

ΤΕΧΝΟΛΟΓΙΚΟ ΕΚΠΑΙΔΕΥΤΙΚΟ ΙΔΡΥΜΑ ΔΥΤΙΚΗΣ ΕΛΛΑΔΑΣ
ΣΧΟΛΗ ΤΕΧΝΟΛΟΓΙΚΩΝ ΕΦΑΡΜΟΓΩΝ
ΤΜΗΜΑ ΜΗΧΑΝΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ Τ.Ε.

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

ΣΧΕΔΙΑΣΜΟΣ ΚΑΙ ΚΑΤΑΣΚΕΥΗ ΗΛΕΚΤΡΟΝΙΚΗΣ ΠΛΑΚΕΤΑΣ ΓΙΑ ΑΠΟΜΑΚΡΥΣΜΕΝΟ ΕΛΕΓΧΟ «ΕΞΥΠΝΟΥ ΣΠΙΤΙΟΥ» ΜΕ ΤΗ ΧΡΗΣΗ ΗΛΕΚΤΡΟΝΙΚΟΥ ΥΠΟΛΟΓΙΣΤΗ



ΣΠΟΥΔΑΣΤΗΣ: ΑΒΡΑΜΠΟΣ ΘΕΟΔΩΡΟΣ (Α.Μ. 6017)

ΕΠΙΒΛΕΠΩΝ ΚΑΘΗΓΗΤΗΣ: ΚΑΛΑΡΑΚΗΣ ΑΛΕΞΑΝΔΡΟΣ

ΚΑΘΗΓΗΤΗΣ ΕΦΑΡΜΟΓΩΝ

ΠΑΤΡΑ 2016

ΠΡΟΛΟΓΟΣ

Το παρόν τεύχος αποτελεί την Πτυχιακή Εργασία που εκπονήθηκε στο Τμήμα Μηχανολόγων Μηχανικών Τ.Ε. του Τεχνολογικού Εκπαιδευτικού Ιδρύματος Δυτικής Ελλάδας και αναφέρεται στην δημιουργία ενός έξυπνου σπιτιού με μικροελεγκτή ARDUINO.

Ο μικροελεγκτής ARDUINO μας δίνει την δυνατότητα να ελέγχουμε διάφορες συσκευές καθώς τον κεντρικό φωτισμό του σπιτιού μας από τον Η/Υ ακόμα και από το έξυπνο κινητό μας (smart phone) ή το tablet μας.

Στην αρχή δημιουργήσαμε και προγραμματίσαμε τον μικροελεγκτή ARDUINO μαζί με την Ethernet shield με σκοπό να ελέγχουμε τις συσκευές μας με ένα τοπικό web server. Στην συνέχεια δημιουργήσαμε και προγραμματίσαμε ένα δεύτερο μικροελεγκτή ARDUINO με σκοπό να ανοίγουμε και να κλείνουμε την κεντρική είσοδο του σπιτιού μας ηλεκτρολογώντας διάφορους κωδικούς που έχουμε επιλέξει. Τέλος δημιουργήσαμε ένα τρίτο μικροελεγκτή με σκοπό να ανάβουν αυτόματα τα φώτα στο έξυπνο σπίτι μας στον εξωτερικό χώρο μετά την δύση του ηλίου. Όλα τα παραπάνω τοποθετήθηκαν σε μία μακέτα.

Ευχαριστώ θερμά τον Επιβλέποντα Καθηγητή μου κ. Αλέξανδρο Καλαράκη Καθηγητή Εφαρμογών του Τμήματος Μηχανολόγων Μηχανικών Τ.Ε., για την πολύτιμη βοήθεια και καθοδήγηση που μου προσέφερε για την πραγματοποίηση της Εργασίας και για την ευκαιρία που μου έδωσε να δημιουργήσω ένα έξυπνο σπίτι.

Αβράμπος Θεόδωρος
Ιούλιος 2016

Υπεύθυνη Δήλωση Σπουδαστή: Ο κάτωθι υπογεγραμμένος σπουδαστής έχω επίγνωση των συνεπειών του Νόμου περί λογοκλοπής και δηλώνω υπεύθυνα ότι είμαι συγγραφέας αυτής της Πτυχιακής Εργασίας, έχω δε αναφέρει στην Βιβλιογραφία μου όλες τις πηγές τις οποίες χρησιμοποίησα και έλαβα ιδέες ή δεδομένα. Δηλώνω επίσης ότι, οποιοδήποτε στοιχείο ή κείμενο το οποίο έχω ενσωματώσει στην εργασία μου προερχόμενο από Βιβλία ή άλλες εργασίες ή το διαδίκτυο, γραμμένο ακριβώς ή παραφρασμένο, το έχω πλήρως αναγνωρίσει ως πνευματικό έργο άλλου συγγραφέα και έχω αναφέρει ανελλιπώς το όνομά του και την πηγή προέλευσης.

Ο σπουδαστής

(Ονοματεπώνυμο)

.....

(Υπογραφή)

ΠΕΡΙΛΗΨΗ

Η παρούσα πτυχιακή εργασία έχει ως σκοπό την κατασκευή ενός έξυπνου σπιτιού με τον μικροϋπολογιστή - μικροελεγκτή ARDUINO με σκοπό τον έλεγχο συσκευών και του κεντρικού φωτισμού από τον ηλεκτρονικό μας υπολογιστή είτε το έξυπνο κινητό μας ή το tablet μας και διάφορων επιπλέον αυτοματισμών.

Η ανάπτυξη του θέματος γίνεται σε 5 κεφάλαια. Στο πρώτο κεφάλαιο εξηγούμε τι είναι ο μικροϋπολογιστής - μικροελεγκτής ARDUINO, οι λόγοι που επιλέξαμε τον μικροελεγκτή ARDUINO, τι είναι ο μικροελεγκτής AT mega328, περιγραφή πλακέτας ARDUINO UNO και την εγκατάσταση του προγράμματος από την επίσημη ιστοσελίδα. Επίσης εξηγούμε τι είναι η ETHERNET Shield και περιγράφουμε την πλακέτα, εξηγούμε διάφορα εξαρτήματα που δέχεται ο μικροελεγκτής και τέλος εξηγούμε πως μπορούμε να βελτιώσουμε το ασύρματο δίκτυό μας.

Στο δεύτερο κεφάλαιο συνδέουμε τον μικροελεγκτή ARDUINO με την ETHERNET Shield, παρουσιάζουμε την συνδεσμολογία, εγκαθιστούμε στο βασικό πρόγραμμα του ARDUINO τις βιβλιοθήκες για το Web Server μας ,αναπτύσσουμε το πρόγραμμα μας και τέλος τον HTML κώδικα ώστε να δημιουργήσουμε ένα κεντρικό πάνελ διαχείρισης συσκευών και φωτισμού στον υπολογιστή μας .

Στο τρίτο κεφάλαιο κατασκευάζουμε μια έξυπνη κλειδαριά για την κεντρική είσοδο στο έξυπνο σπίτι μας, εξηγούμε τι είναι το keypad και τι ο σερβοκινητήρας που χρησιμοποιούμε στη κατασκευή μας. Συνδέουμε το keypad και τον σερβοκινητήρα με το ARDUINO και τέλος εγκαθιστούμε τις βιβλιοθήκες για το keypad και δημιουργούμε το πρόγραμμα μας.

Στο τέταρτο κεφάλαιο δημιουργούμε ένα κύκλωμα με αισθητήρα φωτός με σκοπό τον φωτισμό εξωτερικού χώρου μετά την δύση του ηλίου, εξηγούμε επίσης τι είναι ο αισθητήρας φωτός, παρουσιάζουμε την συνδεσμολογία και τέλος δημιουργούμε το πρόγραμμα μας.

Στο πέμπτο και τελευταίο κεφάλαιο μας κατασκευάζουμε μια μονοκατοικία σε μακέτα με σκοπό να εγκαταστήσουμε όλα τα παραπάνω σε αυτή ώστε να την κάνουμε λειτουργική.

ΠΕΡΙΕΧΟΜΕΝΑ

ΠΡΟΛΟΓΟΣ

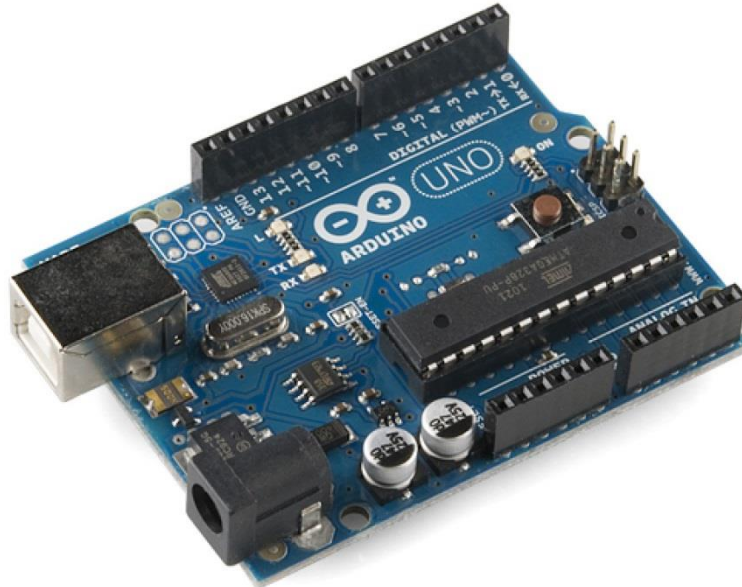
ΠΕΡΙΛΗΨΗ

ΠΕΡΙΕΧΟΜΕΝΑ

1.	Ο μικροελεγκτής Arduino	1
1.1	Τι είναι το Arduino	1
1.2	Λόγοι επιλογής Arduino	2
1.3	Τι είναι το έξυπνο σπίτι	3
1.4	Τι είναι ο μικροελεγκτής AT mega328.....	4
1.5	Εγκατάσταση περιβάλλοντος Arduino IDE στον υπολογιστή.....	5
1.6	Ethernet Shield	9
1.7	Πλακέτα Ethernet Shield.....	10
1.8	Εξαρτήματα μικροελεγκτή Arduino.....	11
1.8.1	Αισθητήρες.....	12
1.8.2	LED.....	12
1.8.3	Button	13
1.8.4	Αντιστάσεις	15
1.8.4.1	Πως διαβάζονται οι αντιστάσεις	16
1.9	Breadboard	17
1.10	Ηλεκτρικός διακόπτης	18
1.11	Βελτίωση ασύρματου δικτύου	19
2.	Σύνδεση Arduino με την Ethernet Shield	20
2.1	Εγκατάσταση βιβλιοθηκών για Web Server και δοκιμή λειτουργίας	20
2.2	Τι θα χρειαστούμε για το κύκλωμά μας.....	23
2.3	Ανάπτυξη προγράμματος για το Arduino.....	24
2.4	Ανάπτυξη κώδικα HTML	27
3.	Δημιουργία έξυπνης κλειδαριάς	30
3.1	Τι θα χρειαστούμε για το κύκλωμά μας	30
3.2	Τι είναι το Keypad πληκτρολόγιο	32
3.3	Τι είναι ο σερβοκινητήρας	32
3.4	Σύνδεση Arduino με το Keypad και τον σερβοκινητήρα	33
3.5	Εγκατάσταση βιβλιοθηκών για το Keypad και για το σερβοκινητήρα	34

3.6 Ανάπτυξη προγράμματος έξυπνης κλειδαριάς.....	35
4. Τοποθέτηση αισθητήρα φωτός σε πλακέτα Arduino με σκοπό τον φωτισμό εξωτερικού χώρου μετά την δύση του ηλίου	39
4.1 Τι είναι ο αισθητήρας φωτός	39
4.2 Τι θα χρειαστούμε για την κατασκευή μας και για την συνδεσμολογία του αισθητήρα στη πλακέτα Arduino	40
4.3 Ανάπτυξη προγράμματος για τον αισθητήρα φωτός	43
5. Κατασκευή μονοκατοικίας σε μακέτα και εγκατάσταση όλων των παραπάνω	44
5.1 Υλικά και αξεσουάρ που χρησιμοποιήσαμε για την κατασκευή της μακέτας	44
5.2 Εξοπλισμός για την υλοποίηση της κατασκευής όλων των παραπάνω	44
6. Συμπεράσματα - Προτάσεις για μελλοντική εργασία	47
7. Βιβλιογραφία	48
8. Εικόνες	49

ΚΕΦΑΛΑΙΟ 1. Ο Μικροελεγκτής Arduino



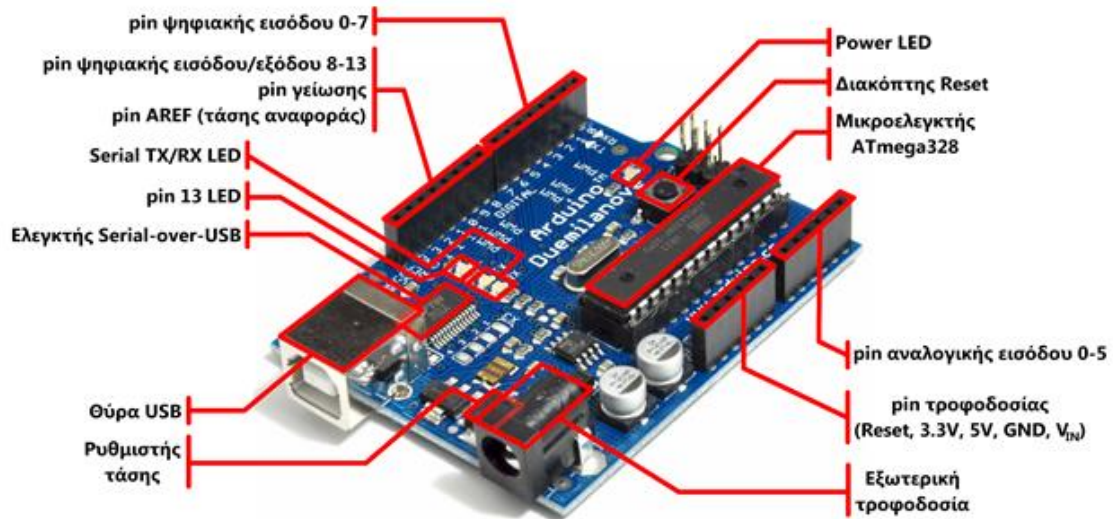
Εικόνα 1: Μικροελεγκτής Arduino UNO [1]

1.1 ΤΙ ΕΙΝΑΙ ΤΟ ARDUINO

Το Arduino είναι ένας μικροελεγκτής ATmega. Με λίγα λόγια διαθέτει εισόδους και εξόδους που αντιδρούν βάση του προγραμματισμού που φορτώνονται στο μικροελεγκτή με τη βοήθεια του υπολογιστή. Η γλώσσα προγραμματισμού που χρησιμοποιεί είναι η Wiring που είναι αντιγραφή της γλώσσας C++ και επομένως είναι αρκετά εύκολη στη σύνταξη. Το πρόγραμμα Arduino τρέχει σε πλατφόρμες λειτουργικών συστημάτων, όπως Linux, MAC και Windows με άδεια χρήσης GPL.

Το Arduino διατίθεται με άδεια χρήσης Creative Commons, πράγμα που σημαίνει πως ο καθένας έχει τη δυνατότητα να κατασκευάσει την δική του πλακέτα όπως αυτός επιθυμεί και σύμφωνα με τις ανάγκες του κάθε χρήστη.

Αν και μικροσκοπικό (7x5 cm) οι δυνατότητες που προσφέρει στον εκάστοτε χρήστη – προγραμματιστή, αρχάριο και μη είναι πάρα πολλές. Μπορούμε να το χρησιμοποιήσουμε σε πολλές εφαρμογές όπως, ρομποτική και γενικότερα σε αυτοματισμούς καταφέροντας έτσι πάρα πολλά όπως: την κίνηση servo, stepper και DC κινητήρων, τη λήψη πληροφοριών από διάφορους αισθητήρες (θερμοκρασίας, υγρασίας, υπέρυθρων κ.α.), την αμφίδρομη σειριακή επικοινωνία μεταξύ Arduino και PC χρησιμοποιώντας γλώσσες προγραμματισμού (όπως Java και python), όπως επίσης την αναπαραγωγή και αντίληψη ήχων.



Εικόνα 2: Arduino UNO R3 [2]

1.2 ΛΟΓΟΙ ΕΠΙΛΟΓΗΣ ARDUINO

Ο κυριότερος λόγος που όλο και περισσότεροι άνθρωποι επιλέγουν να χρησιμοποιήσουν έναν μικροελεγκτή Arduino είναι ότι δίνει την δυνατότητα να εφαρμοστεί σε χιλιάδες διαφορετικά προγράμματα και εφαρμογές.

Το Arduino μπορεί να χρησιμοποιηθεί εύκολα από αρχάριους χρήστες, διότι είναι αρκετά ευέλικτο, για τη δημιουργία μικροεφαρμογών αλλά και από προχωρημένους χρήστες για πιο σύνθετες και επαγγελματικές εφαρμογές.

Κάποιοι από τους λόγους επιλογής της συσκευής είναι:

- **Χαμηλό κόστος**

Η πλακέτα Arduino σχετικά χαμηλού κόστους σε σχέση με άλλες πλακέτες μικροελεγκτών. Υπάρχουν φθηνές πλακέτες Arduino που την συναρμολόγηση την κάνουμε εμείς και υπάρχουν και έτοιμες συναρμολογημένες και το μέσο κόστος μιας τέτοιας πλακέτας κυμαίνεται περίπου στα 25 ευρώ.

- **Απλό, σαφές περιβάλλον προγραμματισμού**

Ο προγραμματισμός του Arduino είναι αρκετά εύκολο για αρχάριους προγραμματιστές, αλλά και αρκετά ευέλικτο για προχωρημένους προγραμματιστές. Το λογισμικό του Arduino IDE είναι ανοιχτού κώδικα και επεκτάσιμο λογισμικό. Ο προγραμματισμός γίνεται εύκολα με την χρήση της γλώσσας προγραμματισμού c++, επίσης μπορεί το λογισμικό να επεκταθεί με βιβλιοθήκες c++.

- **Το Arduino βασίζεται σε ATmega μικροελεγκτές της Atmel**

Τα Sketch (σχεδιασμός - ανάπτυξη προγράμματος) δημοσιεύονται με την άδεια Creative Commons, έμπειροι προγραμματιστές μπορούν να δημιουργήσουν και να τρέξουν το δικό τους Sketch, επεκτείνοντας και βελτιώνοντάς το. Ακόμα, σχετικά άπειροι προγραμματιστές μπορούν να δημιουργήσουν οποιοδήποτε κύκλωμα πάνω

στη breadboard, προκειμένου να κατανοήσουν πώς λειτουργεί και να εξοικονομήσουν χρήματα με διάφορα LED και αισθητήρες.

1.3 ΤΙ ΕΙΝΑΙ ΤΟ ΕΞΥΠΝΟ ΣΠΙΤΙ

Ο όρος “ΕΞΥΠΝΟ ΣΠΙΤΙ” εμπεριέχει το σύνολο των αυτοματισμών, με τους οποίους ομαδοποιούνται, οργανώνονται και αυτοματοποιούνται οι λειτουργίες μιας κατοικίας, ανάλογα με τις ανάγκες που έχει ο κάθε ιδιοκτήτης. Αυτό πρακτικά σημαίνει ότι τα ίδια συστήματα μπορούν να χρησιμοποιηθούν για περισσότερες εφαρμογές (π.χ. ένας διακόπτης μπορεί να ελέγχει το φωτισμό ενός δωματίου ή και περισσότερων, ενώ ίδιος διακόπτης μπορεί να ανοιγοκλείνει και τις πόρτες, η οθόνη της τηλεόρασης μπορεί να δέχεται και την εικόνα της θυροτηλεόρασης, το τηλέφωνο μπορεί να χτυπά και όταν κάποιος μας χτυπάει το κουδούνι της εξώπορτας κ.α.).

Το βασικό πλεονέκτημα της τεχνολογίας «έξυπνο σπίτι» είναι η δυνατότητα παρακολούθησης και διαχείρισης όλων των χώρων ή μεμονωμένων και εγκαταστάσεων μιας κατοικίας με οποιοδήποτε τρόπο επικοινωνίας (μέσω σταθερού τηλεφώνου ή κινητού αλλά και του διαδικτύου). Συγκεκριμένα ορισμένες από τις λειτουργίες ενός σπιτιού που μπορούν να αυτοματοποιηθούν με στόχο να γίνει εύχρηστο, άνετο και γρήγορο είναι:

- Κεντρικός ελεγκτής αυτοματισμού
- Έλεγχος φωτισμού
- Κεντρικό σύστημα συναγερμού
- Κεντρικό σύστημα θέρμανσης
- Κεντρικό σύστημα διανομής εικόνας και ήχου
- Σύστημα ποτίσματος
- Σύστημα παρακολούθησης από κάμερες
- Έλεγχος ζεστού νερού
- Έλεγχος ηλεκτρικών συσκευών
- Έλεγχος καιρικών συνθηκών
- Εφαρμογές προγραμματισμού
- Υπολογιστικά προγράμματα
- Έλεγχος μέσω κινητού
- Έλεγχος μέσου ασύρματου δικτύου

Παράλληλα, με την δημιουργία σεναρίων, επιτυγχάνεται ο πλήρης έλεγχος όλων των εγκαταστάσεων. Ενδεικτικά κάποια από τα σενάρια που μπορούν να τεθούν σε εφαρμογή είναι:

- Όταν αποχωρώ από το σπίτι το σύστημα κλείνει όλες τις ηλεκτρικές καταναλώσεις, θέρμανση, ύδρευση, ρολά, τέντες, συναγερμός, φυσικό αέριο κ.α.).
- Σενάρια φωτισμού κατοικίας (party mode, home cinema , κτλ).
- Κλείσιμο, άνοιγμα όλων των ρολών ταυτόχρονα / ασφάλιση της κατοικίας (π.χ. το βράδυ).
- Κατά την επιστροφή προς το σπίτι να τίθεται σε εφαρμογή η θέρμανση ή να ανάβει ο φωτισμός σε συγκεκριμένους χώρους του σπιτιού.

1.4 ΤΙ ΕΙΝΑΙ Ο ΜΙΚΡΟΕΛΕΓΚΤΗΣ AT MEGA328



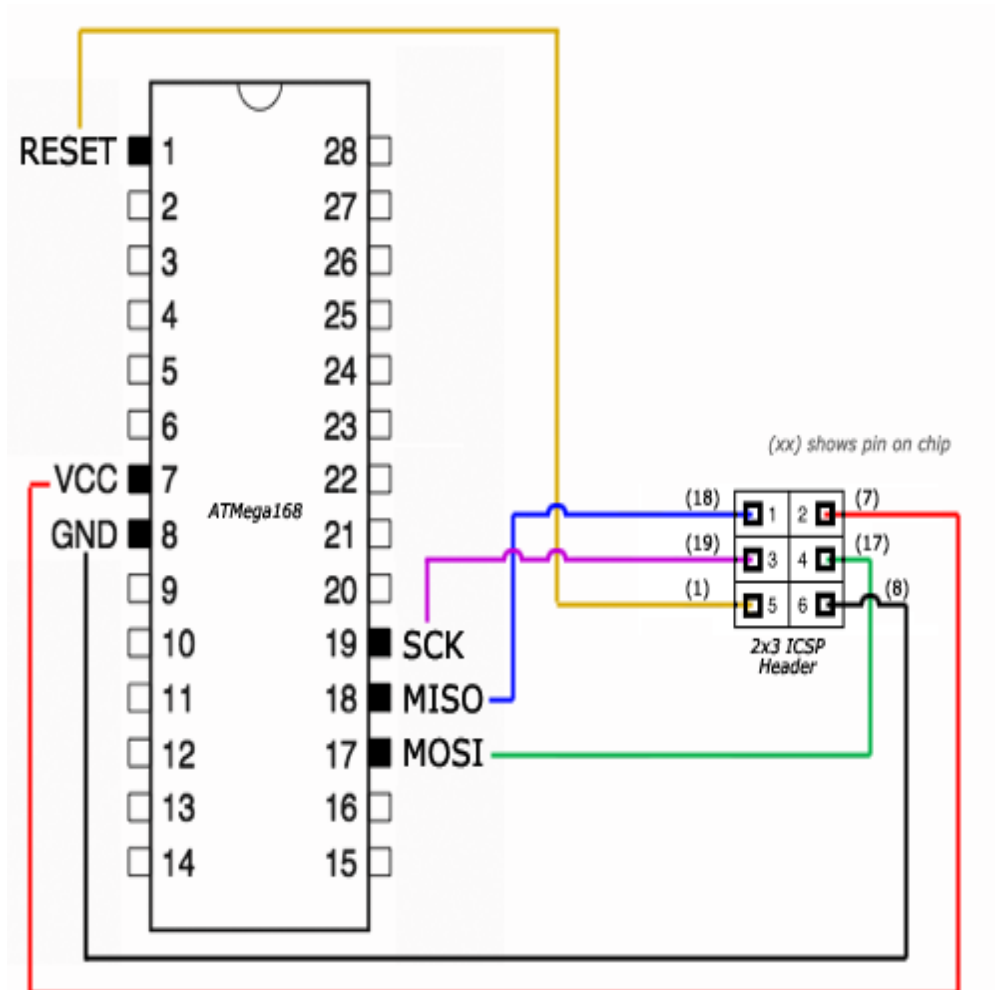
Εικόνα 3: Μικροελεγκτής AT mega 328 [3]

Το ARDUINO διαθέτει έναν μικροελεγκτή ATmega328 είναι ένας 8-bit μικροελεγκτής, ο οποίος λειτουργεί στα 16MHz. Ο ATmega328 έχει δικιά του μνήμη τριών διαφορετικών τύπων:

-Η πρώτη έχει 2Kb μνήμης SRAM, είναι η μνήμη στην οποία αποθηκεύουμε τα Sketch μας. Η μνήμη αυτή χάνει τα Sketch μας όταν γίνει reset ή πάψει να τροφοδοτείται πια.

-Η δεύτερη έχει 1Kb μνήμης EEPROM, είναι η μνήμη που χρησιμοποιείται για ανάγνωση/εγγραφή δεδομένων, σε σχέση με την μνήμη SPAM δεν χάνει τα δεδομένα της.

-Η Τρίτη έχει 32Kb μνήμης Flash, από τα οποία τα 2Kb χρησιμοποιούνται από το Firmware (το firmware είναι ένα είδος λογισμικού το οποίο είναι γραμμένο σε γλώσσα μηχανής και είναι φτιαγμένο αποκλειστικά και μόνο για ένα μοντέλο συσκευής) του Arduino που έχει εγκαταστήσει ο κατασκευαστής του. Το Firmware, αυτό που στην ορολογία του Arduino ονομάζεται Bootloader είναι αναγκαίο για την εγκατάσταση των δικών μας Sketch στον μικροελεγκτή μέσω της θύρας USB, χωρίς δηλαδή να χρειάζεται εξωτερικός Hardware Programmer. Τα υπόλοιπα 30Kb της μνήμης Flash χρησιμοποιούνται για την αποθήκευση αυτών των προγραμμάτων. Η μνήμη Flash, όπως και η EEPROM δε χάνει τα δεδομένα της με απώλεια τροφοδοσίας ή Reset.



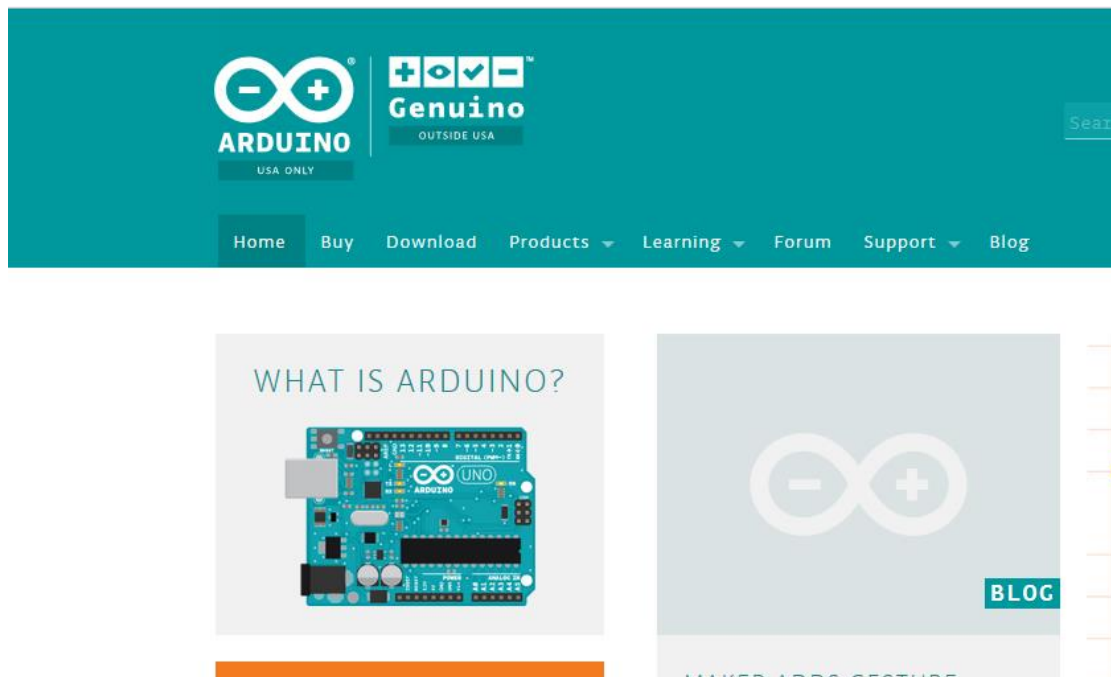
Εικόνα 4: Διάγραμμα μικροελεγκτή AT mega328 [4]

1.5 ΕΓΚΑΤΑΣΤΑΣΗ ΠΕΡΙΒΑΛΛΟΝΤΟΣ ARDUINO IDE ΣΤΟΝ ΥΠΟΛΟΓΙΣΤΗ

Για να προγραμματιστεί η μονάδα θα χρειαστείτε το περιβάλλον προγραμματισμού Arduino. Στο περιβάλλον αυτό ο χρήστης γράφει κώδικα που βασίζεται στη γλώσσα προγραμματισμού C/C++ τον οποίο μετά μεταγλωττίζει και μεταφορτώνει στη μονάδα που θέλει να την εφαρμόσει. Το Arduino IDE υπάρχει σε εκδόσεις για Windows, Mac και Linux και μπορείτε να το κατεβαστεί εντελώς δωρεάν από την επίσημη ιστοσελίδα «www.arduino.cc».

Παρακάτω ακολουθούν τα βήματα που πρέπει να ακολουθήσει ο χρήστης προκειμένου να εγκαταστήσει το περιβάλλον του Arduino στον υπολογιστή του.

Βήμα 1^ο Επισκεπτόμαστε το URL www.arduino.cc



Εικόνα 5: Κεντρική σελίδα επίσημης σελίδας Arduino [5]

Βήμα 2^ο Από το κεντρικό μενού επιλέγουμε την υπερσύνδεση “Download”



Εικόνα 6: Download 1.6.9 [6]

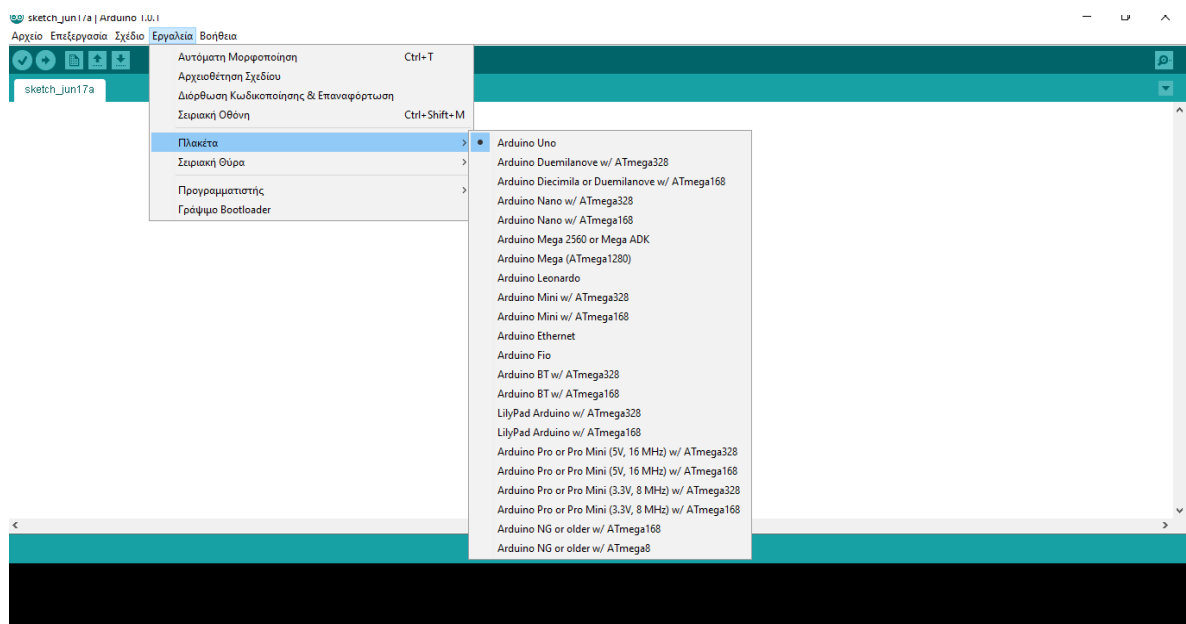
Βήμα 3^ο Στο δεξιό μενού επιλέγουμε “Windows installer” και κατεβάζουμε το λογισμικό μας στον τοπικό δίσκο του υπολογιστή μας.

Το περιβάλλον αυτό έχει Ελληνικό μενού, καθώς και αρκετά έτοιμα παραδείγματα χρήσης.



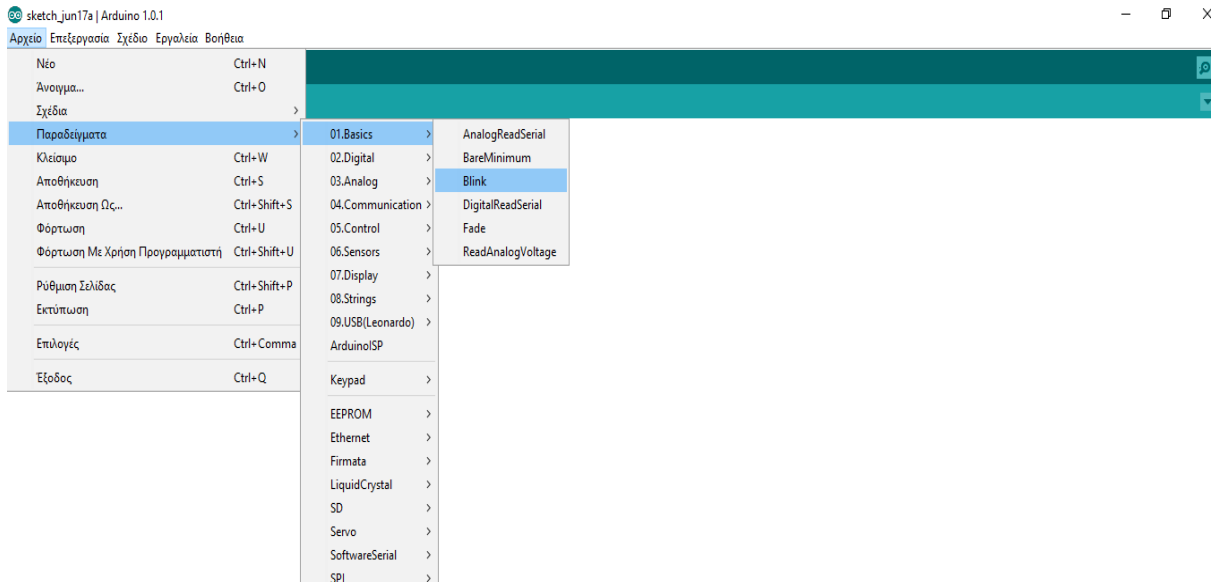
Εικόνα 7: Περιβάλλον προγραμματισμού Arduino IDE

Βήμα 4^ο Το επόμενο βήμα είναι να επιλέξουμε πλακέτα, στην περίπτωση μας επιλέγουμε την πλακέτα Arduino UNO.

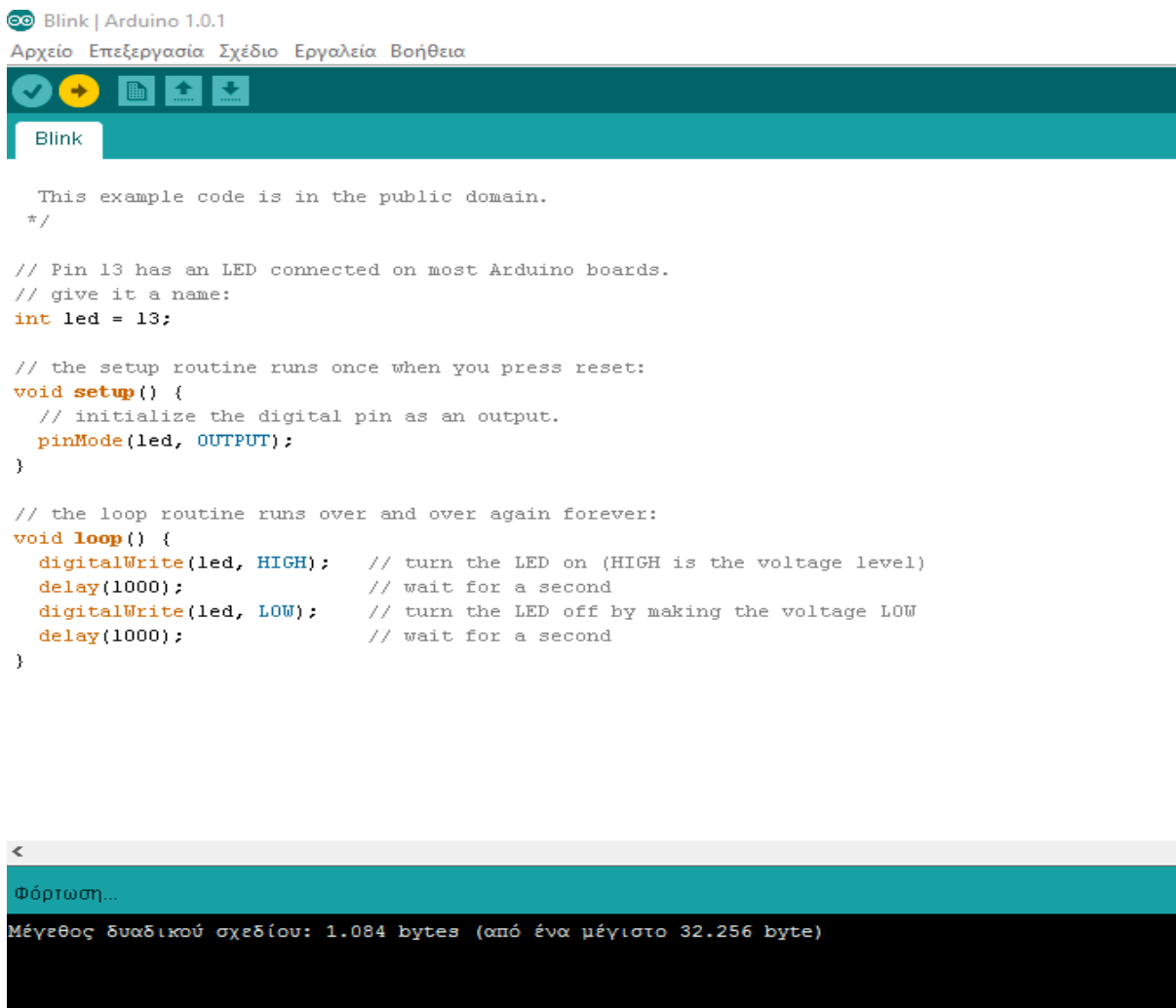


Εικόνα 8: Επιλογή πλακέτας

Βήμα 5^ο Τέλος ελέγχουμε αν έχει γίνει σωστά η εγκατάσταση της πλακέτας, με χρήση κάποιου από τα παραδείγματα που βρίσκονται έτοιμα στο λογισμικό μας.



Εικόνα 9: Έλεγχος λειτουργίας



Εικόνα 10: Φόρτωση προγράμματος στο Arduino

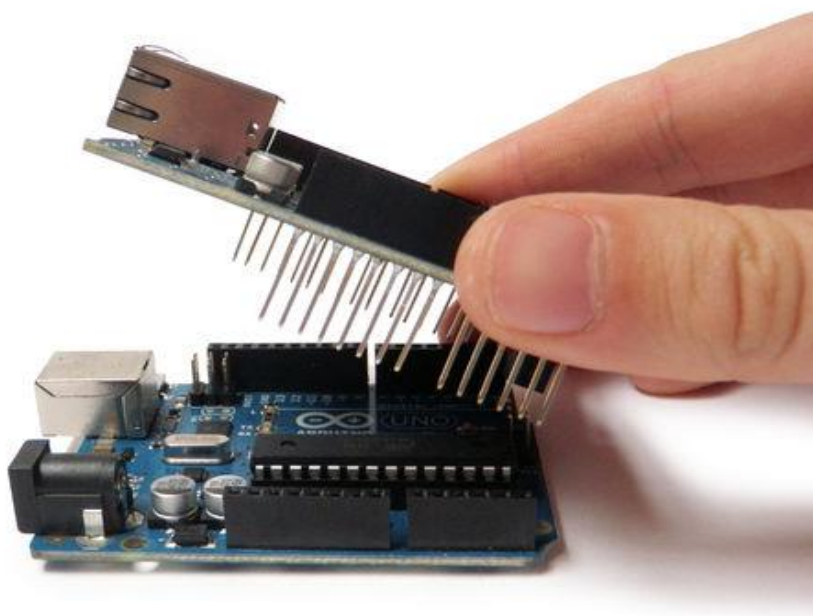
1.6 ETHERNET SHIELD

Η Ethernet Shield είναι μια πλακέτα συνδέσεις του Arduino με το Router μας και ταυτόχρονα και με το διαδίκτυο σε ελάχιστα λεπτά. Η Ethernet Shield τοποθετείται πάνω στη πλακέτα Arduino. Στη συνέχεια συνδέουμε ένα καλώδιο RJ45 στο Router μας και είναι η Ethernet Shield είναι έτοιμη για χρήση. Υπάρχουν επίσης έτοιμα προγράμματα που μας δείχνουν ακριβώς πως μπορούμε να δημιουργήσουμε ένα Web Server και να τα αναπτύξουμε για το σχεδιασμό ενός δικού μας κυκλώματος.

Για να τεθεί σε λειτουργία :

- Απαιτείται μια πλακέτα Arduino
- 5V Τάση λειτουργίας
- Ταχύτητα σύνδεσης: 10 / 100Mb
- Σύνδεση με Arduino στη θύρα SPI

Η Wiznet W5100 (η Wiznet W5100 παρέχει μια στοίβα δικτύου ικανή να ανταποκριθεί τόσο σε TCP όσο και σε UDP, τα πρωτόκολλα TCP και UDP είναι τα δύο βασικά πρωτόκολλα του διαδικτύου και χρησιμοποιούνται για την ανταλλαγή δεδομένων μεταξύ δύο υπολογιστών) παρέχει ένα δίκτυο (IP). Υποστηρίζει έως τέσσερις συνδέσεις μαζί. Χρησιμοποιούμε τη βιβλιοθήκη Ethernet για να συνδεθούμε στο Internet χρησιμοποιώντας την πλακέτα Ethernet. Η Ethernet Shield συνδέεται με την πλακέτα Arduino εφαρμόζοντας όλα της τα Pin στα Pin του Arduino(Εικόνα 11). Αυτό κρατά τη διάταξη τους άθικτη και επιτρέπει η μια στην άλλη άμεση επικοινωνία. Η Ethernet Shield έχει ένα πρότυπο RJ-45 σύνδεσης, με ενσωματωμένο μετασχηματιστή γραμμής η RJ-45 είναι η μια κλασική θύρα όπως έχουμε στον υπολογιστή μας για σύνδεση με το διαδίκτυο και η σύνδεση της Ethernet με το Router επιτυγχάνεται με καλώδιο τύπου RJ-45.



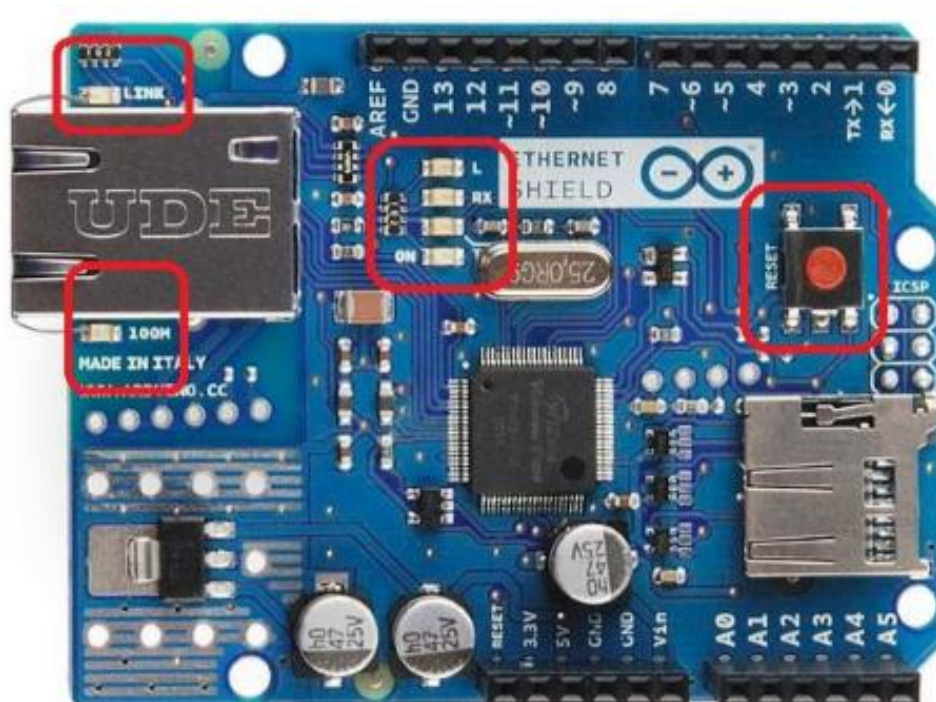
Εικόνα 11: Τοποθέτηση Ethernet shield στη πλακέτα Arduino [7]

1.7 ΠΛΑΚΕΤΑ ETHERNET SHIELD

Το Arduino επικοινωνεί με την W5100 και την κάρτα μνήμης SD (συσκευή αποθήκευσης ψηφιακών δεδομένων), χρησιμοποιώντας το SPI (είναι ένα σύγχρονο σειριακό πρωτόκολλο δεδομένων που χρησιμοποιούνται από μικροελεγκτές για την επικοινωνία με μία ή περισσότερες περιφερειακές συσκευές γρήγορα σε μικρές αποστάσεις). Αυτό είναι για τα ψηφιακά Pin 10, 11, 12, και 13 για το UNO που εμείς χρησιμοποιούμε στη κατασκευή μας. Επίσης στο Arduino το Pin 10 χρησιμοποιείται για την επιλογή του W5100 και το Pin 4 για την κάρτα SD. Αυτά τα Pin δεν μπορούν να χρησιμοποιηθούν για οποιαδήποτε ανάγκη μας. Η Ethernet Shield μας παρέχει ένα πρότυπο, το RJ45 Ethernet Jack.

Η Ethernet Shield περιλαμβάνει:

- Το κουμπί Reset πάνω στην Ethernet Shield επαναφέρει τον W5100 αλλά μας επαναφέρει μαζί και την πλακέτα Arduino.
- PWR: υποδεικνύει ότι η πλακέτα τροφοδοτείτε με ρεύμα.
- LINK: δείχνει την παρουσία ενός δικτύου και αναβοσβήνει όταν η ασπίδα μεταδίδει ή λαμβάνει δεδομένα.
- FULLD: υποδεικνύει ότι η σύνδεση δικτύου είναι Full Duplex.
- 100M: δείχνει την ταχύτητα σύνδεσης με το δίκτυο αν είναι 100 Mb / s .
- RX: αναβοσβήνει όταν η πλακέτα λαμβάνει δεδομένα που εμείς δίνουμε.
- TX: αναβοσβήνει όταν η πλακέτα στέλνει δεδομένα.
- COLL: αναβοσβήνει όταν υπάρχουν δυσλειτουργίες στο δίκτυο μας.



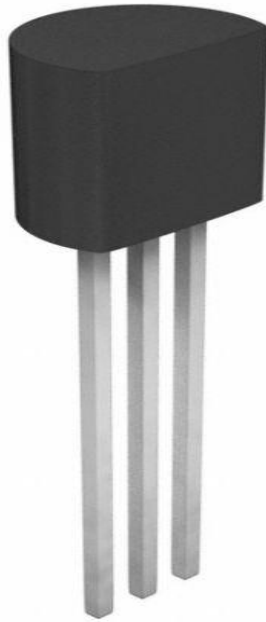
Εικόνα 12: Περιγραφή πλακέτας Ethernet Shield

1.8 ΕΞΑΡΤΗΜΑΤΑ ΜΙΚΡΟΕΛΕΓΚΤΗ ARDUINO

Ένας μικροελεγκτής Arduino δέχεται μια μεγάλη γκάμα από αισθητήρες και διαφορά εξαρτήματα, με αποτέλεσμα να έχουμε δυνατότητα να επιλέξουμε από έναν απλό αισθητήρας θερμοκρασίας μέχρι έναν κινητήρα συνεχούς ρεύματος ανάλογα με το κύκλωμά που θέλουμε να επιτύχουμε.

1.8.1 Αισθητήρες

Αισθητήρας είναι ένας μετατροπέας που το είδος του εξαρτάται από τον τρόπο δημιουργίας και τελικής χρήσης του. Ο αισθητήρας ανιχνεύει γεγονότα ή αλλαγές στο περιβάλλον και δίνει μια αντίστοιχη έξοδο, γενικά ως ένα ηλεκτρικό ή οπτικό σήμα ανάλογα τον αισθητήρα. Για παράδειγμα, ένας αισθητήρας θερμοκρασίας μετατρέπει την θερμοκρασία σε μια τάση που λαμβάνουμε στα άκρα του αισθητήρα. Ο τρόπος σχεδιασμού των αισθητήρων θα πρέπει να στοχεύει στη μικρή ευαισθησία της μέτρησης η οποία πρέπει φυσικά να είναι αξιόπιστη και να λαμβάνεται με μικρή απόκριση.



Εικόνα 13 : Αισθητήρας θερμοκρασίας

1.8.2 Led

Ένα Led είναι μια δίοδος εκπομπής (LED, Light Emitting Diode), αποκαλείται ένας ημιαγωγός ο οποίος εκπέμπει φωτεινή ακτινοβολία, το LED αποτελείται από μόλυβδο. Όταν μια κατάλληλη τάση εφαρμόζεται στα άκρα του, αυτό έχει σαν αποτέλεσμα τα ηλεκτρόνια να απελευθερώνουν ενέργεια με τη μορφή φωτονίων. Αυτό το φαινόμενο ονομάζεται ηλεκτροφωταύγεια και το χρώμα του φωτός καθορίζεται από το ενεργειακό χάσμα ζώνης του ημιαγωγού, στο εμπόριο υπάρχει μια ευρεία γκάμα χρωμάτων και μεγεθών με χαμηλό κόστος. Τα Led έχουν πολλά πλεονεκτήματα σε σχέση με το φως πυρακτώσεως, που στις μέρες όλο και λιγότεροι χρησιμοποιούν, ένα από αυτά είναι η χαμηλή κατανάλωση ενέργειας, η διάρκεια ζωής και το μικρό τους μέγεθος σε σχέση με τις κλασικές λάμπες πυρακτώσεως.

Ο τρόπος σύνδεσης ενός Led είναι συγκεκριμένος υπάρχουν 2 ακροδέκτες στα άκρα του LED ακροδέκτης που είναι και ο μακρύτερος τοποθετούμε τροφοδοσία (+) και στον άλλον που είναι μικρότερος τοποθετούμε την γείωση (-).



Εικόνα 14: LED σε διάφορα χρώματα [8]

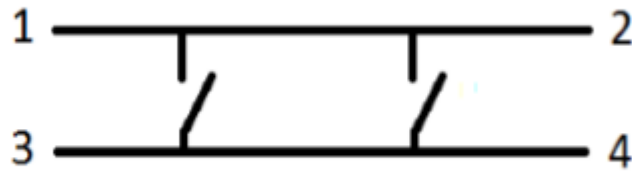
1.8.3 Button

Το κουμπί(Εικόνα 15) με τους 4 ακροδέκτες αποτελείται από δύο διακόπτες που κλείνουν με το πάτημα τους. Για να το χρησιμοποιήσουμε θα πρέπει να γνωρίζουμε το διάγραμμά του. Το Button το χρησιμοποιούμε όταν θέλουμε να επέμβουμε στο κύκλωμα μας, είτε όταν θέλουμε να ενεργοποιήσουμε κάτι ή όταν θέλουμε να διακόψουμε το κύκλωμα μας.



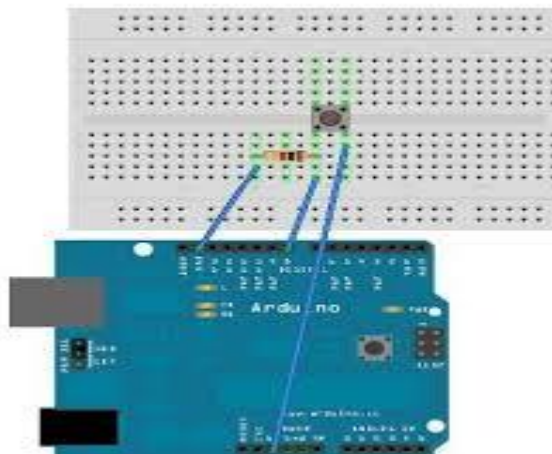
Εικόνα 15: Button Arduino [9]

Αν κοιτάξουμε το μπουτόν θα παρατηρήσουμε ότι έχει δύο ζεύγη ακροδεκτών σε αντίθετες πλευρές. Το ζεύγος των ακροδεκτών το οποίο είναι σε κάθε πλευρά, είναι συνδεδεμένο με ένα διακόπτη. Σε αυτήν την περίπτωση διαφορετικό διακόπτη έχουν οι ακροδέκτες 2 - 4 και άλλο οι 1 - 3. Το μπουτόν κλείνει και τους δύο διακόπτες ταυτόχρονα όταν πιέζεται στη κεφαλή του, ενώ όταν το αφήσουμε επανέρχεται και μένει ανοιχτός.



Εικόνα 16: Διάγραμμα Button

Από το διάγραμμα μπορούμε να συμπεράνουμε ότι, οι ακροδέκτες 1 – 2 είναι βραχυκυκλωμένοι το ίδιο και οι 3 – 4. Οπότε θα πρέπει να είμαστε αρκετά προσεκτικοί για τη συνδεσμολογία και πλέον όταν θα έχουμε ολοκληρώσει το κύκλωμά μας, να είμαστε σίγουροι πριν το τροφοδοτήσουμε. Σε περίπτωση λάθους συνδεσμολογίας του μπουτόν, υπάρχει μεγάλος κίνδυνος να καταστρέψουμε κάποια από τα εξαρτήματά μας, εφόσον πλέον θα υπάρχει βραχυκύκλωμα.



Εικόνα 17: Παράδειγμα συνδεσμολογίας Arduino με Button [10]

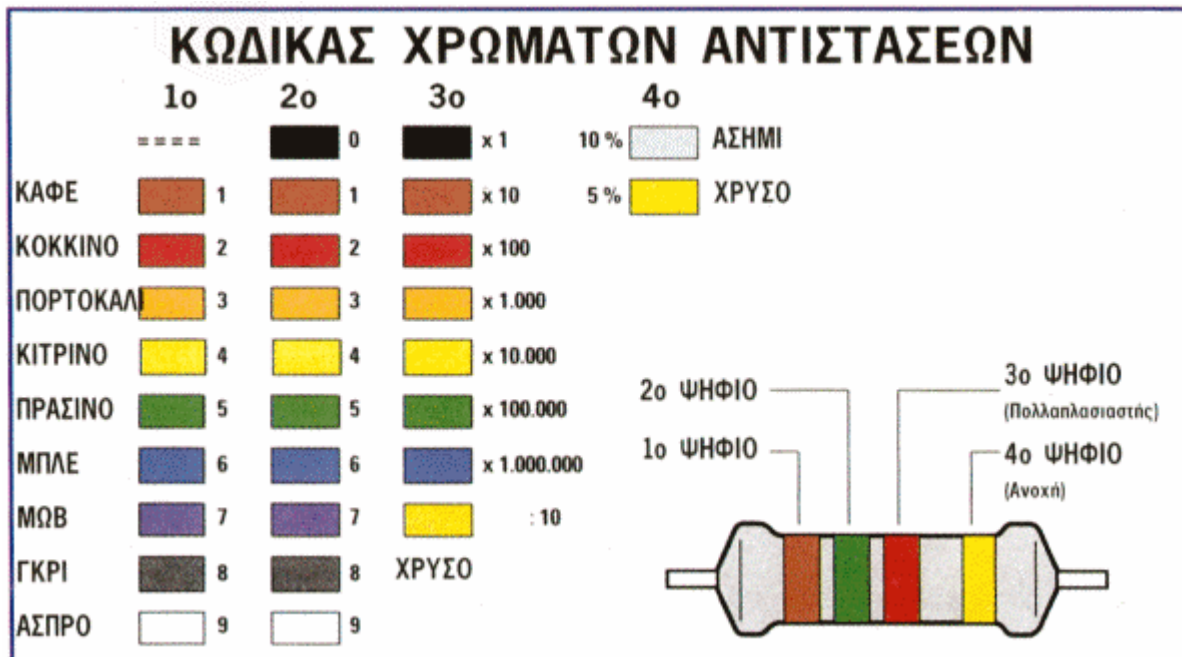
1.8.4 Αντιστάσεις



Εικόνα 18: Αντιστάσεις [11]

Η αντίσταση είναι ένα εξάρτημα - αγωγός, που ο ρόλος της σε ένα κύκλωμα είναι να μειώνει τη ροή της ηλεκτρικής ενέργειας ώστε να προστατέψουμε διάφορα εξαρτήματα από την καταστροφή τους. Οι αντιστάσεις διακρίνονται από το υλικό το οποίο είναι παρασκευασμένες, μετατρέποντας ένα μέρος της ηλεκτρικής ενέργειας σε θερμότητα και άλλα. Για παράδειγμα αν βάλουμε μια αντίσταση σε σειρά με ένα εξάρτημα, όπως ένα Led, η αντίσταση αυτή θα δυσχεραίνει τη διέλευση της ηλεκτρικής ενέργειας η οποία θα περνούσε και ως αποτέλεσμα είναι το Led να δεχθεί λιγότερη ενέργεια. Με αυτό τον τρόπο πετυχαίνουμε να αυξήσουμε τον χρόνο ζωής του LED, μιας και αν δεν είχαμε συνδέσει αντίσταση στο κύκλωμά μας, το LED θα ήταν πιο φωτεινό και γρήγορα θα καιγόταν. Η ικανότητα αυτή της αντίστασης, μας δίνει τη δυνατότητα να προμηθεύσουμε τα εξαρτήματά μας με το ποσοστό ενέργειας που χρειάζονται, κάθε εξάρτημα δέχεται και την σωστή αντίσταση. Στο εμπόριο συναντάμε πολλών ειδών αντιστάσεις από διαφορετικά υλικά μεγέθη και διαφορετικών τιμών σύμφωνα πάντα με τις ανάγκες και προδιαγραφές τις εκάστοτε εφαρμογής.

1.8.4.1 Πως διαβάζονται οι αντιστάσεις



Εικόνα 19: Πίνακας χρωμάτων αντιστάσεων [12]

Κάθε αντίσταση αποτελείται από 4 ψηφία, τα οποία αντιστοιχούν και σε κάποιο χρώμα. Η τιμή της αντίστασης σημειώνεται πάνω της χρησιμοποιώντας χρώματα. Όπως μπορούμε να δούμε παρακάτω είναι:

ΜΑΥΡΟ: 0

ΜΠΛΕ: 6

ΚΑΦΕ: 1

ΜΩΒ: 7

ΚΟΚΚΙΝΟ: 2

ΓΚΡΙ: 8

ΠΟΡΤΟΚΑΛΙ: 3

ΛΕΥΚΟ: 9

ΚΙΤΡΙΝΟ: 4

ΠΡΑΣΙΝΟ: 5

Από δεξιά προς τα αριστερά, το δεύτερο χρώμα της αντίστασης είναι ο πολλαπλασιαστής. Ο αριθμός από τον πρώτο χρωματισμό που βρίσκεται πάνω στην αντίσταση πρέπει να πολλαπλασιάζεται με τον αριθμό αυτού του χρώματος:

ΜΑΥΡΟ: 1

ΠΡΑΣΙΝΟ: 10000

ΚΑΦΕ: 10

ΜΠΛΕ: 100000

ΚΟΚΚΙΝΟ: 100

ΧΡΥΣΟ: 0,1

ΠΟΡΤΟΚΑΛΙ: 1000

ΑΣΗΜΙ: 0,01

ΚΙΤΡΙΝΟ: 10000

Το επόμενο χρώμα είναι η ανοχή. Η ανοχή είναι η ακρίβεια του αντιστάτη και δίνεται ως ποσοστό. Για παράδειγμα, μια αντίσταση 400 Ω. με ανοχή $\pm 10\%$ θα έχει μια τιμή, μεταξύ $400 - 40 = 360$ και $400 + 40 = 440$ (40 είναι το 10% των 400 Ω).

ΚΑΦΕ: 1%

ΚΟΚΚΙΝΟ: 2%

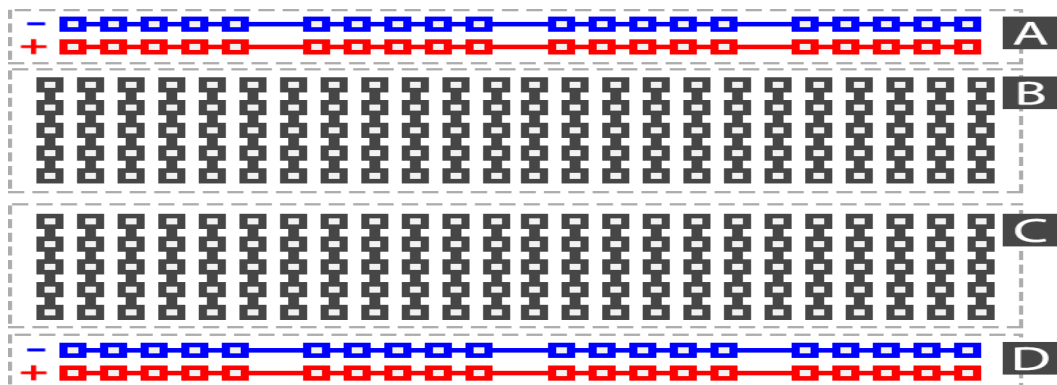
ΧΡΥΣΟ: 5%

ΑΣΗΜΙ: 10%

ΚΕΝΟ: 20%

1.9 Breadboard

Η Breadboard είναι μια πλακέτα που μας επιτρέπει να συνδέσουμε διάφορα εξαρτήματα και να κάνουμε διάφορες συνδέσεις, χωρίς να κάνουμε συγκόλληση αυτών απλά με την βοήθεια καλωδίων (τύπου καρφιά). Ανάλογα τον τύπο Breadboard που χρησιμοποιούμε έχουμε και την κατάλληλη διάταξη στα pins.



Εικόνα 20: Παράδειγμα διάταξης Breadboard [13]

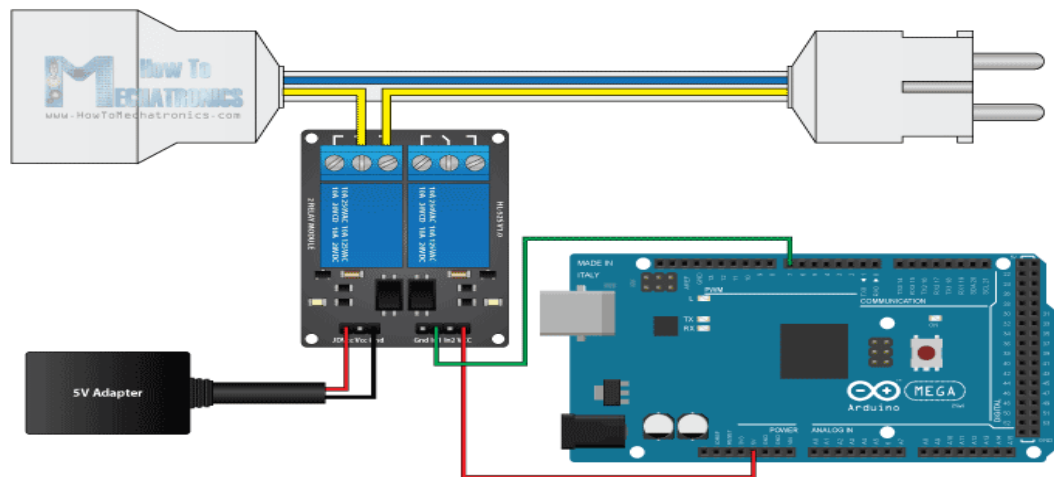
Στην παραπάνω εικόνα, στο πλαίσιο «A ,D», όλα τα Pin δεξιά και αριστερά είναι βραχυκυκλωμένα όπως είναι η κόκκινη και μπλε σημείωση. Ενώ, στο πλαίσιο «B,C», όλα τα Pin στη μέση είναι βραχυκυκλωμένα με διαφορετικό τρόπο και τον τρόπο αυτόν μπορούμε να τον διακρίνουμε επίσης από την μαύρη σημείωση που βρίσκεται εντός πλαισίου «B,C».

1.10 ΗΛΕΚΤΡΙΚΟΣ ΔΙΑΚΟΠΤΗΣ (ΡΕΛΕΣ)



Εικόνα 21: Ηλεκτρικός διακόπτης [14]

Ο ηλεκτρονόμος ή ρελές είναι ένας ηλεκτρικός διακόπτης που ανοίγει και κλείνει ένα ηλεκτρικό κύκλωμα κάτω από τον έλεγχο ενός άλλου ηλεκτρικού κυκλώματος. Δίνοντας ρεύμα στον ηλεκτρικό διακόπτη ενεργοποιούμε ένα κύκλωμα. Ο ρελές χρησιμοποιείται όπου είναι απαραίτητο για τον έλεγχο κυκλώματος από ένα σήμα χαμηλής ισχύος. Τα πρώτα ρελέ χρησιμοποιήθηκαν σε κυκλώματα τηλεγραφημάτων ως ενισχυτές: επανέλαβαν το σήμα που προέρχεται από ένα κύκλωμα και εκ νέου μεταδοθεί σε άλλο κύκλωμα. Ρελέ χρησιμοποιήθηκαν εκτενώς και σε τηλεφωνικά κέντρα και στους πρώτους υπολογιστές ώστε να εκτελούν λογικές πράξεις. Στο μικροελεγκτή Arduino χρησιμοποιούμε ηλεκτρικούς διακόπτες που ως σκοπό έχουν την αποστολή σήματος ώστε να ανοίγει ηλεκτρικές συσκευές, φώτα καθώς και πολλά αλλά ηλεκτρικά κυκλώματά.



Εικόνα 22: Παράδειγμα σύνδεσης Arduino με ηλεκτρικό διακόπτη [15]

1.11 ΒΕΛΤΙΩΣΗ ΑΣΥΡΜΑΤΟΥ ΔΙΚΤΥΟΥ

Όπως αναφέραμε η σύνδεση του Arduino με το διαδίκτυο γίνεται μέσω της Ethernet Shield. Επειδή επιθυμούμε να κάνουμε απομακρυσμένο έλεγχο στο έξυπνο σπίτι θα πρέπει να βελτιώσουμε το ασύρματο δίκτυό μας. Μερικοί τρόποι βελτίωσης ασύρματου δικτύου είναι να τοποθετήσουμε μια εξωτερική κεραία Wi-Fi συνδεδεμένη με το router μας, η κεραία είναι στα 2,4GHz εξωτερική και αδιάβροχη, στα 58dBi και ταχύτητες δεδομένων 150 Mbps και η ελάχιστη απόσταση κεραίας και router είναι στα 5 μέτρα η σύνδεση της κεραίας και του router γίνεται με καλώδιο. Τοποθετούμε την κεραία Wi-Fi , σε εξωτερικά σημεία του σπιτιού μας. Έτσι έχουμε μια εκπομπή ασύρματου σήματος πάνω από 200μέτρα απόσταση.



Εικόνα 23: Εξωτερική κεραία ασύρματου δικτύου [16]



Εικόνα 24: Router [17]

ΚΕΦΑΛΑΙΟ 2. Σύνδεση Arduino με την Ethernet Shield

Αφού συνδέσουμε την Ethernet Shield με το Arduino και με το δίκτυό μας τότε για την Ethernet Shield θα πρέπει να καθοριστεί μέσα στο Sketch για την κάθε μία ξεχωριστά, μια διεύθυνση Mac και μια σταθερή διεύθυνση IP η οποία είναι η εσωτερική διεύθυνση του Router χρησιμοποιώντας την εντολή `Ethernet.begin()`. Η διεύθυνση Mac (Μία διεύθυνση Media Access Control - ελέγχου προσπέλασης στο μέσο, που καλείται επίσης και φυσική διεύθυνση, είναι μία μοναδική ταυτότητα που αποδίδεται στις διασυνδέσεις δικτύου για την επικοινωνία στο φυσικό τμήμα του δικτύου) είναι ένα μοναδικό αναγνωριστικό για μια συγκεκριμένη συσκευή, ωστόσο για τις πιο παλιές Ethernet Shield που δεν γνωρίζουμε την ειδική διεύθυνση Mac, θα πρέπει να υπάρξουν κάποιες δοκιμές για να την εφεύρουμε. Έπειτα όσον αφορά την διεύθυνση IP είμαστε υποχρεωμένοι να την γνωρίζουμε και να είναι έγκυρη, αυτό εξαρτάται από τη διαμόρφωση του δικτύου μας.

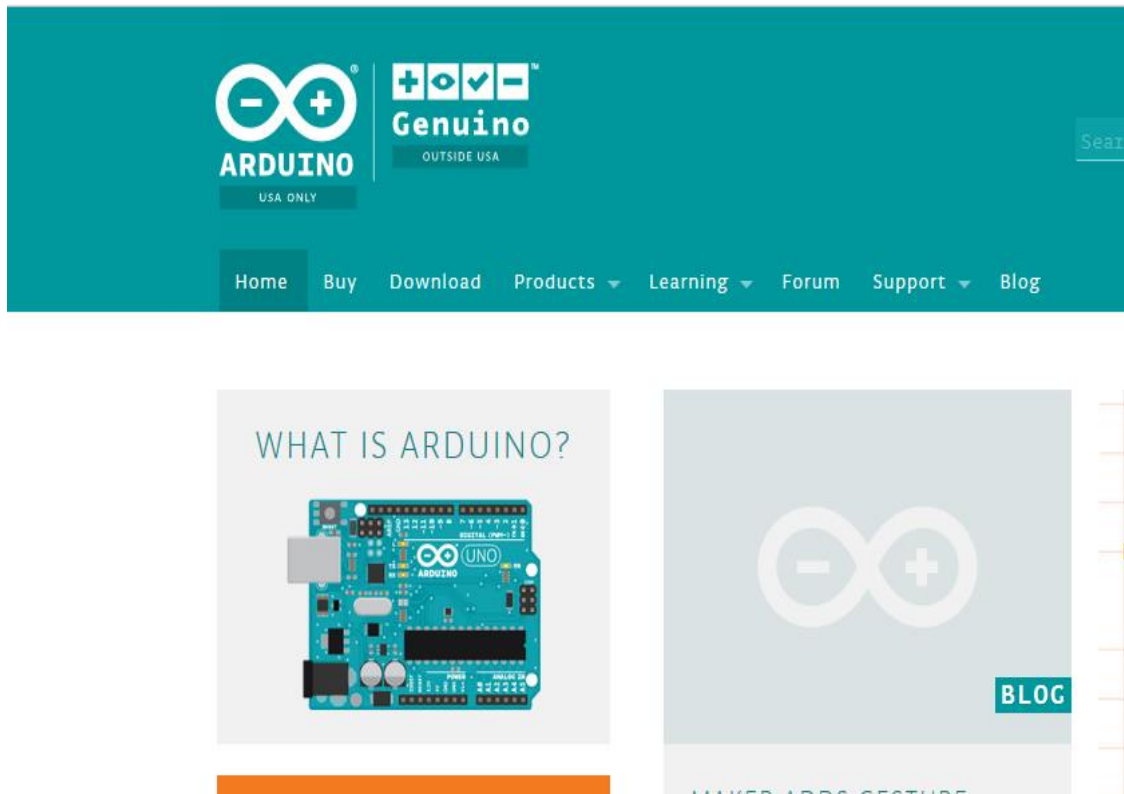
Είναι δυνατόν να χρησιμοποιήσουμε και το πρωτόκολλο DHCP (το DHCP πρωτόκολλο είναι ουσιαστικά ένα λογισμικό που τρέχει σε έναν router και σε υπολογιστή και διευθετεί όλα τα θέματα επικοινωνίας με αυτόν τον υπολογιστή και άλλους που χρησιμοποιούν αυτό το πρωτόκολλο ως γλώσσα), για να βάλουμε δυναμική IP. Μία ακόμα σημαντική πληροφορία για την Ethernet Shield είναι ότι μπορούμε να προσδιορίσουμε μονό μία IP . Για παράδειγμα αν επιθυμούμε να δημιουργήσουμε ένα πρόγραμμα το οποίο στον ιστότοπο θα περιέχει πολλαπλά interfaces και ρυθμίσεις, τότε θα αναγκαστούμε να δουλέψουμε σε μία IP και πάνω σε αυτήν να ξεχωρίσουμε σελίδες (όπως `192.168.178.1/paradeigma`) αλλιώς θα μας εμφανίζει το μήνυμα “EPIC FAIL”.

2.1 ΕΓΚΑΤΑΣΤΑΣΗ ΒΙΒΛΙΟΘΗΚΩΝ ΓΙΑ ΤΟΝ WEB SERVER ΚΑΙ ΔΟΚΙΜΗ ΛΕΙΤΟΥΡΓΙΑΣ

Στην πληροφορική ονομάζουμε βιβλιοθήκη (library) μια συλλογή από έτοιμα υποπρογράμματα που χρησιμοποιείται για τη δημιουργία λογισμικού. Οι βιβλιοθήκες περιέχουν υποβοηθητικό κώδικα καθώς επίσης και δεδομένα, παρέχοντας, με αυτόν τον τρόπο, διάφορες υπηρεσίες- λειτουργίες σε προγράμματα. Αυτό μας δίνει την δυνατότητα να διαμοιράζουμε τόσο τον κώδικα όσο και τα δεδομένα με αρθρωτό τρόπο. Η έννοια της βιβλιοθήκης είναι απαραίτητη έννοια στον δομημένο προγραμματισμό αφού αναπτύχθηκε παράλληλα με αυτόν.

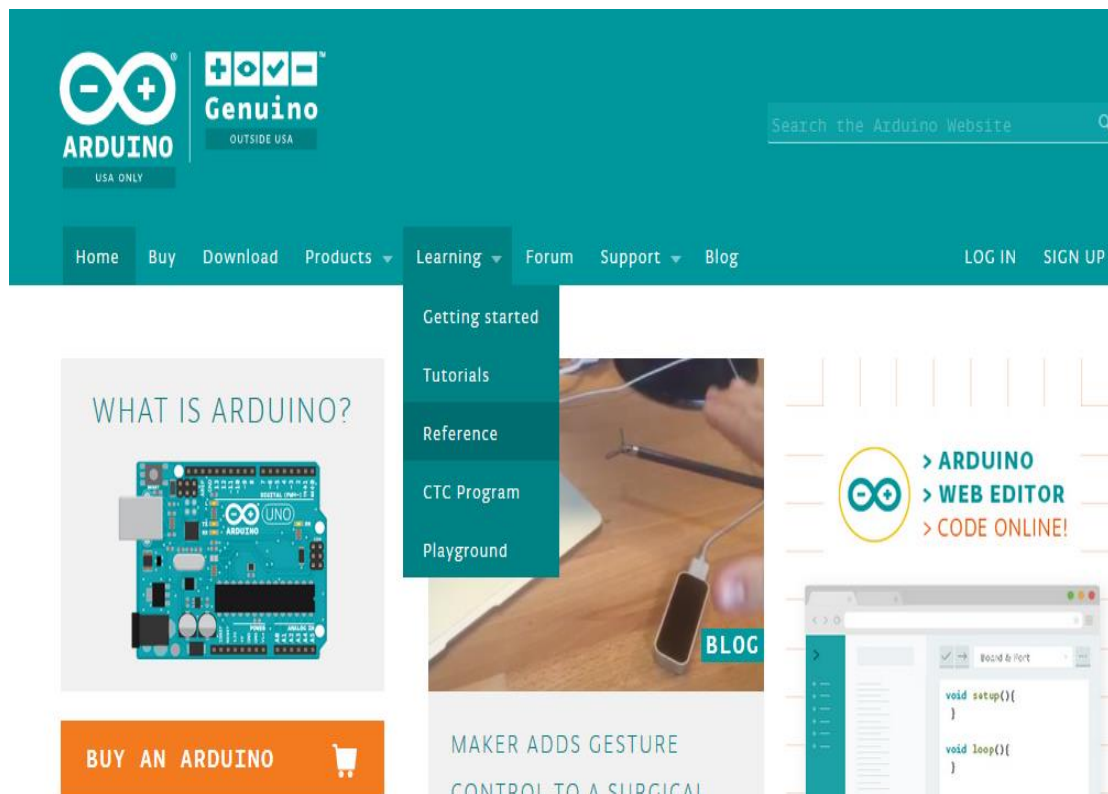
Κάποια εκτελέσιμα αρχεία (executables) είναι προγράμματα και βιβλιοθήκες ταυτόχρονα. Οι περισσότερες βιβλιοθήκες όμως δεν είναι εκτελέσιμες. Τα εκτελέσιμα αρχεία και οι βιβλιοθήκες αναφέρονται το ένα στον κώδικα και τα δεδομένα του άλλου μέσω μιας διαδικασίας που ονομάζεται σύνδεση και την πραγματοποιεί ο συνδέτης. Τα σύγχρονα λειτουργικά συστήματα μας παρέχουν βιβλιοθήκες που πραγματοποιούν την πλειονότητα των υπηρεσιών του συστήματος. Έτσι, ο περισσότερος κώδικας που χρησιμοποιούν οι σύγχρονες εφαρμογές παρέχεται από αυτές τις βιβλιοθήκες και δεν χρειάζεται να γραφεί από την αρχή για κάθε νέο πρόγραμμα. Για την εγκατάσταση του Arduino είναι απαραίτητη η χρήση των βιβλιοθηκών που μας παρέχει το λογισμικό.

Για την εισαγωγή τους επισκεπτόμαστε ξανά την επίσημη ιστοσελίδα του Arduino στο «www.arduino.cc».



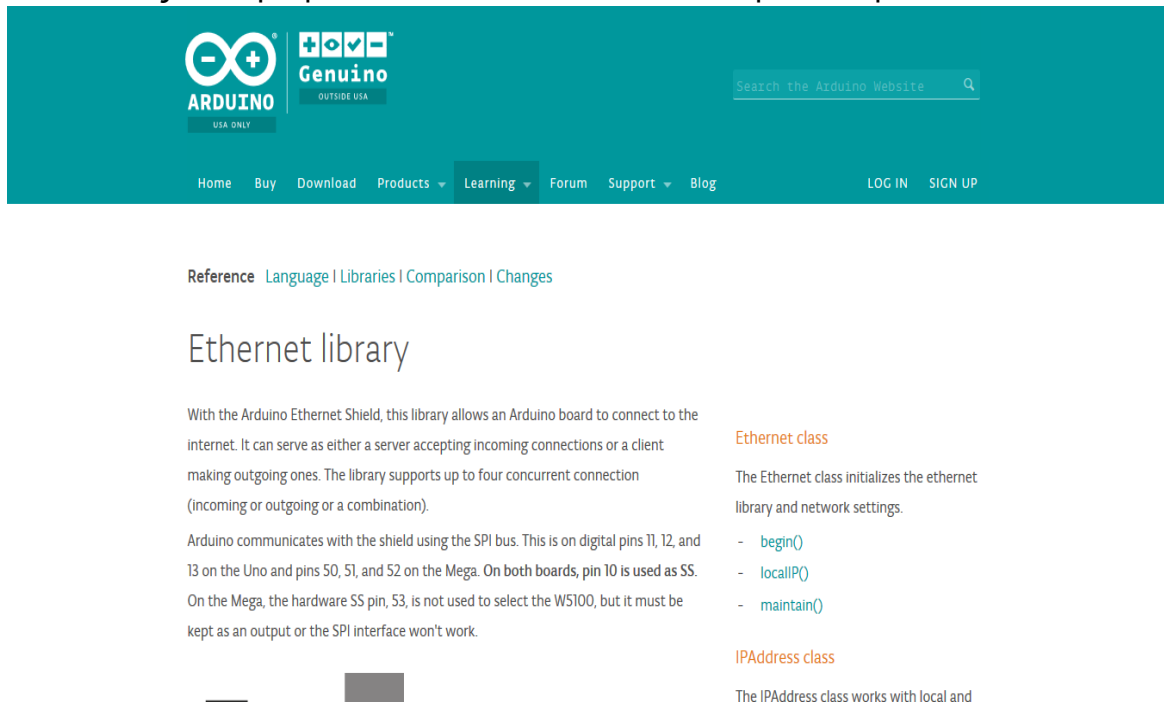
Εικόνα 25: Κεντρική σελίδα του Arduino [18]

Έπειτα πάμε στην εντολή Learning και στην εντολή Reference.



Εικόνα 26: Επιλογή Reference [19]

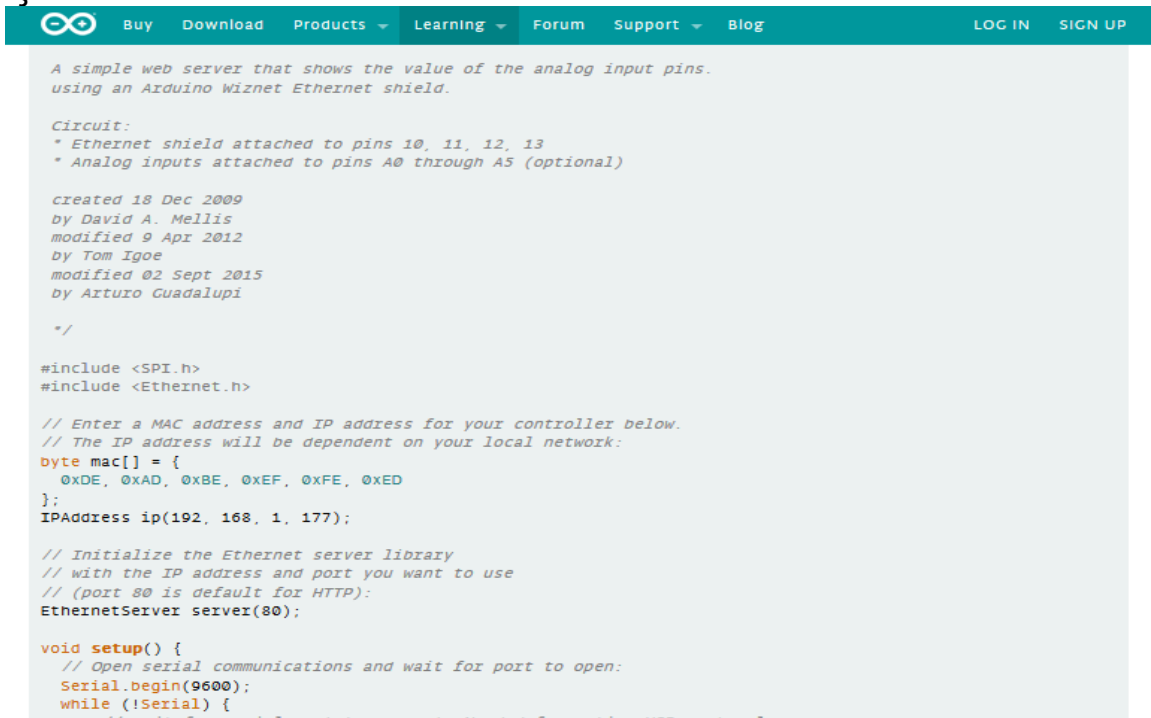
Τέλος επιλέγουμε από το Standard Libraries την εντολή Ethernet.



The screenshot shows the Arduino website's 'Reference' section for the 'Ethernet library'. The page title is 'Ethernet library'. The main text explains that the library allows an Arduino board to connect to the internet via an Ethernet Shield. It details the SPI bus connection for Uno and Mega boards. On the right side, there are sections for 'Ethernet class' and 'IPAddress class'. The 'Ethernet class' section states it initializes the library and network settings, listing methods: begin(), localIP(), and maintain(). The 'IPAddress class' section notes it works with local and

Εικόνα 27: Ethernet library [20]

Στη συνέχεια αντιγράφουμε το πρόγραμμα από τα παραδείγματα που μας έχει για τον Web Server δημιουργούμε ένα νέο sketch και κάνουμε επικόλληση τον κώδικα μας.



The screenshot shows a code example for a 'Web Server' on the Arduino website. The code is as follows:

```
A simple web server that shows the value of the analog input pins.
using an Arduino Wiznet Ethernet shield.

Circuit:
* Ethernet shield attached to pins 10, 11, 12, 13
* Analog inputs attached to pins A0 through A5 (optional)

created 18 Dec 2009
by David A. Mellis
modified 9 Apr 2012
by Tom Igoe
modified 02 Sept 2015
by Arturo Guadalupi

*/

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

// Enter a MAC address and IP address for your controller below.
// The IP address will be dependent on your local network:
byte mac[] = {
  0xDE, 0xAD, 0xBE, 0xEF, 0xFE, 0xED
};
IPAddress ip(192, 168, 1, 177);

// Initialize the Ethernet server library
// with the IP address and port you want to use
// (port 80 is default for HTTP):
EthernetServer server(80);

void setup() {
  // Open serial communications and wait for port to open:
  Serial.begin(9600);
  while (!Serial) {
    // wait for serial port to connect. Needed for native USB port only
  }
}
```

Εικόνα 28: Έτοιμο παράδειγμα για Web Server [21]

Αντιγράφουμε το παράδειγμα του κώδικα σε ένα νέο sketch και αλλάζουμε την IP του παραδείγματος με την δική μας και θέτουμε την Ethernet shield σε λειτουργία.

```
sketch_jun18a | Arduino 1.0.1
Αρχείο Επέξεργασία Σχέδιο Εργαλεία Βοήθεια

sketch_jun18a $

// Enter a MAC address and IP address for your controller below.
// The IP address will be dependent on your local network:
byte mac[] = {
  0xDE, 0xAD, 0xBE, 0xEF, 0xFE, 0xED
};
IPAddress ip(192, 168, 1, 177);

// Initialize the Ethernet server library
// with the IP address and port you want to use
// (port 80 is default for HTTP):
EthernetServer server(80);

void setup() {
  // Open serial communications and wait for port to open:
  Serial.begin(9600);
  while (!Serial) {
    ; // wait for serial port to connect. Needed for native USB port only
  }

  // start the Ethernet connection and the server:
  Ethernet.begin(mac, ip);
  server.begin();
  Serial.print("server is at ");
  Serial.println(Ethernet.localIP());
}


```

Εικόνα 29: Sketch παράδειγμα για Web Server

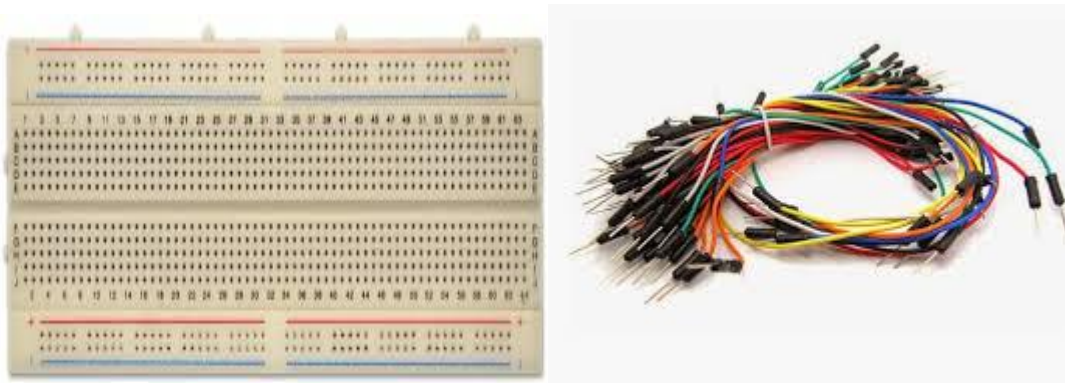
2.2 ΤΙ ΘΑ ΧΡΕΙΑΣΤΟΥΜΕ ΓΙΑ ΤΟ ΚΥΚΛΩΜΑ ΜΑΣ

Για την κατασκευή μας θα χρειαστούμε 3 LED σε λευκό χρώμα, έναν ανεμιστήρα 5Volt, 3 αντιστάσεις 100Ω (αν δεν χρησιμοποιήσουμε αντιστάσεις στο κύκλωμα μας τα LED θα καταστραφούν), καλώδια για τις μεταξύ τους συνδέσεις κατάλληλα για Arduino και τέλος ένα καλώδιο USB για την τροφοδοσία του Arduino μας.



Εικόνα 30: Αντίσταση 100Ω, λευκό LED [22]

Επίσης όλα τα παραπάνω θα τοποθετηθούν στο Arduino και σε μια Bread Board πλακέτα.



Εικόνα 31: Bread Board πλακέτα, καλώδια κατάλληλα για Arduino [23]



Εικόνα 32: Ανεμιστήρας 5volt, καλώδιο USB για το Arduino [24]

2.3 ΑΝΑΠΤΥΞΗ ΠΡΟΓΡΑΜΜΑΤΟΣ ΓΙΑ ΤΟ ARDUINO

Όταν τελειώσουμε με την συνδεσμολογία είμαστε έτοιμοι να ετοιμάσουμε τον κώδικα στο πρόγραμμα μας, το οποίο θα περάσουμε στο Arduino Uno. Θα δούμε μερικά βασικά μέρη του κώδικα, για να κατανοήσουμε την λειτουργία του. Αρχικά πρέπει να καλέσουμε της βιβλιοθήκες που θα χρησιμοποιήσουμε, οι οποίες είναι:

```
#include "SPI.h"  
#include "Ethernet.h"  \ βιβλιοθήκες για την Ethernet shield ώστε να την διαβάξει το  
Arduino.
```

```
#include "WebServer.h" \ βιβλιοθήκες για την δημιουργία web server
```

Στη συνέχεια τοποθετούμε την IP μας για να δημιουργήσουμε τον Web Server μας.

```
static uint8_t mac[6] = { 0xDE, 0xDE, 0xDE, 0xDE, 0xDE, 0xDE };
```

```
static uint8_t ip[4] = { 192, 168, 2, 12 }; // εδώ τοποθετούμε την δική μας ip
```

```
#define PREFIX "/remote"
```

Η Ethernet Shield server(80) δημιουργεί ένα Web Server στην πόρτα 80 για την μεταφορά δεδομένων.

```
WebServer webserver(PREFIX, 80);
```

Δηλώνουμε τις εισόδους για τα led και τον ανεμιστήρα μας, στην είσοδο 5,6,3 που έχουμε δηλώσει τα led μας και στην είσοδο 2 τον ανεμιστήρα μας

```
#define LED1 5 //εδώ έχουμε τοποθετήσει το πρώτο μας λευκό led  
#define LED2 3 //εδώ έχουμε τοποθετήσει το δεύτερο μας λευκό led
```

```
#define LED3 6 //εδώ έχουμε τοποθετήσει το τρίτο μας λευκό led
```

```
#define ANEMISTHRAS 2 //εδώ έχουμε τοποθετήσει τον ανεμιστήρα μας
```

```
int led1= 0; //pin 5 // δηλώνουμε το πρώτο μας led στην είσοδο 5  
int led2= 0; //pin 6 // δηλώνουμε το δεύτερο μας led στην είσοδος 6  
int led3= 0; //pin 3 // δηλώνουμε το τρίτο μας led στην είσοδο 3  
int anemisthras = 2; //pin 2 // δηλώνουμε τον ανεμιστήρα μας στην είσοδο 2  
void rgbCmd(WebServer &server, WebServer::ConnectionType type, char *, bool)  
{  
if (type == WebServer::POST)  
{  
bool repeat;  
char name[16], value[16];  
do  
{  
repeat = server.readPOSTparam(name, 16, value, 16);
```

Δημιουργία συνθήκης με το αν είναι ή όχι ενεργοποιημένα τα led μας και ο ανεμιστήρας μας.

```
if (strcmp(name, "led1") == 0)
```

```
{  
red = strtoul(value, NULL, 10);  
}  
if (strcmp(name, "led2") == 0)  
{
```

```

green = strtoul(value, NULL, 10);
}
if (strcmp(name, "led3") == 0)
{
blue = strtoul(value, NULL, 10);
}
if (strcmp(name, "led4") == 0)
{
brown = strtoul(led5, NULL, 10);

}
} while (repeat);

server.httpSeeOther(PREFIX);

return;
}

```

```
server.httpSuccess();
```

```
//Εξοδοι
```

```

void setup()
{
pinMode(LED1, OUTPUT);
pinMode(LED2, OUTPUT);
pinMode(LED3, OUTPUT);
pinMode(ANEMISTHRAS, OUTPUT);
Ethernet.begin(mac, ip);

webservice.setDefaultCommand(&rgbCmd);

webservice.begin();
}

```

Επαναλαμβανόμενη διαδικασία ώστε να τρέχει συνεχώς στο Arduino μας.

```

void loop()
{
webservice.processConnection();
analogWrite(LED1, led1);
analogWrite(LED2, led2);
analogWrite(LED3, led3);
analogWrite(ANEMISTHRAS, anemisthras);
}

```


2.4 ΑΝΑΠΤΥΞΗ ΚΩΔΙΚΑ HTML

Η HTML (HyperText Markup Language, ελλ. Γλώσσα Σήμανσης Υπερκειμένου) είναι η κύρια γλώσσα κατασκευής για τις ιστοσελίδες, και τα στοιχεία της είναι τα βασικά στοιχεία των τους.

Η HTML γράφεται σε μορφή στοιχείων HTML τα οποία αποτελούνται από ετικέτες (tags), οι οποίες περιλαμβάνουν μέσα σε σύμβολα «μεγαλύτερο από» και «μικρότερο από» (για παράδειγμα <html>), μέσα στο περιεχόμενο της ιστοσελίδας. Οι ετικέτες HTML συνήθως λειτουργούν σε ζεύγη (για παράδειγμα <h1> και </h1>), με την πρώτη να ονομάζεται ετικέτα έναρξης και τη δεύτερη ετικέτα λήξης. Ενδιάμεσα αυτών των ετικετών, οι κατασκευαστές ιστοσελίδων μπορούν να τοποθετήσουν κείμενο, πίνακες, εικόνες κλπ. Ένας web browser στόχο έχει να διαβάζει τα έγγραφα HTML και να τα συνθέτει σε σελίδες που μπορεί κανείς να διαβάσει αλλά και να ακούσει. Στον browser δεν είναι ορατές οι ετικέτες HTML, αλλά τις χρησιμοποιεί για να «καταλάβει» το περιεχόμενο της σελίδας.

Τα στοιχεία της HTML χρησιμοποιούνται για να σχεδιάσουν όλους του ιστότοπους. Η HTML επιτρέπει την εισαγωγή εικόνων και άλλων αντικειμένων μέσα στη σελίδα, και μπορεί να χρησιμοποιηθεί για να εμφανίσει διαδραστικές φόρμες με σκοπό την επικοινωνία με τον χρήστη. Παρέχει τις μεθόδους δημιουργίας δομημένων εγγράφων (δηλαδή εγγράφων που αποτελούνται από το περιεχόμενο που μεταφέρουν και από τον κώδικα μορφοποίησης του περιεχομένου) καθορίζοντας δομικά σημαντικά στοιχεία για το κείμενο, όπως κεφαλίδες, παραγράφους, λίστες, συνδέσμους, σχόλια και άλλα. Μπορούν επίσης να ενσωματώνονται ομάδες εντολών σε γλώσσες όπως η JavaScript, τα οποία επηρεάζουν τη σύσταση και τη δομή των ιστοσελίδων HTML.

Στην δική μας εφαρμογή – εγκατάσταση θα δημιουργήσουμε έναν HTML κώδικα που σκοπό θα έχει την δημιουργία ενός κεντρικού πάνελ διαχείρισης των τριών led και του ανεμιστήρα με κουμπί «ON-OFF» από τον ηλεκτρονικό μας υπολογιστή καθώς και από το έξυπνο κινητό μας ή το tablet μας.

Στο πρώτο Head έχουμε γράψει ένα τίτλο (Smart Home), όπου αυτός ο τίτλος θα εμφανίζεται στην καρτέλα του Web Browser που θα χρησιμοποιήσουμε. Ανάμεσα στα Script θα μπει ο κώδικας, ο οποίος θα επικοινωνεί με τον κώδικα του Arduino.

```
P(message) =
"<!DOCTYPE html><html><head>"
  "<meta http-equiv='Content-Type' content='text/html; charset=utf-8'>"
  "<title>Smart Home</title>"
  "<link rel='stylesheet' href='http://code.jquery.com/mobile/1.3.1/jquery.mobile-1.3.1.min.css' />"
  "<script src='http://code.jquery.com/jquery-1.9.1.min.js'></script>"
  "<script src='http://code.jquery.com/mobile/1.3.1/jquery.mobile-1.3.1.min.js'></script>"
  "<script>"
// va
  "$(document).ready(function(){ $('#led1, #led2, #led3, #anemisthras').slider;
$('#led1, #led2, #led3, #anemisthras').bind('change', function(event, ui) {
jQuery.ajaxSetup({timeout: 1000}); /*not to DDoS the Arduino, you might have to
change this to some threshold value that fits your setup*/ var id = $(this).attr('id'); var
strength = $(this).val(); if (id == 'led1') $.post('/remote', { led1: strength } ); if (id ==
```

```
'led2') $.post('/remote', { led2: strength } ); if (id == 'led3') $.post('/remote', { led3:
strength } ); if (id=='anemisthras') $.post('/remote', { anemisthras: strength }); });});"
"</script>"
```

Ορισμός τίτλου που θα εμφανίζεται στο πάνελ μας ο οποίος θα είναι «Smart Home by TEIWEST»

```
"</head>"
"<body>"
"<div data-role=\"header\" data-position=\"inline\"><h1>Smart Home by
TEIWEST</h1></div>"
"<div class=\"ui-body ui-body-a\">"
"<div data-role=\"fieldcontain\">"
```

Εδώ δημιουργούμε για το led1, led2, led3 και τον ανεμιστήρα μας κουμπί «ON-OFF» διακόπτη στο πάνελ μας.

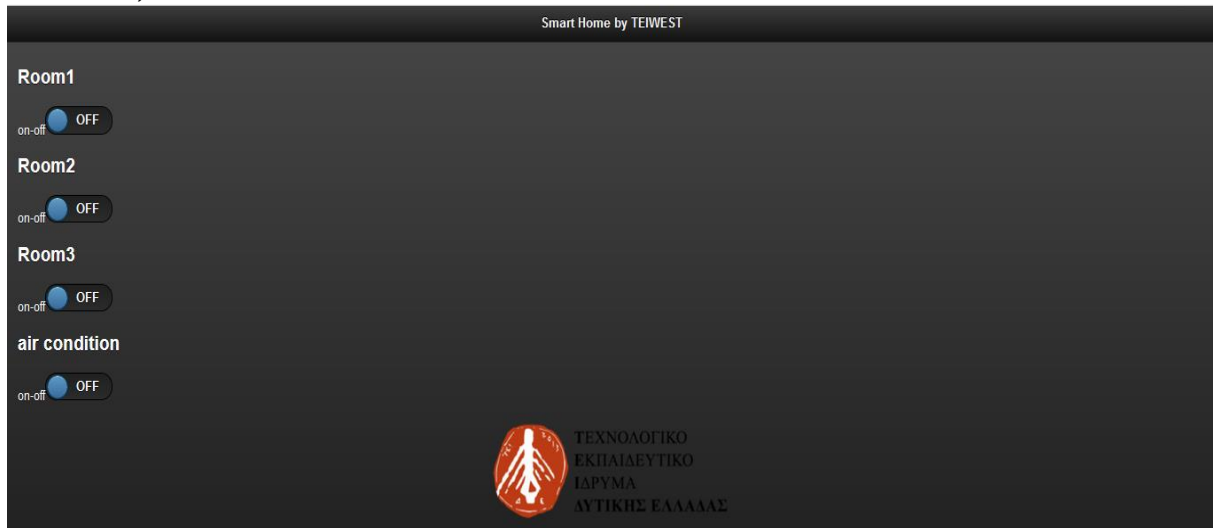
```
"<h2>Room1</h2>"
"<label for=\"flip-3\">on-off</label>"
"<select name=\"flip-3\" id=\"led1\" data-role=\"slider\" data-theme=\"b\">"
"<option value=\"0\">OFF</option>"
"<option value=\"255\">ON</option>"
"</select> "
"<h2>Room2</h2>"
"<label for=\"flip-3\">on-off</label>"
"<select name=\"flip-3\" id=\"led2\" data-role=\"slider\" data-theme=\"b\">"
"<option value=\"0\">OFF</option>"
"<option value=\"255\">ON</option>"
"</select> "
"<h2>Room3</h2>"
"<label for=\"flip-3\">on-off</label>"
"<select name=\"flip-3\" id=\"led3\" data-role=\"slider\" data-theme=\"b\">"
"<option value=\"0\">OFF</option>"
"<option value=\"255\">ON</option>"
"</select> "
"<h2>air condition</h2>"
"<label for=\"flip-3\">on-off</label>"
"<select name=\"flip-3\" id=\"anemisthras\" data-role=\"slider\" data-
theme=\"b\">"
"<option value=\"0\">OFF</option>"
"<option value=\"255\">ON</option>"
```

Εισαγωγή εικόνας στο κάτω μέρος του πάνελ μας από το διαδίκτυο και στη περίπτωση μας έχουμε τοποθετήσει το λογότυπο του Τεχνολογικού Εκπαιδευτικού Ιδρύματος Δυτικής Ελλάδας.

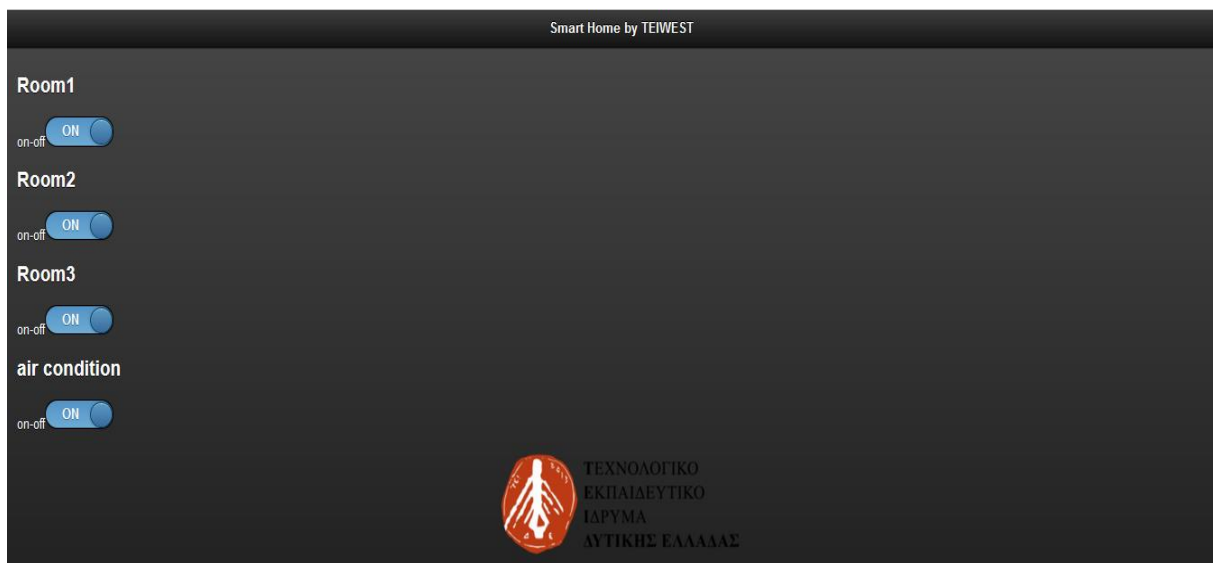
```

"/div>"
"<center><img src=\"http://fluids.mech.teiwest.gr/Fluids/wp-
content/uploads/2016/04/TEIWEST_DOC_HEADER-GR-COLOR-1-
e1460225692392.png\"></center>"
"/div>"
"/body>"
"/html>";

```



Εικόνα 33: Περιβάλλον html OFF



Εικόνα 34: Περιβάλλον html ON

ΚΕΦΑΛΑΙΟ 3. Δημιουργία έξυπνης κλειδαριάς

Η πρόσβαση χωρίς κλειδιά στις μέρες μας είναι ο πιο ασφαλής και εύκολος τρόπος για την είσοδο μας στο σπίτι. Επομένως η δημιουργία μια έξυπνης κλειδαριάς έχει ως σκοπό την εύκολη πρόσβαση μας από την κεντρική είσοδο της κατοικίας μας, αυτό το πετυχαίνουμε πληκτρολογώντας έναν συνδυασμό από αριθμούς και γράμματα που έχουμε ρυθμίσει εμείς οι ίδιοι και είναι επίσης εύκολο να γίνεται συχνή αλλαγή σε αυτόν για την καλύτερη ασφάλειά μας.

3.1 ΤΙ ΘΑ ΧΡΕΙΑΣΤΟΥΜΕ ΓΙΑ ΤΟ ΚΥΚΛΩΜΑ ΜΑΣ

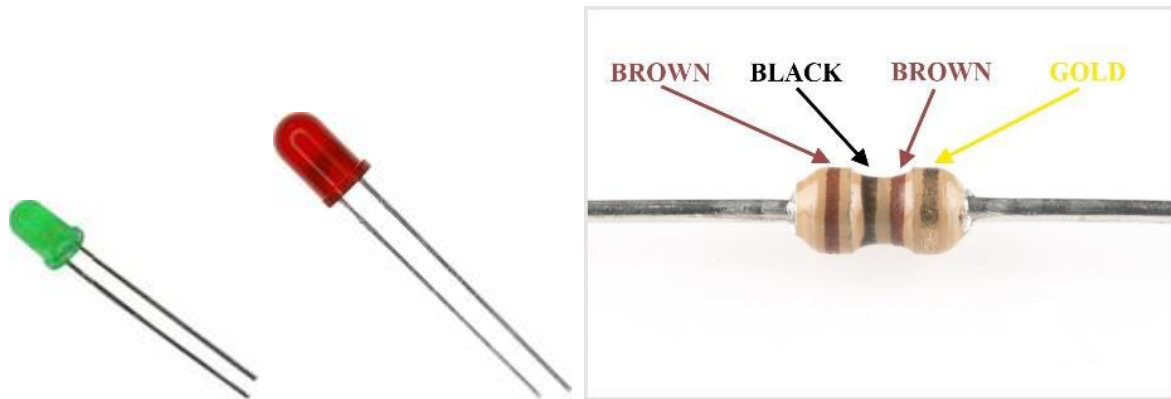
Για την κατασκευή μας θα χρειαστούμε ένα Arduino UNO, 2 LED χρώματος πράσινο και κόκκινο, μια Bread Board πλακέτα, ένα Keypad, 2 αντιστάσεις 100Ω, ένα σερβοκινητήρα, καλώδια και τέλος ένα κουτί κατασκευών για την τοποθέτηση των παραπάνω.



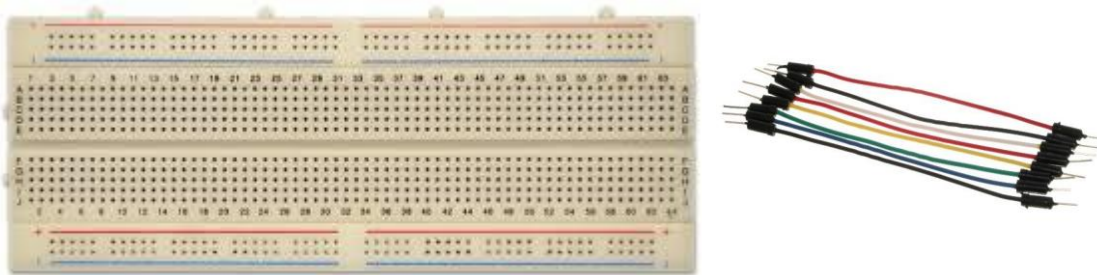
Εικόνα 35: Πλακέτα Arduino UNO [25]



Εικόνα 36: Πληκτρολόγιο Keypad [26]



Εικόνα 37: LED πράσινο, LED κόκκινο, αντίσταση 100Ω [27]



Εικόνα 38: Bread board πλακέτα και καλώδια για Arduino [28]



Εικόνα 39: Πλαστικό κουτί κατασκευών [29]



Εικόνα 40: Σερβοκινητήρας [30]

3.2 ΤΙ ΕΙΝΑΙ ΤΟ ΚΕΥΡΑΔ ΠΛΗΚΤΡΟΛΟΓΙΟ

Το Keypad είναι ένα αριθμητικό πληκτρολόγιο. Αυτό το πληκτρολόγιο έχει 16 κουμπιά, αποτελείται από 10 αριθμούς και από 6 χαρακτήρες τοποθετημένα σε μια τηλεφωνική γραμμή 4x4 πλέγμα. Είναι κατασκευασμένο από ένα λεπτό, εύκαμπτο υλικό μεμβράνης με αυτοκόλλητη υποστήριξη ώστε να μπορείτε να το συνδέσουμε εύκολα. Τα κλειδιά συνδέονται με 7 καλώδια τύπου καρφίτσες στον μικροελεγκτή Arduino (4-στήλες και 4 σειρές).

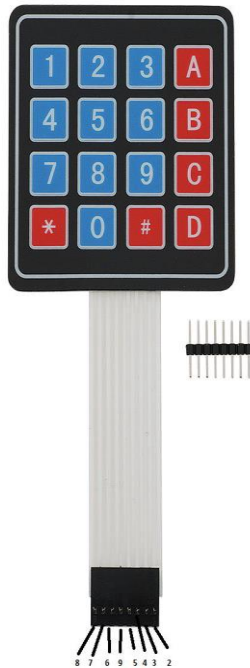
3.3 ΤΙ ΕΙΝΑΙ Ο ΣΕΡΒΟΚΙΝΗΤΗΡΑΣ

Ο σερβοκινητήρας θα χρησιμοποιηθεί για το άνοιγμα και το κλείσιμο της κεντρικής εισόδου στο έξυπνο σπίτι μας. Ο σερβοκινητήρας είναι ένας περιστροφικός ενεργοποιητής ή γραμμικός ενεργοποιητής που επιτρέπει τον ακριβή έλεγχο της γωνιακής ή γραμμικής θέσης, ταχύτητας και επιτάχυνσης. Αποτελείται από ένα κατάλληλο κινητήρα συνδέεται με έναν αισθητήρα για ανάδραση της θέσης.

Σερβοκινητήρες δεν αποτελούν ειδική κατηγορία κινητήρων, αν και ο όρος σερβοκινητήρας συχνά χρησιμοποιείται για να αναφερθεί σε έναν κινητήρα κατάλληλο για χρήση σε ένα σύστημα ελέγχου κλειστού βρόχου. Οι σερβοκινητήρες χρησιμοποιούνται σε εφαρμογές όπως η ρομποτική, CNC μηχανήματα ή αυτοματοποιημένη παραγωγή. Ένας σερβοκινητήρας είναι κλειστού βρόχου σερβομηχανισμού που χρησιμοποιεί ανάδραση θέσης για να μπορεί να ελέγξει την κίνησή του και την τελική του θέση. Η είσοδος ελέγχου του είναι κάποιο σήμα, είτε αναλογικό είτε ψηφιακό, που αντιπροσωπεύει τη θέση εντολή για τον άξονα εξόδου. Στην περίπτωση μας η κίνηση που έχουμε προγραμματίσει είναι από την αρχική θέση που είναι η θέση 0 μοίρες, στην τελική θέση που είναι η 90 μοίρες.

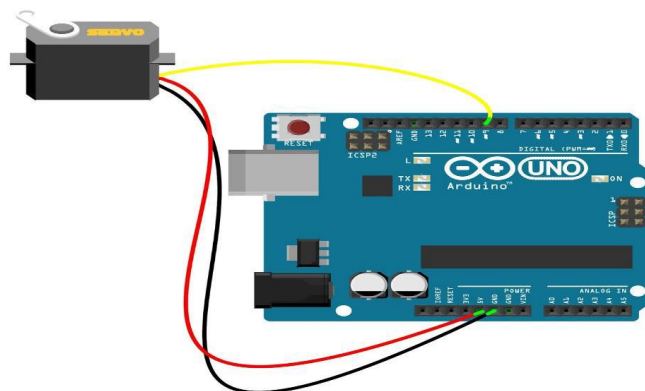
3.4 ΣΥΝΔΕΣΗ ARDUINO ΜΕ ΤΟ KEYPAD ΚΑΙ ΤΟΝ ΣΕΡΒΟΚΙΝΗΤΗΡΑ

Συνδέουμε το Keypad μας με το Arduino στα ψηφιακά μας pins στους αριθμούς 8,7,6,9,5,4,3,2 (είναι αριθμοί είναι με την σειρά και η σύνδεση γίνεται από τα αριστερά προς στα δεξιά από το Keypad μας στο Arduino).



Εικόνα 41: Σύνδεση Keypad με το Arduino

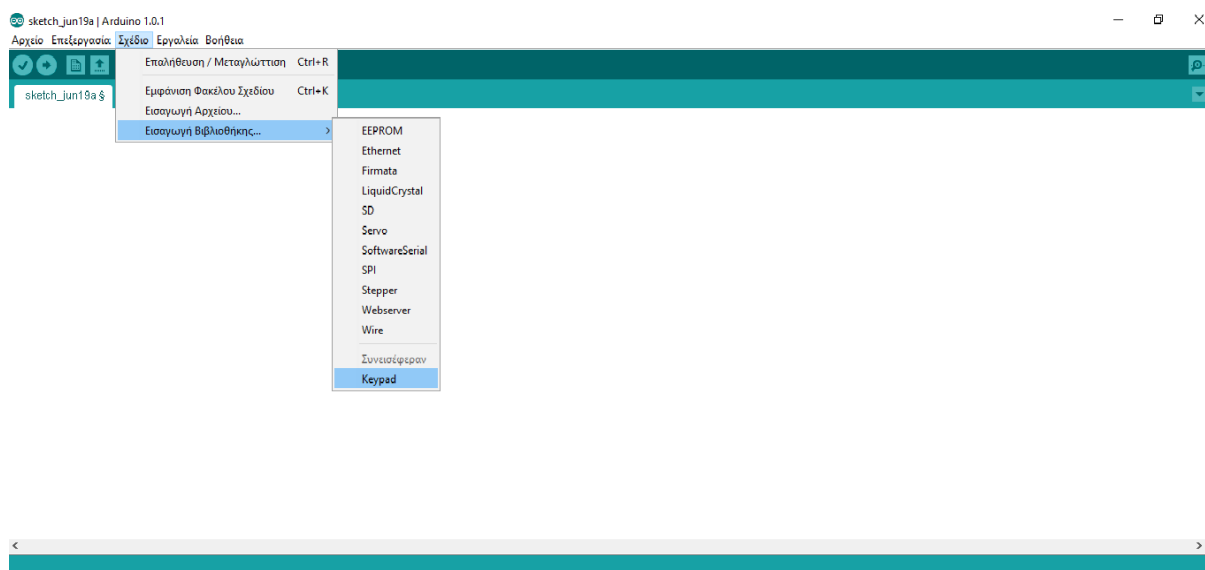
Στη συνέχεια συνδέσουμε τον σερβοκινητήρα με το Arduino μας στο ψηφιακό pin 11 και τα δυο LED μας στα ψηφιακά pins 12 και 13. Το κόκκινο LED το τοποθετούμε στο pin 12 και το πράσινο στο pin 13.



Εικόνα 42: Παράδειγμα σύνδεσης σερβοκινητήρα με Arduino [31]

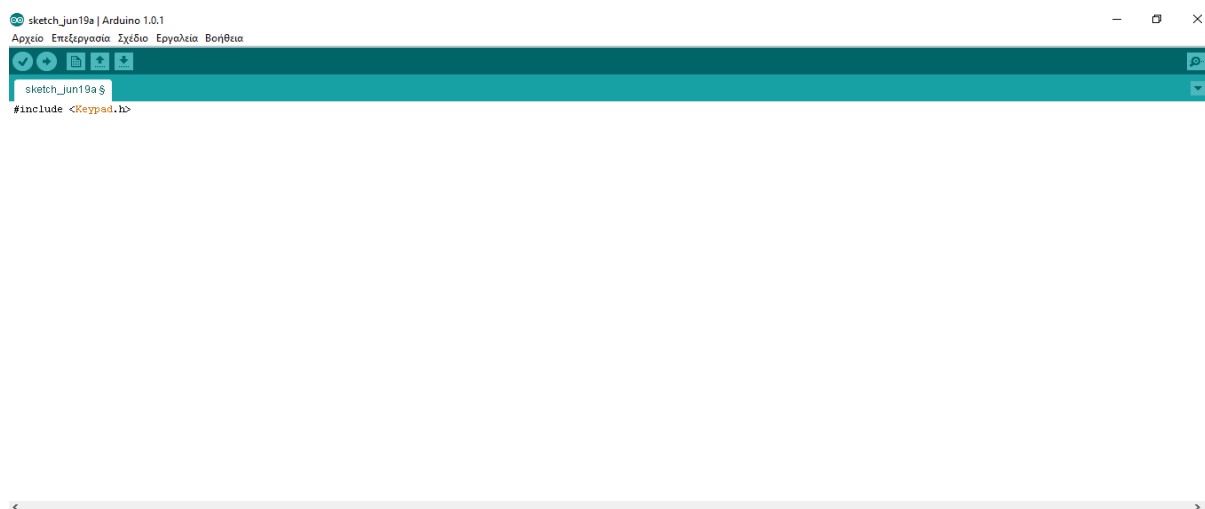
3.5 ΕΓΚΑΤΑΣΤΑΣΗ ΒΙΒΛΙΟΘΗΚΩΝ ΓΙΑ ΤΟ KEYPAD ΚΑΙ ΓΙΑ ΤΟ ΣΕΡΒΟΚΙΝΗΤΗΡΑ

Για να δουλέψει το Keypad μας θα πρέπει να επιλέξουμε από τη βιβλιοθήκη μας στο περιβάλλον προγραμματισμού Arduino IDE. Δημιουργούμε ένα sketch, από το μενού μας επιλέγουμε Σχέδιο – Εισαγωγή Βιβλιοθήκης – Keypad.



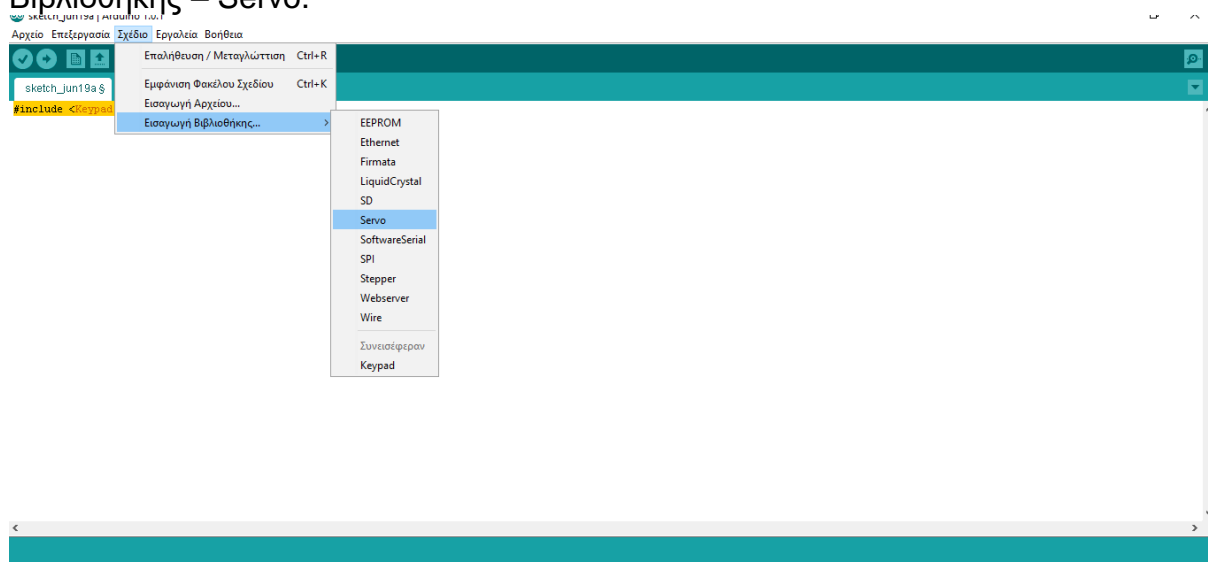
Εικόνα 43: Επιλογή εγκατάστασης βιβλιοθήκης για το Keypad

Μετά την επιλογή βιβλιοθήκης για το Keypad μας εμφανίζει στο sketch μας την εντολή «`#include <Keypad.h>`» το Keypad έχει οριστεί στο sketch.



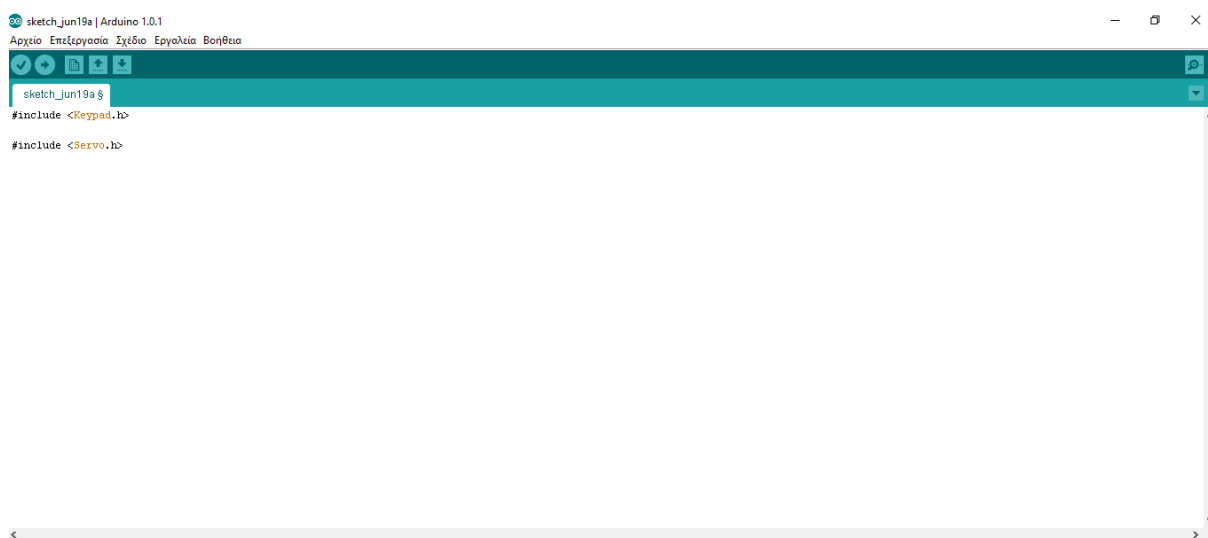
Εικόνα 44: Εμφάνιση Keypad στο sketch

Μετά την εγκατάσταση και επιλογή βιβλιοθηκών για το Keypad μας, θα πρέπει να κάνουμε ακριβώς τα ίδια βήματα και για τον σερβοκινητήρα. Άρα στο ίδιο sketch που εγκαταστήσαμε και επιλέξαμε τη βιβλιοθήκη για το keypad, επιλέγουμε από το μενού στο περιβάλλον προγραμματισμού Arduino IDE, Σχέδιο – Εισαγωγή Βιβλιοθήκης – Servo.



Εικόνα 45: Επιλογή εγκατάστασης βιβλιοθήκης για τον σερβοκινητήρα μας

Μετά την επιλογή βιβλιοθήκης για το σερβοκινητήρα μας εμφανίζει στο sketch μας την εντολή «`#include <Servo.h>`» ο σερβοκινητήρας έχει οριστεί στο sketch μας κάτω από το «`#include <Keypad.h>`».



Εικόνα 46: Εμφάνιση σερβοκινητήρα στο sketch μας

3.6 ΑΝΑΠΤΥΞΗ ΠΡΟΓΡΑΜΜΑΤΟΣ ΕΞΥΠΝΗΣ ΚΛΕΙΔΑΡΙΑΣ

Αφού εγκαταστήσαμε τις βιβλιοθήκες για το Keypad και τον σερβοκινητήρα μας, είμαστε έτοιμοι να γράψουμε το πρόγραμμά μας.

Οι πρώτες γραμμές του κώδικά μας περιλαμβάνουν τις βιβλιοθήκες, για το Keypad και τον σερβοκινητήρα, έπειτα επιλέγουμε τον κωδικό που επιθυμούμε και την θέση του σερβοκινητήρα.

```
#include <Keypad.h>
#include <Servo.h>
Servo servo_Motor;
char* password = "2580A"; //κωδικός για το keypad
int position = 0;
Στη συνέχεια δημιουργούμε έναν πίνακα 4Χ4 για το Keypad.
const byte ROWS = 4;
const byte COLS = 4;
char keys[ROWS][COLS] = {
  {'1','2','3','A'},
  {'4','5','6','B'},
  {'7','8','9','C'},
  {'*','0','#','D'}
};
byte rowPins[ROWS] = { 8, 7, 6, 9};
byte colPins[COLS] = { 5, 4, 3, 2 };
Keypad keypad = Keypad( makeKeymap(keys), rowPins, colPins, ROWS, COLS );
Δηλώνουμε τα LED μας στα pin 12 και 13, τους εξόδους για τα LED και τον
σερβοκινητήρα.
int redPin = 12;
int greenPin = 13;
void setup()
{
  pinMode(redPin, OUTPUT);
  pinMode(greenPin, OUTPUT);
  servo_Motor.attach(11);
  setLocked(true);
}
```

Εδώ έχουμε δημιουργήσει μια συνθήκη, ώστε αν επιθυμούμε να κλείσει η πόρτα μας μετά από κάποια δευτερόλεπτα, μπορούμε να επιλέξουμε είτε το '*' ή '#' και η πόρτα μας θα κλείσει σε ελάχιστα δευτερόλεπτα που εμείς έχουμε ρυθμίσει.

```
void loop()
{
  char key = keypad.getKey();
  if (key == '*' || key == '#') //για να κλεισουμε σε καποια deyterolepta
```

```

{
delay(10000);
position = 0;
setLocked(true);
}

```

Στη παρακάτω συνθήκη μας, διαβάζει διάφορους κωδικούς που πληκτρολογούμε στο Keypad. Αν ο κωδικός που έχουμε πληκτρολογήσει στο Keypad είναι λάθος τότε το LED μας είναι παραμένει κόκκινο και ο σερβοκινητήρας στη θέση 0 μοίρες που έχουμε ορίσει. Στην αντίθετη περίπτωση που ο κωδικός που πληκτρολογήσαμε είναι σωστός, τότε το LED μας γίνεται πράσινο και ο σερβοκινητήρας μας τοποθετείται στη θέση 90 μοίρες που έχουμε ορίσει.

```

if (key == password[position])
{
position ++;
}
if (position == 5)
{
setLocked(false);
}
delay(100);
}
void setLocked(int locked)
{
if (locked)
{
digitalWrite(redPin, HIGH);
digitalWrite(greenPin, LOW);
servo_Motor.write(11);
}
else
{
digitalWrite(redPin, LOW);
digitalWrite(greenPin, HIGH);
servo_Motor.write(90);
}
}

```

Αφού τελειώσουμε με το πρόγραμμά μας, είμαστε έτοιμοι να τοποθετήσουμε το Arduino, Keypad, τα LED και τέλος την Bread board πλακέτα μας στο πλαστικό κουτί κατασκευών που έχουμε προμηθευτεί. Τέλος η έξυπνη κλειδαριά θα τοποθετηθεί στην μακέτα που θα δημιουργήσουμε στο τελευταίο μας κεφάλαιο.



Εικόνα 47: Τελική μορφή έξυπνης κλειδαριάς

ΚΕΦΑΛΑΙΟ 4. Τοποθέτηση αισθητήρα φωτός σε πλακέτα Arduino με σκοπό το φωτισμό εξωτερικού χώρου μετά τη δύση του ηλίου

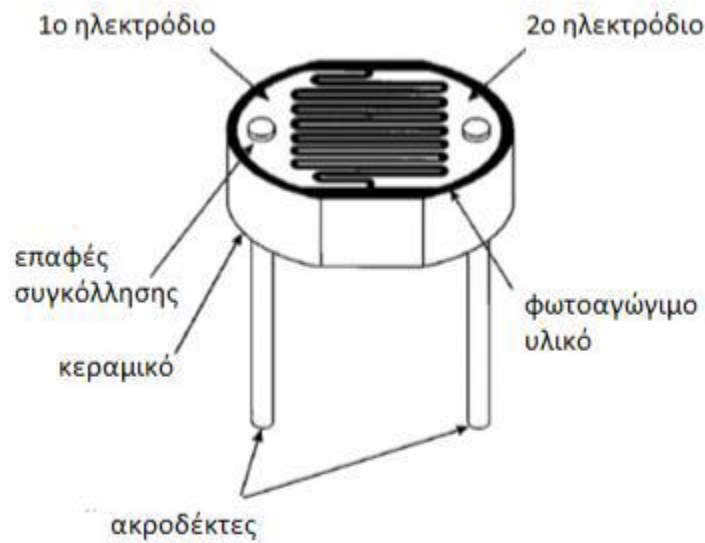
Η τοποθέτηση ενός αισθητήρα φωτός θα μας βοηθήσει να έχουμε ένα αυτόματο σύστημα φωτισμού στον εξωτερικό χώρο του σπιτιού μας χωρίς καμία παρέμβαση από εμάς με τη χρήση κάποιου διακόπτη ή χρονοδιακόπτη. Ο αισθητήρας θα ενεργοποιείται όταν ο ήλιος θα δύει και θα απενεργοποιείται μετά την ανατολή του ηλίου. Με την τοποθέτηση αισθητήρα φωτός στον εξωτερικό χώρο του σπιτιού μας πετυχαίνουμε και εξοικονόμηση ενέργειας.

4.1 ΤΙ ΕΙΝΑΙ Ο ΑΙΣΘΗΤΗΡΑΣ ΦΩΤΟΣ

Ο αισθητήρας φωτός είναι ένα ηλεκτρονικό εξάρτημα που ανιχνεύει την παρουσία του ορατού φωτός, την μετάδοση υπεριώθρου φωτός ή της υπεριώδους ενέργειας. Οι περισσότεροι φωτοαισθητήρες αποτελούνται από ημιαγωγούς που έχει μια ιδιότητα που ονομάζεται photoconductivity (photoconductivity είναι ένα οπτικό και ηλεκτρικό φαινόμενο στο οποίο ένα υλικό γίνεται πιο ηλεκτρικά αγώγιμο λόγω της απορρόφησης της ηλεκτρομαγνητικής ακτινοβολίας, όπως το ορατό φως, υπεριώδες φως, υπέρυθρο φως στα οποία η ηλεκτρική αγωγιμότητα μεταβάλλεται ανάλογα με την ένταση του φωτός που κοιτάζει τον αισθητήρα). Με λίγα λόγια ο αισθητήρας φωτός είναι μια αντίσταση που η τιμή της μεταβάλλεται συνέχεια ανάλογα με το πόσο φως πέφτει πάνω στην επιφάνειά του αισθητήρα μας. Το κόστος ενός τέτοιου αισθητήρα είναι πολύ χαμηλό, προσαρμόζεται σχεδόν σε οποιαδήποτε ανάγκη μας εφόσον παρασκευάζεται σε πολλά μεγέθη και προδιαγραφές και η επιλογή γίνεται ανάλογα με της ανάγκες που έχουμε, σημαντικό είναι να τονίσουμε ότι κάθε αισθητήρας φωτός θα λειτουργεί λίγο διαφορετικά από τον άλλον ακόμα και αν είναι του ίδιου τύπου.



Εικόνα 48: Αισθητήρας φωτός [32]



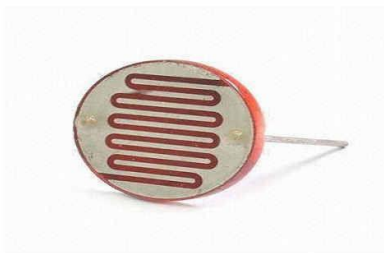
Εικόνα 49: Ανάλυση του αισθητήρα φωτός [33]

4.2 ΤΙ ΘΑ ΧΡΕΙΑΣΤΟΥΜΕ ΓΙΑ ΤΗΝ ΚΑΤΑΣΚΕΥΗ ΜΑΣ ΚΑΙ ΓΙΑ ΤΗΝ ΣΥΝΔΕΣΜΟΛΟΓΙΑ ΤΟΥ ΑΙΣΘΗΤΗΡΑ ΣΤΗ ΠΛΑΚΕΤΑ ARDUINO

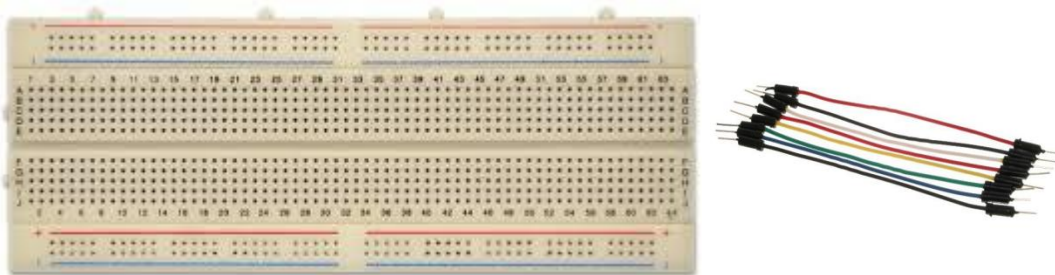
Για την δημιουργία της κατασκευής μας, δηλαδή την ενεργοποίηση του εξωτερικού φωτισμού ανάλογα με την ηλιακή ακτινοβολία, θα χρειαστούμε έναν μικροελεγκτή Arduino UNO, 3 LED λευκού χρώματος, 3 αντιστάσεις 100Ω, 1 αντίσταση 10KΩ, καλώδια κατάλληλα για Arduino και USB καλώδιο για την τροφοδοσία του Arduino, μια πλακέτα Bread Board και φυσικά τον αισθητήρα φωτός.



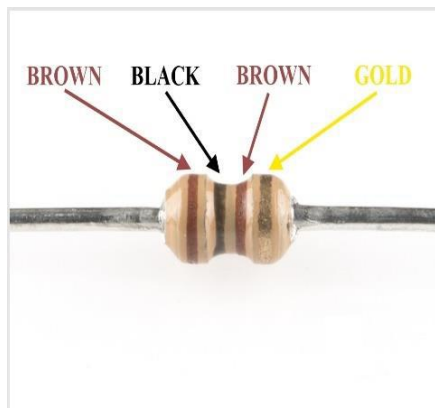
Εικόνα 50: Μικροελεγκτής Arduino UNO και καλώδιο USB τροφοδοσίας. [34]



Εικόνα 51: Αισθητήρας φωτός και αντίσταση 10KΩ [35]

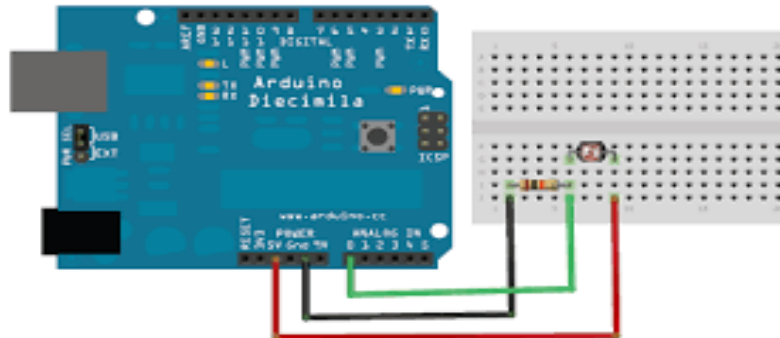


Εικόνα 52: Bread board πλακέτα και καλώδια για Arduino [36]



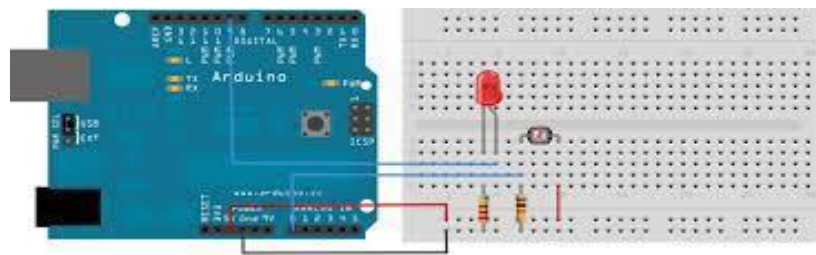
Εικόνα 53: Λευκό LED, αντίσταση 100Ω [37]

Ο αισθητήρας φωτός συνδέεται στο Arduino στα αναλογικά του pins του Arduino μαζί με την αντίσταση 10KΩ. Τοποθετούμε την αντίσταση στον αισθητήρα ώστε να τον θέσουμε σε λειτουργία.



Εικόνα 54: Παράδειγμα συνδεσμολογίας του αισθητήρα φωτός με τον μικροελεγκτή Arduino [38]

Τα LED τα τοποθετούμε στις ψηφιακές εξόδους του μικροελεγκτή, ώστε να ενεργοποιούνται όταν ο αισθητήρας τεθεί σε λειτουργία.



Εικόνα 55: Παράδειγμα συνδεσμολογίας του αισθητήρα φωτός και ενός LED στον μικροελεγκτή Arduino [39]

4.3 ΑΝΑΠΤΥΞΗ ΠΡΟΓΡΑΜΜΑΤΟΣ ΓΙΑ ΤΟΝ ΑΙΣΘΗΤΗΡΑ ΦΩΤΟΣ

Ο προγραμματισμός για τον αισθητήρα φωτός γίνεται σε ένα sketch στο περιβάλλον προγραμματισμού Arduino IDE. Στις πρώτες γραμμές του προγράμματός μας δηλώνουμε τα LED στα ψηφιακά pins 11, 10 , 9 και τον αισθητήρα φωτός στο αναλογικό pin A0.

```
{  
int light_sensor = A0; //analogiko pin 0  
int led1 = 11; //digital pin 11  
int led2=10; //digital pin 10  
int led3=9; // digital pin 9  
void setup(){  
Serial.begin(9600); //serial gia debug  
pinMode(led1, OUTPUT);  
pinMode(led2 ,OUTPUT);  
pinMode(led3,OUTPUT);  
}
```

Στη συνέχεια δημιουργούμε μια επανάληψη με μια συνθήκη, στην οποία αν ο αισθητήρας δέχεται μικρότερη από 600 lux που είναι η μονάδα μέτρησης φωτός τότε τα LED μας ενεργοποιούνται.

```
void loop()  
{  
int light_reading = analogRead(light_sensor);  
if (light_reading<600) {  
digitalWrite(led1, HIGH); //anavw to led  
digitalWrite(led2, HIGH); // anavw to led  
digitalWrite(led3, HIGH); //anavw to led  
}  
else  
{  
digitalWrite(led1,LOW); //svinw to led  
digitalWrite(led2, LOW) ; //svinw to led  
digitalWrite(led3, LOW); //svinw to led  
}  
Serial.println(light_reading); //serial gia debug  
delay(300); //ena mikro delay gia kalitero output  
}
```

ΚΕΦΑΛΑΙΟ 5. Κατασκευή μονοκατοικίας σε μακέτα και εγκατάσταση όλων των παραπάνω

Τέλος δημιουργήσαμε μια μακέτα που σκοπό έχει την εγκατάσταση του κεντρικού φωτισμού, την έξυπνη κλειδαριά για το άνοιγμα της κεντρικής εισόδου και τον αισθητήρα φωτός για την ενεργοποίηση του εξωτερικού φωτισμού.

Η μακέτα θα είναι προσέγγιση μιας μονοκατοικίας, που στο εσωτερικό της θα αποτελείται από 3 δωμάτια όπου σε κάθε δωμάτιο θα εγκαταστήσουμε από ένα LED και στο μεγαλύτερο δωμάτιο θα τοποθετηθεί και ο ανεμιστήρας. Ο έλεγχος σε όλα αυτά θα γίνεται από τον ηλεκτρονικό μας υπολογιστή επίσης ο έλεγχος μπορεί να γίνει και με έξυπνο κινητό. Στο μπροστινό μέρος της μακέτας έχουμε τοποθετήσει τον σερβοκινητήρα για το άνοιγμα της πόρτας και ένα LED στην είσοδο, στο πίσω μέρος έχουμε τοποθετήσει 2 LED και η ενεργοποίησή τους θα γίνεται με τον αισθητήρα φωτός.

5.1 Υλικά και αξεσουάρ που χρησιμοποιήσαμε για την κατασκευή της μακέτας.

Τα υλικά που χρησιμοποιήθηκαν για την μακέτα είναι τα παρακάτω :

- Χαρτόνι Μακέτας 70X100εκ./ 5mm (3 τεμάχια)
- Χαρτόνι Μακέτας 50X70εκ./3mm (1 τεμάχιο)

Τα αξεσουάρ που χρησιμοποιήθηκαν για την μακέτα είναι τα παρακάτω :

- Τοίχος Πέτρινος 24x15.5cm (2 τεμάχια)
- Γρασίδι Άνοιξης (1 τεμάχιο)
- Θάμνος, Πράσινο Σκούρο & Ανοιχτό χρώμα. (1 τεμάχιο)
- Όλα τα παραπάνω κολλήθηκαν με πιστόλι θυροκόλλησης.

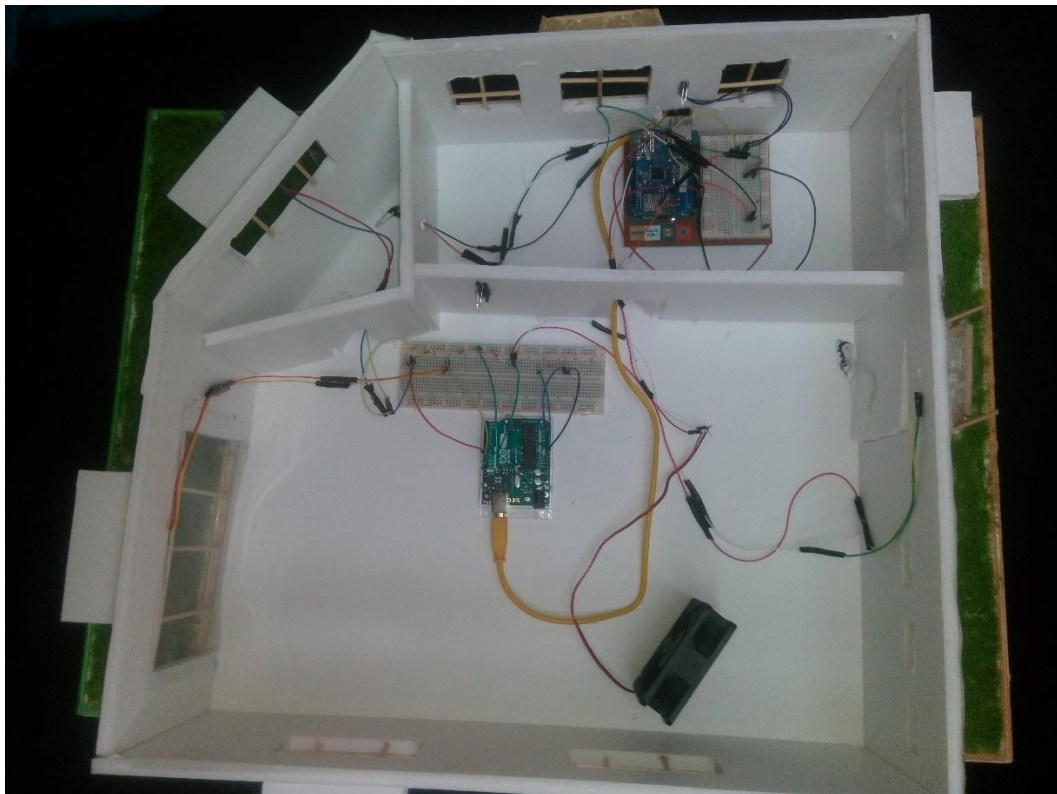
5.2 Εξοπλισμός για την υλοποίηση της κατασκευής

Για την κατασκευή χρειαστήκαμε :

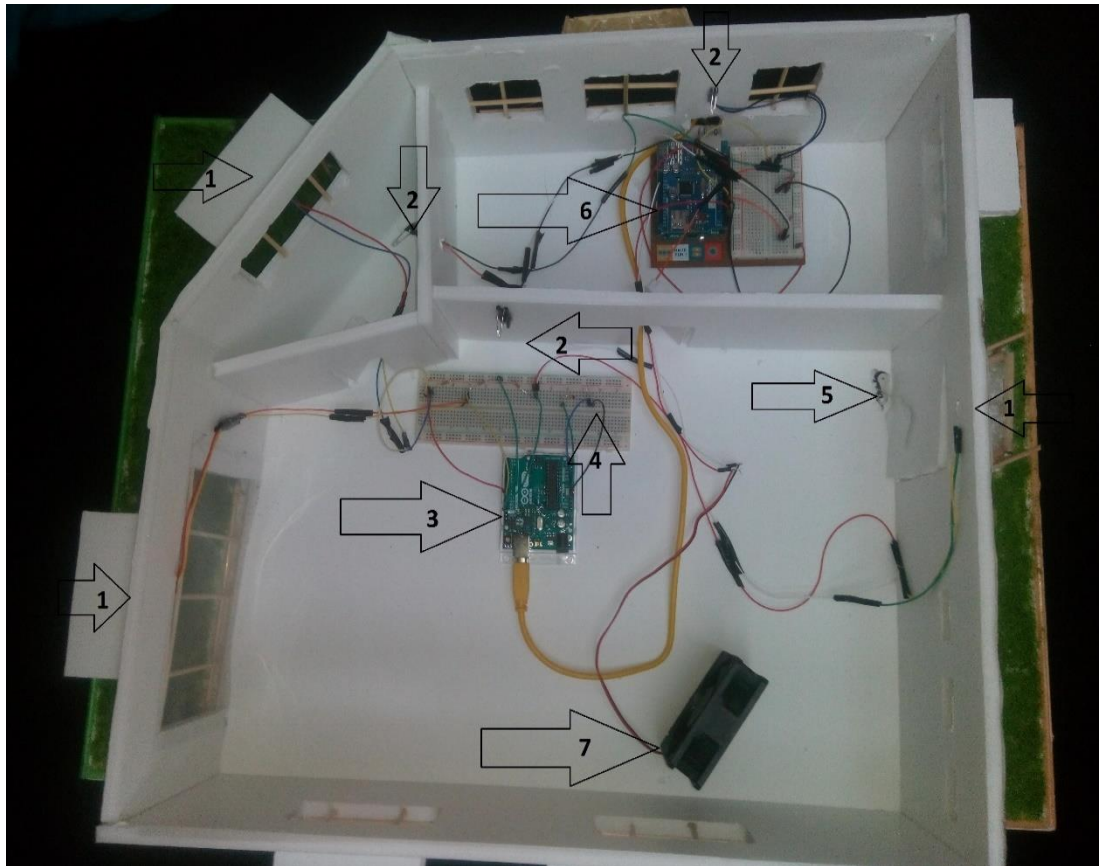
- 1) 3 μικροελεγκτές Arduino UNO.
- 2) Ethernet Shield για την σύνδεση στο δίκτυο.
- 3) Καλώδια και Breadboard πλακέτα.
- 4) Διάφορα LED.
- 5) Ανεμιστήρα 5Volt και σερβοκινητήρα.



Εικόνα 56: Αρχικό στάδιο κατασκευής



Εικόνα 57: Κάτοψη μακέτας και εγκατάσταση Arduino



Εικόνα 58: Περιγραφή εγκατάστασης Arduino στην μακέτα.

Όπως παρατηρούμε στην παραπάνω εικόνα έχουμε δημιουργήσει κάποια βελάκια ώστε να εξηγήσουμε τι είναι το καθετί από αυτά.

Στα βελάκια με τον αριθμό 1 έχουμε εγκαταστήσει τα LED που χρησιμοποιεί ο αισθητήρας φωτισμού τον οποίο και παρατηρούμε στο βελάκι με τον αριθμό 4, επίσης στα βελάκια με τον αριθμό 2 έχουμε εγκαταστήσει LED και ο έλεγχος τους θα γίνεται μέσω του υπολογιστή μας, στο βελάκι με τον αριθμό 3 παρατηρούμε τον μικροελεγκτή Arduino για τον έλεγχο του αισθητήρα φως, στο βελάκι με τον αριθμό 5 έχουμε εγκαταστήσει τον σερβοκινητήρα ώστε να ανοίγει η κεντρική μας πόρτα. Στο βελάκι με τον αριθμό 6 παρατηρούμε τον μικροελεγκτή Arduino και ακριβώς από πάνω του την Ethernet Shield.

Τέλος στο βελάκι με τον αριθμό 7 έχουμε εγκαταστήσει τον ανεμιστήρα μας για να προσομοιάσουμε το Air condition το οποίο έχουμε τοποθετήσει στο κεντρικό πάνελ διαχείρισης.

ΚΕΦΑΛΑΙΟ 6. Συμπεράσματα - Προτάσεις για μελλοντική εργασία

Σκοπός της παρούσας πτυχιακής εργασίας ήταν η κατασκευή ενός έξυπνου σπιτιού με την βοήθεια του μικροελεγκτή Arduino.

Ο στόχος μας ήταν να προγραμματίσουμε και να θέσουμε σε λειτουργία διάφορους αισθητήρες αυτοματοποιημένα χωρίς την παρέμβαση από εμάς, να ελέγχουμε τον κεντρικό φωτισμό και διάφορες συσκευές με τον υπολογιστή μας, επίσης ο έλεγχος κάλλιστα μπορεί να γίνει από το έξυπνο κινητό μας ή το τάμπλετ μας. Όλη η κατασκευή κόστισε περίπου 500 με 600 ευρώ.

Η κατασκευή που δημιουργήσαμε κάλλιστα μπορεί να εγκατασταθεί σε μια συμβατική κατοικία έχοντας περισσότερους μικροελεγκτές ώστε να καλύψει ολόκληρο το σπίτι, ιδιαίτερα τέτοιες κατασκευές προτείνονται σε άτομα με ειδικές ανάγκες, ώστε να τα βοηθήσουμε να έχουν τον πλήρη έλεγχο του σπιτιού τους στο κινητό τους. Μπορούν επίσης να ελέγχουν διάφορες ράμπες πατώντας πάνω τους, πόρτες ώστε να ανοίγουν και να κλείνουν από το έξυπνο κινητό τους και πολλές ακόμα συσκευές και μηχανήματα. Το κόστος μιας τέτοιας εγκατάστασης σε μια υπάρχουσα κατοικία ξεκινάει από το ποσό των 10.000 ευρώ

Τέλος, η παρούσα πτυχιακή μπορεί να επεκταθεί με την δημιουργία Android εφαρμογής για το έξυπνο κινητό μας ή το τάμπλετ μας, η εφαρμογή θα λειτουργεί και από 3G-4G δίκτυο. Η εφαρμογή που θα δημιουργηθεί θα μπορεί να χρησιμοποιείται παραπάνω από έναν χρήστη και η είσοδος στην εφαρμογή θα μπορεί να γίνεται με προσωπικό «όνομα χρήστη» και προσωπικό «κωδικό».

ΒΙΒΛΙΟΓΡΑΦΙΑ

1. Πως διαβάζονται οι αντιστάσεις :
Λουτρίδης Ι. Σπυρίδων, Ηλεκτροτεχνία για μηχανολόγους, Εκδόσεις ΙΩΝ, Λάρισα, 2011.
2. Ανάπτυξη κώδικα HTML :
Λεόντιος Μάνος, Γαβάνα Αναστασία, Μανωλάκος Νίκος, ECDL Webstarter, Β. ΓΚΙΟΥΡΔΑΣ ΕΚΔΟΤΙΚΗ , Αθήνα 2006.
3. Συνδέσεις Arduino με εξαρτήματα και ανάπτυξη προγραμμάτων :
Scott Fitzgerald, Michael Shiloh, ARDUINO PROJECTS BOOK, Officine Arduino Torino, Italy 2012
4. Εγκατάσταση περιβάλλοντος Arduino IDE και εγκατάσταση βιβλιοθηκών :
Επίσημη ιστοσελίδα Arduino, <http://www.arduino.cc>, (Ανάκτηση την 20/3/2016).
5. Ιστοσελίδα Wiki :
<https://en.wikipedia.org/wiki/Wiki> , (Ανάκτηση την 5/4/2016).
6. Πρόγραμμα Arduino :
<http://www.arduino.cc/en/Main/Software>, (Ανάκτηση την 20/3/2016).
7. Βιβλιοθήκες Arduino :
<http://www.arduino.cc/en/Reference/Libraries>, (Ανάκτηση την 20/3/16).
8. Επίσημη ιστοσελίδα για βελτίωση ασύρματου δικτύου :
TP-LINK, <http://www.tp-link.gr/> , (Ανάκτηση την 5/5/16).

EΙΚΟΝΕΣ

- [1-2]: Ιστοσελίδα πωλήσεων Arduino, <http://www.internetnow.gr/>, (Ανάκτηση την 20/3/2016).
- [3]: Ιστοσελίδα Atmel, <http://www.atmel.com>, (Ανάκτηση την 20/3/2016).
- [4]: Ελληνικό forum για ηλεκτρονικά κυκλώματα, <http://www.hlektronika.gr>, (Ανάκτηση την 20/3/2016).
- [5-6]: Επίσημη ιστοσελίδα Arduino, <http://www.arduino.cc>, (Ανάκτηση την 25/3/2016).
- [7]: Ιστοσελίδα αυτοματισμών, <http://www.instructables.com>, (Ανάκτηση την 20/3/2016).
- [8-9]: Ιστοσελίδα πωλήσεων ηλεκτρονικών, <http://www.grobotronics.com>, (Ανάκτηση την 20/5/2016).
- [10]: Ελληνικό Forum Android, <http://www.greeceandroid.gr/>, (Ανάκτηση την 2/6/16).
- [11]: Ιστοσελίδα πωλήσεων ηλεκτρονικών, <http://www.grobotronics.com>, (Ανάκτηση την 20/5/2016).
- [12], Ελληνικό forum για ηλεκτρονικά κυκλώματα, <http://www.hlektronika.gr>, (Ανάκτηση την 15/4/2016).
- [13], Ιστοσελίδα υπολογιστών, <http://www.tweaking4all.com/>, (Ανάκτηση την 20/4/2016).
- [14]: Ιστοσελίδα πωλήσεων ηλεκτρονικών, <http://www.grobotronics.com>, (Ανάκτηση την 20/5/2016).
- [15], Ιστοσελίδα κατασκευών ηλεκτρονικών, <http://www.howtomechatronics.com>, (Ανάκτηση την 17/5/2016).
- [16-17]: Επίσημη ιστοσελίδα για βελτίωση ασύρματου δικτύου, <http://www.tp-link.gr/>, (Ανάκτηση την 5/5/16).
- [18-21]: Επίσημη ιστοσελίδα Arduino, <http://www.arduino.cc>, (Ανάκτηση την 25/3/2016).
- [22-29]: Ιστοσελίδα πωλήσεων ηλεκτρονικών, <http://www.grobotronics.com>, (Ανάκτηση την 20/5/2016).
- [30]: Ιστοσελίδα ηλεκτρονικών κατασκευών, <https://electrosome.com>, (Ανάκτηση την 5/5/16).
- [31-32]: Ιστοσελίδα πωλήσεων ηλεκτρονικών, <http://www.grobotronics.com>, (Ανάκτηση την 20/5/2016).
- [33]: Ελληνικό Forum Android, <http://www.greeceandroid.gr/>, (Ανάκτηση την 2/6/16).
- [34-37]: Ιστοσελίδα πωλήσεων ηλεκτρονικών, <http://www.grobotronics.com>, (Ανάκτηση την 20/5/2016).
- [38-39]: Ελληνικό Forum Android, <http://www.greeceandroid.gr>, (Ανάκτηση την 2/6/16).