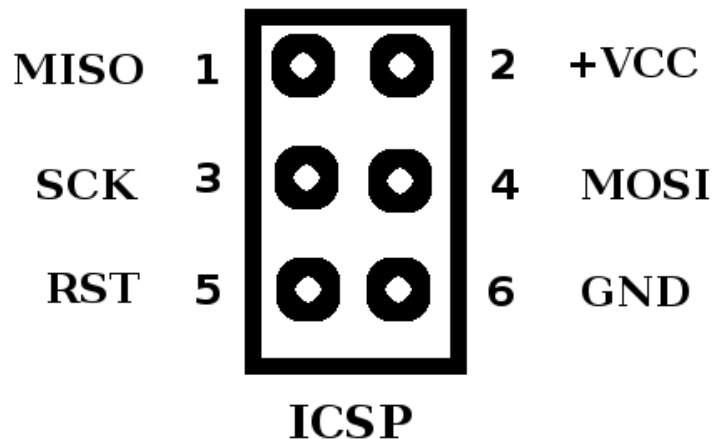


	Έλεγχος του κώδικα για λάθη.
	Μεταγλώττιση του κώδικα και ανέβασμα του στο Arduino.
	Δημιουργία νέου έργου (sketch).
	Παρουσίαση μενού με όλα τα αποθηκευμένα έργα. Πατώντας σε ένα από αυτά ανοίγει για επεξεργασία.
	Αποθήκευση του έργου.

Πίνακας 3- Βασικές λειτουργίες του IDE



3.1.1 Βιβλιοθήκη SPI



Εικόνα 4 – Arduino pins SPI

Μια από της βιβλιοθήκες που θα χρειαστούμε ώστε να υλοποιήσουμε το κομμάτι του κώδικα είναι η SPI (Serial Peripheral Interface) η βιβλιοθήκη αυτή επιτρέπει την επικοινωνία μέσω της συσκευής SPI, ουσιαστικά είναι ένα σύγχρονο σειριακό πρωτόκολλο δεδομένων που χρησιμοποιείται από μικροελεγκτές για τη διασύνδεση μίας ή περισσότερων περιφερειακών συσκευών σε μικρές απόστασης. Επίσης μας δίνει τη δυνατότητα να επικοινωνήσουμε και μεταξύ δυο μικροελεγκτών.

Σε μια σύνδεση SPI έχουμε πάντα μια κυρία συσκευή (master) συνήθως ένα μικροελεγκτή, ο οποίος ελέγχει τις περιφερικές συσκευές.

Τυπικά ακολουθούνται τρεις κοινές γραμμές για όλες τις συσκευές:

- MISO (Master in Slave Out) Η γραμμή δευτερεύον (Slave) για την αποστολή δεδομένων προς την κυρία συσκευή (Master).
- MOSI (Master Out Slave In) Η κυρία γραμμή (Master) για την αποστολή δεδομένων με τα περιφερειακά (Slave).
- SCK (Serial Clock) οι παλμοί του ρολογιού που συγχρονίζουν τη μετάδοση δεδομένων οι παλμοί δίνονται από την master συσκευή στην περίπτωση μας, ο μικροελεγκτής.

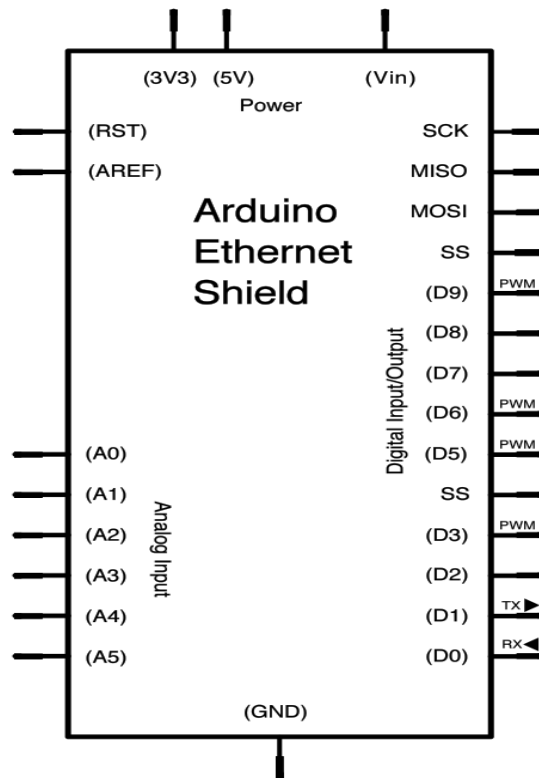
Και μια συγκεκριμένη γραμμή για κάθε συσκευή

- SS (Slave Select) Η ακίδα σε κάθε συσκευή που ο master μπορεί να χρησιμοποιήσει για να ενεργοποιήσει και να απενεργοποιήσει συγκεκριμένες συσκευές.

Όταν μια συσκευή δευτερεύον (slave) βρίσκετε σε κατάσταση όπου η επιλεγόμενη ακίδα είναι LOW, τότε παρέχει επικοινωνία με τον master, όταν είναι HIGH αγνοεί τον master, αυτό μας επιτρέπει να κάνουμε πολλαπλές συσκευές SPI που να μοιράζονται ίδια MISO, MOSI και τις γραμμές CLK.

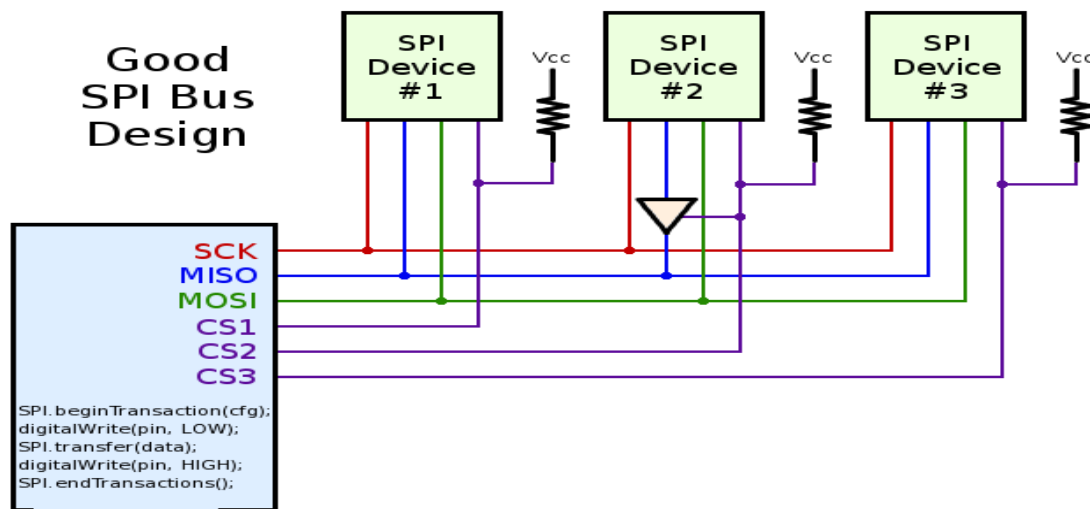


3.1.2 Βιβλιοθήκη Ethernet



Εικόνα 5 – Arduino Ethernet Shield schematic

Με το Arduino Ethernet Shield, και την Ethernet.h βιβλιοθήκη επιτρέπετε σε μια πλατφόρμα Arduino να έχει πρόσβαση στο internet. Μπορούμε τώρα να χρησιμοποιήσουμε την πλατφόρμα μας ως server που να δέχεται εισερχόμενες συνδέσεις ή σαν clients (πελάτη) για να κάνει εξερχόμενες συνδέσεις. Η βιβλιοθήκη μπορεί να υποστηρίζει ως και τέσσερις ταυτόχρονες συνδέσεις (εισερχομένη, εξερχομένη η συνδυασμούς). Η επικοινωνία της πλατφόρμας με το Shield πραγματοποιείται μέσω SPI bus.



Εικόνα 6 – SPI Bus connection



Αυτό ισχύει για της ακίδες (Pins) 11,12 και 13 στο Uno και για την πλατφόρμα Due μας τα pins 50,51 και 52. Και στις δυο πλακέτες το pin 10 είναι για να συνδέσουμε το SS. Στην Due, το pin 53 δεν πρέπει να το δεσμεύσουμε, αλλά πρέπει να το κρατήσουμε σαν έξοδο ειδάλλως το SPI interface δεν θα δουλέψει.

Θα χρειαστούμε τις παραπάνω βιβλιοθήκες κατά την έναρξη του κώδικα

```
#include <SPI.h>
#include <Ethernet.h>
```

3.2 Σύνδεση στο δίκτυο (web)

```
// Enter a MAC address and IP.
// The IP address will be dependent on your local network:
byte mac[] = { 0xDE, 0xAD, 0xBE, 0xEF, 0xFE, 0xED };
IPAddress ip(192, 168, 1,8);
// Initialize the Ethernet server library
// with the IP address and port you want to use
// (port 80 is default for HTTP):
EthernetServer server(80);
```

Για να πραγματώσουμε την σύνδεση μας δίνουμε τι Mac διεύθυνση DE,AD,BE,EF,FE,ED και έπειτα την IP διεύθυνση του εκάστοτε δικτύου, ένα βασικό κομμάτι είναι η δήλωση Ethernet server και ποια πύλη θα πάρουμε για την σύνδεση μας στο arduino due η πύλη για την http είναι το 80.

```
Ethernet.begin(mac, ip);
server.begin();
```

Με της παραπάνω εντολές βλέπουμε ότι καλούμε την συνάρτηση Ethernet.begin() προκειμένου να ξεκινήσει ο server. Επίσης να πούμε ότι Gateway και Subnet είναι επιλογής του χρήστη αν τα χρησιμοποιήσει.

3.4 Προστασία της σελίδας

```
if (c == '\n' && currentLineIsBlank) {
    //dXNlcjowMDA='user:000' (user:password) base64 encode
    // send a standard http response header
    if (strstr(header, "dXNlcjowMDA=") != NULL) {
client.println("HTTP/1.1 200 OK");
        client.println("Content-Type: text/html");
        client.println();
        .
        .
        .
    } else {
        // wrong user/pass
        client.println("HTTP/1.1 401 Unauthorized");
```



Ντάκος Δημήτρης : Έλεγχος τάσης φορτίου μέσω ιστοσελίδας ανακτημένη από πλατφόρμα Arduino DUE.

```
//client.println("HTTP/1.1 401 Authorization Required");
client.println("WWW-Authenticate: Basic realm=\"Secure\"");
client.println("Content-Type: text/html");
client.println();
client.println("<html>Text to send if user hits Cancel button</html>");
// really need this for the popup!
}
```

Πως λειτουργεί ο Web έλεγχος ταυτότητας η ροή της επικοινωνίας είναι:

Το πρόγραμμα περιήγησης κάνει μια αίτηση HTTP.

Ο διακομιστής web(web server)εκτελεί έναν έλεγχο ταυτότητας. Εάν αυτό δεν είναι επιτυχές απαιτείται ξανά έλεγχος ταυτότητας, ο διακομιστής στέλνει πίσω ένα μήνυμα λάθους (401 απαγόρευση) πανόμοιο με τα εξής:

a) Δεν έχετε δικαίωμα προβολής αυτής της σελίδας.

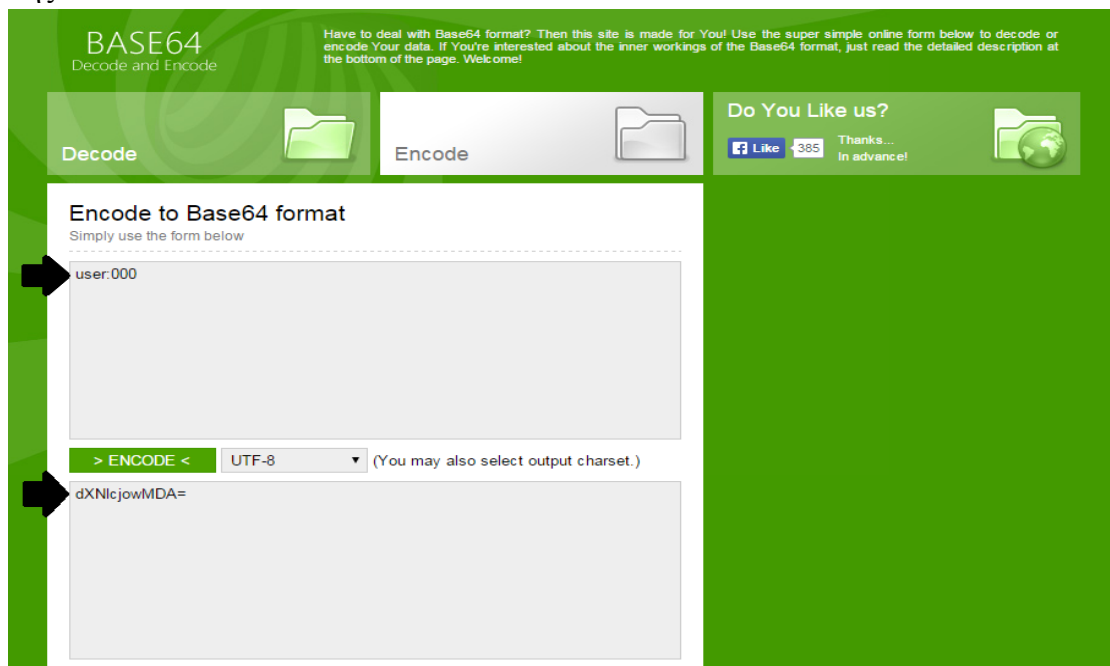
b) Δεν έχετε δικαίωμα προβολής αυτού του καταλόγου η της σελίδας με τα στοιχεία που πληκτρολογήσατε.

Το πρόγραμμα περιήγησης χρησιμοποιεί τον server για να δημιουργήσει μια νέα αίτηση που περιέχει πληροφορίες ελέγχου ταυτότητας.

Ο web server εκτελεί έναν έλεγχο ταυτότητας. Εάν ο έλεγχος είναι επιτυχής, ο web server στέλνει τα δεδομένα που ζητήθηκαν αρχικά από τον web server.

Αυτό το κομμάτι του κώδικα είναι αυτό που μας προστατεύει σε περιπτώσεις που κάποιος άγνωστος προσπαθήσει να εισέλθει στην σελίδα μας. Η επιλογή του κωδικού που επιθυμούμε για την σελίδα μας π.χ (όνομα: user κωδικός: 000).

Με βοήθεια του base64 encode έτσι κατασκευάσαμε μια κεφαλίδα εξουσιοδότησης.



Εικόνα 7 – Κατασκευή κεφαλίδας εξουσιοδότησης



Μια 401 απαγόρευση σημαίνει ότι δεν έχετε δικαιώματα να δείτε το αρχείο που ζητήθηκε. Μερικές φορές είναι σκόπιμο, άλλες φορές αυτό οφείλεται σε εσφαλμένα δικαιώματα.

3.5 Αρχείο CSS

Ένα αρχείο CSS είναι απαραίτητο για την εμφάνιση του περιβάλλοντος, έτσι θα μπορούμε να ελέγχουμε την εμφάνιση της σελίδας μας, για να μπορέσουμε να καλέσουμε το περιεχομένων του CSS από τον web hosting στον οποίο τοποθετήσαμε το έγγραφο, μας βοηθάει το παρακάτω απόσπασμα του κώδικα.

Μια μικρή αναφορά στο πως λειτουργεί το Dropbox ένας νέος χρήστης πρέπει να δημιουργήσει ένα λογαριασμό στην σελίδα της υπηρεσίας και έπειτα να κατεβάσει το λογισμικό του Dropbox. Μετά την ολοκλήρωση της εγκατάστασης ένας ειδικός φάκελος δημιουργείται, εκεί τοποθετούμε στην περίπτωση μας το CSS. Ο χρήστης μπορεί να έχει πρόσβαση ανά πασά στιγμή από Windows, Mac Linux, iPhone, iPad, Android και BlackBerry στα αρχεία του. Η απόφαση του συγκεκριμένου web hosting ήταν λόγω της οικειότητας του περιβάλλοντος. Ο καθένας μπορεί να χρησιμοποιήσει έναν άλλον.

Η εντολή που μας παρέχει την επικοινωνία με το αρχείο CSS από τον web hosting.

```
client.println("<link rel='shortcut icon'  
href='https://dl.dropboxusercontent.com/u/106805737/test.ico' />");
```

Το κομμάτι αυτό του κώδικα μας παρέχει την σύνδεση της σελίδας μας με Windows, Mac Linux, iPhone, iPad, Android και BlackBerry για τον έλεγχο της από την προσωπική μας συσκευή. Ουσιαστικά μπορούμε πλέον μέσω των εντολών που ακολουθούν να έχουμε καλύτερο περιβάλλον λειτουργίας και έλεγχου από τις παραπάνω συσκευές.

```
client.println("<meta name='apple-mobile-web-app-capable' content='yes'char  
set=utf-8 />");
```

```
client.println("<meta name='apple-mobile-web-app-status-bar-style' con-  
tent='black-translucent' />");
```

Ανεβάζαμε με τον ίδιο τρόπο το λογότυπο του TEI Κρήτης.

```
client.println("<link rel='stylesheet' type='text/css'  
href='https://dl.dropboxusercontent.com/s/yikaa27gj9r6ew4/Style.css'/>");
```



Ο κώδικας στο αρχείο CSS του dropbox:

```
#top{
  position: fixed;
  left: 0;
  right: 0;
  height: 117px
}
#top {
  top: 0;
  background-color: #778188;
}

body{
background-color: #C0C0C0;
text-align:center;
}
h1
{
text-align: center;
font-family:Arial, "Trebuchet MS", Helvetica, sans-serif;
font-size: 200% ;
}
h2
{
text-align: center;
font-family:Arial, "Trebuchet MS", Helvetica, sans-serif;
}

footer{
  position:absolute;
  bottom:0;
  opacity: 0.75;
}

div.relay{
font-family:Arial, "Trebuchet MS", Helvetica, sans-serif;
}
div.bot{
background-color: #778188;
}

div.alignRight{
text-align: right;
float:right;
margin-right:300;
}
div.alignLeft{
text-align: left;
float:left;
margin-left:300;
}

a.a1
{
text-decoration:none;
```



```
width:75px;
height:50px;
border-color:black;
border-top:2px solid;
border-bottom:2px solid;
border-right:2px solid;
border-left:2px solid;
border-radius:10px 10px 10px;
-o-border-radius:10px 10px 10px;
-webkit-border-radius:10px 10px 10px;
font-family:"Trebuchet MS",Arial, Helvetica, sans-serif;
-moz-border-radius:10px 10px 10px;
background-color:#000000;
padding:8px;
text-align:center;
}
a.a1:link {color:white;} /* unvisited link */
a.a1:visited {color:white;} /* visited link */
a.a1:hover {color:white;} /* mouse over link */
a.a1:active {color:white;} /* selected link */

a:link {color:black;} /* unvisited link */
a:visited {color:white;} /* visited link */
a:hover {color:white} /* mouse over link */
a:active {color:black;} /* selected link */

a.a2
{
text-decoration:none;
width:75px;
height:50px;
border-color:black;
border-top:2px solid;
border-bottom:2px solid;
border-right:2px solid;
border-left:2px solid;
border-radius:10px 10px 10px;
-o-border-radius:10px 10px 10px;
-webkit-border-radius:10px 10px 10px;
font-family:"Trebuchet MS",Arial, Helvetica, sans-serif;
-moz-border-radius:10px 10px 10px;
background-color:#000000;
padding:8px;
text-align:center;
}
a.a1:link {color:white;} /* unvisited link */
a.a1:visited {color:white;} /* visited link */
a.a1:hover {color:white;} /* mouse over link */
a.a1:active {color:white;} /* selected link */

a:link {color:black;} /* unvisited link */
a:visited {color:white;} /* visited link */
a:hover {color:white} /* mouse over link */
a:active {color:black;} /* selected link */
```



3.6 Έλεγχος του Arduino μέσω κουμπιών

Όπως θα δούμε στον κώδικα μας ο έλεγχος των κουμπιών αποτελείται από συνεχείς else if που βρίσκονται μέσα στην επανάληψη μας (void loop) έτσι σε κάθε αλλαγή της κατάστασης των κουμπιών ενημερώνετε και η σελίδα μας σε περίπτωση που επιλέγουμε τα κουμπιά 2 και 4 και είχαμε στην έξοδο ένα μετρητή έντασης θα βλέπαμε ότι στα ρελέ 2 και 4 θα είχε εφαρμοστεί η τάση των 5V DC που είναι επαρκείς για να διεγερθούν.

```
//Controls the Arduino if you press the buttons
if (readString.indexOf("?button-1-OFF") > 0) {
  digitalWrite(r1, HIGH);
  Serial.print("Relay 1 is OFF.");
  Serial.println();
}
else if (readString.indexOf("?button-1-ON") > 0) {
  digitalWrite(r1, LOW);
  Serial.print("Relay 1 is ON.");
  Serial.println();
}
else if (readString.indexOf("?button-2-OFF") > 0) {
  digitalWrite(r2, HIGH);
  Serial.print("Relay 2 is OFF.");
  Serial.println();
}
else if (readString.indexOf("?button-2-ON") > 0) {
  digitalWrite(r2, LOW);
  Serial.print("Relay 2 is ON.");
  Serial.println();
}
else if (readString.indexOf("?button-3-OFF") > 0) {
  digitalWrite(r3, HIGH);
  Serial.print("Relay 3 is OFF.");
  Serial.println();
}
else if (readString.indexOf("?button-3-ON") > 0) {
  digitalWrite(r3, LOW);
  Serial.print("Relay 3 is ON.");
  Serial.println();
}
if (readString.indexOf("?button-4-OFF") > 0) {
  digitalWrite(r4, HIGH);
  Serial.print("Relay 4 is OFF.");
  Serial.println();
}
```



```
}  
else if (readString.indexOf("?button-4-ON") > 0) {  
    digitalWrite(r4, LOW);  
    Serial.print("Relay 4 is ON.");  
    Serial.println();  
}  
    readString = ""; }  
}
```

3.7 Σελίδα HTML

Η γλώσσα του arduino βασίζεται στη γλώσσα Wiring, μια παραλλαγή της C/C++ για μικροελεγκτές αρχιτεκτονικής AVR, έτσι ξέρουμε ότι υποστηρίζει όλες τις βασικές δομές της C και C++. Για να μπορέσουμε να γράψουμε την ιστοσελίδα μας θα χρειαστούμε δυο βασικές εντολές την client.println() και την println() έτσι θα μπορούμε να εκτυπώσουμε στο πρόγραμμα περιήγησης web του πελάτη.

```
client.println("HTTP/1.1 200 OK");  
    client.println("Content-Type: text/html");  
    client.println();  
    client.println("<TITLE>Πτυχιακή Εργασία</TITLE>");  
    client.println("</HEAD>");  
    client.println("<BODY>");  
    client.println("<div id='top'>");  
    client.println("<H1>Έλεγχος τάσης φορτίου μέσω ιστοσελίδας αναρτημένη  
<br /> από πλατφόρμα Arduino DUE. </H1>");  
    client.println("<hr />");  
    client.println("<br />");  
    client.println("<H2>Πίνακας Ελέγχου.</H2>");  
    client.println("<br />");  
    client.println("<div class='alignLeft'>");  
    client.println("<div class='relay'>");  
    client.println("    Relay 1 ");  
    client.println("    <a class='a1' href='\"/?button-1-ON\"'>Turn ON</a>");  
    client.println("        <a class='a1' href='\"/?button-1-OFF\"'>Turn  
OFF</a><br /><br /><br />");  
    client.println("</div>");  
    client.println("<div class='relay'>");  
    client.println("    Relay 2 ");  
    client.println("    <a class='a1' href='\"/?button-2-ON\"'>Turn ON</a>");  
    client.println("        <a class='a1' href='\"/?button-2-OFF\"'>Turn  
OFF</a><br /><br /><br />");  
    client.println("</div>");  
    client.println("<br />");  
    client.println("</div>");
```



```
client.println("<div class='relay'>");
client.println("  Relay 3 ");
client.println("  <a class='a1' href='\"/?button-3-ON\"'>Turn ON</a>");
client.println("      <a class='a1' href='\"/?button-3-OFF\"'>Turn
OFF</a><br /><br /><br />");
client.println("</div>");
client.println("<div class='relay'>");
client.println("  Relay 4 ");
client.println("  <a class='a1' href='\"/?button-4-ON\"'>Turn ON</a>");
client.println("      <a class='a1' href='\"/?button-4-OFF\"'>Turn
OFF</a><br /><br /><br />");
client.println("</div>");
client.println("<br />");
client.println("</div>");
client.println("<footer>");
client.println("  Created by: Ntakos Dimitrios ");
client.println("</footer>");
client.println("</BODY>");
client.println("</HTML>");
```

3.8 Συνδεσμολογία με 4 relay και σχηματικό 4 relay

Για να συνδέσετε τον πίνακα 4 relay σε ένα Arduino είναι πολύ εύκολο και σας επιτρέπει να ενεργοποιήσετε και να απενεργοποιήσετε ένα ευρύ φάσμα συσκευών, τόσο AC και DC. Η πρώτη κίνηση είναι να συνδέσεις τις ακίδες (pins) του εδάφους (GND) και τη τροφοδοσίας, θα χρειαστεί να συνδέσετε από το Arduino τα + 5v στο σύμβολο VCC του 4 relay και το έδαφος (GND) του Arduino στο GND του 4 relay. Τότε είναι απλά θέμα να συνδέεις τις ακίδες επικοινωνίας, IN1, IN2, IN3 και IN4, της πλακέτας στα 4 pins δεδομένων του Arduino.

Στον κώδικα παραπάνω χρησιμοποιήσαμε τα Arduino pins 5, 6, 7, 8. αποφεύγουμε να χρησιμοποιούμε τα pins των δεδομένων 0 και 1, για τον λόγο ότι αυτές χρησιμοποιούνται από την Arduino για σειριακή επικοινωνία και μπορεί να προκαλέσει προβλήματα κατά τη μεταφόρτωση κώδικα στο Arduino.

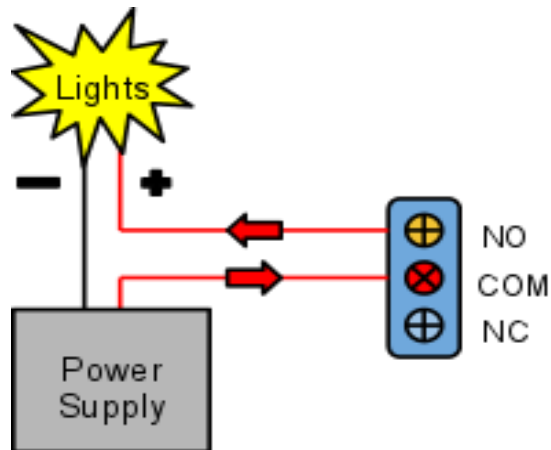
Η προεπιλεγμένη κατάσταση του ρελέ, όταν η συσκευή είναι απενεργοποιημένη πρέπει να είναι σε συνδέσει με NO (κανονικά ανοικτές) επαφές, πρόκειται για ένα χαρακτηριστικό ασφαλείας έτσι αν χρησιμοποιηθεί η υποδοχή NO και το Arduino χάνει την εξουσία θα απενεργοποιούνται αυτόματα όλες τις συσκευές που συνδέονται

στο ρελέ σε NO επαφές.

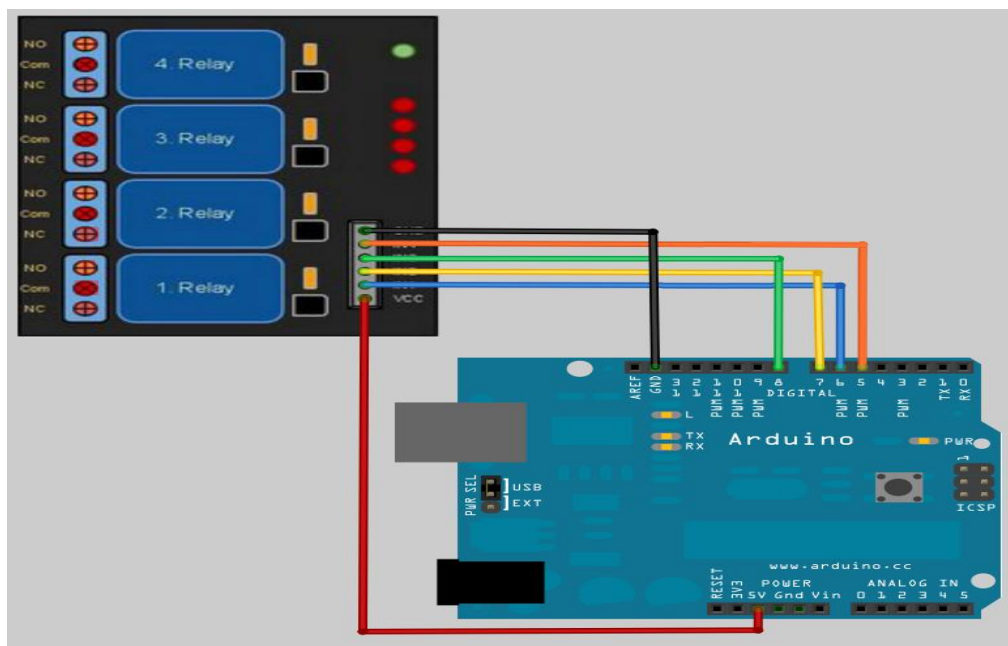
Όταν έχετε κάτι που συνδέεται στο ρελέ με NC (κανονικά κλειστή) επαφές τότε όταν το Arduino χάνει την εξουσία θα ενεργοποιούνται αυτόματα όλες τις συσκευές που συνδέονται στο ρελέ σε NC επαφές.



Ντάκος Δημήτρης : Έλεγχος τάσης φορτίου μέσω ιστοσελίδας ανακτημένη από πλατφόρμα Arduino DUE.



Πάνω: Παράδειγμα σύνδεσης ρεύματος με τα φώτα σε ένα ρελέ

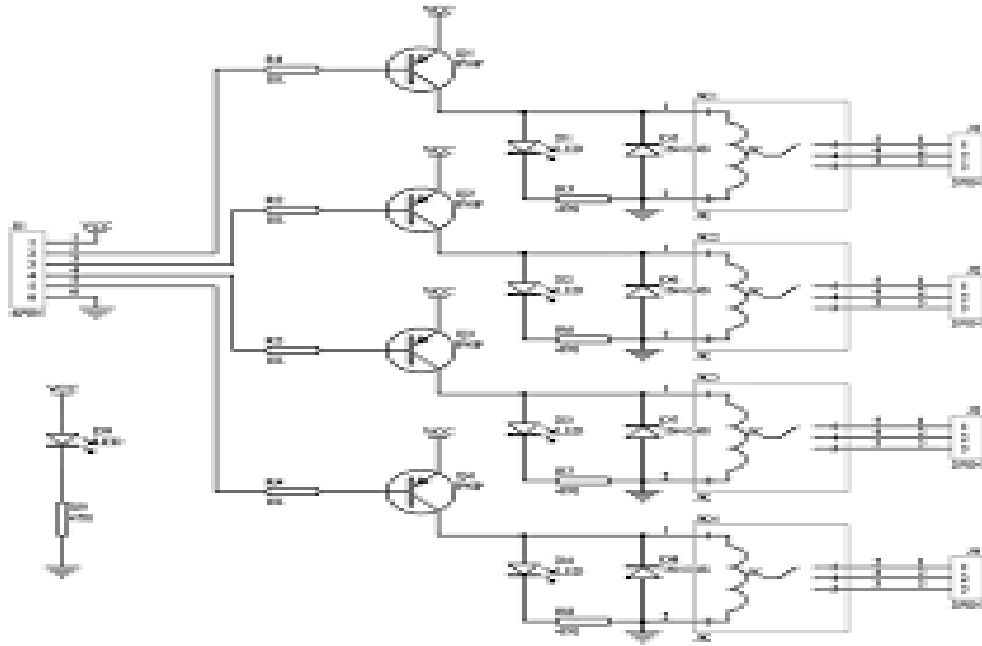


Εικόνα 8 – Συνδεσμολογία με 4 relay

Μπορεί να ελέγχεται άμεσα από ένα ευρύ φάσμα μικροελεγκτών

- Τροφοδοσία: 5V Εξωτερικού
- 4 μηχανικά ρελέ με LED κατάστασης
- "NC" και "NO" θύρες για κάθε ρελέ





Εικόνα 9 – Σχηματικό του 4 relay

3.9 Τεχνολογίες αιχμής

Η υπολογιστές έγιναν μέρος τις καθημερινότητας μας αλλά ακόμα για ορισμένους είναι μεγάλη η πολυπλοκότητα και η δυσκολία στην εκμάθηση του προγραμματισμού. Η intel, παγκοσμίως γνωστή ως η μεγαλύτερη εταιρία κατασκευής επεξεργαστών, συνεργάζεται με τους ιδρυτές του arduino, για την υλοποίηση του arduino 101, πρόκειται για μια πλακέτα οικονομική και είναι ιδανική για τους νέους, αλλά και για εκπαιδευτικούς σκοπούς. Η νέα αυτή πλατφόρμα θα είναι η πρώτη με το μικροσκοπικό, χαμηλής κατανάλωσης ολοκληρωμένο (intel curie).

Το arduino είναι μια πλατφόρμα που αφορά ένα ερευνητικό στάδιο και έχει τα τελευταία χρόνια πολλές επιτυχίες σε διάφορους κλάδους της ερευνάς αυτό μπορούμε να το δούμε και από της πτυχιακές των νέων που έχουν δείξει ότι στρέφονται σε τεχνολογίες αιχμής τα τελευταία χρονιά που δεν έχουν να κάνουν μόνο με ένα συγκεκριμένο κομμάτι ερευνάς. Μερικά παραδείγματα από τμήματα στην Ελλάδα που ασχολούνται και χρησιμοποίησαν ίδια πλατφόρμα με εμάς είναι:

Το τμήμα βιοϊατρικής τεχνολογίας, μερικές από της εφαρμογές:

A) Ηλεκτρονική πρότυπη συσκευή με μικροελεγκτή για την καταγραφή και απεικόνιση μετρήσεων πίεσης και υποπίεσης κατά την διάρκεια μηχανικής υποστήριξης της αναπνοής σε ασθενείς μερικά υποστηριζόμενους.

B) Κατασκευή συστήματος εντοπισμού θέσης ασθενών με άνοια.

Γ) Κατασκευή αυτοκινούμενης συσκευής (ρομπότ) με δυνατότητα αναγνώρισης εμποδίων.



Εικόνα 10 – Personal Energy Orb

Αλλά τα περισσότερα τμήματα που είδαμε ήταν τα τμήματα ρομποτικής με πληθώρα εφαρμογών άλλωστε είναι και από τους κλάδους που το arduino διαπρέπει, μερικά από τα πολλά παραδείγματα είναι: ρομποτικά οχήματα, ρομποτικά χέρια, Quadcopter, μηχανές διαλογής ένα παράδειγμα είναι και το Personal Energy Orb βασισμένο στο Arduino μετράει πόση ώρα περνάει κάποιος μπροστά στον υπολογιστή. Η ενέργεια του Personal Energy Orb υπολογίζεται με το πόσες ώρες έχει περάσει κάποιος μπροστά στον υπολογιστή σε σχέση με τις ώρες που έχει περάσει εκτός. Μόλις η ενέργεια τελειώσει, ο δείκτης του υπολογιστή γίνεται ολοένα και πιο αργός. Για να επανέλθει θα πρέπει ο χρήστης να φορτίσει το Personal Energy Orb, πηγαίνοντας απλά μια βόλτα με το ποδήλατό του. Μια αρκετά έξυπνη ιδέα που μας δείχνει τι μπορεί να κάνει κάποιος χρησιμοποιώντας open source hardware και εργαλεία όπως το Arduino.

Όπως το περιγράφει ο δημιουργός του, το Arduino είναι μια «ανοικτού κώδικα» πλατφόρμα «πρωτοτυποποίησης» ηλεκτρονικών βασισμένη σε ευέλικτο και εύκολο στη χρήση hardware και software που προορίζεται για οποιονδήποτε έχει λίγη προγραμματιστική εμπειρία, στοιχειώδεις γνώσεις ηλεκτρονικών και ενδιαφέρεται να δημιουργήσει διαδραστικά αντικείμενα ή περιβάλλοντα. Στην ουσία, πρόκειται για ένα ηλεκτρονικό κύκλωμα που βασίζεται στον μικροελεγκτή ATmega της Atmel και του οποίου όλα τα σχέδια, καθώς και το software που χρειάζεται για την λειτουργία του, διανέμονται ελεύθερα και δωρεάν ώστε να μπορεί να κατασκευαστεί από τον καθένα (απ' όπου και ο περίεργος -για hardware- χαρακτηρισμός «ανοικτού κώδικα»). Αφού κατασκευαστεί, μπορεί να συμπεριφερθεί σαν ένας μικροσκοπικός υπολογιστής, αφού ο χρήστης μπορεί να συνδέσει επάνω του πολλαπλές μονάδες εισόδου/εξόδου και να προγραμματίσει τον μικροελεγκτή να δέχεται δεδομένα από τις μονάδες εισόδου, να τα επεξεργάζεται και να στέλνει κατάλληλες εντολές στις μονάδες εξόδου. Μάλιστα κάποιος θα μπορούσε να ισχυριστεί - και θα ήταν ένας αρκετά πετυχημένος παραλληλισμός - ότι λειτουργικά το Arduino μοιάζει πολύ με το NXT Brick των Lego Mindstorms NXT. Άλλωστε η ρομποτική είναι μια από τις πολλές εφαρμογές στις οποίες το Arduino διαπρέπει.

Το Arduino βέβαια, δεν είναι ούτε ο μοναδικός, ούτε και ο καλύτερος δυνατός τρόπος για την δημιουργία μιας οποιασδήποτε διαδραστικής ηλεκτρονικής συσκευής. Όμως το κύριο πλεονέκτημά του είναι η τεράστια κοινότητα που το υποστηρίζει και η οποία



Ντάκος Δημήτρης : Έλεγχος τάσης φορτίου μέσω ιστοσελίδας ανακτημένη από πλατφόρμα Arduino DUE.

έχει δημιουργήσει, συντηρεί και επεκτείνει μια ανάλογου μεγέθους online γνωσιακή βάση. Έτσι, παρότι ένας έμπειρος ηλεκτρονικός μπορεί να προτιμήσει διαφορετική πλατφόρμα ή εξαρτήματα ανάλογα με την εφαρμογή που έχει στον νου του, το Arduino, με το εκτενές documentation, καταφέρνει να κερδίσει όλους αυτούς των οποίων οι γνώσεις στα ηλεκτρονικά περιορίζονται στα όσα λίγα έμαθαν στο σχολείο.



Κεφάλαιο 4⁰

4.ΣΥΜΠΕΡΑΣΜΑΤΑ

4.1 Αποτελέσματα

Κατά την ολοκλήρωση της εφαρμογής μας ,είχαμε σαν στόχο να γίνει προσιτή προς τον χρήστη η επικοινωνία με τον μικροελεγκτή, με λίγα λόγια ο τελικός χρήστης να μπορεί μέσω κινητού η μέσω διαφορών συσκευών που συνδέονται στο διαδίκτυο και έχουν πρόσβαση σε κάποιον φυλλομετρήτη (Web Browser) να έχει τον έλεγχο τον ρελέ στα οποία είναι συνδεδεμένα ς τα φορτία της επιλογής του.

Ο στόχος μας κατά την ολοκλήρωση του συστήματος επιτεύχθηκε, κατασκευάσαμε ένα σύστημα αυτοματοποιημένο και εύχρηστο σε πολλές και ποικίλες εφαρμογές, έχοντας πρόσβαση απλά σε ένα υπολογιστή η ένα κινητό. Η συσκευή δοκιμάστηκε εφαρμόζοντας φορτίο 40W υπό τάση 220V(οικιακού φωτισμού), υπάρχει περιθώριο επέκτασης και ευελιξίας στην χρήση του προγράμματος και σε ποιο εξειδικευμένες χρήσεις, όπως βιομηχανική, αγροτική, οικιακή και πολλά αλλά πάντα με της κατάλληλες μετατροπές στον κώδικα και τα καταλληλά hardware κομμάτια.

4.2 Δυσκολίες – Προβλήματα

Ξεκινώντας με την ανάπτυξη ενός προγράμματος με μεγάλη πολυπλοκότητα, στο Arduino μπορείς να περιμείνεις ότι θα αποτελέσει πρόβλημα το ποσό περιορισμένου πόρους έχει ο μικροελεγκτής. Έτσι λοιπόν στην πράξη κιόλας αντιληφθήκαμε το ποσό γρηγορά η μνήμη RAM και η Flash εξαντλούνται ανάλογα με το ποσό πολύπλοκο είναι τα συστήματα που επιχειρούμε.

Το μέγεθος της μνήμης Flash στο οποίο γράφουμε και το εκτελέσιμο πρόγραμμα μέσα στο Arduino, επίσης εξαντλείται πολύ γρήγορα λόγω της χρήσης των βιβλιοθηκών του Arduino. Υπάρχουν βιβλιοθήκες που αυξάνουν το μέγεθος του εκτελέσιμου κατά πολλά KB. Κάποιος που θα θελήσει να υλοποιήσει ένα σύστημα που να κάνει τα πάντα πιθανών να βρεθεί σε δύσκολη θέση, ίσως να χρειαστεί και το ενδεχόμενο περικοπών κάποιων χαρακτηριστικών.

Για να κρατήσουμε σε χαμηλά επίπεδα τη χρήση μνήμης RAM, μπορεί να γίνει χρήση του PROGMEM που μας δίνει την δυνατότητα ανάγνωσης αλφαριθμητικών χωρίς να απαιτείται αντιγραφή στην μνήμη RAM, ακόμα περισσότερη μνήμη μπορούμε να εξοικονομήσουμε με χρήση μικρότερων τύπων ακεραίων, π.χ. uint8_t αντί του συνηθισμένου int.

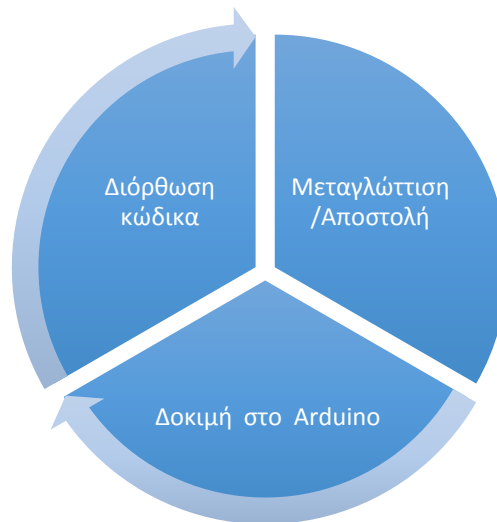
Μια ακόμα παρατήρηση για της βιβλιοθήκες είναι ότι σε χώρους συζητήσεων είναι αποδεκτό ότι οι περισσότερες βιβλιοθήκες δεν είναι γραμμένες από καθαρούς προγραμματιστές, αλλά μάλλον από άλλους προερχόμενους από τον χώρο των ηλεκτρονικών και των ψηφιακών συστημάτων. Ακόμα και σήμερα που το arduino είναι αρκετά διαδεδομένο και έχει αμέτρητους υποστηρικτές περιέχει σφάλματα και ορισμένες φορές δεν προσφέρει την καλύτερη λειτουργικότητα, αυτό αποτελεί και την



επιβράδυνση της διαδικασίας ανάπτυξης, ενώ μερικές φορές είναι απαραίτητη η προσαρμογή τις εφαρμογής.

4.3 Ανάπτυξη και Αποσφαλμάτωση

Η ανάπτυξη του λογισμικού για την πλατφόρμα μας έχει των παρακάτω τυπικό κύκλο:



Εικόνα 11 – Τυπικός κύκλος ανάπτυξης λογισμικού για το Arduino

Η διαδικασία φαίνεται απλή και το ολοκληρωμένο περιβάλλον ανάπτυξης του Arduino την έκανε αρκετά πιο αυτοματοποιημένη. Σε περίπτωση όμως που το σύστημα μας αρχίζει να μεγαλώνει σε μέγεθος, τότε η διαδικασία είναι ιδιαίτερα χρονοβόρα με μεγάλους χρόνους αναμονής, μπορεί να χρειαστώ και μισό λεπτό για να μεταγλωττιστεί και να αποσταλεί μέσω σειριακής στο Arduino. Αυτό μπορεί να καταστήσει την διαδικασία ανάπτυξης χρονοβόρα, ειδικά σε περιπτώσεις που ψάχνουμε κάποιο σφάλμα στο πρόγραμμα μας (bug hunting), οπότε μπορεί να κάνουμε πολλές επαναλήψεις/δοκιμές για μικρές αλλαγές του κώδικα κάθε φορά.

Το έργο της αποσφαλμάτωσης του κώδικα μπορεί να γίνει εξαιρετικά δύσκολο αρκετά συχνά. Το γεγονός ότι ο εκτελέσιμος κώδικας τρέχει σε διαφορετική μηχανή από αυτήν του περιβάλλοντος ανάπτυξης καθιστά αδύνατη τη βήμα προς βήμα εκτέλεση ή παρακολούθηση των μεταβλητών. Η αποσφαλμάτωση έτσι βασίζεται σε μεγάλο βαθμό στην εκτύπωση πληροφοριών στην σειριακή θύρα (print debugging). Πολύ σημαντική είναι η τακτική παρακολούθηση της διαθέσιμης μνήμης RAM, καθώς όπως ήδη αναφέρθηκε, εξαντλείται πολύ γρήγορα.

4.4 Αδυναμίες

Το σύστημα έχει δημιουργηθεί με το πρωτόκολλο HTTP, το οποίο έχει σαν επακολουθώ όταν είναι στην αρχική του μορφή χωρίς δηλαδή την χρήση κρυπτογραφίας είναι ιδιαίτερα ευπαθές σε επιθέσεις υποκλοπής και σε ενδιάμεσους. Αυτό με λίγα λόγια σημαίνει ότι οποιοσδήποτε μπορεί να παρεμβληθεί αναμεσα σε μας και τη μονάδα Arduino, μπορεί να υποκλέψει τις εντολές που στέλνουμε και τις απατήσεις που λαμβάνουμε, αλλά όχι μόνο αυτό μπορεί και να τις τροποποιήσει συστηματικά. Με άλλα λόγια τίθεται σε κίνδυνο η εμπιστευτικότητα και η ακεραιότητα της μεταξύ επικοινωνίας του πελάτη με το Arduino.

Να επισημανθεί ότι στην μορφή που αναφέρουμε δεν έχει εφαρμοστεί καν πιστοποίηση πρόσβασης (basic access authentication) οπου προσθέσαμε στο τελικό στάδιο για να μην είμαστε εντελώς ευάλωτοι. Βέβαια το στάδιο αυτό θα ήταν αρκετά πιο ικανό στις απαιτήσεις μας αν γίνει από ένα καταρτισμένο άτομο στο πεδίο τις ασφάλειας.

4.5 Διαφορές και προτεινόμενες εκδόσεις του Arduino

Το **Arduino Diecimila** :

1) Βασίζεται στον μικροελεγκτή ATmega168, ο οποίος διαθέτει μνήμη από 1Kb SRAM, 512bytes EEPROM και 16Kb Flash (14 ελεύθερα λόγω του bootloader).

2) Δεν επιλέγει αυτόματα μεταξύ της εξωτερικής τροφοδοσίας και της τροφοδοσίας μέσω της θύρας USB. Το Diecimila διαθέτει ειδικό jumper με το οποίο μπορείτε να επιλέξετε χειροκίνητα την πηγή τροφοδοσίας.

Το **Arduino Mega** είναι η πιο εξελιγμένη έκδοση με τον μικροελεγκτή ATmega1280 και αρκετά μεγαλύτερο μέγεθος:

- Μνήμη 8Kb SRAM, 4Kb EEPROM, 128Kb Flash
- Ψηφιακά pin εισόδου/εξόδου 54
- Pin αναλογικής εισόδου 16
- Υποστήριξη ψευδοαναλογικής εξόδου PWM σε ψηφιακά pin 14 PWM pin
- Υποστήριξη εξωτερικού interrupt σε ψηφιακά pin 6 interrupt
- Σειριακά interface 4 από τα οποία το ένα προωθείται στον ελεγκτή Serial-Over-USB για σύνδεση με τον υπολογιστή.

Σημειώστε ότι το Arduino Mega είναι συμβατό με τα περισσότερα shield που έχουν κυκλοφορήσει για το Arduino αλλά όχι με το Ethernet Shield, το οποίο είναι ένα αρκετά σημαντικό μειονέκτημα για όσους θέλουν να φτιάξουν εφαρμογές με πρόσβαση στο Internet.



4.6 Χρήσης της Εφαρμογής

Όσον αφορά την πρακτικές εφαρμογές της πτυχιακής κατασκευάσαμε μια πλατφόρμα η οποία είναι χρηστικότερη και ιδιαίτερα επεκτάσιμη. Κατά την υλοποίηση της δημιουργηθήκαν πολλές ιδέες για εφαρμογές (όπως έξυπνο σπίτι, αγροτικές). Καθώς ο τρόπος που το κατασκευάσαμε την καθιστά ικανή να δέχεται και να επεξεργάζεται πολλά σήματα. Στην ουσία θα μπορείτε να χειριστείτε τα πάντα από το Android κινητό/τάμπλετ σας. Από τον φωτισμό και την μουσική, μέχρι την καφετιέρα,

Επίσης θα μπορείτε να έχετε ανά πάσα στιγμή εικόνα του τι γίνεται στο σπίτι σας, καθώς και να ορίσετε ενέργειες που θα γίνονται αυτόματα, για παράδειγμα:

Αν η θερμοκρασία ανέβει πάνω από 50 βαθμούς Κελσίου, να ενεργοποιηθεί το σύστημα πυρόσβεσης κτλ. Ο μόνος περιορισμός στο τι μπορείτε να κάνετε είναι ο χρόνος και τα χρήματα που θέλετε να διαθέσετε στο Project.

4.7 Μελλοντικές Επέκτασης

Η Ethernet Shield με την οποία έχουμε την δυνατότητα να συνδεόμαστε στο δίκτυο, περιλαμβάνει και ένα αναγνώστη καρτών Micro-SD (Card Reader).



Εικόνα 12 – Ethernet Shield με Micro-SD Card Reader

Αυτό θα έδινε στην μονάδα μας μεγαλύτερα ποσοστά περιφερειακής μνήμης απλά βάζοντας στην υποδοχή του αναγνώστη την κάρτα Micro-SD, ενώ με της κατάλληλες βιβλιοθήκες ο Arduino μπορεί να αποκτήσει πρόσβαση στο σύστημα αρχείων (file system) της κάρτας. Η συγχρονίες κάρτες Micro-SD διαθέτουν χωρητικότητες της τάξης των GB ενώ κοστίζουν μόλις μερικά ευρώ.

Για να χρησιμοποιήσουμε την εξωτερική Micro-SD κάρτα μέσα από ένα πρόγραμμα για Arduino θα πρέπει να συμπεριλάβουμε πρώτα την βιβλιοθήκη SD.h, η οποία ορίζει τις κλήσεις SD και Files για την πρόσβαση στο σύστημα αρχείων.

```
#include < SD.h>
```



Ο αναγνώστης κάρτας Micro-SD είναι περιφερειακή ομάδα και επικοινωνεί με τον Arduino μέσω του δίαυλου SPI που έχουμε αναφέρει πιο πάνω.

Πέρα από την επέκταση μνήμης μπορούμε να χρησιμοποιήσουμε την κάρτα ως μέσω ανιχνεύσεις της επισκευσιμότητας της σελίδας με τις κατάλληλες ρυθμίσεις και το τι αλλαγές έκανε ο πελάτης στην σελίδα μας. Βέβαια για να είναι εφικτό κάτι τέτοιο θα χρειαστούμε ένα ρολόι πραγματικού χρόνου (Real Time Clock, RTC) το οποίο τροφοδοτείται από ανεξάρτητη μπαταρία και γι' αυτό πρέπει να έχει πολύ χαμηλή κατανάλωση ρεύματος για μεγαλύτερη διάρκεια καταγραφής του χρόνου. Η πλατφόρμα δεν έχει ενσωματωμένο ρολόι.

4.8 Εντυπώσεις

Με την ενασχόληση μου στην παρούσα πτυχιακή εργασία απέκτησα γνώση πάνω σε ένα ενδιαφέρον και εύχρηστο εργαλείο. Με βοήθησε να εμβαθύνω τις γνώσεις μου στον προγραμματισμό του arduino και να καταπιαστώ με πιο προχωρημένα ζητήματα που ξεφεύγουν από τις βασικές χρήσεις του μικροελεγκτή αυτού. Τελικά θα μπορούσαμε να πούμε ότι ήταν μια πετυχημένη ιδέα από την αρχή μέχρι το τέλος της και δίνει την δυνατότητα και σε άλλους φοιτητές να την επεκτείνουν και να την εξειδικεύσουν.

Το Arduino αποτελεί μια μικρή επανάσταση στον κόσμο της ρομποτικής. Η χρήση της Wiring ως γλώσσα προγραμματισμού, η εύκολη διασύνδεσή του με διάφορους αισθητήρες ή άλλες πλακέτες το καθιστούν ελκυστικό από άτομα που δεν είναι εξοικειωμένα με την ρομποτική. Η δημιουργία ενός ασύρματου δικτύου είναι μόνο η κορυφή του παγόβουνου καθώς οι δυνατότητες του εξελίσσονται με τέτοιο τρόπο, που το μόνο πλέον που μπορεί να το περιορίσει είναι η ανθρώπινη φαντασία.



4.9 Βιβλιογραφία – Πηγες Πληροφοριών

Web:

- 1) <http://forum.arduino.cc/index.php/topic,13134.0.htm>
- 2) <https://www.arduino.cc/en/Guide/ArduinoEthernetShield>
- 3) <https://www.processing.org/>
- 4) <https://www.arduino.cc/>
- 5) <https://www.arduino.cc/en/Tutorial/HomePage>
- 6) <https://forum.arduino.cc/>
- 7) <https://github.com/arduino/Arduino/>
- 8) <https://en.wikipedia.org/wiki/Arduino>
- 9) https://en.wikipedia.org/wiki/Basic_access_authentication
- 10) <http://www.ladyada.net/learn/arduino/>
- 11) <http://www.jeremyblum.com/category/arduino-tutorials/>
- 12) <http://deltahacker.gr/arduino-intro/>

Βιβλιογραφία:

- 1) Ανάπτυξη Εφαρμογών με το arduino, Παναγιώτης Παπάζογλου M.Sc., Ph.D. Σπύρος-Πολυχρόνης Λιωνής, M.Sc. εκδόσεις Τζιόλα.
- 2) Programming your home, automate with Arduino, Android, and your computer, Mike Riley
- 3) Building Wireless Sensor Networks , Robert Faludi εκδόσεις O'REILLY
- 4) Getting started with arduino, Massimo Banzì εκδόσεις O'Reilly Media
- 5) Arduino Cookbook, 2nd Edition εκδόσεις O'Reilly Media

