



**ΑΝΩΤΑΤΟ ΕΚΠΑΙΔΕΥΤΙΚΟ ΙΔΡΥΜΑ ΠΕΙΡΑΙΑ
ΤΕΧΝΟΛΟΓΙΚΟΥ ΤΟΜΕΑ**

**ΣΧΟΛΗ ΤΕΧΝΟΛΟΓΙΚΩΝ ΕΦΑΡΜΟΓΩΝ
ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΑΥΤΟΜΑΤΙΣΜΟΥ**

ΘΕΜΑ ΠΤΥΧΙΑΚΗΣ ΕΡΓΑΣΙΑΣ

"ΣΥΣΤΗΜΑ ΟΙΚΙΑΚΟΥ ΑΥΤΟΜΑΤΙΣΜΟΥ ΓΙΑ ΕΛΕΓΧΟ ΑΠΟ ΑΠΟΣΤΑΣΗ ΜΕ ARDUINO"

"DISTANCE AUTOMATED CONTROL OF HOUSE AUTOMATIONS VIA ARDUINO PROCESSOR"



ΟΝΟΜΑΤΑ ΦΟΙΤΗΤΩΝ:

ΑΝΤΩΝΟΠΟΥΛΟΣ ΙΩΑΝΝΗΣ

ΝΟΥΛΑΣ ΒΑΣΙΛΕΙΟΣ

ΕΠΙΒΛΕΠΩΝ ΚΑΘΗΓΗΤΗΣ:

ΜΙΧΑΛΗΣ ΠΑΠΟΥΤΣΙΔΑΚΗΣ

ΑΙΓΑΛΕΩ, ΙΑΝΟΥΑΡΙΟΣ 2018

ΔΗΛΩΣΗ ΣΥΓΓΡΑΦΕΑ ΠΤΥΧΙΑΚΗΣ ΕΡΓΑΣΙΑΣ

Ο κάτωθι υπογεγραμμένος ο Ιωάννης Αντωνόπουλος του Παναγιώτη , με αριθμό μητρώου 44545. φοιτητής του Τμήματος **Μηχανικών Αυτοματισμού Τ.Ε.** του Α.Ε.Ι. Πειραιά Τ.Τ. πριν αναλάβω την εκπόνηση της Πτυχιακής Εργασίας μου, δηλώνω ότι ενημερώθηκα για τα παρακάτω:

«Η Πτυχιακή Εργασία (Π.Ε.) αποτελεί προϊόν πνευματικής ιδιοκτησίας τόσο του συγγραφέα, όσο και του Ιδρύματος και θα πρέπει να έχει μοναδικό χαρακτήρα και πρωτότυπο περιεχόμενο.

Απαγορεύεται αυστηρά οποιοδήποτε κομμάτι κειμένου της να εμφανίζεται αυτούσιο ή μεταφρασμένο από κάποια άλλη δημοσιευμένη πηγή. Κάθε τέτοια πράξη αποτελεί προϊόν λογοκλοπής και εγείρει θέμα Ηθικής Τάξης για τα πνευματικά δικαιώματα του άλλου συγγραφέα. Αποκλειστικός υπεύθυνος είναι ο συγγραφέας της Π.Ε., ο οποίος φέρει και την ευθύνη των συνεπειών, ποινικών και άλλων, αυτής της πράξης.

Πέραν των όποιων ποινικών ευθυνών του συγγραφέα σε περίπτωση που το Ίδρυμα του έχει απονείμει Πτυχίο, αυτό ανακαλείται με απόφαση της Συνέλευσης του Τμήματος. Η Συνέλευση του Τμήματος με νέα απόφαση της, μετά από αίτηση του ενδιαφερόμενου, του αναθέτει εκ νέου την εκπόνηση της Π.Ε. με άλλο θέμα και διαφορετικό επιβλέποντα καθηγητή. Η εκπόνηση της εν λόγω Π.Ε. πρέπει να ολοκληρωθεί εντός τουλάχιστον ενός ημερολογιακού 6μήνου από την ημερομηνία ανάθεσης της. Κατά τα λοιπά εφαρμόζονται τα προβλεπόμενα στο άρθρο 18, παρ. 5 του ισχύοντος Εσωτερικού Κανονισμού.»

Ο Δηλών

Ημερομηνία

Ιωάννης Αντωνόπουλος

10/01/2018

ΔΗΛΩΣΗ ΣΥΓΓΡΑΦΕΑ ΠΤΥΧΙΑΚΗΣ ΕΡΓΑΣΙΑΣ

Ο κάτωθι υπογεγραμμένος ο ΒΑΣΙΛΕΙΟΣ ΝΟΥΛΑΣ του ΚΩΣΤΑΝΤΙΝΟΥ με αριθμό μητρώου 44551. φοιτητής του Τμήματος **Μηχανικών Αυτοματισμού Τ.Ε.** του Α.Ε.Ι. Πειραιά Τ.Τ. πριν αναλάβω την εκπόνηση της Πτυχιακής Εργασίας μου, δηλώνω ότι ενημερώθηκα για τα παρακάτω:

«Η Πτυχιακή Εργασία (Π.Ε.) αποτελεί προϊόν πνευματικής ιδιοκτησίας τόσο του συγγραφέα, όσο και του Ιδρύματος και θα πρέπει να έχει μοναδικό χαρακτήρα και πρωτότυπο περιεχόμενο.

Απαγορεύεται αυστηρά οποιοδήποτε κομμάτι κειμένου της να εμφανίζεται αυτούσιο ή μεταφρασμένο από κάποια άλλη δημοσιευμένη πηγή. Κάθε τέτοια πράξη αποτελεί προϊόν λογοκλοπής και εγείρει θέμα Ηθικής Τάξης για τα πνευματικά δικαιώματα του άλλου συγγραφέα. Αποκλειστικός υπεύθυνος είναι ο συγγραφέας της Π.Ε., ο οποίος φέρει και την ευθύνη των συνεπειών, ποινικών και άλλων, αυτής της πράξης.

Πέραν των όποιων ποινικών ευθυνών του συγγραφέα σε περίπτωση που το Ίδρυμα του έχει απονείμει Πτυχίο, αυτό ανακαλείται με απόφαση της Συνέλευσης του Τμήματος. Η Συνέλευση του Τμήματος με νέα απόφαση της, μετά από αίτηση του ενδιαφερόμενου, του αναθέτει εκ νέου την εκπόνηση της Π.Ε. με άλλο θέμα και διαφορετικό επιβλέποντα καθηγητή. Η εκπόνηση της εν λόγω Π.Ε. πρέπει να ολοκληρωθεί εντός τουλάχιστον ενός ημερολογιακού 6μήνου από την ημερομηνία ανάθεσης της. Κατά τα λοιπά εφαρμόζονται τα προβλεπόμενα στο άρθρο 18, παρ. 5 του ισχύοντος Εσωτερικού Κανονισμού.»

Ο Δηλών

ΒΑΣΙΛΕΙΟΣ ΝΟΥΛΑΣ

Ημερομηνία

10/01/2018

ΠΙΝΑΚΑΣ ΠΕΡΙΕΧΟΜΕΝΩΝ

ΠΙΝΑΚΑΣ ΕΙΚΟΝΩΝ.....	5
ΕΙΣΑΓΩΓΗ	6
ΚΕΦΑΛΑΙΟ 1: ΤΟ ARDUINO	8
1.1. Παρουσίαση	8
1.2. Υλικό.....	12
1.3. Πλεονεκτήματα χρήσης Arduino	14
ΚΕΦΑΛΑΙΟ 2: ΑΙΣΘΗΤΗΡΕΣ	16
2.1. Γενικά για τους αισθητήρες.....	16
2.2. Ταξινόμηση των σφαλμάτων μέτρησης	20
2.3. Αποκλίσεις αισθητήρων	21
2.4. Ανάλυση.....	23
2.5. Ενδεικτικοί τύποι αισθητήρων	23
ΚΕΦΑΛΑΙΟ 3: ΓΝΩΡΙΜΙΑ ΜΕ ΤΟ APP INVENTOR	25
3.1. Στοιχεία για το App Inventor	26
3.2. Δοκιμή εφαρμογής	30
ΚΕΦΑΛΑΙΟ 4: ΠΩΣ ΛΕΙΤΟΥΡΓΕΙ ΤΟ ARDUINO.....	32
4.1. Μικροελεγκτές	32
4.2. Είσοδοι – Έξοδοι.....	33
4.3. Τροφοδοσία και Led.....	35
ΚΕΦΑΛΑΙΟ 5: ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΣ ΣΕ ARDUINO	39
ΣΥΜΠΕΡΑΣΜΑΤΑ.....	55
ΒΙΒΛΙΟΓΡΑΦΙΑ.....	57
ΠΑΡΑΡΤΗΜΑ: BLOCKS ΣΤΟ APPINVENTOR.....	57

ΠΙΝΑ ΚΑΣ ΕΙΚΟΝΩΝ

<i>Εικόνα 1. Διάφοροι τύποι αισθητήρων</i>	16
<i>Εικόνα 2. Διάφορες τυπολογίες αισθητήρων</i>	18
<i>Εικόνα 1. Περιβάλλον του App Inventor</i>	27
<i>Εικόνα 2. Το περιβάλλον των blocks</i>	29
<i>Εικόνα 3. Είσοδοι – Εξοδοι</i>	33
<i>Εικόνα 4. Τροφοδοσία Arduino</i>	36
<i>Εικόνα 7. Block Σαλονιού</i>	44
<i>Εικόνα 8. Block ενεργοποίησης διακόπτη</i>	45
<i>Εικόνα 9. Block Απενεργοποίησης διακόπτη</i>	45
<i>Εικόνα 10. Block ενεργοποίησης πρίζας</i>	46
<i>Εικόνα 11. Block απενεργοποίησης πρίζας</i>	46
<i>Εικόνα 12. LM35</i>	48
<i>Εικόνα 13. Συνδέσεις</i>	48
<i>Εικόνα 14. Μετρήσεις</i>	49
<i>Εικόνα 15. Εφαρμογή της πλακέτας</i>	50
<i>Εικόνα 16. Εικόνα πλακέτας και συνδέσεων</i>	51
<i>Εικόνα 17. Εξαρτήματα (Αντιστάσεις, Φωτοαντιστάσεις και LED)</i>	51
<i>Εικόνα 18. Εικόνα συνδέσεων</i>	52
<i>Εικόνα 19. Τροφοδοτικά</i>	53

ΕΙΣΑΓΩΓΗ

Η Arduino είναι μια εταιρεία υλικού και λογισμικού ηλεκτρονικών υπολογιστών ανοικτού κώδικα, κοινότητας του έργου και των χρηστών της, που σχεδιάζει και κατασκευάζει κιτ βασισμένα σε μικροελεγκτή για την κατασκευή ψηφιακών συσκευών και διαδραστικών αντικείμενων που μπορούν να αισθανθούν και να ελέγξουν αντικείμενα στο φυσικό κόσμο.

Το project βασίζεται σε σχέδια της πλακέτας του μικροελεγκτή, που κατασκευάζονται από διάφορους προμηθευτές, χρησιμοποιώντας διάφορους μικροελεγκτές. Αυτά τα συστήματα. Οι πίνακες διαθέτουν σειριακές διεπαφές επικοινωνίας, συμπεριλαμβανομένων των USB σε ορισμένα μοντέλα, για τα προγράμματα φόρτωσης από τους προσωπικούς υπολογιστές. Για τον προγραμματισμό των μικροελεγκτών, το project Arduino παρέχει ένα ολοκληρωμένο περιβάλλον ανάπτυξης (IDE) με βάση το πρόγραμμα επεξεργασίας, το οποίο περιλαμβάνει υποστήριξη για τις γλώσσες προγραμματισμού C και C ++.

Το πρώτο Arduino εισήχθη το 2005, με στόχο να προσφέρει ένα φθινό και εύκολο τρόπο για αρχάριους και επαγγελματίες για τη δημιουργία συσκευών που αλληλεπιδρούν με το περιβάλλον τους, χρησιμοποιώντας αισθητήρες και ενεργοποιητές. Κοινά παραδείγματα τέτοιων προϊόντων που προορίζονται για αρχάριους περιλαμβάνουν απλά ρομπότ, θερμοστάτες, και ανιχνευτές κίνησης.

Οι πλακέτες Arduino είναι διαθέσιμες στο εμπόριο σε προσυναρμολογημένη μορφή, ή ως εξαρτήσεις “καν’ το μόνος σου”. Οι προδιαγραφές σχεδιασμού υλικού είναι ανοιχτά διαθέσιμες, επιτρέποντας στις πλακέτες Arduino να κατασκευαστούν από τον καθένα. Οι βιομηχανίες Adafruit εκτιμάται ότι στα μέσα του 2011 παρήγαγαν εμπορικά πάνω από 300.000 επίσημα Arduinos και το 2013 700.000 επίσημες πλακέτες ήταν στα χέρια των χρηστών.

Τα προγράμματα Arduino μπορούν να γραφτούν σε οποιαδήποτε γλώσσα προγραμματισμού, με μεταγλωττιστή που παράγει δυαδικό κώδικα μηχανής. Η Atmel παρέχει ένα περιβάλλον ανάπτυξης για τους μικροελεγκτές της, το AVR Studio και το νεότερο Atmel Studio.

Το project Arduino παρέχει το ολοκληρωμένο περιβάλλον ανάπτυξης Arduino (IDE), το οποίο είναι μια εφαρμογή cross-platform γραμμένη σε Java. Προέρχεται από το IDE για το project επεξεργασίας γλώσσας προγραμματισμού και για το project καλωδίωσης. Έχει

σχεδιαστεί για να εισαγάγει τον προγραμματισμό σε καλλιτέχνες και αρχάριους που δεν είναι εξοικειωμένοι με την ανάπτυξη λογισμικού. Περιλαμβάνει ένα πρόγραμμα επεξεργασίας κώδικα με χαρακτηριστικά όπως τονισμό σύνταξης, ταιριάζασμα αγκυλών, και αυτόματη εσοχή, και παρέχει απλό μηχανισμό με ένα κλικ για την κατάρτιση και τη φόρτωση προγραμμάτων σε μια πλακέτα Arduino. Ένα πρόγραμμα γραμμένο με τον IDE για Arduino ονομάζεται «σκίτσο».

Το IDE Arduino υποστηρίζει τις γλώσσες προγραμματισμού C και C ++ με τη χρήση ειδικών κανόνων οργάνωσης κώδικα. Το IDE Arduino παρέχει μια βιβλιοθήκη λογισμικού που ονομάζεται "καλωδίωση" (Wiring) από το σχέδιο συνδεσμολογίας, το οποίο παρέχει πολλές κοινές διαδικασίες εισόδου και εξόδου. Ένα τυπικό σκίτσο Arduino C/ C ++ αποτελείται από δύο λειτουργίες που καταρτίζονται και συνδέονται με ένα στέλεχος προγράμματος main () σε ένα εκτελέσιμο κυκλικό εκτελεστικό πρόγραμμα:

setup (): μια λειτουργία που εκτελείται μία φορά κατά την έναρξη του προγράμματος και μπορεί να προετοιμάσει τις ρυθμίσεις.

loop (): μια λειτουργία που πραγματοποιείται κατ'επανάληψη μέχρι να σβήσει η πλακέτα.

Μετά την κατάρτιση και τη σύνδεση με την εργαλειοθήκη GNU, που περιλαμβάνεται επίσης στην κατανομή του IDE, το IDE Arduino χρησιμοποιεί το πρόγραμμα avrdude για να μετατρέψει τον εκτελέσιμο κώδικα σε ένα αρχείο κειμένου σε δεκαεξαδική κωδικοποίηση που έχει τοποθετηθεί στην πλακέτα Arduino από ένα πρόγραμμα φορτωτή στο firmware του της πλακέτας.

ΚΕΦΑΛΑΙΟ 1: ΤΟ ARDUINO

Το Arduino είναι μια μικρή συσκευή (μικροεπεξεργαστής) που συνδέεται με USB στον υπολογιστή. Με το Arduino καλλιτέχνες, σχεδιαστές, αρχιτέκτονες, χομπίστες, μαθητές και φοιτητές μαθαίνουν προγραμματισμό και κατασκευάζουν ολόκληρα έργα για αυτοματισμούς και εφαρμογές ρομποτικής. Έτσι ειδικά τα παιδιά που ξεκινούν προγραμματισμό μαθαίνουν να αναβοσβήνουν Led σε γλώσσα Wiring που μοιάζει με c/c++ χάρη στο Arduino. Η πλακέτα του Arduino αποτελείται από έναν μικρό επεξεργαστή ανοικτού κώδικα στον οποίο μπορεί κανείς να προγραμματίσει ακόμη κι αν είναι αρχάριος μέσω του δικού του δωρεάν προγράμματος (IDE: Integrated Development Environment) που τρέχει σε Windows, Linux και MAC OS X. Στο Arduino συνδέονται Led, ροοστάτες, θύρα ethernet για να γίνει webserver και επικοινωνία μέσω bluetooth. Επειδή λογισμικό και υλικό είναι δεμένα, ευέλικτα και δοκιμασμένα, με το Arduino κατασκευάζονται εφαρμογές, οι οποίες αλληλεπιδρούν μεταξύ υπολογιστών-χρηστών-περιβάλλοντος.

1.1. Παρουσίαση

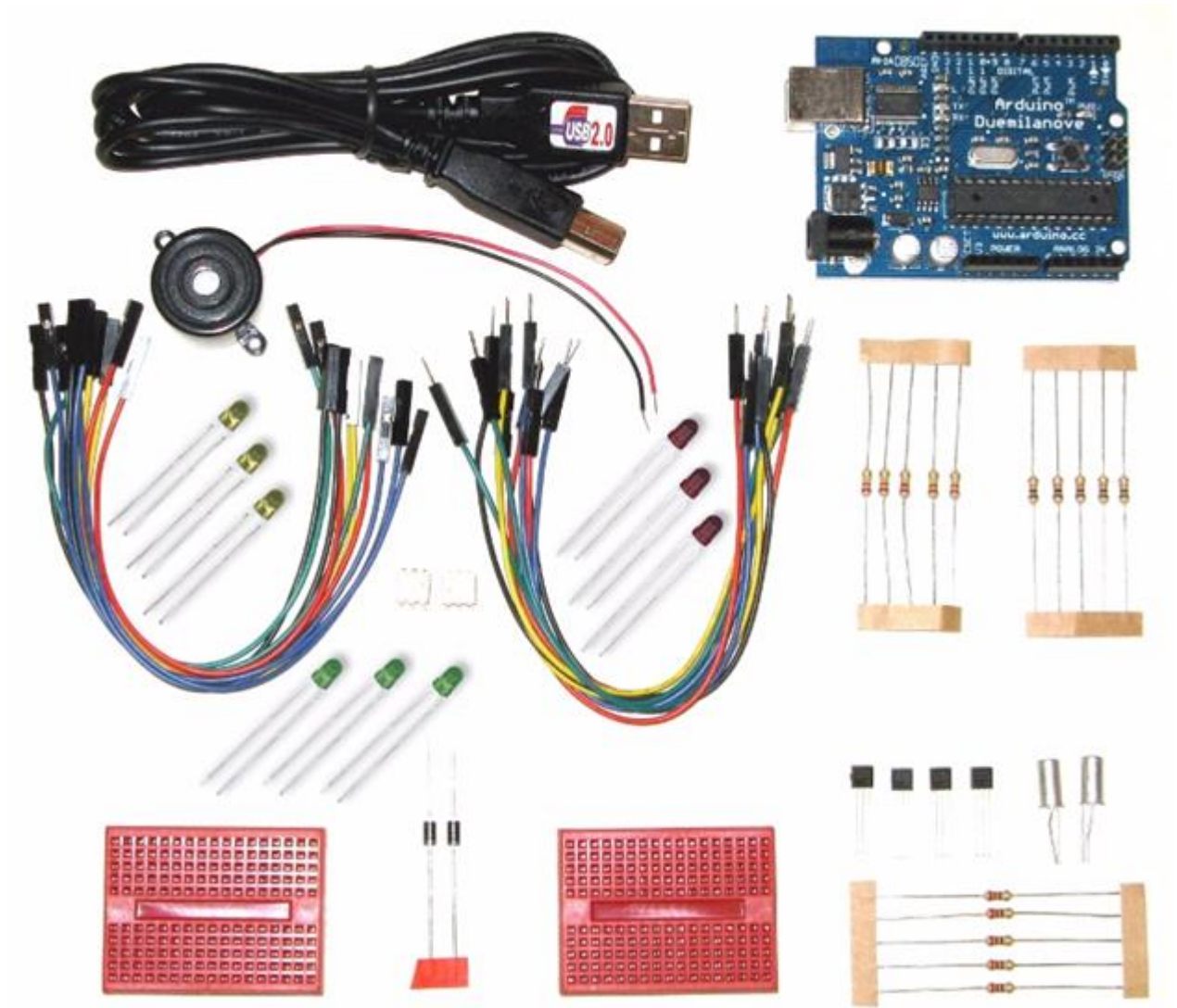
Όπως το περιγράφει ο δημιουργός του, το Arduino είναι μια «ανοικτού κώδικα» πλατφόρμα «πρωτοτυποποίησης» ηλεκτρονικών βασισμένη σε ευέλικτο και εύκολο στη χρήση hardware και software που προορίζεται για οποιονδήποτε έχει λίγη προγραμματιστική εμπειρία, στοιχειώδεις γνώσεις ηλεκτρονικών και ενδιαφέρεται 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.

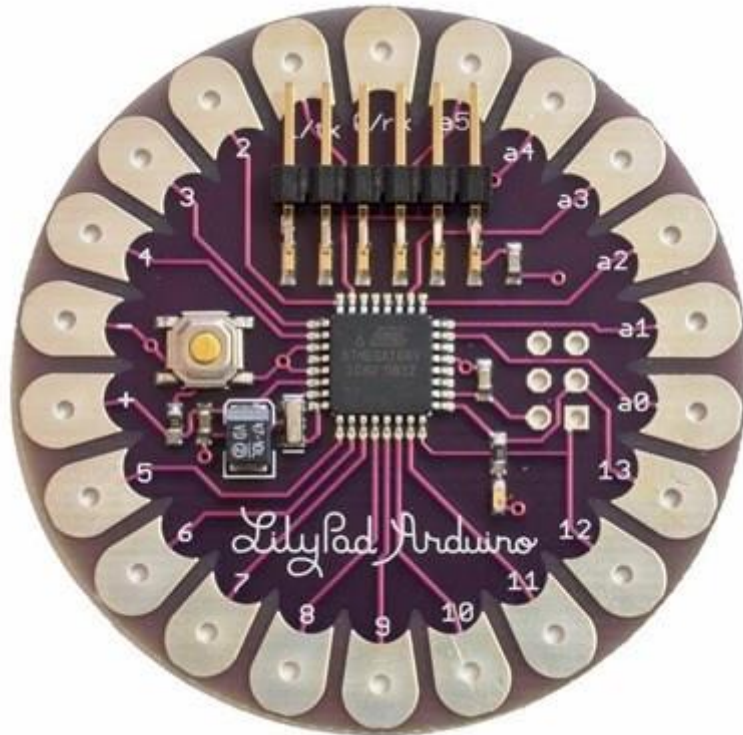
Το Arduino βέβαια, δεν είναι ούτε ο μοναδικός, ούτε και ο καλύτερος δυνατός τρόπος για την δημιουργία μιας οποιασδήποτε διαδραστικής ηλεκτρονικής συσκευής. Όμως το κύριο πλεονέκτημά του είναι η τεράστια κοινότητα που το υποστηρίζει και η οποία

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.



Ακριβώς επειδή απευθύνεται κυρίως σε αρχάριους των ηλεκτρονικών και επειδή, παρά τις αναλυτικότερες οδηγίες που υπάρχουν, δεν έχουν όλοι τις γνώσεις και τα μέσα να κατασκευάσουν μια ηλεκτρονική πλακέτα, κυκλοφορούν έτοιμες, προκατασκευασμένες πλακέτες Arduino που μπορούμε να προμηθευτούμε για περίπου €25. Με λίγα χρήματα παραπάνω μάλιστα, οι περισσότεροι προμηθευτές

διαθέτουν Arduino Starter Kit, τα οποία, εκτός από το ίδιο το Arduino, περιέχουν διάφορα άλλα εξαρτήματα και εργαλεία που μπορεί να χρειαστούμε για τις πρώτες εφαρμογές (όπως το απαραίτητο καλώδιο USB για την σύνδεση με τον υπολογιστή, ράστερ, καλώδια, LED, διακόπτες, ποτενσιόμετρα, αντιστάσεις, διόδους, τρανζίστορ κ.λπ.).



Από τις επίσημες εκδόσεις (Duemilanove, Diecimila, Nano, Mega, WiFi, LilyPad, Mini, Mini USB, Pro, Pro Mini, Serial και Serial SS) συνιστάται κυρίως η αγορά του Arduino Duemilanove ή τουλάχιστον των Diecimila ή Mega επειδή διαθέτουν υποδοχή USB και είναι συμβατές με τα shield. Για τους ίδιους λόγους, από τις ανεπίσημες εκδόσεις (Freeduino, Boarduino, Sanguino, Seeduino, BBB, RBBB κ.α.) συνιστάται μόνο το Freeduino v1.16 και το Seeduino.

1.2. Υλικό

Μια πλακέτα Arduino ιστορικά αποτελείται από ένα μικροελεγκτή AVR Atmel 8-, 16- ή 32-bit (αν και από το 2015 έχουν χρησιμοποιηθεί μικροελεγκτές άλλων κατασκευαστών) με συμπληρωματικά στοιχεία που διευκολύνουν τον προγραμματισμό και την ενσωμάτωση σε άλλα κυκλώματα. Μια σημαντική πτυχή του Arduino είναι οι στάνταρ υποδοχές του, οι οποίες επιτρέπουν στους χρήστες να συνδέουν την πλακέτα της CPU σε μια ποικιλία εναλλάξιμων add-on modules που είναι γνωστά ως ασπίδες. Μερικές ασπίδες επικοινωνούν με την πλακέτα Arduino άμεσα σε διάφορα pins, αλλά σε πολλές ασπίδες απευθύνονται μοναδικά μέσω ενός σειριακού bus I²C, και με αυτό τον τρόπο πολλές ασπίδες μπορούν να στοιβάζονται και να χρησιμοποιούνται παράλληλα. Πριν από το 2015 τα επίσημα Arduinos είχαν χρησιμοποιήσει τη σειρά chip Atmel mega AVR, ειδικά τα ATmega8, ATmega168, ATmega328, ATmega1280, και ATmega2560 και το 2015 προστέθηκαν μονάδες από άλλους κατασκευαστές. Αρκετοί άλλοι επεξεργαστές έχουν επίσης χρησιμοποιηθεί με συμβατές συσκευές Arduino. Οι περισσότερες πλακέτες περιλαμβάνουν γραμμικό ρυθμιστή 5V και ένα ταλαντωτή κρυστάλλου 16 MHz (ή κεραμικό αντηχείο σε ορισμένες παραλλαγές), αν και ορισμένα σχέδια όπως η LilyPad λειτουργούν σε 8 MHz και να άρουν τον onboard ρυθμιστή τάσης που οφείλεται σε συγκεκριμένους περιορισμούς μορφής-παράγοντα. Ο μικροελεγκτής ενός Arduino είναι επίσης προ-προγραμματισμένος με ένα φορτωτή εκκίνησης που απλοποιεί το φόρτωμα των προγραμμάτων στη μνήμη flash on-chip, σε σύγκριση με άλλες συσκευές που χρειάζονται συνήθως έναν εξωτερικό προγραμματιστή. Αυτό καθιστά τη χρήση ενός Arduino απλούστερη, επιτρέποντας τη χρήση ενός συνηθισμένου υπολογιστή ως προγραμματιστή. Επί του παρόντος, το optiboot bootloader είναι το προεπιλεγμένο πρόγραμμα εκκίνησης εγκατεστημένο στον Arduino UNO.

Σε εννοιολογικό επίπεδο, όταν χρησιμοποιούμε το ολοκληρωμένο περιβάλλον ανάπτυξης Arduino, όλες οι κάρτες είναι προγραμματισμένες σε μια σειριακή σύνδεση. Η εφαρμογή του ποικίλλει ανάλογα με την έκδοση υλικού. Μερικές σειριακές πλακέτες Arduino περιλαμβάνουν ένα κύκλωμα επιλογέα επιπέδου, για τη μετατροπή μεταξύ του επιπέδου λογικής RS-232 και των σημάτων επιπέδου TTL. Οι τρέχουσες πλακέτες Arduino έχουν προγραμματιστεί μέσω του Universal Serial Bus (USB), που υλοποιείται με τη χρήση τσιπ προσαρμογέα USB-to-serial, όπως το FTDI

FT232. Ορισμένες πλακέτες, όπως οι πλακέτες UNO των μετέπειτα μοντέλων, αντικατέστησαν το τσιπ FTDI με ένα ξεχωριστό τσιπ AVR περιέχοντας firmware USB-to-serial, το οποίο είναι επαναπρογραμματιζόμενο μέσω του δικού του header της ICSP. Άλλες παραλλαγές, όπως το Arduino Mini και το ανεπίσημο Boarduino, χρησιμοποιούν μια αποσπώμενη κάρτα προσαρμογέα ή καλώδιο USB-to-serial, Bluetooth ή άλλες μεθόδους. Όταν χρησιμοποιούνται με τα παραδοσιακά εργαλεία μικροελεγκτή αντί του Arduino IDE, χρησιμοποιείται πρότυπος προγραμματισμός AVR ISP.

Η πλακέτα Arduino εκθέτει τις περισσότερες από τις ακίδες I/ O του μικροελεγκτή για χρήση από άλλα κυκλώματα. Οι Diecimila, Duemilanove, και η τρέχουσα Uno παρέχουν 14 ψηφιακές ακίδες I/ O, έξι από τις οποίες μπορεί να παράγουν διαμορφωμένα σήματα πλάτους παλμού, και έξι αναλογικές εισόδους, οι οποίες μπορούν επίσης να χρησιμοποιηθούν ως έξι ψηφιακές ακίδες I/ O. Αυτές οι καρφίτσες είναι στην κορυφή της πλακέτας, μέσω θυληκών υποδοχέων 2,5 mm. Αρκετές plug-in ασπίδες εφαρμογής είναι επίσης εμπορικά διαθέσιμα. Οι πλακέτες Arduino Nano, το Boarduino, και η πλακέτα Bare Bones συμβατή με Arduino, μπορούν να παρέχουν ακίδες αρσενικής κεφαλίδας στην κάτω πλευρά της πλακέτας που μπορούν να συνδεθούν σε solderless breadboards.

Υπάρχουν πολλές πλακέτες συμβατές με Arduino και με προέλευση Arduino. Μερικές είναι λειτουργικά ισοδύναμες με μια Arduino και μπορούν να χρησιμοποιηθούν εναλλακτικά. Πολλές ενισχύουν τη βασική Arduino με την προσθήκη οδηγών εξόδου, συχνά για χρήση στην εκπαίδευση σε επίπεδο σχολείου για την απλοποίηση της κατασκευής μικρών ρομπότ. Άλλες είναι ηλεκτρικά ισοδύναμες, αλλά αλλάζουν τον παράγοντα μορφής, μερικές φορές διατηρώντας τη συμβατότητα με τις ασπίδες, και άλλες φορές, όχι. Μερικές παραλλαγές χρησιμοποιούν εντελώς διαφορετικούς επεξεργαστές, με διαφορετικά επίπεδα συμβατότητας.

1.3.Πλεονεκτήματα χρήσης Arduino

Χάρη στην απλή και προσιτή εμπειρία του χρήστη της, η Arduino έχει χρησιμοποιηθεί σε χιλιάδες διαφορετικά προγράμματα και εφαρμογές. Το λογισμικό Arduino είναι εύκολο στη χρήση για αρχάριους, αλλά και αρκετά ευέλικτο για προχωρημένους χρήστες. Τρέχει σε Mac, Windows και Linux. Οι εκπαιδευτικοί και οι μαθητές μπορούν να την χρησιμοποιήσουν για την κατασκευή χαμηλού κόστους επιστημονικών οργάνων, για να αποδείξει τις αρχές της χημείας και της φυσικής, ή για να ξεκινήσουν με τον προγραμματισμό και τη ρομποτική. Σχεδιαστές και αρχιτέκτονες δημιουργούν διαδραστικά πρωτότυπα, μουσικοί και καλλιτέχνες την χρησιμοποιούν για ρυθμίσεις και για να πειραματιστούν με νέα μουσικά όργανα. Η Arduino είναι ένα βασικό εργαλείο για την απόκτηση γνώσεων πάνω σε νέα πράγματα. Ο οποιοσδήποτε μπορεί να χρησιμοποιήσει τις οδηγίες του kit βήμα προς βήμα, ή να μοιραστεί τις του ιδέες του με τα άλλα μέλη της κοινότητας Arduino.

Υπάρχουν πολλοί άλλοι μικροελεγκτές και πλατφόρμες μικροελεγκτή που διατίθενται για τη φυσικό computing. Οι Parallax Basic Stamp, Netmedia BX24, Phidgets, Handyboard του MIT, και πολλοί άλλοι προσφέρουν παρόμοια λειτουργικότητα. Όλα αυτά τα αφαιρούν τις δύσκολες λεπτομέρειες του προγραμματισμού μικροελεγκτών και το επαναπροσδιορίζουν σε ένα πακέτο εύκολο στη χρήση. Η Arduino απλοποιεί επίσης τη διαδικασία της εργασίας με μικροελεγκτές, αλλά προσφέρει επιπλέον πλεονεκτήματα στους εκπαιδευτικούς, τους μαθητές, και τους ενδιαφερόμενους ερασιτέχνες σε σχέση με άλλα συστήματα:

Είναι ανέξοδη – Οι πλακέτες Arduino είναι σχετικά φθηνές σε σύγκριση με άλλες πλατφόρμες μικροελεγκτή. Η λιγότερο ακριβή έκδοση μονάδας Arduino μπορεί να συναρμολογηθεί με το χέρι, και ακόμη και οι προ-συναρμολογημένες μονάδες Arduino κοστίζουν λιγότερο από 50 δολάρια.

Είναι Cross-platform - Το λογισμικό της Arduino (IDE) τρέχει σε λειτουργικά συστήματα Windows, Macintosh OSX και Linux. Τα περισσότερα συστήματα μικροελεγκτή περιορίζονται στα Windows.

Απλό, σαφές περιβάλλον προγραμματισμού - Το λογισμικό της Arduino (IDE) είναι εύκολο στη χρήση για αρχάριους, αλλά και αρκετά ευέλικτο για προχωρημένους

χρήστες ώστε να επωφεληθούν από αυτό και εκείνοι. Για τους εκπαιδευτικούς, είναι ιδιαίτερα σημαντικό το ότι βασίζεται στο περιβάλλον προγραμματισμού επεξεργασίας, έτσι ώστε οι μαθητές που μαθαίνουν να προγραμματίζουν σε αυτό το περιβάλλον θα είναι εξοικειωμένοι με το πώς λειτουργεί το IDE της Arduino.

Λογισμικό ανοικτού κώδικα και επεκτάσιμο - Το λογισμικό της Arduino δημοσιεύεται ως εργαλείο ανοικτού κώδικα, που διατίθεται για επέκταση από έμπειρους προγραμματιστές. Η γλώσσα μπορεί να επεκταθεί μέσω των βιβλιοθηκών της C ++, και όσοι θέλουν να κατανοήσουν τις τεχνικές λεπτομέρειες μπορούν να κάνουν το άλμα από την Arduino στη γλώσσα προγραμματισμού AVR C στην οποία είναι βασισμένη. Ομοίως, μπορεί να προσθέσει τον κώδικα AVR-C απευθείας στο πρόγραμμα Arduino.

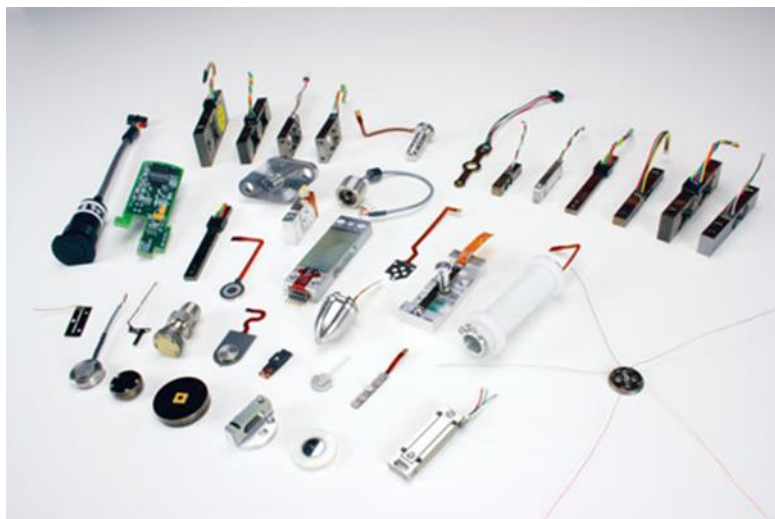
Υλικό ανοικτού κώδικα και επεκτάσιμο - Τα σχέδια των πλακετών Arduino δημοσιεύονται με άδεια Creative Commons, έτσι ώστε οι έμπειροι σχεδιαστές κυκλωμάτων να μπορούν να δημιουργήσουν τη δική τους έκδοση μονάδας, επεκτείνοντάς την και βελτιώνοντας την. Ακόμα και οι σχετικά άπειροι χρήστες μπορούν να δημιουργήσουν πάνω στην έκδοση breadboard της μονάδας, προκειμένου να κατανοήσουν πώς λειτουργεί και να εξοικονομήσουν χρήματα.

ΚΕΦΑΛΑΙΟ 2: ΑΙΣΘΗΤΗΡΕΣ

Στο κεφάλαιο αυτό θα αναλύσουμε την έννοια των αισθητήρων γενικότερα. Τι τύπου αισθητήρες μπορούμε να συναντήσουμε γενικά στο εμπόριο, ποιος είναι ο ρόλος τους σε γενικές γραμμές, ποια είναι τα πλεονεκτήματα της χρήσης τους και ποια τα μειονεκτήματα. Θα δούμε επίσης τους περισσότερο χρησιμοποιούμενους αισθητήρες και θα καταλήξουμε στα είδη των αισθητήρων που χρησιμοποιούνται στις διάφορων τύπων καλλιέργειες.

2.1.Γενικά για τους αισθητήρες

Οι αισθητήρες είναι εξελιγμένες συσκευές που χρησιμοποιούνται συχνά για να ανιχνεύσουν και να ανταποκριθούν σε ηλεκτρικά ή οπτικά σήματα. Ένας αισθητήρας μετατρέπει τη φυσική παράμετρο (για παράδειγμα: την θερμοκρασία, την αρτηριακή πίεση, την υγρασία, την ταχύτητα, κ.λπ.) σε ένα σήμα που μπορεί να μετρηθεί ηλεκτρικά. Ας εξηγήσουμε το παράδειγμα της θερμοκρασίας. Ο υδράργυρος στο θερμόμετρο γυαλιού, διαστέλλει και συστέλλει το υγρό για να μετατρέψει τη μετρημένη θερμοκρασία που μπορεί να διαβαστεί από ένα άτομο στον βαθμονομημένο γυάλινο σωλήνα.



Εικόνα 1. Διάφοροι τύποι αισθητήρων

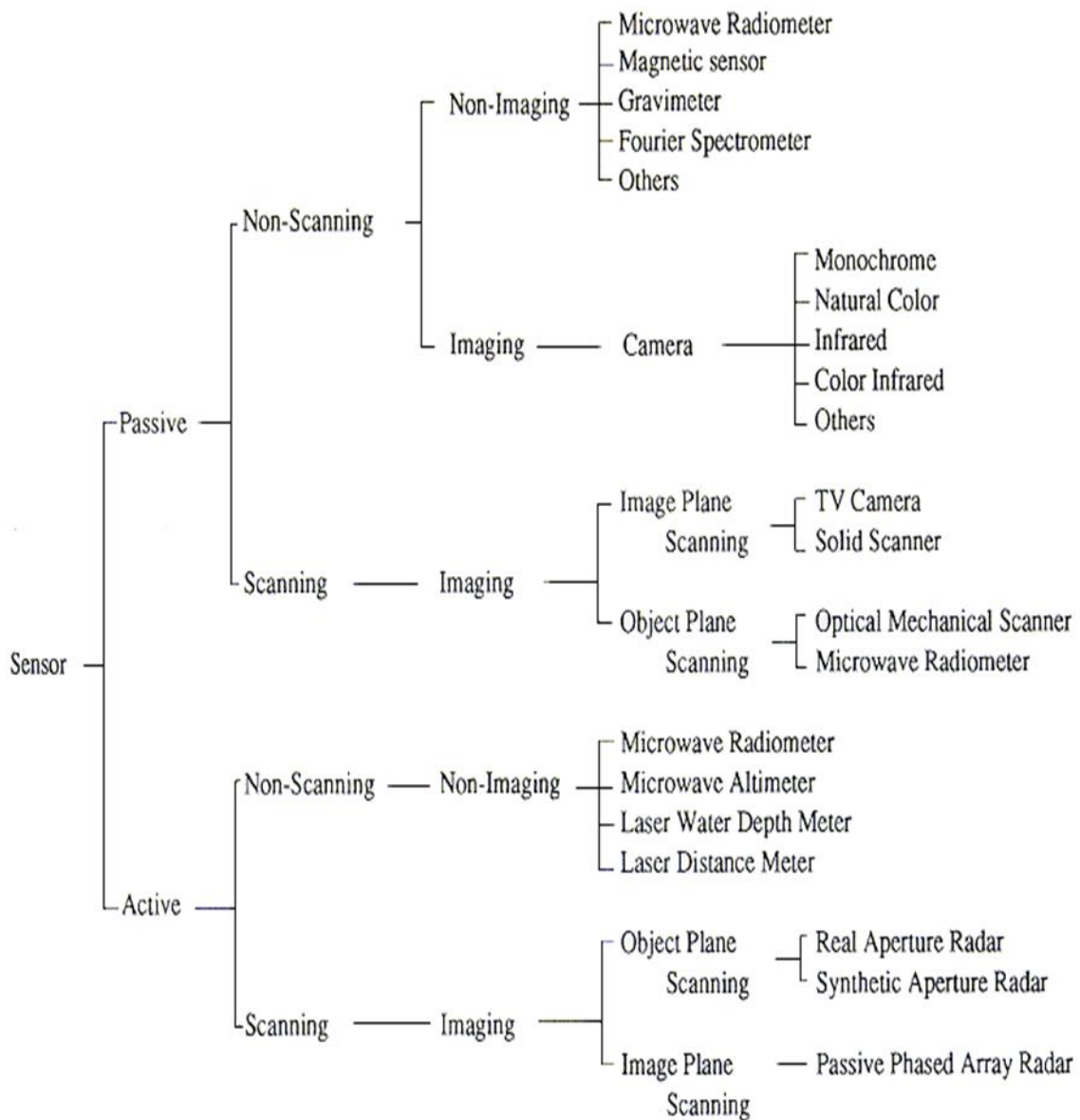
Υπάρχουν ορισμένα χαρακτηριστικά που πρέπει να ληφθούν υπόψη όταν θα επιλέξει κανείς ένα αισθητήρα. Τα χαρακτηριστικά αυτά είναι:

1. Η Ακρίβεια
2. Η περιβαλλοντική κατάσταση – συνήθως έχει όρια θερμοκρασίας/ υγρασίας
3. Το φάσμα όριο μέτρησης του αισθητήρα
4. Η βαθμονόμηση – Ουσιαστική για τις περισσότερες από τις συσκευές μέτρησης, καθώς ο τρόπος με τον οποίο διαβάζεται κάτι αλλάζει με τον καιρό
5. Ανάλυση – Μικρότερη προσαύξηση ανιχνεύεται από τον αισθητήρα
6. Κόστος
7. Επαναληψιμότητα – Η ανάγνωση, η οποία ποικίλλει, μετράται επανειλημμένως στο ίδιο περιβάλλον.

Οι αισθητήρες κατατάσσονται στα ακόλουθα κριτήρια:

1. Κύρια ποσότητα εισόδου (μετρητέα)
2. Αρχές μεταγωγής (Χρησιμοποιώντας φυσικές και χημικές επιδράσεις)
3. Υλικό και Τεχνολογία
4. Ιδιότητα
5. Εφαρμογή

Η αρχή μεταγωγής είναι τα βασικά κριτήρια που ακολουθούνται για την αποτελεσματική προσέγγιση. Συνήθως, τα κριτήρια υλικών και τεχνολογίας έχουν επιλεγεί από την ομάδα ανάπτυξης των μηχανικών.



Εικόνα 2. Διάφορες τυπολογίες αισθητήρων

Η κατάταξη με βάση την ιδιότητα γίνεται ως κάτωθι:

1. Θερμοκρασία – θερμίστορες, θερμοστοιχεία και πολλά άλλα.
2. Πίεση – οπτικές ίνες, κενό, μανόμετρα με βάση ελαστικό υγρό, κ.λπ.
3. Ροή – Ηλεκτρομαγνητική, διαφορικής πίεσης, μετατόπισης θέσης, θερμικής μάζας, κ.λπ.

4. Αισθητήρες Επίπεδου – διαφορικής πίεσης, υπερήχων ραδιοσυχνοτήτων, ραντάρ, θερμικής μετατόπισης, κ.λπ.
5. Εγγύτητα και μετατόπιση – φωτοηλεκτρική, χωρητική, μαγνητική, με υπερήχους.
6. Βιοαισθητήρες – αντηχητικό κάτοπτρο, ηλεκτροχημικός, επιφάνεια συντονισμού Plasma, ποτενσιομετρικός προσπελάσιμος από φως.
7. Εικόνας
8. Φυσικού αερίου και χημικών – ημιαγωγός, υπερύθρων, αγωγιμότητας Ηλεκτροχημικός.
9. Επιτάχυνση – Γυροσκόπιο, επιταχυνσιόμετρο.
10. Άλλα – Υγρασία, αισθητήρας υγρασίας, αισθητήρας ταχύτητας, μάζας, αισθητήρας κλίσης, δύναμης, το ιξώδους.

Η κατάταξη με βάση την εφαρμογή είναι όπως δίνεται παρακάτω:

- Ελέγχου βιομηχανικής διαδικασίας, μέτρησης και αυτοματισμού
- Μη-βιομηχανικής χρήσης σε αεροσκάφη, ιατρικά προϊόντα, αυτοκίνητα, ηλεκτρονικά είδη ευρείας κατανάλωσης, και άλλους τύπους αισθητήρων.

Οι αισθητήρες μπορούν να ταξινομηθούν ανάλογα με την ισχύ ή την απαίτηση παροχής ενέργειας των αισθητήρων:

- Ενεργός αισθητήρας: Οι αισθητήρες που απαιτούν τροφοδοσία ρεύματος, ονομάζονται ενεργοί αισθητήρες.
- Παθητικός αισθητήρας: Οι αισθητήρες που δεν απαιτούν παροχή ηλεκτρικού ρεύματος, ονομάζονται παθητικοί αισθητήρες.

Υπό τις σημερινές και μελλοντικές εφαρμογές, οι αισθητήρες μπορούν να ταξινομηθούν σε ομάδες ως εξής:

- Επιταχυνσιόμετρα - Αυτά βασίζονται στην τεχνολογία μικροηλεκτρομηχανικού αισθητήρα.
- Βιοαισθητήρες - Αυτοί βασίζονται στην ηλεκτροχημική τεχνολογία.
- Αισθητήρες Εικόνας - Αυτοί βασίζονται στην τεχνολογία CMOS. Χρησιμοποιούνται σε ηλεκτρονικά είδη ευρείας κατανάλωσης, τη χρήση βιομετρικών στοιχείων, την κυκλοφορία και την εποπτεία της ασφάλειας και της απεικόνισης του υπολογιστή.
- Ανιχνευτές Κίνησης - Αυτοί βασίζονται στην υπέρυθη ακτινοβολία, στους υπερήχους, και την τεχνολογία μικροκυμάτων/ ραντάρ. Χρησιμοποιούνται σε βιντεοπαιχνίδια και προσομοιώσεις, στην ενεργοποίηση φωτός και στην ανίχνευση ασφαλείας (Vetelino & Reghu, 2010, Webster, 1999, Sinclair, 2000).

2.2. Ταξινόμηση των σφαλμάτων μέτρησης

Ένας καλός αισθητήρας υπακούει στους ακόλουθους κανόνες:

- Είναι ευαίσθητος στην μετρούμενη ιδιότητα και μόνο
- Δεν είναι ευαίσθητος σε κάθε άλλη ιδιότητα που ενδέχεται να συναντήσει κατά την εφαρμογή του
- Δεν επηρεάζει τη μετρούμενη ιδιότητα

Οι ιδανικοί αισθητήρες έχουν σχεδιαστεί για να είναι γραμμικοί ή γραμμικοί ως προς κάποια απλή μαθηματική συνάρτηση μέτρησης, τυπικά λογαριθμική. Η έξοδος ενός τέτοιου αισθητήρα είναι ένα αναλογικό σήμα και γραμμικά ανάλογο με την αξία ή την απλή συνάρτηση της μετρούμενης ιδιοκτησίας (Καλοβρέκτης & Κατέβας, 2012: 78). Η ευαισθησία στη συνέχεια ορίζεται ως ο λόγος μεταξύ του σήματος εξόδου και της μετρούμενης ιδιότητας. Για παράδειγμα, εάν ένας αισθητήρας μετρά την

θερμοκρασία και έχει έξοδο τάσης, η ευαισθησία είναι μία σταθερά με μονάδα V/K. Ο αισθητήρας αυτός είναι γραμμικός, επειδή η αναλογία είναι σταθερή σε όλα τα σημεία μέτρησης.

Για να τεθεί σε επεξεργασία ή χρήση σε ψηφιακό εξοπλισμό, ένας αισθητήρας αναλογικού σήματος, αυτό θα πρέπει να μετατραπεί σε ένα ψηφιακό σήμα, με τη χρήση ενός μετατροπέα αναλογικού προς ψηφιακό.

2.3.Αποκλίσεις αισθητήρων

Εάν ο αισθητήρας δεν είναι ιδανικός, μπορούν να παρατηρηθούν διάφοροι τύποι αποκλίσεων όπως:

- Η ευαισθησία μπορεί να διαφέρει στην πράξη από την καθορισμένη τιμή. Αυτό ονομάζεται σφάλμα ευαισθησίας, αλλά ο αισθητήρας είναι ακόμη γραμμικός.
- Δεδομένου ότι το φάσμα του σήματος εξόδου είναι πάντα περιορισμένο, το σήμα εξόδου θα φτάσει τελικά στο ελάχιστο ή στο μέγιστο όταν η μετρούμενη ιδιότητα υπερβαίνει τα όρια. Το εύρος πλήρους κλίμακας καθορίζει τις μέγιστες και ελάχιστες τιμές της μετρούμενης ιδιότητας.
- Εάν το σήμα εξόδου δεν είναι μηδέν, όταν η μετρούμενη ιδιότητα είναι μηδέν, ο αισθητήρας έχει ένα αντιστάθμισμα. Αυτό ορίζεται ως η έξοδος του αισθητήρα σε μηδενική είσοδο.
- Εάν η ευαισθησία δεν είναι σταθερή σε όλο το εύρος του αισθητήρα, αυτό ονομάζεται μη γραμμικότητα. Συνήθως αυτό ορίζεται από το ποσό η έξοδος διαφέρει από την ιδανική συμπεριφορά σε όλο το εύρος του αισθητήρα, που συχνά σημειώνεται ως ποσοστό του πλήρους φάσματος.
- Εάν η απόκλιση προκαλείται από μια ταχεία μεταβολή της μετρούμενης ιδιότητας με την πάροδο του χρόνου, υπάρχει ένα δυναμικό σφάλμα. Συχνά,

αυτή η συμπεριφορά περιγράφεται με ένα προσχέδιο που παρουσιάζει το σφάλμα της ευαισθησίας και την μετατόπιση φάσης ως συνάρτηση της συχνότητας ενός περιοδικού σήματος εισόδου.

- Εάν το σήμα εξόδου αλλάζει αργά ανεξάρτητα από την μετρούμενη ιδιότητα, αυτό ορίζεται ως εκτροπή.
- Η μακροπρόθεσμη μεταβολή συνήθως δείχνει μια αργή υποβάθμιση των ιδιοτήτων του αισθητήρα πάνω σε ένα μεγάλο χρονικό διάστημα.
- Ο θόρυβος είναι μια τυχαία απόκλιση του σήματος που μεταβάλλεται στο χρόνο.
- Η υστέρηση είναι ένα σφάλμα που προκαλείται όταν η μετρούμενη ιδιότητα αντιστρέφει την κατεύθυνση, αλλά υπάρχει κάποια πεπερασμένη υστέρηση στο χρόνο για τον αισθητήρα ώστε να ανταποκριθεί, δημιουργώντας μια διαφορετική αντιστάθμιση σφάλματος σε μία συγκεκριμένη κατεύθυνση από ό,τι στην άλλη.
- Αν ο αισθητήρας έχει ψηφιακή έξοδο, η έξοδος είναι ουσιαστικά μια προσέγγιση της μετρούμενης ιδιότητας. Το σφάλμα προσέγγισης επίσης ονομάζεται σφάλμα ψηφιοποίησης.
- Εάν το σήμα παρακολουθείται ψηφιακά, ο περιορισμός της συχνότητας δειγματοληψίας μπορεί επίσης να προκαλέσει δυναμικό λάθος, ή εάν η μεταβλητή ή ο προστιθέμενος θόρυβος αλλάξει περιοδικά σε μία συχνότητα κοντά σε ένα πολλαπλάσιο του ρυθμού δειγματοληψίας μπορεί να προκαλέσει σφάλματα παραποίησης.
- Ο αισθητήρας μπορεί σε κάποιο βαθμό να είναι ευαίσθητος σε ιδιότητες, εκτός από την ιδιότητα που μετράται. Για παράδειγμα, οι περισσότεροι αισθητήρες επηρεάζονται από τη θερμοκρασία του περιβάλλοντος τους.

Όλες αυτές οι αποκλίσεις μπορούν να ταξινομηθούν ως συστηματικά ή τυχαία σφάλματα. Τα συστηματικά σφάλματα μπορούν μερικές φορές να αντισταθμιστούν με τη βοήθεια κάποιου είδους στρατηγικής βαθμονόμησης. Ο θόρυβος είναι ένα τυχαίο σφάλμα που μπορεί να μειωθεί με επεξεργασία του σήματος, όπως το

φιλτράρισμα, συνήθως σε βάρος της δυναμικής συμπεριφοράς του αισθητήρα (Yamasaki, 1996, Λουτρίδης, 2008).

2.4.Ανάλυση

Η ανάλυση του αισθητήρα είναι η μικρότερη αλλαγή που μπορεί να ανιχνεύσει στην ποσότητα που μετρά. Συχνά σε μια ψηφιακή οθόνη, το λιγότερο σημαντικό ψηφίο θα παρουσιάσει διακυμάνσεις, που δείχνουν ότι οι αλλαγές αυτού του μεγέθους έχουν μόλις αναλυθεί. Η ανάλυση σχετίζεται με την ακρίβεια με την οποία γίνεται η μέτρηση. Για παράδειγμα, ένας ανιχνευτής σάρωσης σήραγγας (μια λεπτή άκρη κοντά σε μια επιφάνεια συλλέγει ένα ρεύμα ηλεκτρονίων σηράγγων) μπορεί να αναλύσει τα άτομα και τα μόρια (Λουτρίδης, 2008: 92).

2.5.Ενδεικτικοί τύποι αισθητήρων

Στην φάση αυτή μπορούμε να αναφέρουμε τις βασικές κατηγορίες αισθητήρων και στην συνέχεια μπορούμε να αναφερθούμε σε ορισμένους ενδεικτικούς τύπους.

Οι βασικές κατηγορίες είναι:

- Επιτάχυνσης/ Δόνησης
- Ακουστικοί/ Υπερήχων
- Χημικοί/ Αερίου
- Ηλεκτρικοί/ Μαγνητικοί
- Ροής
- Δύναμης/ Φορτίου/ Ροπής/ Καταπόνησης

- Υγρασίας
- Διαρροής/ Επίπεδου
- Τεχνητής Όρασης
- Οπτικοί
- Κίνησης/ Ταχύτητας/ Κυβισμού
- Θέσης/ Παρουσίας/ Εγγύτητας
- Πίεσης
- Θερμοκρασίας

ΚΕΦΑΛΑΙΟ 3: ΓΝΩΡΙΜΙΑ ΜΕ ΤΟ APP INVENTOR

Το App Inventor για Android είναι ένα περιβάλλον οπτικού προγραμματισμού που παρεχόταν αρχικά από την Google, και τώρα διατίθεται και υποστηρίζεται από το MIT. Το Google App Inventor διατέθηκε στο κοινό για πρώτη φορά το τέλος του 2010 και μετά από ένα χρόνο στις 31 Δεκεμβρίου, 2011 η Google τερμάτισε την υποστήριξη του προγράμματος και το έκανε open source. Το App Inventor ξαναδιατέθηκε στο κοινό, πλέον με όνομα MIT App Inventor ως beta service (τον Μάρτιο του 2012), από το MIT Center for Mobile Learning μετά την ανάληψη του έργου από την Google.

Το App Inventor είναι ένα on-line περιβάλλον προγραμματισμού. Πιο συγκεκριμένα για την ανάπτυξη εφαρμογών με το App Inventor είναι απαραίτητη η χρήση ενός φυλλομετρητή (web browser) και ενός συνδεδεμένου τηλεφώνου με Android ή ενός εξομοιωτή τηλεφώνου.

Το περιβάλλον του App Inventor επιτρέπει σε οποιονδήποτε είναι εξοικειωμένος με τον προγραμματισμό ηλεκτρονικών υπολογιστών να δημιουργεί εφαρμογές για λειτουργικό σύστημα Android. Χρησιμοποιεί ένα γραφικό περιβάλλον αλληλεπίδρασης με τον χρήστη, παρόμοιο με του Scratch και της StarLogo TNG, το οποίο επιτρέπει στους χρήστες να σέρνουν και να αφήνουν πλακίδια δημιουργώντας έτσι εφαρμογές που μπορούν να τρέξουν σε Android, το λειτουργικό σύστημα σήμερα πολλών φορητών συσκευών.

Ειδικότερα το γραφικό περιβάλλον προγραμματισμού του App Inventor αποτελείται από:

Τον App Inventor Designer, όπου επιλέγονται τα βασικά δομικά στοιχεία της εφαρμογής που θα δημιουργηθεί

Τον App Inventor Blocks Editor, όπου λαμβάνει χώρα ο καθορισμός – προγραμματισμός της συμπεριφοράς των βασικών δομικών στοιχείων της εφαρμογής με σύρσιμο πλακιδίων. Εδώ ο προγραμματισμός συμβαίνει με δόμηση των προγραμμάτων με οπτικό τρόπο, ταιριάζοντας πλακίδια μαζί όπως τα κομμάτια ενός πάζλ.

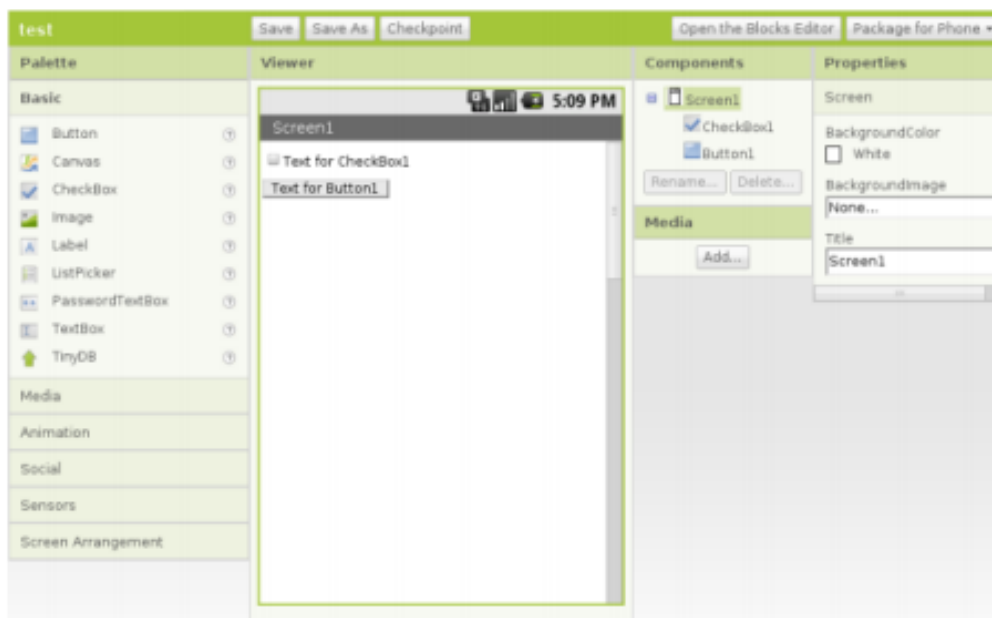
Η εφαρμογή που δημιουργείται εμφανίζεται στο συνδεδεμένο τηλέφωνο ή στον εξομοιωτή τηλεφώνου, βήμα-προς-βήμα, όπως προστίθενται κομμάτια σε αυτή, ώστε να είναι δυνατός ο έλεγχος της εφαρμογής καθώς αυτή χτίζεται. Με την ολοκλήρωση της εφαρμογής, υπάρχει η δυνατότητα πακεταρίσματος της εφαρμογής έτσι ώστε να είναι δυνατή η εγκατάσταση της σε Android συσκευές.

3.1.Στοιχεία για το App Inventor

Όπως αναφέρθηκε και προηγουμένως το App Inventor είναι web εφαρμογή και αυτό είναι ιδιαίτερα σημαντικό αφού παρέχει τρία σημαντικά πλεονεκτήματα. Μπορεί να λειτουργήσει από οποιοδήποτε Η/Υ που έχει πρόσβαση στο διαδίκτυο αρκεί να έχει εγκατεστημένο ένα περιηγητή. Ο περιορισμός που υπάρχει είναι, ότι οι μόνοι περιηγητές που είναι συμβατοί προς το παρόν με το App Inventor, είναι ο Mozilla Firefox και ο Google Chrome. Ο χρήστης δε χρειάζεται να εγκαταστήσει την εφαρμογή στον Η/Υ του. Επίσης, οποτεδήποτε ενσωματώνεται μια αναβάθμιση, δε χρειάζεται ο χρήστης να την κατεβάσει και να την εγκαταστήσει, αφού όλα γίνονται αυτόματα χωρίς εκείνος να επέμβει στη διαδικασία. Όταν θα μπει στο App Inventor θα ενημερωθεί ότι έχει γίνει αναβάθμιση. Ακόμα, όλα τα προγράμματα που γράφει ο χρήστης αποθηκεύονται στο cloud και έτσι μπορεί να έχει πρόσβαση σε αυτά, από οποιαδήποτε συσκευή (υπολογιστής, ταμπλέτα, έξυπνο κινητό τηλέφωνο) αρκεί να συνδέεται με το internet. Παράλληλα η αποθήκευση στο cloud του επιτρέπει το χρήστη να έχει αντίγραφα ασφαλείας των εφαρμογών του.

Το App Inventor 2 έχει αρκετές διαφορές από τον προκάτοχό του. Προστέθηκαν νέες λειτουργίες και βελτιώθηκε το μενού των επιλογών στο οποίο έχει πρόσβαση ο χρήστης. Στη συνέχεια περιγράφονται τα πιο ενδιαφέροντα. Η βασική αλλαγή είναι το web περιβάλλον, σε αντίθεση με την έκδοση 1 που απαιτούσε από το χρήστη την εγκατάσταση της εφαρμογής στον υπολογιστή του. Ο τύπος αρχείου του πηγαίου κώδικα έγινε aia σε αντίθεση με το zip που ίσχυε. Ο χρήστης μπορεί να προσθέσει νέες στο Designer Mode ή στο Blocks Mode (που στην προηγούμενη έκδοση δεν επιτρεπόταν). Στη σελίδα των μπλοκ, η ταξινόμηση τους έχει αλλάξει και τοποθετήθηκαν στα αριστερά της οθόνης. Οι διαδικασίες (μοιάζουν πολύ με τις

μεθόδους), απέκτησαν δικιά τους κατηγορία. Επιπλέον μικρές αλλά σημαντικές αλλαγές, που βοηθούν το χρήστη έγιναν στα χρώματα, την προσομοίωση, στις μεταβλητές αλλά και σε ορισμένα μπλοκ.

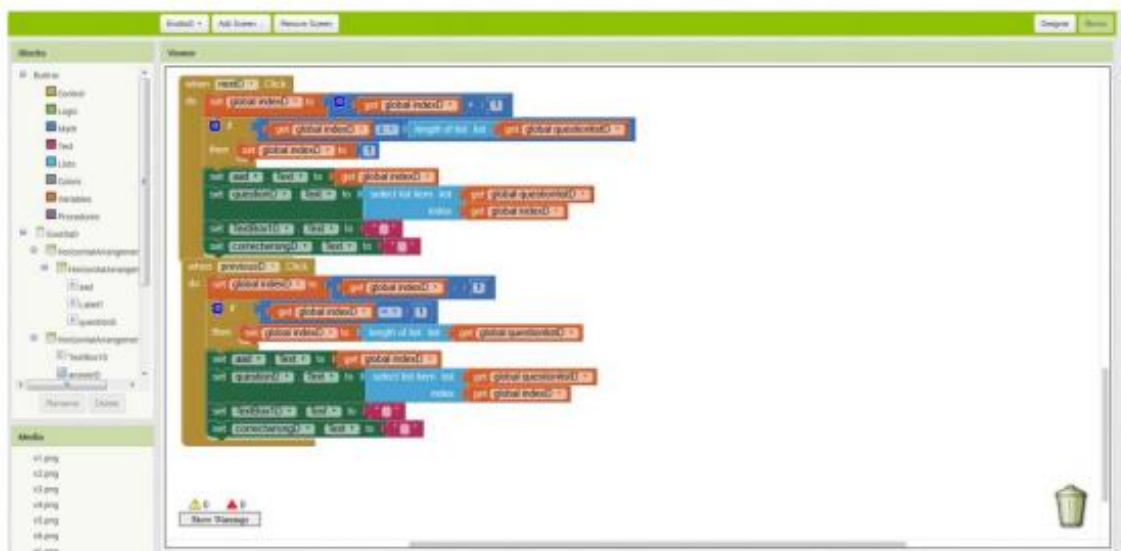


Εικόνα 3. Περιβάλλον του App Inventor

Το μενού της εφαρμογής, περιλαμβάνει ένα γραφικό περιβάλλον που χωρίζεται σε τρία τμήματα. Στο κεντρικό τμήμα, κυριαρχεί η (υποθετική) οθόνη του smartphone. Στα αριστερά, υπάρχουν οι κατηγορίες με τα αντικείμενα που μπορεί να προσθέσει στην οθόνη, με τη χρήση της μεθόδου drag and drop και στα δεξιά υπάρχουν οι καρτέλες ιδιοτήτων και συστατικών όπου ο χρήστης μπορεί να ρυθμίσει το κάθε αντικείμενο, που έχει προσθέσει, ξεχωριστά. Η λογική που πρέπει να ακολουθήσει ο χρήστης, είναι να φτιάξει πρώτα το γραφικό περιβάλλον της εφαρμογής του, επιλέγοντας τα χαρακτηριστικά που θέλει και στη συνέχεια να μεταβεί από το χώρο σχεδίασης στο χώρο των μπλοκ όπου εκεί θα δηλώσει μεταβλητές και λειτουργίες. Στο πάνω μέρος της σελίδας υπάρχει ένα μενού από drop down τα οποία παρέχουν επιπρόσθετες λειτουργίες. Τα μενού αυτά είναι Project από το οποίο δημιουργείται νέο project, εισαγωγή project από τον υπολογιστή του χρήστη, διαγραφή και αποθήκευση. Το μενού Connect επιτρέπει τη δοκιμή της εφαρμογής σε προσομοιωτή, ή σε φορητή συσκευή. Το μενού Build παρέχει στο χρήστη την εφαρμογή σε δύο

μορφές. Η πρώτη δείχνει ένα κώδικα QR ώστε ο χρήστης να κατεβάσει την εφαρμογή στη συσκευή (άμεσα) και η δεύτερη παράγει το apk ώστε να μεταφερθεί στη συσκευή έμμεσα, π.χ. μέσω καλωδίου. Το Help εμφανίζει πληροφορίες έκδοσης του App Inventor και συντομεύσεις για τις βιβλιοθήκες, το φόρουμ και τα μαθήματα εκμάθησης. Το My Projects εμφανίζει μια λίστα με όλα τα projects που έχει δημιουργήσει ο χρήστης και εμφανίζει ημερομηνίες δημιουργίας και τελευταίας τροποποίησης. Το Guide ανοίγει ένα νέο παράθυρο με το documentation της εφαρμογής και τέλος, το Report an Issue χρησιμεύει στην εμφάνιση των πιο συχνών προβλημάτων που αντιμετωπίζουν οι χρήστες και στην επικοινωνία με τον κατασκευαστή του App Inventor.

Στα δεξιά της οθόνης υπάρχουν τα Συστατικά, οι Ιδιότητες και τα Μέσα. Τα συστατικά, είναι μια λίστα που περιλαμβάνει όλα τα αντικείμενα που έχει τοποθετήσει ο χρήστης στην οθόνη του. Αυτά, ταξινομούνται ανάλογα με τη θέση τους και είναι δυνατή η αλλαγή της προεπιλεγμένης ονομασίας σε αυτή που επιθυμεί ο χρήστης. Οι ονομασίες αυτές είναι πολύ χρήσιμες γιατί αυτές θα χρησιμοποιηθούν στο προγραμματισμό. Οι ιδιότητες, αναφέρονται σε κάθε ένα από τα αντικείμενα που έχει βάλει στην οθόνη ο χρήστης. Οι ιδιότητες περιλαμβάνουν χρώματα, μεγέθη, αλλαγή διάταξης, βοηθητικά μηνύματα για το χρήστη. Τα Μέσα, περιλαμβάνουν όλες τις εικόνες που χρησιμοποιεί η εφαρμογή. Αυτές τις ανεβάζει ο χρήστης και στη συνέχεια τις τοποθετεί εκεί που επιθυμεί μέσω του προγράμματος. Ο χρήστης σχεδιάζει το περιβάλλον της κάθε οθόνης όπως επιθυμεί με τη χρήση των συστατικών που αναφέρθηκαν παραπάνω. Στη συνέχεια, για την κάθε μια οθόνη μεταβαίνει στην καρτέλα των Blocks.



Εικόνα 4. Το περιβάλλον των blocks

Η καρτέλα χωρίζεται σε δύο τμήματα όπου στο αριστερό βρίσκονται τα μπλοκ και στην υπόλοιπη οθόνη είναι το περιβάλλον στο οποίο ο χρήστης τα τοποθετεί. Τα μπλοκ χωρίζονται σε δύο κατηγορίες. Στην πρώτη είναι τα ανώνυμα μπλοκ τα οποία ο χρήστης τα χρησιμοποιεί για να εκτελέσει διάφορες λειτουργίες. Έτσι λοιπόν, αυτά αφορούν τον έλεγχο με τη χρήση της επαναληπτικής δομή while αλλά και των συνθηκών διακλάδωσης if, if-then-else. Επίσης στη σελίδα αυτή, είναι διαθέσιμα μπλοκ όπως το άνοιγμα άλλης οθόνης, κλείσιμο οθόνης και κλείσιμο εφαρμογής. Όλα τα προαναφερθέντα μπλοκ έχουν πορτοκαλί χρώμα. Στη συνέχεια είναι τα μπλοκ των λογικών πράξεων. And, or, not, =, >, <, και οι τιμές true, false. Το χρώμα των μπλοκ είναι πράσινο. Ακολουθούν τα μαθηματικά μπλοκ, με όλες τις πράξεις, τιμές, τυχαίους αριθμούς, στρογγυλοποιήσεις, τριγωνομετρικούς αριθμούς, όπως επίσης και έλεγχος αν το όρισμα είναι αριθμός. Το χρώμα αυτών των μπλοκ είναι μπλε. Στη συνέχεια είναι τα μπλοκ κειμένου. Οι εντολές ελέγχουν για αλφαριθμητικά, μετατρέπουν κεφαλαία σε πεζά και το αντίστροφο, διασπών ένα αλφαριθμητικό στα κενά του και αντικαθιστούν ένα κείμενο. Το χρώμα που αντιστοιχεί στα μπλοκ αυτά, είναι το φούξια. Ακολουθούν οι λίστες. Ο χρήστης μπορεί να δημιουργήσει κενές λίστες, ή να τοποθετήσει αντικείμενα σ' αυτές εξ αρχής. Μπορεί να γίνει έλεγχος αν ένα αντικείμενο υπάρχει ήδη στη λίστα, όπως επίσης και να μετρηθούν τα αντικείμενα μιας λίστας. Ο χρήστης με κατάλληλες εντολές μπορεί να προσθέσει και να αλλάξει κάποια από τα στοιχεία της λίστας, να αντιγράψει τα στοιχεία μιας λίστας

σε μια άλλη και να μεταφερθούν δεδομένα από πίνακες CSV και ακόμα να κάνει αναζήτηση ενός αντικειμένου σε μια λίστα. Το χρώμα των λιστών είναι γαλάζιο. Το μπλοκ των χρωμάτων επιτρέπει την εμφάνιση των 13 πιο γνωστών χρωμάτων. Τα μπλοκ των μεταβλητών, επιτρέπουν τη δημιουργία, αρχικοποίηση, τοπικών και καθολικών μεταβλητών. Το χρώμα των μεταβλητών είναι πορτοκαλί. Τελευταίο μπλοκ είναι οι διαδικασίες. Το χρώμα τους είναι το μωβ.

Η δεύτερη κατηγορία των μπλοκ, σχετίζεται άμεσα με τα αντικείμενα που τοποθετήθηκαν σε μια οθόνη από τον ίδιο το χρήστη. Για το κάθε ένα αντικείμενο της οθόνης, διατίθεται ένα σετ εντολών. Για παράδειγμα, υπάρχουν μπλοκ που αφορούν τη διαχείριση της οθόνης που προβάλλεται στο χρήστη και οι εντολές αφορούν την εμφάνιση μηνυμάτων και την εκτέλεση άλλων διαδικασιών. Τέτοιες είναι οι τίτλοι, αλλαγές χρωμάτων, άνοιγμα και κλείσιμο μια οθόνης και χρήση εικόνας ως φόντο. Η ύπαρξη τουλάχιστον μιας οθόνης σε μια εφαρμογή είναι δεδομένη. Τα μπλοκ που αφορούν την κάθε οθόνη σχετίζονται με το άνοιγμα και κλείσιμο μιας άλλης, ή με λεπτομέρειες που καθορίζουν την ίδια. Το τελευταίο σημαίνει ότι ρυθμίζονται παράμετροι, όπως το φόντο, οι εικόνες, τα χρώματα και η σχεδιοκίνηση. Μπορούν να ρυθμιστούν ο προσανατολισμός, το ύψος και το πλάτος της οθόνης. Για τα αντικείμενα εικόνας, τα μπλοκ αφορούν τη μεταβολή του μεγέθους της, αντικατάστασή της από άλλη και προσθήκη σχεδιοκίνησης. Ο χρήστης επιπλέον, μπορεί να επιλέξει πότε η εικόνα θα είναι ορατή και πότε όχι, όπως επίσης και ποια μηνύματα θα εμφανίζονται στην οθόνη. Τα αντικείμενα διάταξης έχουν μπλοκ που σχετίζονται με τον τρόπο εμφάνισης της πληροφορίας.

3.2.Δοκιμή εφαρμογής

Για να δοκιμάσει ο χρήστης την εφαρμογή που έχει φτιάξει, μπορεί να επιλέξει έναν από τους τρεις τρόπους που περιγράφονται. Ο πρώτος από αυτούς είναι η σύνδεση του App Inventor με τη συσκευή μέσω Wi-Fi. Ο χρήστης βλέπει την εκτέλεση της εφαρμογής του και σε πραγματικό χρόνο μπορεί να επέμβει στον κώδικα και να κάνει αλλαγές. Οι αλλαγές περνούν αυτόματα στη συσκευή. Για να λειτουργήσει αυτή η μέθοδος, πρέπει στη συσκευή να εγκατασταθεί το MIT AI2 Companion App που είναι διαθέσιμο στο Google Play. Αμέσως μετά πραγματοποιείται η σύνδεση μεταξύ

του App Inventor και της συσκευής μέσω κώδικα QR ή με την εισαγωγή στη συσκευή ενός κωδικού που παρέχει το App Inventor. Αμέσως μετά, η εφαρμογή πρέπει να εκτελεστεί κανονικά και παράλληλα η διεύθυνση IP της εμφανίζεται και στον υπολογιστή. Σε οποιαδήποτε οθόνη της εφαρμογής και αν βρίσκεται ο χρήστης, ο κώδικας της εμφανίζεται στην οθόνη του υπολογιστή του.

Ο δεύτερος τρόπος δοκιμής είναι η χρήση του προσομοιωτή. Αυτός ο τρόπος επίσης προτείνεται γι' αυτούς που θέλουν να φτιάξουν εφαρμογές χωρίς να διαθέτουν κάποια συσκευή που να εκτελεί το ΛΣ Android. Το μειονέκτημα του προσομοιωτή είναι ότι αργεί να δείξει άλλες οθόνες και πολλές φορές κολλάει. Η δε εκκίνηση λειτουργία του προσομοιωτή χρειάζεται περίπου δύο λεπτά για να εκκινήσει. Για να γίνει η χρήση του προσομοιωτή πρέπει ο χρήστης να εγκαταστήσει στον υπολογιστή του το aiStarter, που είναι διαθέσιμο μόνο για Windows και Linux. Οποιαδήποτε στιγμή, ο χρήστης μπορεί να δημιουργήσει το apk αρχείο και να το μεταφέρει στη συσκευή του για να δοκιμάσει την εφαρμογή. Ο τρίτος τρόπος δοκιμής είναι η σύνδεση της συσκευής μέσω USB με τον υπολογιστή.

Ο χρήστης πρέπει να κατεβάσει το κατάλληλο πρόγραμμα, ανάλογα με το ΛΣ του υπολογιστή του και στη συνέχεια να ακολουθήσει τα ίδια βήματα με τον προηγούμενο τρόπο. Στο τέλος, πρέπει να δοκιμάσει αν λειτουργεί η σύνδεση. Ο έλεγχος της σύνδεσης γίνεται μέσω του περιηγητή. Ενδεικτική σύνδεση μέσω USB

Οι απαιτήσεις που έχει το App Inventor είναι: Λειτουργικό σύστημα windows XP ή ανώτερο, GNU/Linux Ubuntu 8 ή ανώτερο, Debian 5 ή ανώτερο και Mac OS X 10.5 ή ανώτερο. Οι περιηγητές ιστοσελίδων που υποστηρίζονται είναι οι Firefox 3.6, Safari 5.0, Chrome 4.0 και όλες οι μεταγενέστερες εκδόσεις. Ο internet Explorer προς το παρόν δεν υποστηρίζεται. Σχετικά με τη συσκευή (τηλέφωνο ή ταμπλέτα), πρέπει να έχει ΛΣ Android 2.3 ή ανώτερο.

ΚΕΦΑΛΑΙΟ 4: ΠΩΣ ΛΕΙΤΟΥΡΓΕΙ ΤΟ ARDUINO

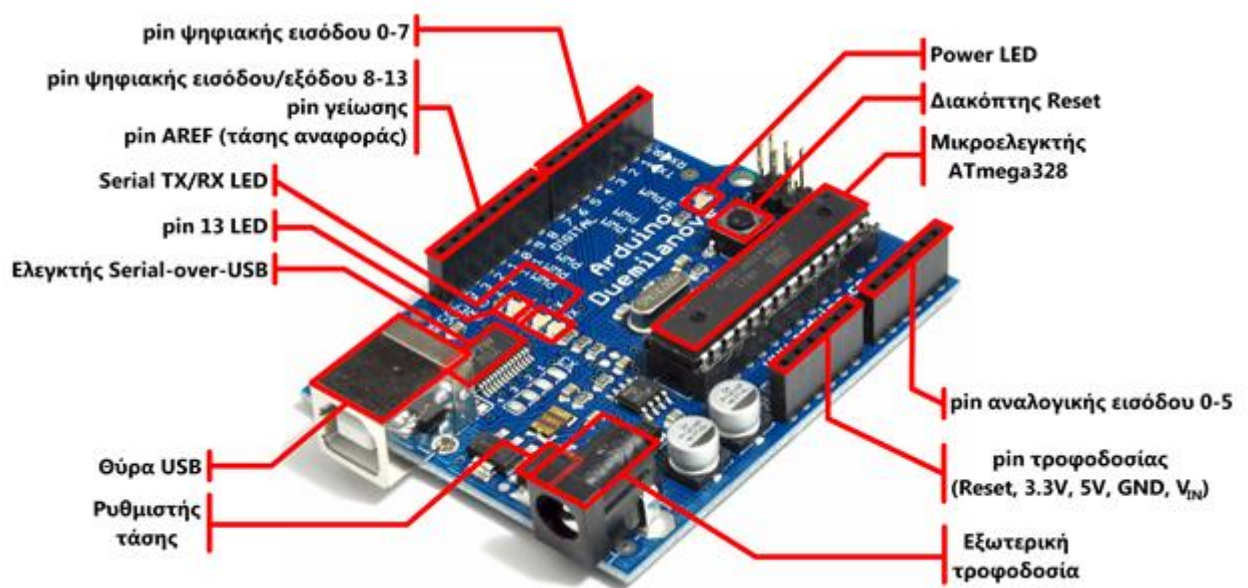
4.1.Μικροελεγκτές

Το Arduino βασίζεται στον ATmega328, έναν 8-bit RISC μικροελεγκτή, τον οποίο χρονίζει στα 16MHz. Ο ATmega328 διαθέτει ενσωματωμένη μνήμη τριών τύπων:

- 2Kb μνήμης SRAM που είναι η ωφέλιμη μνήμη που μπορούν να χρησιμοποιήσουν τα προγράμματά για να αποθηκεύουν μεταβλητές, πίνακες κ.λπ. κατά το runtime. Όπως και σε έναν υπολογιστή, αυτή η μνήμη χάνει τα δεδομένα της όταν η παροχή ρεύματος στο Arduino σταματήσει ή αν γίνει reset.
- 1Kb μνήμης EEPROM η οποία μπορεί να χρησιμοποιηθεί για «ωμή» εγγραφή/ανάγνωση δεδομένων (χωρίς datatype) ανά byte από τα προγράμματά κατά το runtime. Σε αντίθεση με την SRAM, η EEPROM δεν χάνει τα περιεχόμενά της με απώλεια τροφοδοσίας ή reset οπότε είναι το ανάλογο του σκληρού δίσκου.
- 32Kb μνήμης Flash, από τα οποία τα 2Kb χρησιμοποιούνται από το firmware του Arduino που έχει εγκαταστήσει ήδη ο κατασκευαστής του. Το firmware αυτό που στην ορολογία του Arduino ονομάζεται bootloader είναι αναγκαίο για την εγκατάσταση των δικών προγραμμάτων στον μικροελεγκτή μέσω της θύρας USB, χωρίς δηλαδή να χρειάζεται εξωτερικός hardware programmer. Τα υπόλοιπα 30Kb της μνήμης Flash χρησιμοποιούνται για την αποθήκευση αυτών ακριβώς των προγραμμάτων, αφού πρώτα μεταγλωττιστούν στον υπολογιστή. Η μνήμη Flash, όπως και η EEPROM δεν χάνει τα περιεχόμενά της με απώλεια τροφοδοσίας ή reset. Επίσης, ενώ η μνήμη Flash υπό κανονικές συνθήκες δεν προορίζεται για χρήση runtime μέσα από τα προγράμματά , λόγω της μικρής συνολικής μνήμης που είναι διαθέσιμη σε αυτά (2Kb SRAM + 1Kb EEPROM), έχει σχεδιαστεί μια βιβλιοθήκη που επιτρέπει την χρήση όσου χώρου περισσεύει (30Kb μείον το μέγεθος του προγράμματός σε μεταγλωττισμένη μορφή).

4.2.Είσοδοι – Έξοδοι

Καταρχήν το Arduino διαθέτει σειριακό interface. Ο μικροελεγκτής ATmega υποστηρίζει σειριακή επικοινωνία, την οποία το Arduino προωθεί μέσα από έναν ελεγκτή Serial-over-USB ώστε να συνδέεται με τον υπολογιστή μέσω USB. Η σύνδεση αυτή χρησιμοποιείται για την μεταφορά των προγραμμάτων που σχεδιάζονται από τον υπολογιστή στο Arduino αλλά και για αμφίδρομη επικοινωνία του Arduino με τον υπολογιστή μέσα από το πρόγραμμα την ώρα που εκτελείται.



Εικόνα 5. Είσοδοι – Έξοδοι

Επιπλέον, στην πάνω πλευρά του Arduino βρίσκονται 14 θηλυκά pin, αριθμημένα από 0 ως 13, που μπορούν να λειτουργήσουν ως ψηφιακές εισοδοι και έξοδοι. Λειτουργούν στα 5V και καθένα μπορεί να παρέχει ή να δεχτεί το πολύ 40mA.

Ως ψηφιακή έξοδος, ένα από αυτά τα pin μπορεί να τεθεί από το πρόγραμμα σε κατάσταση HIGH ή LOW, οπότε το Arduino θα ξέρει αν πρέπει να διοχετεύσει ή όχι ρεύμα στο συγκεκριμένο pin. Με αυτόν τον τρόπο μπορούμε λόγω χάρη να ανάψουμε και να σβήσουμε ένα LED που έχουμε συνδέσει στο συγκεκριμένο pin. Αν πάλι ρυθμίσουμε ένα από αυτά τα pin ως ψηφιακή είσοδο μέσα από το πρόγραμμα, μπορούμε με την κατάλληλη εντολή να διαβάσουμε την κατάστασή του (HIGH ή

LOW) ανάλογα με το αν η εξωτερική συσκευή που έχουμε συνδέσει σε αυτό το pin διοχουλεύει ή όχι ρεύμα στο pin (με αυτόν τον τρόπο λόγω χάρη μπορούμε να «διαβάσουμε» την κατάσταση ενός διακόπτη).

Μερικά από αυτά τα 14 pin, εκτός από ψηφιακές είσοδοι/έξοδοι έχουν και δεύτερη λειτουργία. Συγκεκριμένα:

- Τα pin 0 και 1 λειτουργούν ως RX και TX της σειριακής όταν το πρόγραμμα ενεργοποιεί την σειριακή θύρα. Έτσι, όταν λόγω χάρη το πρόγραμμα στέλνει δεδομένα στην σειριακή, αυτά προωθούνται και στην θύρα USB μέσω του ελεγκτή Serial-Over-USB αλλά και στο pin 0 για να τα διαβάσει ενδεχομένως μια άλλη συσκευή (π.χ. ένα δεύτερο Arduino στο δικό του pin 1). Αυτό φυσικά σημαίνει ότι αν στο πρόγραμμα ενεργοποιήσουμε το σειριακό interface, χάνουμε 2 ψηφιακές εισόδους/εξόδους.
- Τα pin 2 και 3 λειτουργούν και ως εξωτερικά interrupt (interrupt 0 και 1 αντίστοιχα). Με άλλα λόγια, μπορούμε να τα ρυθμίσουμε μέσα από το πρόγραμμα ώστε να λειτουργούν αποκλειστικά ως ψηφιακές είσοδοι στις οποίες όταν συμβαίνουν συγκεκριμένες αλλαγές, η κανονική ροή του προγράμματος σταματάει *άμεσα* και εκτελείται μια συγκεκριμένη συνάρτηση. Τα εξωτερικά interrupt είναι ιδιαίτερα χρήσιμα σε εφαρμογές που απαιτούν συγχρονισμό μεγάλης ακρίβειας.
- Τα pin 3, 5, 6, 9, 10 και 11 μπορούν να λειτουργήσουν και ως ψευδοαναλογικές έξοδοι με το σύστημα PWM (Pulse Width Modulation), δηλαδή το ίδιο σύστημα που διαθέτουν οι μητρικές των υπολογιστών για να ελέγχουν τις ταχύτητες των ανεμιστήρων. Έτσι, μπορούμε να συνδέσουμε λόγω χάρη ένα LED σε κάποιο από αυτά τα pin και να ελέγξουμε πλήρως την φωτεινότητά του με ανάλυση 8bit (256 καταστάσεις από 0-σβηστό ως 255-πλήρως αναμμένο) αντί να έχουμε απλά την δυνατότητα αναμμένο-σβηστό που παρέχουν οι υπόλοιπες ψηφιακές έξοδοι. Είναι σημαντικό να καταλάβουμε ότι το PWM δεν είναι πραγματικά αναλογικό σύστημα και ότι θέτοντας στην έξοδο την τιμή 127, δεν σημαίνει ότι η έξοδος θα δίνει 2.5V αντί της κανονικής τιμής των 5V, αλλά ότι θα δίνει ένα παλμό που θα

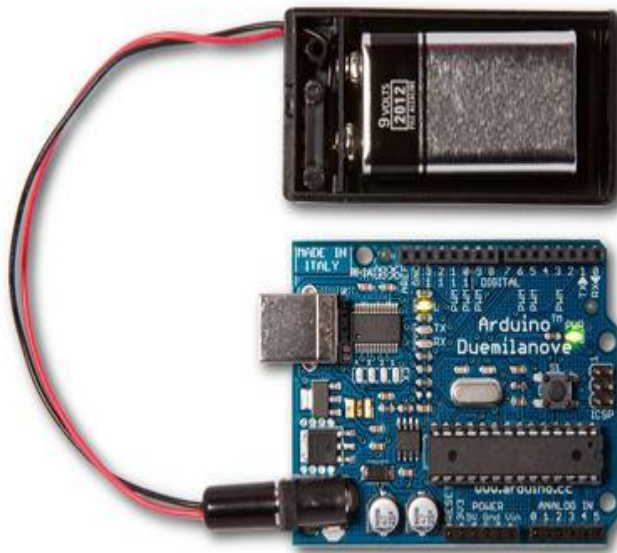
εναλλάσσεται με μεγάλη συχνότητα και για ίσους χρόνους μεταξύ των τιμών 0 και 5V.

Στην κάτω πλευρά του Arduino, με τη σήμανση ANALOG IN, θα βρούμε μια ακόμη σειρά από 6 pin, αριθμημένα από το 0 ως το 5. Το καθένα από αυτά λειτουργεί ως αναλογική είσοδος κάνοντας χρήση του ADC (Analog to Digital Converter) που είναι ενσωματωμένο στον μικροελεγκτή. Για παράδειγμα, μπορούμε να τροφοδοτήσουμε ένα από αυτά με μια τάση την οποία μπορούμε να κυμάνουμε με ένα ποτενσιόμετρο από 0V ως μια τάση αναφοράς V_{ref} η οποία, αν δεν κάνουμε κάποια αλλαγή είναι προρυθμισμένη στα 5V. Τότε, μέσα από το πρόγραμμα μπορούμε να «διαβάσουμε» την τιμή του pin ως ένα ακέραιο αριθμό ανάλυσης 10-bit, από 0 (όταν η τάση στο pin είναι 0V) μέχρι 1023 (όταν η τάση στο pin είναι 5V). Η τάση αναφοράς μπορεί να ρυθμιστεί με μια εντολή στο 1.1V, ή σε όποια τάση επιθυμούμε (μεταξύ 2 και 5V) τροφοδοτώντας εξωτερικά με αυτή την τάση το pin με την σήμανση AREF που βρίσκεται στην απέναντι πλευρά της πλακέτας. Έτσι, αν τροφοδοτήσουμε το pin AREF με 3.3V και στην συνέχεια δοκιμάσουμε να διαβάσουμε κάποιο pin αναλογικής εισόδου στο οποίο εφαρμόζουμε τάση 1.65V, το Arduino θα επιστρέψει την τιμή 512.

Τέλος, καθένα από τα 6 αυτά pin, με κατάλληλη εντολή μέσα από το πρόγραμμα μπορεί να μετατραπεί σε ψηφιακό pin εισόδου/εξόδου όπως τα 14 που βρίσκονται στην απέναντι πλευρά και τα οποία περιγράφηκαν πριν. Σε αυτή την περίπτωση τα pin μετονομάζονται από 0~5 σε 14~19 αντίστοιχα.

4.3. Τροφοδοσία και Led

Το Arduino μπορεί να τροφοδοτηθεί με ρεύμα από τον υπολογιστή μέσω της σύνδεσης USB, ή από εξωτερική τροφοδοσία που παρέχεται μέσω μιας υποδοχής φισ των 2.1mm (θετικός πόλος στο κέντρο) και βρίσκεται στην κάτω-αριστερή γωνία του Arduino.



Εικόνα 6. Τροφοδοσία Arduino

Για να μην υπάρχουν προβλήματα, η εξωτερική τροφοδοσία πρέπει να είναι από 7 ως 12V και μπορεί να προέρχεται από ένα κοινό μετασχηματιστή του εμπορίου, από μπαταρίες ή οποιαδήποτε άλλη πηγή DC. Δίπλα από τα pin αναλογικής εισόδου, υπάρχει μια ακόμα συστοιχία από 6 pin με την σήμανση POWER. Η λειτουργία του καθενός έχει ως εξής:

- Το πρώτο, με την ένδειξη RESET, όταν γειωθεί (σε οποιοδήποτε από τα 3 pin με την ένδειξη GND που υπάρχουν στο Arduino) έχει ως αποτέλεσμα την επανεκκίνηση του Arduino.
- Το δεύτερο, με την ένδειξη 3.3V, μπορεί να τροφοδοτήσει τα εξαρτήματά με τάση 3.3V. Η τάση αυτή δεν προέρχεται από την εξωτερική τροφοδοσία αλλά παράγεται από τον ελεγκτή Serial-over-USB και έτσι η μέγιστη ένταση που μπορεί να παρέχει είναι μόλις 50mA.
- Το τρίτο, με την ένδειξη 5V, μπορεί να τροφοδοτήσει τα εξαρτήματά με τάση 5V. Ανάλογα με τον τρόπο τροφοδοσίας του ίδιου του Arduino, η τάση αυτή προέρχεται ούμε άμεσα από την θύρα USB (που ούτως ή άλλως λειτουργεί στα 5V), ούμε από την εξωτερική τροφοδοσία αφού αυτή περάσει από ένα ρυθμιστή τάσης για να την «φέρει» στα 5V.
- Το τέταρτο και το πέμπτο pin, με την ένδειξη GND, είναι φυσικά γειώσεις.
- Το έκτο και τελευταίο pin, με την ένδειξη Vin έχει διπλό ρόλο. Σε συνδυασμό με το pin γείωσης δίπλα του, μπορεί να λειτουργήσει ως μέθοδος εξωτερικής τροφοδοσίας του Arduino, στην περίπτωση που δεν βολεύει να χρησιμοποιήσουμε την υποδοχή του φισ των 2.1mm. Αν όμως έχουμε ήδη συνδεδεμένη εξωτερική τροφοδοσία μέσω του φισ, μπορούμε να χρησιμοποιήσουμε αυτό το pin για να τροφοδοτήσουμε εξαρτήματα με την πλήρη τάση της εξωτερικής τροφοδοσίας (7~12V), πριν αυτή περάσει από τον ρυθμιστή τάσης όπως γίνεται με το pin των 5V.

Πάνω στην πλακέτα του Arduino υπάρχει ένας διακόπτης micro-switch και 4 μικροσκοπικά LED επιφανειακής στήριξης. Η λειτουργία του διακόπτη (που έχει την σήμανση RESET) και του ενός LED με την σήμανση POWER είναι μάλλον προφανής.

Τα δύο LED με τις σημάνσεις TX και RX, χρησιμοποιούνται ως ένδειξη λειτουργίας του σειριακού interface, καθώς ανάβουν όταν το Arduino στέλνει ή λαμβάνει (αντίστοιχα) δεδομένα μέσω USB. Τα LED αυτά ελέγχονται από τον ελεγκτή Serial-over-USB και συνεπώς δεν λειτουργούν όταν η σειριακή επικοινωνία γίνεται αποκλειστικά μέσω των ψηφιακών pin 0 και 1.

Τέλος, υπάρχει το LED με την σήμανση L. Η βασική δοκιμή λειτουργίας του Arduino είναι να του αναθέσουμε να αναβοσβήνει ένα LED (θα το δούμε αυτό στην συνέχεια όταν θα φτιάξουμε την πρώτη εφαρμογή). Οι κατασκευαστές του σκέφτηκαν να ενσωματώσουν ένα LED στην πλακέτα, το οποίο σύνδεσαν στο ψηφιακό pin 13. Έτσι, ακόμα και αν δεν έχει κανείς συνδέσει τίποτα πάνω στο φυσικό pin 13, αναθέτοντάς του την τιμή HIGH μέσα από το πρόγραμμά , θα ανάψει αυτό το ενσωματωμένο LED.

ΚΕΦΑΛΑΙΟ 5: ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΣ ΣΕ ARDUINO

Το Arduino Mega έχει το μικροελεγκτής ATmega328 που το κάνει να είναι και η «καρδιά» του κυκλώματος που ελέγχει όλες τις εισόδους και εξόδους του. Το Arduino Nano 328 είναι μια μικρή, πλήρης πλακέτα που έχει τον μικροελεγκτή ATmega328. Τροφοδοτείται μέσω mini B USB. Ο ανιχνευτής ήχου αφουγκράζεται τους εξωτερικούς ήχους και τους δημιουργεί ως εισόδους για την πλακέτα του Arduino ώστε να τις επεξεργαστεί και να δώσει τις αντίστοιχες εξόδους κάθε φορά. Οι MQ2 αισθητήρες αερίου χρησιμοποιούν ένα μικρό καλοριφέρ με ένα ηλεκτροχημικό αισθητήρα. Αυτά είναι ευαίσθητα και χρησιμοποιούνται για την ανίχνευση αερίων σε περίπτωση διαρροής. Η έξοδος του είναι αναλογική και μπορεί να συνδεθεί σε μία αναλογική είσοδο του Arduino.

Ψηφιακός αισθητήρας θερμοκρασίας και υγρασίας. Ανιχνεύει την θερμοκρασία και την υγρασία του περιβάλλοντα χώρου και τα αποτελέσματα τα βγάζει σε ψηφιακή έξοδο. Οι αισθητήρες κίνησης είναι κατασκευασμένοι από ένα πυροηλεκτρικό αισθητήρα, ο οποίος μπορεί να ανιχνεύσει επίπεδα υπέρυθρης ακτινοβολίας (όλα εκπέμπουν υπέρυθρη ακτινοβολία, όσο πιο θερμό τόσο περισσότερη ακτινοβολία εκπέμπει). Το WiFi Connector είναι μία συσκευή WiFi η οποία επιτρέπει στο κύκλωμα να συνδεθεί με την Android συσκευή και να δέχεται αλλά και να στέλνει δεδομένα από και προς την εφαρμογή Smart Home. Το TEA5767 είναι ένας δέκτης FM, που ελέγχεται μέσω I2C οθόνες. Το ραδιόφωνο μπορεί να συντονιστεί με Ευρώπη, ΗΠΑ και ιαπωνικές συχνότητες FM.

Ο HC-SR04 είναι ένας αισθητήρας υπερήχων που μπορεί και υπολογίζει την απόσταση που έχει ένα αντικείμενο. Είναι εξαιρετικής ακρίβειας. Ο πυκνωτής (capacitor) χρησιμοποιείται για την προσωρινή αποθήκευση ηλεκτρικής ενέργειας σε ένα ηλεκτρικό πεδίο. Το Motor Shield επιτρέπει τον έλεγχο της ταχύτητας και της κατεύθυνσης 2 κινητήρων συνεχούς ρεύματος, ξεχωριστά. Το LDR είναι αισθητήρας φωτός, μπορεί «καταλάβει» αν έχει φώς η σκοτάδι και να στείλει την αντίστοιχη έξοδο στο Arduino. Το DS1307 είναι ένα σειριακό ρολόι πραγματικού χρόνου. Είναι σε δυαδικό κώδικα και παρέχει πληροφορίες δευτερολέπτων, λεπτών, ώρας και ημερομηνίας. Το DS18B20 είναι ένα ψηφιακό θερμόμετρο που παρέχει μετρήσεις θερμοκρασίας σε βαθμούς Κελσίου αλλά και ειδοποιήσεις. Οι διακόπτες εισόδων

Arduino τοποθετούνται στο κύκλωμα και μπορούμε χειροκίνητα να δώσουμε τιμές 1 και 0 για εισόδους. Το Dimmer ελέγχει την ένταση της φωτεινότητας ενός Led. Το Ψηφιακό ποτενσιόμετρο, μας δίνει την δυνατότητα να ελέγχουμε την αντίσταση ενός κυκλώματος χωρίς να είναι απαραίτητο να τη ρυθμίζουμε χειροκίνητα.

Το Transmitter Receiver είναι πομπός δέκτης σαν το WiFi σε συχνότητες 477Hz. Το Relay Board έχει I2C πρωτόκολλο επικοινωνίας. Χρησιμοποιεί τον εγκέφαλο MCP23017. Έχει την δυνατότητα για interrupts.

Η λογική είναι η εξής, οι αισθητήρες παίρνουν δεδομένα τα οποία στέλνουν στο Arduino, το οποίο με την σειρά του τα ανεβάζει μέσω της Ethernet Shield σε έναν HTTP Server. Στην συνέχεια η Android εφαρμογή διαβάζει τα δεδομένα αυτά από τον HTTP Server, και τα προβάλλει.

Ο κώδικας java που χρησιμοποιούμε είναι:

```
package com.sql.greekapp;

import android.app.Activity;

import android.os.Build;

import android.os.Bundle;

import android.view.KeyEvent;

import android.view.Window;

import android.webkit.WebChromeClient;

import android.webkit.WebSettings;

import android.webkit.WebView;

import android.webkit.WebViewClient;

import com.sql.greekapp.R;
```



```

public class GreekNewsActivity extends Activity {

    WebView webview;

    public void onCreate(Bundle savedInstanceState) {

        super.onCreate(savedInstanceState);

        this.getWindow().requestFeature(Window.FEATURE_PROGRESS);

        setContentView(R.layout.main);

        getWindow().setFeatureInt(    Window.FEATURE_PROGRESS,
Window.PROGRESS_VISIBILITY_ON);

        webview = (WebView) findViewById(R.id.web_engine);

        webview.setWebViewClient(new HelloWebViewClient());

        webview.getSettings().setJavaScriptEnabled(true);

        webview.getSettings().setPluginsEnabled(true); //depriciated

        if (Integer.parseInt(Build.VERSION.SDK) >= Build.VERSION_CODES.FROYO)

            webview.getSettings().setPluginState(WebSettings.PluginState.ON);

        webview.getSettings().setAllowFileAccess(true);

        webview.getSettings().setBuiltInZoomControls(true); //multitouch an ipostirizetai

        webview.setScrollBarStyle(WebView.SCROLLBARS_OUTSIDE_OVERLAY);

        webview.loadUrl("http://192.168.10.250"); //o server tou Arduino

        //webview.loadUrl("http://google.com"); //gia debug

        final Activity MyActivity = this;

        webview.setWebChromeClient(new WebChromeClient() {

            public void onProgressChanged(WebView view, int progress)

```

```

{

    //Na svisw tin mpara afou fortwsei to url

    MyActivity.setTitle("Φόρτωση...");

    MyActivity.setProgress(progress * 100); //svisimo

    // Kai na emfanisw to app name

    if(progress == 100)

        MyActivity.setTitle(R.string.app_name);

    }

});

}

//Na krataw ta clicks within the app

private class HelloWebViewClient extends WebViewClient {

    @Override

    public boolean shouldOverrideUrlLoading(WebView view, String url) {

        return false;

    }

}

@Override

public boolean onKeyDown(int keyCode, KeyEvent event) {

    if ((keyCode == KeyEvent.KEYCODE_BACK) && webview.canGoBack()) {

        webview.goBack();

    }

}

```

```
    return true;

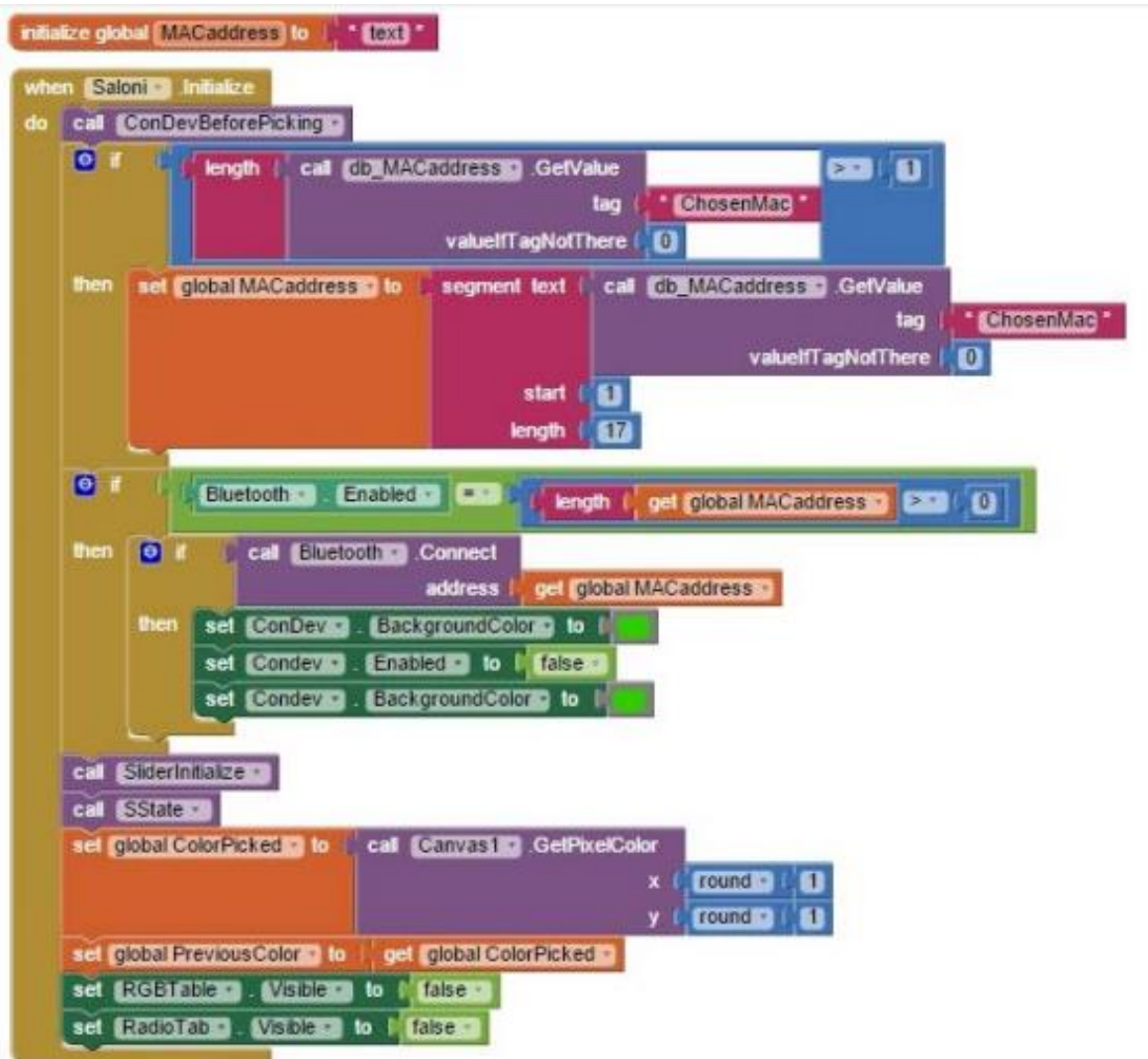
}

return super.onKeyDown(keyCode, event);

}

}
```

Για την κατασκευή της εφαρμογής Smart Home χρησιμοποιήθηκε η πλατφόρμα AppInventor, που με απλό και γρήγορο τρόπο μας επιτρέπει να σχεδιάσου με το interface της εφαρμογής, καθώς επίσης και τον κώδικα που τρέχει από πίσω σε μορφή μπλοκ-διαγράμματος (block diagram) Στην ενότητα που ακολουθεί παρουσιάζονται τα βασικότερα κομμάτια του κώδικα.

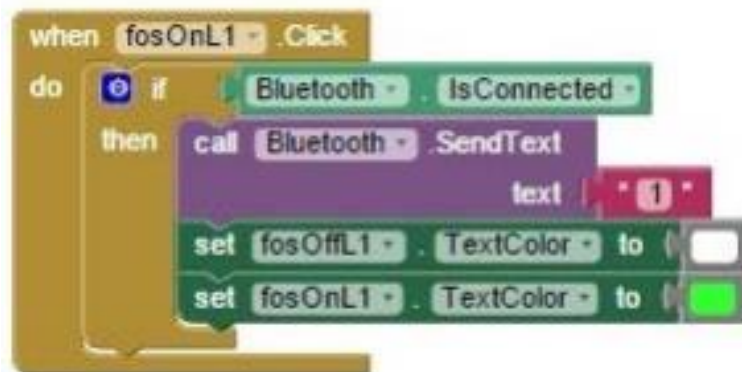


Εικόνα 7. Block Σαλονιού

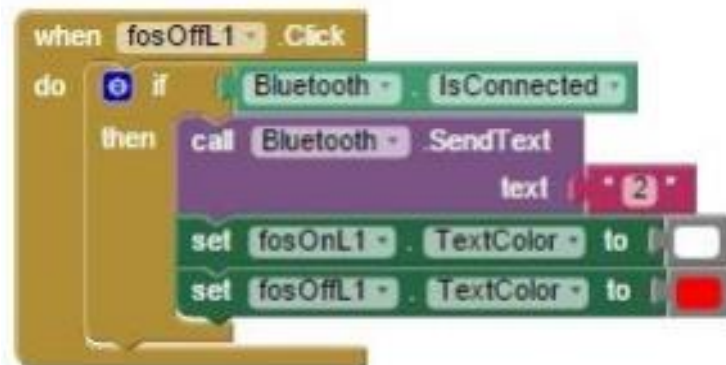
Κατά την είσοδο στο «Σαλόνι» η εφαρμογή αναζητά στην βάση δεδομένων την επιθυμητή MAC του WiFi. Στην συνέχεια ελέγχει αν το WiFi του κινητού συνδέθηκε με το Arduino και πρασινίζει το αντίστοιχο εικονίδιο στο interface, αλλά και αρχικοποιεί όλο το interface της εφαρμογής. Κατά την αρχικοποίηση του Σαλονιού εμφανίζονται τα κείμενα με τα με τα αντίστοιχα κουμπιά (slider buttons, buttons) και αποκρύπτει το interface του RGB πίνακα και του HiFi και εφόσον επιλεγθούν από τον χρήστη τότε θα εμφανιστούν και θα «κρυφτούν» τα μη επιθυμητά κείμενα και κουμπιά.

Για να μπορέσει να ανάψει το φως εκτελούνται οι εξής διαδικασίες: Μόλις πατηθεί το κουμπί on η εφαρμογή ελέγχει αν το WiFi είναι συνδεδεμένο με την πλακέτα εάν ΝΑΙ τότε στέλνει το κείμενο “1” στην πλακέτα του Arduino και τέλος αλλάζει το

χρώμα του κειμένου στο interface στιγμιαία σε γκρι και αμέσως μετά σε πράσινο. Για να μπορέσει να σβήσει το φως εκτελούνται οι εξής διαδικασίες: Μόλις πατηθεί το κουμπί off η εφαρμογή ελέγχει αν το WiFi είναι συνδεδεμένο με την πλακέτα εάν ΝΑΙ τότε στέλνει το κείμενο “2” στην πλακέτα του Arduino και τέλος αλλάζει το χρώμα του κειμένου στο interface στιγμιαία σε γκρι και αμέσως μετά σε κόκκινο.



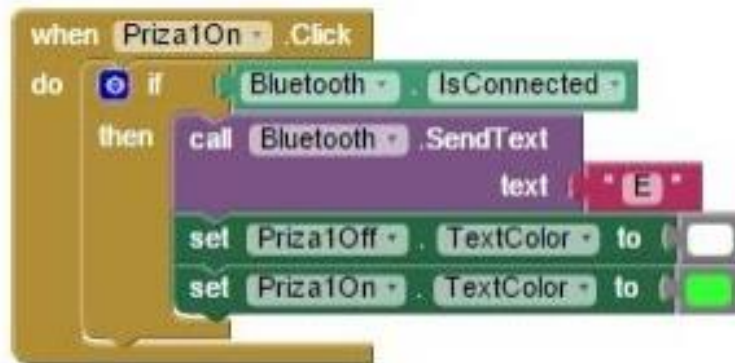
Εικόνα 8. Block ενεργοποίησης διακόπτη



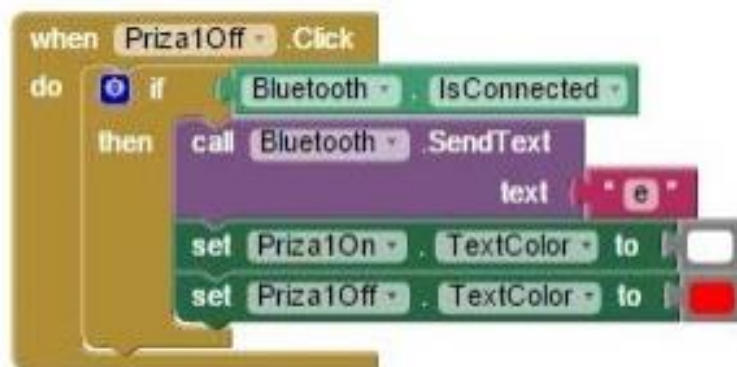
Εικόνα 9. Block Απενεργοποίησης διακόπτη

Για να μπορέσει να ενεργοποιήσει μία πρίζα εκτελούνται οι εξής διαδικασίες: Μόλις πατηθεί το κουμπί on η εφαρμογή ελέγχει αν το WiFi είναι συνδεδεμένο με την πλακέτα εάν ΝΑΙ τότε στέλνει το κείμενο “E” στην πλακέτα του Arduino και τέλος αλλάζει το χρώμα του κειμένου στο interface στιγμιαία σε γκρι και αμέσως μετά σε

πράσινο. Για να μπορέσει να απενεργοποιήσει μία πρίζα εκτελούνται οι εξής διαδικασίες: Μόλις πατηθεί το κουμπί off η εφαρμογή ελέγχει αν το WiFi είναι συνδεδεμένο με την πλακέτα εάν ΝΑΙ τότε στέλνει το κείμενο “e” στην πλακέτα του Arduino και τέλος αλλάζει το χρώμα του κειμένου στο interface στιγμιαία σε γκρι και αμέσως μετά σε κόκκινο.



Εικόνα 10. Block ενεργοποίησης πρίζας



Εικόνα 11. Block απενεργοποίησης πρίζας

Η πλακέτα στην οποία βασιστήκαμε είναι το Arduino για το οποίο χρειαστήκαμε και την Ethernet Shield του.

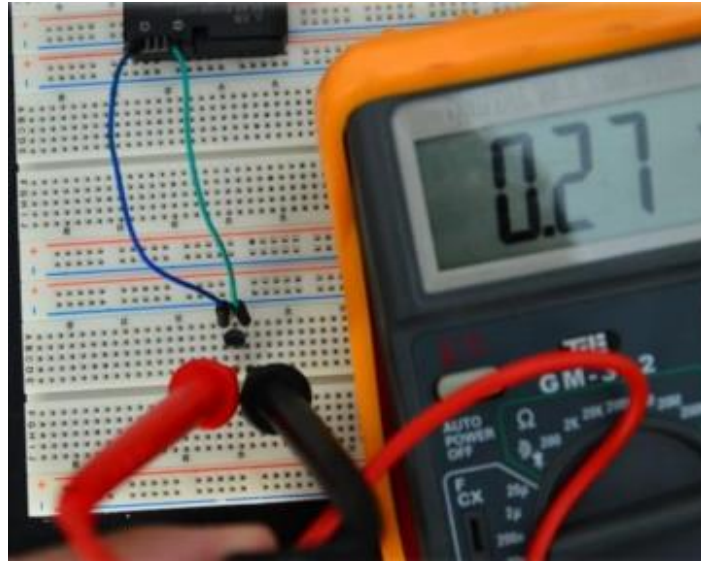
Σε περίπτωση που το Arduino πρέπει να τοποθετηθεί σε σημείο μακριά από το router του σπιτιού (και δεν θέλουμε να τραβάμε καλώδιο ethernet) τότε χρησιμοποιούμε ένα A/P το οποίο ρυθμίζουμε σε Client Mode.

Σε κάθε συσκευή (ή πρίζα/διακόπτη) που θέλουμε να μπορούμε να ελέγχουμε από το Android, θα χρειαστεί να τοποθετήσουμε ένα Relay, όπου ενδεικτικά χρησιμοποιήσαμε αυτό. Φυσικά βεβαιωθείτε ότι η αντοχή σε ρεύμα του Relay που θα επιλέξετε υπερκαλύπτει τις ανάγκες της συσκευής σε ρεύμα.

Λοιπά πράγματα που χρησιμοποιήσαμε: breadboard, jumper wires, αισθητήρας θερμοκρασίας LM35, photocells, λοιποί αισθητήρες, αντιστάσεις, μονωτική, πολύμετρο, κατσαβίδια κτλ.

Στην πιο απλή μορφή του Project, (όπου απλά ελέγχουμε 4 συσκευές/φωτιστικά από το Android), θα μας κοστίσει 23.37 (Arduino)+ 35.60 (Ethernet Shield)+ 37.72 (4x9.43) (relays)+ 10 (λοιπά, καλώδια, μονωτική ταινία κτλ)=106.69 ευρώ.

Όλη η διαδικασία είναι χωρισμένη σε επιμέρους τμήματα, αφενός για να είναι πιο εύκολο να στηθεί η τελική πλατφόρμα και αφετέρου για να είναι ευκολότερη η διαχείριση. Έτσι λοιπόν το πρώτο στάδιο είναι να γίνουν οι απαραίτητες ενέργειες όσον αφορά το hardware για να συνδεθεί ο ελεγκτής με τον αισθητήρα θερμότητας και ακολούθως να συγγραφεί ο απαραίτητος κώδικας για την καταγραφή θερμότητας στο χώρο που έχει εγκατασταθεί ο αισθητήρας. Ο LM35 είναι ένας αισθητήρας θερμοκρασίας, ο οποίος έχει 3 pins, ένα V(in), ένα V(out), και ένα Ground.

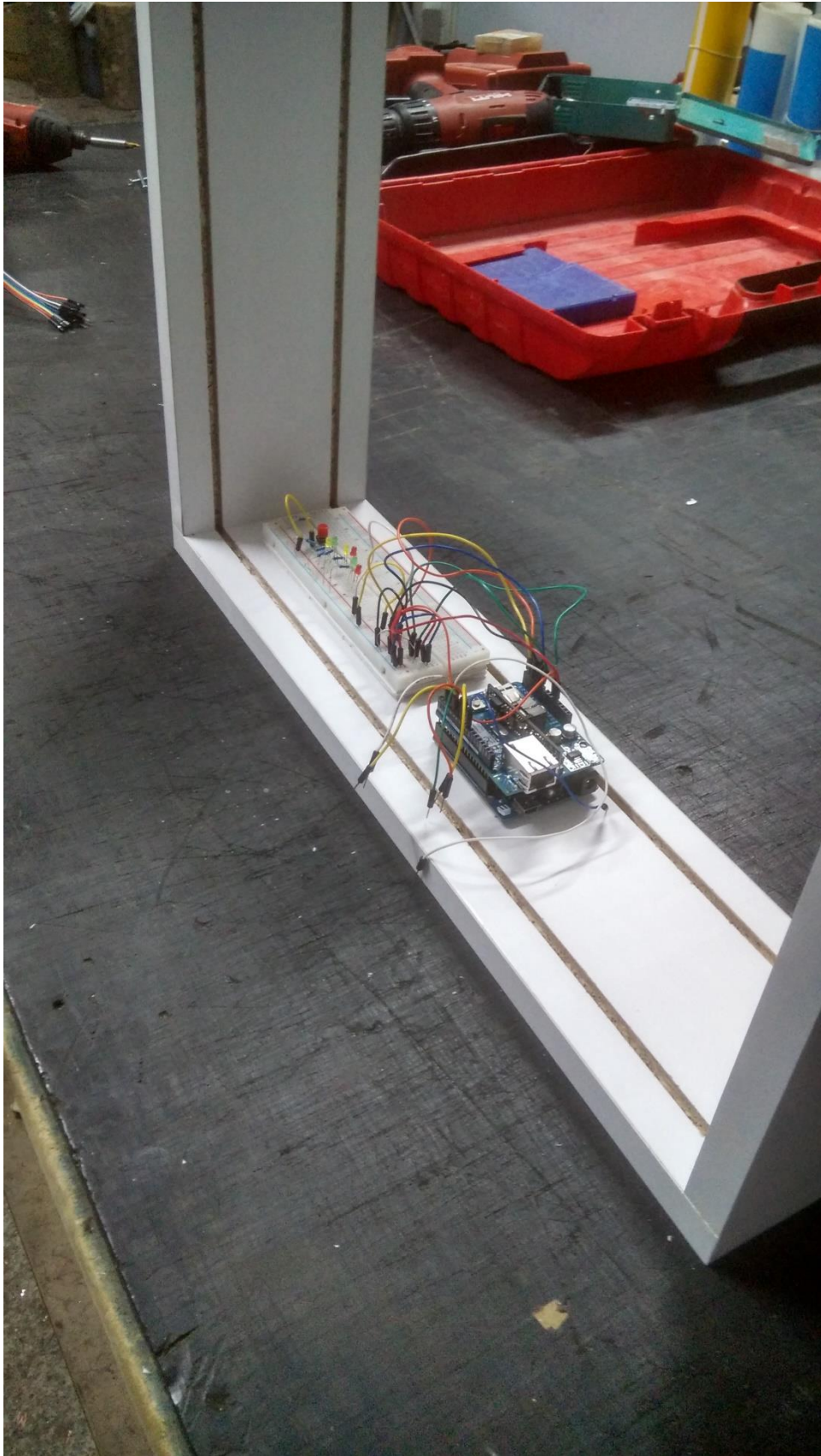


Εικόνα 14. Μετρήσεις

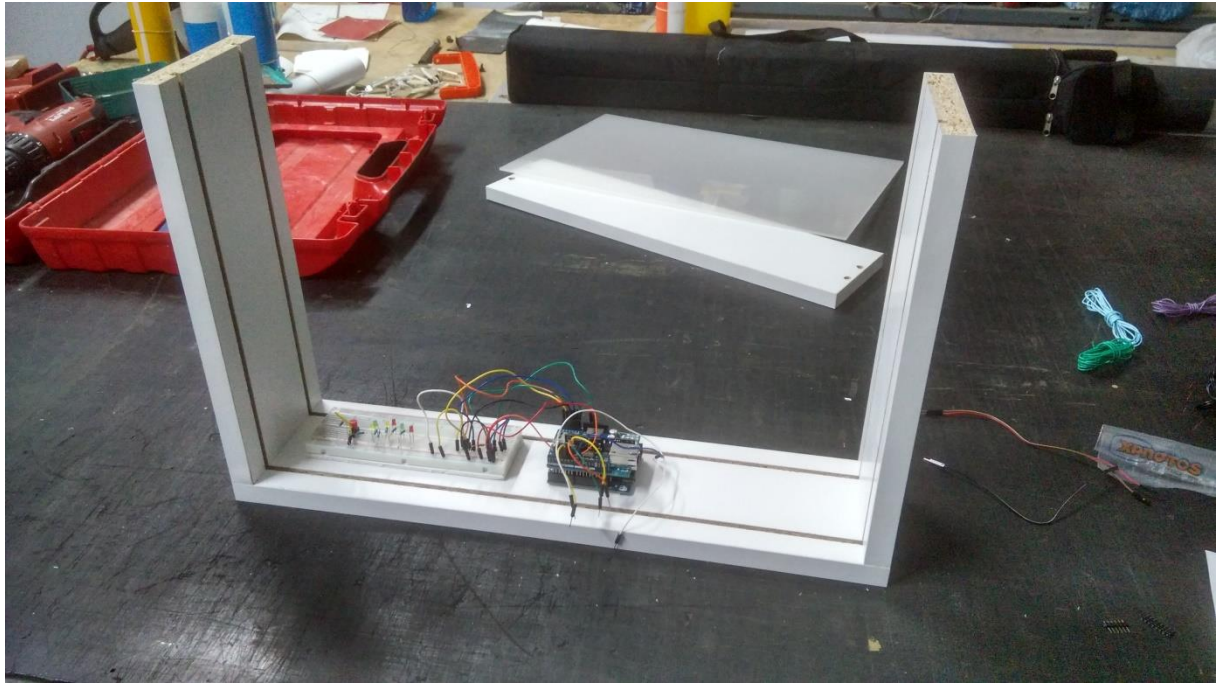
Για να συνδεθεί ο αισθητήρας που φαίνεται παραπάνω (LM35) στο Arduino, λαμβάνουμε την τάση εισόδου από την πλακέτα και την τάση εξόδου στέλνουμε πίσω σε αυτήν. Δίνονται 5V στα ακριανά pins του LM35 (μπλε και πράσινο καλώδιο στην εικόνα 20), και παίρνεται η έξοδος από το μεσαίο pin, η οποία οδηγείται στο Analog Input 0 του Arduino.

Ο λόγος που προγραμματίστηκε ο ελεγκτής να κάνει καταγραφή της θερμοκρασίας είναι κατά πρώτον για να επιτρέπεται να του ορίζεται πότε θα ανοίγει ο κλιματισμός και κατά δεύτερον για να ενεργοποιείται το σύστημα πυρόσβεσης σε περίπτωση που υπάρξει φωτιά. Πειραματικά προστέθηκαν στον παραπάνω κώδικα οι απαραίτητες εντολές έτσι ώστε όταν η θερμοκρασία ξεπεράσει τους 30 βαθμούς Κελσίου να ηχεί ένα buzzer, το οποίο συνδέθηκε στο digital pin 4 και στο ground.

Σε πρακτική εφαρμογή φυσικά θα αντικατασταθεί με ένα relay ισχύος το οποίο θα ενεργοποιεί και θα απενεργοποιεί αντίστοιχα τον κλιματισμό.



Εικόνα 15. Εφαρμογή της πλακέτας



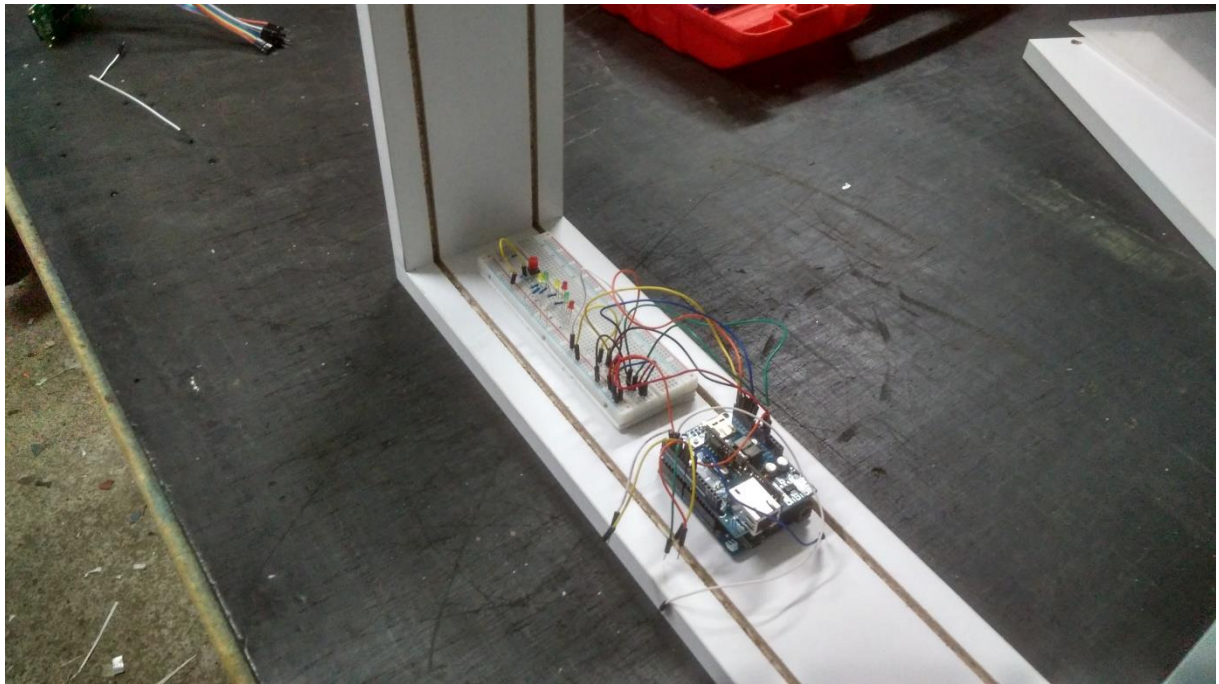
Εικόνα 16. Εικόνα πλακέτας και συνδέσεων

Ακολούθως, προστίθεται η χρήση ενός ακόμα αισθητήρα, και συγκεκριμένα ενός photoresistor που μετράει το φως σε ένα δωμάτιο.



Εικόνα 17. Εξαρτήματα (Αντιστάσεις, Φωτοαντιστάσεις και LED)

Σε αυτό το σημείο είναι επιθυμητό να φτιάξουμε ένα σύστημα, το οποίο θα μετρά την φωτεινή ένταση σε ένα δωμάτιο. Αν η τιμή της πέσει κάτω από κάποιο όριο που οριστεί, θα ενεργοποιεί αυτόματα τον φωτισμό του δωματίου (τεχνητό ή φυσικό). Στην πιλοτική πειραματική διάταξη ενεργοποιείται ένα led, ενώ στην πλήρη ανάπτυξη θα στέλνεται trigger 5V σε ένα ac relay ώστε να ανάβει τα φώτα σε ένα δωμάτιο ή να ανοίγει τα καλύμματα των παραθύρων ή θα κλείνει τα σκίαστρα.



Εικόνα 18. Εικόνα συνδέσεων

Το τελευταίο βήμα που απομένει για τον ελεγκτή είναι να επικοινωνεί με το router ώστε να είναι εφικτή η διαχείριση των δυνατοτήτων που έχουν εγκατασταθεί μέσω της εφαρμογής που θα αναπτυχθεί. Έτσι λοιπόν, επιτρέπεται να δημιουργήσουμε ένα HTTP server μέσω ενός shield που παρέχει η κατασκευάστρια εταιρεία, στο οποίο θα παρέχονται οι πληροφορίες από τους αισθητήρες. Συνδέεται το Arduino στον δρομολογητή με καλώδιο Ethernet, στήνεται ο WebServer, και προγραμματίζεται το Arduino να δείχνει την ένδειξη από τον αισθητήρα θερμότητας. Πλέον, ανοίγοντας έναν οποιονδήποτε περιηγητή είναι εφικτή η πρόσβαση στην IP διεύθυνση που ορίστηκε στο Arduino, για να ελεγχθεί η ένδειξη της θερμοκρασίας.

Το Arduino με την Ethernet shield πρέπει να συνδεθεί στο router του σπιτιού με καλώδιο Ethernet. Στην περίπτωση όμως που κάτι τέτοιο δεν είναι επιθυμητό, υπάρχει η δυνατότητα να χρησιμοποιηθεί ένα A/P σε client mode, ώστε το Arduino να αποκτήσει πρόσβαση στο δίκτυο από "απόσταση". Φυσικά πριν συνδεθεί ασύρματα το A/P σε client mode πρέπει να ληφθούν μέτρα για την ασφάλεια του ασύρματου δικτύου, καθώς δεν είναι επιθυμητό ο καθένας να έχει πρόσβαση στο αναπτυχθέν σύστημα.

Το Arduino τροφοδοτείται από την θύρα USB του υπολογιστή κατά τη διάρκεια του προγραμματισμού του. Η χρήση του όμως (αφού προγραμματιστεί) θα πρέπει να είναι ανεξάρτητη από τον υπολογιστή. Συνεπώς πρέπει να πραγματοποιηθεί η τροφοδοσία του με διαφορετικό τρόπο. Η μία επιλογή είναι να χρησιμοποιηθεί η κλασσική διεπαφή που υπάρχει σχεδόν σε όλα τα κινητά, η οποία δίνει σύνδεση USB female. Εναλλακτικά μπορεί να χρησιμοποιηθεί τροφοδοτικό DC 12V, με βύσμα 2.1mm. Όποια και από τις δύο εναλλακτικές και αν επιλεγεί, θα πρέπει να είναι ικανή να δώσει ρεύμα τουλάχιστον 500mA έως 1A, καθότι στην συνέχεια πρέπει να τροφοδοτηθούν διάφορα ηλεκτρονικά κυκλώματα από το Arduino.



Εικόνα 19. Τροφοδοτικά

Η σύνδεση των relay με τις συσκευές που κάνουν χρήση 220V AC είναι μια απλή συνδεσμολογία αλλά απαιτεί μεγάλη προσοχή δεδομένου ότι μια λανθασμένη σύνδεση μπορεί να προκαλέσει τραυματισμό ή να δημιουργήσει πυρκαγιά από κάποιο βραχυκύκλωμα. Το relay συνδέεται με τρία καλώδια στο Arduino. Το μαύρο καλώδιο συνδέεται στο PIN GND, το πράσινο στο PIN τροφοδοσίας +5V, ενώ το

κόκκινο στο PIN 6 το οποίο θα λειτουργεί ως trigger. Όταν ορίζεται η κατάσταση του PIN6 ως "HIGH", το οποίο δίνει +5V, το relay κλείνει το κύκλωμα, άρα αρχίζει να λειτουργεί η συσκευή. Αντιθέτως ορίζοντας την κατάστασή του ως "LOW", το relay ανοίγει το κύκλωμα και η συσκευή απενεργοποιείται.

Για να συνδεθεί μια συσκευή που διαθέτει διακόπτη ακολουθείται η εξής διαδικασία:

- Αρχικά αποσυνδέεται η συσκευή από την τροφοδοσία.
- Στην συνέχεια ανοίγεται η θήκη του διακόπτη και αφαιρείται ο μηχανισμός.
- Έπειτα για συγκεκριμένη τοποθέτηση του φως, εντοπίζεται το καλώδιο φάσης και το neutral.
- Ακολούθως πρέπει να "παρακαμφθεί" το καλώδιο φάσης, το οποίο συνδέεται στο relay. Το καλώδιο που "έρχεται" από την πρίζα, συνδέεται στο "NO", ενώ αυτό που πηγαίνει στην συσκευή συνδέεται στο COM.

ΣΥΜΠΕΡΑΣΜΑΤΑ

Ο αρχικός στόχος που τέθηκε κατά την υλοποίηση του συστήματος επιτεύχθηκε. Υλοποιήθηκε, δηλαδή, ένα σύστημα οικιακού αυτοματισμού πλήρως λειτουργικό. Ο τελικός χρήστης, έχοντας πρόσβαση στο internet, έχει τη δυνατότητα να ελέγχει οικιακές συσκευές. Επιπλέον, το σύστημα είναι πλήρως επεκτάσιμο. Για τη συγκεκριμένη εφαρμογή χρησιμοποιήθηκε σαν συσκευή προς έλεγχο ένα φωτιστικό χώρου. Με τον ίδιο τρόπο θα μπορούσε να ελεγχθεί οποιαδήποτε οικιακή συσκευή που συνδέεται στα 220V του δικτύου της Δ.Ε.Η, δηλαδή, σε μια οικιακή πρίζα. Οι δυνατότητες του συστήματος βέβαια δεν περιορίζονται στον έλεγχο μόνο μίας συσκευής, αφού ο μικροελεγκτής διαθέτει πλήθος γραμμών εισόδου/εξόδου, τόσο ψηφιακών όσο και αναλογικών. Οι αναλογικές εισοδοί μπορούν να χρησιμοποιηθούν είτε σαν ψηφιακές I/O, οπότε έτσι μπορούν να ελεγχθούν και επιπλέον συσκευές, είτε σαν εισοδοί αισθητήρων αναλογικών μεγεθών που θα επιτρέπουν στο χρήστη να ελέγχει αναλογικά μεγέθη όπως πίεση, υγρασία, φωτεινότητα κ.τ.λ. Το σύστημα, συνεπώς, μπορεί να τροποποιηθεί και να ικανοποιήσει διαφορετικές ανάγκες χρηστών.

Η τροποποίηση αυτή περιλαμβάνει τρία στάδια. Το στάδιο της αλλαγής της συνδεσμολογίας του μικροελεγκτή με τις συσκευές προς έλεγχο. Το στάδιο της τροποποίησης του σχεδίου (sketch) που θα φορτωθεί στον μικροελεγκτή, καθώς και το στάδιο της τροποποίησης των τελικών εφαρμογών του χρήστη. Στο σημείο αυτό θα πρέπει να επισημανθεί ότι οι κώδικες που έχουν γραφεί σίγουρα δεν είναι βέλτιστοι οπότε κάποιος μπορεί να τους τροποποιήσει και να έχει καλύτερα αποτελέσματα. Επιπλέον, θα πρέπει να επισημανθεί και να εξεταστεί το θέμα της ασφάλειας. Η ασφάλεια της επικοινωνίας δεν αποτέλεσε αντικείμενο της παρούσας υλοποίησης. Σίγουρα ένας ειδικός στην ασφάλεια των επικοινωνιών θα βρει τρωτά σημεία στον τομέα της επικοινωνίας, όπως για παράδειγμα το ότι το password ταξιδεύει «φανερά» στη σύνδεση με πιθανό κίνδυνο υποκλοπής.

Θα πρέπει παρόλα αυτά να τονιστεί ότι οι προαναφερθείσες βελτιώσεις και επεκτάσεις δεν αυξάνουν (παρά μόνο ελάχιστα ίσως) το κόστος του συστήματος, το οποίο δεν ξεπέρασε τα 70€, χρησιμοποιώντας παντού καινούρια υλικά. Αυτός ήταν και ένας επιπλέον στόχος της παρούσας διπλωματικής, η ανάπτυξη δηλαδή ενός

πλήρως λειτουργικού συστήματος οικιακού αυτοματισμού με το μικρότερο δυνατό κόστος. Συνοψίζοντας, η βελτίωση των κωδίκων που έχουν γραφεί (π.χ. μέσω της ανάπτυξης ενός ολοκληρωμένου πρωτοκόλλου επικοινωνίας των συσκευών του τελικού χρήστη με τον μικροελεγκτή), καθώς και το θέμα της ασφάλειας της επικοινωνίας αποτελούν τα σημεία μελλοντικής ανάπτυξης του συστήματος που υλοποιήθηκε. Έχοντας ως βάση τον επιτυχημένο απομακρυσμένο έλεγχο συσκευών που παρέχει το συγκεκριμένο σύστημα, μπορεί πλέον κάποιος να το επεκτείνει, να το βελτιώσει και σίγουρα να το κάνει πιο ασφαλές με απώτερο στόχο τη λειτουργικότητα, τη βελτίωση της καθημερινότητας του ανθρώπου και τελικά την πρόοδο της επιστήμης.

ΠΑΡΑΡΤΗΜΑ: BLOCKS ΣΤΟ APPINVENTOR

```

when ConDev AfterPicking
do
  call db_MACAddress StoreValue
  tag ChosenMac
  valueToStore ConDev Selection
  if call Bluetooth Connect
  address ConDev Selection
  then
    set ConDev BackgroundColor to
    set ConDev Enabled to false
    set ConDev BackgroundColor to
  
```

```

to ConDevBeforePicking
do
  if Bluetooth IsConnected
  then call Bluetooth Disconnect
  else if not Bluetooth Enabled
  then call as_TurnOnBT StartActivity
  else set ConDev Elements to Bluetooth AddressesAndNames
  
```

```

when ClockSaloni Timer
do
  if not Bluetooth Enabled or not Bluetooth IsConnected
  then
    set ConDev BackgroundColor to
    set ConDev Enabled to false
    set ConDev BackgroundColor to
  else if Bluetooth IsConnected
  then
    if call Bluetooth BytesAvailableToReceive > 0
    then
      set LblTunePresetFreq Text to
      call Bluetooth ReceiveText
      numberOfBytes call Bluetooth BytesAvailableToReceive
    
```

```

initialize global value_dimmer1 to
when SliderS1 PositionChanged
thumbPosition
do
  set global value_dimmer1 to round get thumbPosition
  set sL1 Text to get global value_dimmer1
  if Bluetooth IsConnected
  then
    call Bluetooth SendText
    text get global value_dimmer1
    call Bluetooth SendText1
  call SdbState StoreValue
  tag 1 Slider Position
  valueToStore round SliderS1 ThumbPosition
  call SdbState StoreValue
  tag sL1%
  valueToStore sL1 Text
  
```

```

initialize global value_dimmer2 to
when SliderS2 PositionChanged
thumbPosition
do
  set global value_dimmer2 to round get thumbPosition
  set sL2 Text to get global value_dimmer2
  if Bluetooth IsConnected
  then
    call Bluetooth SendText
    text get global value_dimmer2
    call Bluetooth SendText1
  call SdbState StoreValue
  tag 2 Slider Position
  valueToStore round SliderS2 ThumbPosition
  call SdbState StoreValue
  tag sL2%
  valueToStore sL2 Text
  
```

```
when green flag clicked  
do  
  [ ]  
  replace all text with [ ]  
  replace all text with [ ]  
  replace all text with [ ]  
  replace all text with [ ]  
  replace all text with [ ]  
  replace all text with [ ]  
  replace all text with [ ]  
  replace all text with [ ]  
  replace all text with [ ]  
  replace all text with [ ]  
  get global ColorPicked  
  set global ColorPicked to [ ]
```

```
when Canvas touch down  
do  
  set X to [ ]  
  set Y to [ ]  
  set global ColorPicked to [ ]  
  call [ ]  
  set ColorCode to [ ]  
  call [ ]
```

```
when Canvas drag  
start: x, y, color  
current: x, y, color  
do  
  set [ ] to [ ]  
  set [ ] to [ ]  
  set global ColorPicked to [ ]  
  set ColorCode to [ ]  
  call [ ]
```

```
when Bluetooth click  
do  
  call [ ]  
  set [ ] visible to [ ]  
  set [ ] visible to [ ]  
  set [ ] background color to [ ]  
  set [ ] visible to [ ]  
  set [ ] visible to [ ]  
  set [ ] visible to [ ]  
  set [ ] visible to [ ]  
  set [ ] visible to [ ]  
  set [ ] visible to [ ]  
  set [ ] visible to [ ]  
  set [ ] visible to [ ]  
  set [ ] visible to [ ]  
  set [ ] visible to [ ]  
  set [ ] visible to [ ]  
  set [ ] visible to [ ]  
  set [ ] visible to [ ]
```

```
when Bluetooth click  
do  
  call [ ]  
  set [ ] visible to [ ]  
  set [ ] visible to [ ]  
  set [ ] visible to [ ]  
  set [ ] visible to [ ]  
  set [ ] visible to [ ]  
  set [ ] visible to [ ]  
  set [ ] visible to [ ]  
  set [ ] visible to [ ]  
  set [ ] visible to [ ]  
  set [ ] visible to [ ]  
  set [ ] visible to [ ]  
  set [ ] visible to [ ]  
  set [ ] visible to [ ]  
  set [ ] visible to [ ]  
  set [ ] visible to [ ]
```

```
when back click  
do  
  set [ ] visible to [ ]  
  set [ ] visible to [ ]  
  set [ ] visible to [ ]  
  set [ ] visible to [ ]  
  set [ ] visible to [ ]  
  set [ ] visible to [ ]  
  set [ ] visible to [ ]  
  set [ ] visible to [ ]  
  set [ ] visible to [ ]  
  set [ ] visible to [ ]  
  set [ ] visible to [ ]  
  set [ ] visible to [ ]  
  set [ ] visible to [ ]  
  set [ ] visible to [ ]
```

```

when HiFi On Click
do
  call FRadio1 ReadFrom
  fileName Station.txt
  set LblTunePresetFreq Text to call DBRadio GetValue
  tag Station
  valueIfTagNotThere

  call PRadio ReadFrom
  fileName tempStation.txt
  set LblTunePresetFreq Text to call DBRadio GetValue
  tag tempStation
  valueIfTagNotThere

  set HiFi On TextColor to
  set HiFi Off TextColor to
  set LblTable Visible to false
  set RGBTable Visible to false
  set RadioTab Visible to true
  set Salon BackgroundColor to
  set Light1 Visible to false
  set Light2 Visible to false
  set SpracTable Visible to false
  set DimmerTable Visible to false

  when FRadio GetText
  text
  do set LblTunePresetFreq Text to get text

  when FRadio1 GetText
  text
  do set LblTunePresetFreq Text to get text

```

```

when back1 Click
do
  set LblTable Visible to true
  set RGBTable Visible to false
  set Light1 Visible to true
  set Light2 Visible to true
  set SpracTable Visible to true
  set DimmerTable Visible to true
  set RadioTab Visible to false

  call DBRadio StoreValue
  tag tempStation
  valueToStore LblTunePresetFreq Text

  call FRadio SaveFile
  text LblTunePresetFreq Text
  fileName tempStation.txt

  call DBRadio StoreValue
  tag Station
  valueToStore LblTunePresetFreq Text

  call FRadio1 SaveFile
  text LblTunePresetFreq Text
  fileName Station.txt

```

BIBΛΙΟΓΡΑΦΙΑ

1. Brock, J. D., Bruce, R. F., & Reiser, S. L. (2009). Using Arduino for introductory programming courses. *Journal of Computing Sciences in Colleges*, 25(2), 129-130.
2. Kato, Y. (2010). Splish: a visual programming environment for Arduino to accelerate physical computing experiences. In *Creating Connecting and Collaborating through Computing (C5)*, 2010 Eighth International Conference on (pp. 3-10). IEEE.
3. Evans, B. (2011). *Beginning Arduino Programming*. Apress.
4. Noble, J. (2009). *Programming Interactivity: A Designer's Guide to Processing, Arduino, and Openframeworks*. " O'Reilly Media, Inc."
5. Buechley, L., & Eisenberg, M. (2008). The LilyPad Arduino: Toward wearable engineering for everyone. *Pervasive Computing, IEEE*, 7(2), 12-15.
6. Monk, S. (2012). *Programming Arduino*. United States of America: McGraw-Hill Companies.
7. Booth, T., & Stumpf, S. (2013). End-user experiences of visual and textual programming environments for Arduino. In *End-User Development* (pp. 25-39). Springer Berlin Heidelberg.
8. Buechley, L., Eisenberg, M., Catchen, J., & Crockett, A. (2008). The LilyPad Arduino: using computational textiles to investigate engagement, aesthetics, and diversity in computer science education. In *Proceedings of the SIGCHI conference on Human factors in computing systems* (pp. 423-432). ACM.
9. Wolber, D. (2011). App inventor and real-world motivation. In *Proceedings of the 42nd ACM technical symposium on Computer science education* (pp. 601-606). ACM.

10. Wolber, D., Abelson, H., Spertus, E., & Looney, L. (2011). App Inventor. " O'Reilly Media, Inc."
11. Gray, J., Abelson, H., Wolber, D., & Friend, M. (2012, March). Teaching CS principles with app inventor. In Proceedings of the 50th Annual Southeast Regional Conference (pp. 405-406). ACM.
12. Morelli, R., De Lanerolle, T., Lake, P., Limardo, N., Tamotsu, E., & Uche, C. (2011). Can android app inventor bring computational thinking to k-12. In Proc. 42nd ACM technical symposium on Computer science education (SIGCSE'11) (pp. 1-6).
13. Hsu, Y. C., Rice, K., & Dawley, L. (2012). Empowering educators with Google's Android App Inventor: An online workshop in mobile app design. *British Journal of Educational Technology*, 43(1)
14. EIKONA ΕΞΩΦΥΛΛΟΥ: <http://aftodioikisi.gr/diethni/erxetai-to-exipno-spiti-tha-kanei-kafe-tha-psinei-kai-tha-mas-xipna/>

