



ΠΑΝΕΠΙΣΤΗΜΙΟ  
ΠΑΤΡΩΝ  
UNIVERSITY OF PATRAS

ΣΧΟΛΗ ΟΙΚΟΝΟΜΙΚΩΝ ΕΠΙΣΤΗΜΩΝ ΚΑΙ ΔΙΟΙΚΗΣΗΣ ΕΠΙΧΕΙΡΗΣΕΩΝ

ΤΜΗΜΑ ΔΙΟΙΚΗΤΙΚΗΣ ΕΠΙΣΤΗΜΗΣ ΚΑΙ ΤΕΧΝΟΛΟΓΙΑΣ

## ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

ΕΞΥΠΝΟ ΣΠΙΤΙ ΜΕ ΤΗ ΧΡΗΣΗ ΜΙΚΡΟΕΛΕΓΚΤΗ  
ARDUINO

ΚΑΔΟΥΛΑΣ ΧΡΙΣΤΟΦΟΡΟΣ  
ΣΚΟΥΠΙΑΣ ΒΑΣΙΛΕΙΟΣ

Πάτρα 2021

Επιβλέπων καθηγητής  
Δημήτρης Φ. Παπαδόπουλος

**UNIVERSITY OF PATRAS**

SCHOOL OF ECONOMICS & BUSINESS

DEPARTMENT OF MANAGEMENT SCIENCE AND  
TECHNOLOGY

**FORMER DEPARTMENT OF BUSINESS  
ADMINISTRATION AT PATRAS**

**THESIS**

**SMART HOUSE WITH THE USE OF  
ARDUINO MICROCONTROLLER**

KADOULAS CHRISTOFOROS

SKOUPAS VASILIOS

PATRAS 2021

Η έγκριση της πτυχιακής εργασίας από το Τμήμα Διοικητικής Επιστήμης και Τεχνολογίας του Πανεπιστημίου Πατρών δεν υποδηλώνει απαραίτητως και αποδοχή των απόψεων του συγγραφέα εκ μέρους του Τμήματος.

## ΠΕΡΙΛΗΨΗ

Σκοπός της εργασίας αυτής είναι η κατασκευή ενός έξυπνου σπιτιού με την χρήση μικροελεγκτών Arduino. Η παρούσα πτυχιακή χωρίζεται σε τρία μέρη.

Στο πρώτο μέρος, το οποίο αποτελεί το γενικό μέρος της πτυχιακής αναφέρεται στην ιστορία των μικροελεγκτών Arduino και την αλληλεπίδραση τους μέσω του λογισμικού Arduino IDE.

Στο δεύτερο μέρος, θα δούμε τον σχεδιασμό και την περιγραφή των περιφερειακών υλικών που έχουμε χρησιμοποιήσει προκειμένου να υλοποιηθεί το πρακτικό μέρος της εργασίας.

Το τρίτο και τελευταίο μέρος, αποτελείται από τον κώδικα της εφαρμογής, με τον οποίο "μιλήσαμε" με τον μικροελεγκτή Arduino UNO.

## **SUMMARY**

The purpose of this work is the construction of a smart house using Arduino microcontrols. The present thesis is divided into three parts.

In the first part, which is the general part of the thesis, it refers to the history of arduino microcontrolists and their interaction through the Arduino IDE software.

In the second part, we will look at the design and description of the regional materials that we have used in order to implement the practical part of the work.

The third and final part consists of the code of the application, with which we "talked" with the microcontroler Arduino UNO.

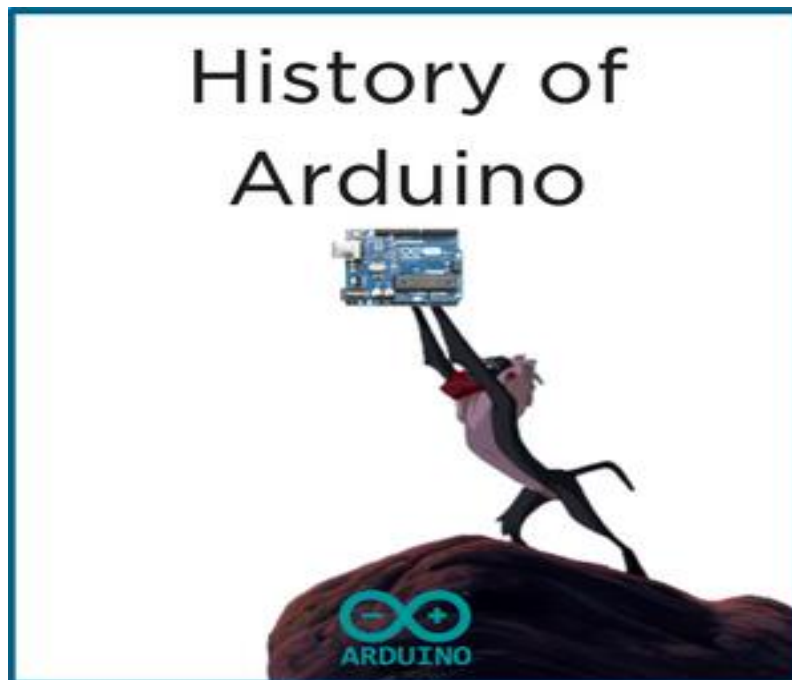
## ΠΙΝΑΚΑΣ ΠΕΡΙΕΧΟΜΕΝΩΝ

<b>ΠΕΡΙΛΗΨΗ.....</b>	<b>1</b>
<b>SUMMARY .....</b>	<b>2</b>
<b>1.ΙΣΤΟΡΙΑ ΤΟΥ ARDUINO .....</b>	<b>4</b>
<b>1.2.ΤΙ ΕΙΝΑΙ ΤΟ ARDUINO .....</b>	<b>5</b>
<b>1.3 ΜΟΝΤΕΛΑ ARDUINO .....</b>	<b>6</b>
<b>1.4 Η ΠΛΑΤΦΟΡΜΑ ARDUINO .....</b>	<b>10</b>
<b>1.5 ΓΛΩΣΣΑ WIRING .....</b>	<b>11</b>
<b>1.6 ΛΟΓΙΣΜΙΚΟ IDE ARDUINO.....</b>	<b>12</b>
<b>2.ΣΧΕΔΙΑΣΜΟΣ ΕΝΟΣ ΕΞΥΠΝΟΥ ΣΠΙΤΙΟΥ ΜΕ ΤΗΝ ΧΡΗΣΗ ΜΙΚΡΟΕΛΕΓΧΤΩΝ ARDUINO .....</b>	<b>15</b>
<b>2.1 ΑΝΑΛΥΣΗ ΣΥΣΤΗΜΑΤΟΣ.....</b>	<b>15</b>
<b>2.2 ΠΕΡΙΓΡΑΦΗ ΥΛΙΚΩΝ .....</b>	<b>16</b>
<b>2.3 ΚΑΤΑΣΚΕΥΗ ΜΑΚΕΤΑΣ ΚΑΙ ΣΥΝΔΕΣΗ ΜΕ ΑΙΣΘΗΤΗΡΕΣ .....</b>	<b>21</b>
<b>3. OUTPUT &amp; ΚΩΔΙΚΑΣ ΕΦΑΡΜΟΓΗΣ.....</b>	<b>31</b>
<b>3.1 OUTPUT.....</b>	<b>31</b>
<b>3.2 ΚΩΔΙΚΑΣ ΕΦΑΡΜΟΓΗΣ .....</b>	<b>32</b>
<b>4.ΣΥΜΠΕΡΑΣΜΑΤΑ .....</b>	<b>38</b>
<b>5.ΒΙΒΛΙΟΓΡΑΦΙΑ .....</b>	<b>40</b>

## 1.ΙΣΤΟΡΙΑ ΤΟΥ ARDUINO

Τα τελευταία χρόνια έχει αναπτυχθεί σε μεγάλο βαθμό ο τομέας της τεχνολογίας. Διάφοροι σπουδαίοι επιστήμονες ,νέοι ταλαντούχοι φοιτητές έχουν αφιερώσει την ζωή τους, ώστε η ζωή μας μέρα με την μέρα να γίνεται καλύτερη και ευκολότερη. Μια ομάδα το 2005 ξεκίνησε ένα project ,προκειμένου να κατασκευαστεί μια συσκευή για τον έλεγχο προγραμμάτων διαδραστικών σχεδίων από μαθητές, η οποία θα ήταν φθηνότερη από άλλα πρωτότυπα συστήματα διαθέσιμα εκείνη την περίοδο.

Οι ιδρυτές MassimoBanzi και DavidCueartielles, ονόμασαν το σχέδιο αυτό arduino (Arduin of Ivrea ) που μεταφράζεται ελεύθερα ως «γενναίος φίλος» και ξεκίνησαν να παράγουν πλακέτες σε ένα εργοστάσιο στην Ιβρέα ,μια κωμόπολη κοντά στο Τορίνο της Ιταλίας.Το σχέδιο arduino είναι μια διακλάδωση της πλατφόρμας Wiring για λογισμικό ανοικτού κώδικα και προγραμματίζεται χρησιμοποιώντας μια γλώσσα βασισμένη στο Wiring, παρόμοια με την γλώσσα προγραμματισμού C++ με απλοποιήσεις και αλλαγές, καθώς και ένα ολοκληρωμένο περιβάλλον ανάπτυξης. Τα αποτελέσματα ήταν άμεσα και το Σεπτέμβριο του 2006 εκδόθηκε το ArduinoMini. Στη σημερινή εποχή πολλές ευκολίες βασίζονται στον Arduino.



## 1.2.ΤΙ ΕΙΝΑΙ ΤΟ Arduino

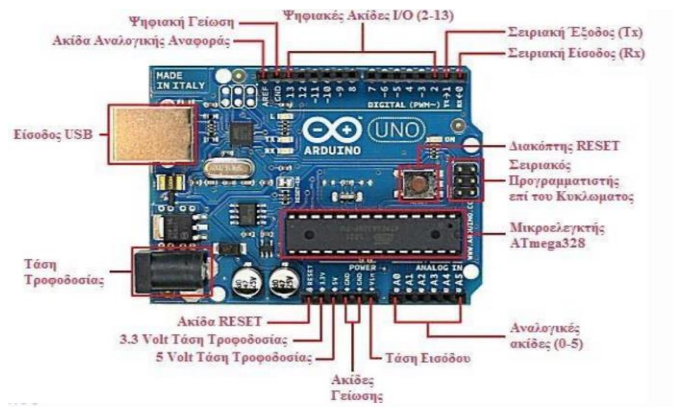
Το Arduino είναι ικανό να διαβάσει καταστάσεις από το περιβάλλον χρησιμοποιώντας διαφόρους αισθητήρες αλλά και να δράσει σε αυτό με την βοήθεια κατάλληλων εξαρτημάτων. Είναι μια πλατφόρμα ανοικτού κώδικα και η αρχιτεκτονική του, το πώς είναι σχεδιασμένο δηλαδή, καθιστά την χρήση και την τροποποίηση του φιλική και εύκολη για τον οποιονδήποτε. Μπορεί να χρησιμοποιηθεί για την ανάπτυξη ανεξαρτήτων διαδραστικών αντικειμένων αλλά και να συνδεθεί με υπολογιστή μέσω προγραμμάτων σε Processing, Max/MSP, PureData, Super Collider .Οι περισσότερες εκδόσεις του Arduino μπορούν να αγοραστούν αποσυναρμολογημένες

Είναι χρήσιμο να αναφερθεί ότι, το διάγραμμα και πληροφορίες για το υλικό είναι ελεύθερα και διαθέσιμα στην αγορά. Arduino λοιπόν, καλείται ένας μικροελεγκτής μονής πλακέτας,δηλαδή μια απλή μητρική πλατφόρμα ανοικτού κώδικα με ενσωματωμένο μικροελεγκτή και εισόδους – εξόδους, η οποία μπορεί να προγραμματιστεί με την γλώσσα Wiring. Χαρακτηριστικά παραδείγματα τα οποία λειτουργούν με Arduino είναι τα φανάρια στους δρόμους, συστήματα που ελέγχουν θερμοκρασία, κίνηση, υγρασία τα drones κ.α.



### 1.3 ΜΟΝΤΕΛΑ Arduino

Η πιο δημοφιλής πλατφόρμα Arduino είναι το Arduino Uno που χρησιμοποιεί την πλατφόρμα ATmega328P. Όλοι οι άλλοι μικροελεγκτές arduino βασίζονται πάνω στην ίδια φιλοσοφία. Παρακάτω απεικονίζονται κάποιοι από αυτούς καθώς και αναλυτικά τα εξαρτήματα του arduino uno.



- Arduino UNO



- Arduino LilyPadUsb



- Arduino Ethernet



- Arduino Leonard



- Arduino Mega 2560



- Arduino Mega ADK



- REV3



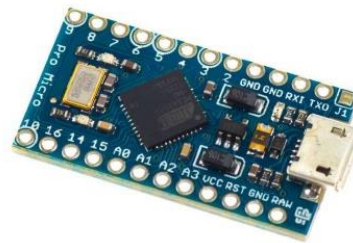
- Arduino Frio



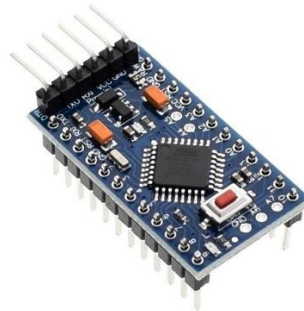
- Arduino Nano



- Arduino Pro



- Arduino Pro Mini



- Arduino Micro





- Arduino Due



- Arduino Esplora

## 1.4 Η ΠΛΑΤΦΟΡΜΑ ARDUINO

Όπως αναφέρθηκε παραπάνω το Arduino λειτουργεί με την γλώσσα προγραμματισμού Wiring βασισμένη στην γλώσσα προγραμματισμού c++. Έτσι ο προγραμματισμός είναι φιλικότερος ευκολότερος και επιτρέπει την γρήγορη ανάπτυξη εφαρμογών ανεξάρτητα από την λεπτομερή γνώση της αρχιτεκτονικής του μικροελεγκτή. Η διαφορά του με τον προγραμματισμό ενός οποιουδήποτε άλλου μικροελεγκτή είναι ότι, δεν προϋποθέτει την ύπαρξη είτε στοιχειωδών κυκλωμάτων, είτε συγκεκριμένων διασυνδέσεων με ένα ειδικό προγραμματιστή ή με ένα υπολογιστικό σύστημα. Οτιδήποτε είναι απαραίτητο για τον προγραμματισμό του μικροελεγκτή βρίσκεται επάνω στην κάρτα, στην οποία είναι τοποθετημένος. Το πρόγραμμα αναπτύσσεται στον υπολογιστή, ενώ η αλλαγή του εκτελέσιμου κώδικα γίνεται μέσω της θύρας USB.

Είναι άξιο αναφοράς ότι όλοι οι μικροελεγκτές Arduino είναι εφοδιασμένοι με ένα ειδικό λογισμικό, που λέγεται bootloader πράγμα που δίνει την δυνατότητα του επαναπρογραμματισμού. Παρακάτω απεικονίζεται ένα σχήμα για να κατανοήσουμε καλύτερα την διαφορά ενός μικροελεγκτή Arduino με ένα οποιοδήποτε άλλο μικροελεγκτή.

ΕΠΙΠΕΔΟ 3		Wiring c
ΕΠΙΠΕΔΟ 2	Συναρμολόγηση	Συναρμολόγηση
ΕΠΙΠΕΔΟ 1	Γλώσσα μηχανής	Γλώσσα Μηχανής
ΕΠΙΠΕΔΟ 0	Μικροελεγκτής (Υλικό)	Μικροελεγκτής (Υλικό)
	Κλασικός Προγραμματισμός Μικροελεγκτη	Προγραμματισμός ARDUINO

## 1.5 ΓΛΩΣΣΑ WIRING

Εφόσον το Arduino λειτουργεί με την γλώσσα Wiring για την ανάπτυξη και το προγραμματισμό του θα χρησιμοποιήσουμε τις ίδιες εντολές, τους ίδιους τύπους δεδομένων, συναρτήσεις, σύνταξη και ίδιους τελεστές όπως παρατηρείται και στη γλώσσα C.

Με βάση την ιστοσελίδα του arduino <https://www.ardumotive.com> ξεκινώντας να προγραμματίζω χωρίζω το πρόγραμμα σε τρία μέρη: την δομή, μεταβλητές και σταθερές και τις συναρτήσεις. Είναι σημαντικό να δηλώσουμε τις μεταβλητές, να γράψουμε το κομμάτι κώδικα που περιέχει την αρχικοποίηση καταστάσεων και μεταβλητών, τον κώδικα που θέλουμε να τρέξει μια φορά στο πρόγραμμα καθώς και το κώδικα loop. Ο κώδικας loop ουσιαστικά είναι το κύριο πρόγραμμα που θα τρέχει συνέχεια μέχρι να βγάλουμε το Arduino από το ρεύμα. Οι μεταβλητές μπορούν να πάρουν διάφορες τιμές νούμερα ή και τα δύο ή να έχουν λογική τιμή True False. Αναλόγως με την τιμή αυτή την αρχικοποιούμε ή την δηλώνουμε αντίστοιχα στο πρώτο τμήμα της δομής τους προγράμματος. Κάθε όνομα που δίνουμε σε μια μεταβλητή θα πρέπει να υπάρχει μόνο μια φορά στο πρόγραμμα μας. Όπως στις μεταβλητές έτσι και στις σταθερές η διαδικασία γίνεται με τον ίδιο τρόπο απλά πριν το τύπο βάζουμε το χαρακτηριστικό const.

## 1.6 ΛΟΓΙΣΜΙΚΟ IDE ARDUINO

Το Arduino IDE αποτελεί το περιβάλλον ανάπτυξης του κώδικα για τον μικροελεγκτή κάθε πλακέτας του Arduino. Το ολοκληρωμένο περιβάλλον ανάπτυξης IDE είναι μία εφαρμογή γραμμένη σε Java και μπορεί να χρησιμοποιηθεί σε πολλές εφαρμογές. Ένα IDE περιλαμβάνει έναν επεξεργαστή πηγαίου κώδικα, έναν μεταγλωττιστή, εργαλεία αυτόματης παραγωγής κώδικα, αποσφαλματωτή, συνδέτη, σύστημα ελέγχου εκδόσεων και εργαλεία κατασκευής γραφικών διασυνδέσεων χρήστη για τις υπό ανάπτυξη εφαρμογές. Πριν γράψουμε οποιοδήποτε πρόγραμμα λοιπόν θα πρέπει να επιλέξουμε την έκδοση arduino, καθώς και να συνδέσουμε, να επιλέξουμε δηλαδή την σωστή θύρα επικοινωνίας.

Παρακάτω απεικονίζονται τα κύρια εργαλεία και η περιγραφή λειτουργίας τους.



**Verify:** Εφόσον τελειώσουμε με το πρόγραμμα ελέγχει για τυχόν συντακτικά λάθη



**Upload:** μεταγλωττίζει τον κώδικα και φορτώνει το πρόγραμμα εφόσον ο κώδικας είναι σωστός



**New:** Δημιουργεί ένα νέο sketch



**Open:** Παραθέτει ένα menu με όλα τα scetch.Ενεργοποιώντας ένα από αυτά θα ανοίξει αυτόματα σε νέο παράθυρο.



**Save:** Αποθηκεύει ένα scetch



```
boolean flag = false;

void setup() {
  Serial.begin(9600);
  |
  pinMode(inPin, INPUT);    // sets the digital pin 7 as input

  // ultrasonic
  pinMode(trigPin, OUTPUT); // Sets the trigPin as an OUTPUT
  pinMode(echoPin, INPUT); // Sets the echoPin as an INPUT

  //end ultrasonic

  //LIVING ROOM
  pinMode(led_pin1, OUTPUT);
  pinMode(led_pin2, OUTPUT);
  pinMode(led_pin3, OUTPUT);
  // end living room
  pinMode(buzzer, OUTPUT);
  digitalWrite(buzzer,LOW);

  // light sensor
  pinMode(led_light1, OUTPUT);
  pinMode(led_light2, OUTPUT);
  // end light sensor

}
#include <dht.h>

dht DHT;
```

---



## **2.ΣΧΕΛΙΑΣΜΟΣ ΕΝΟΣ ΕΞΥΠΝΟΥ ΣΠΙΤΙΟΥ ΜΕ ΤΗΝ ΧΡΗΣΗ ΜΙΚΡΟΕΛΕΓΧΤΩΝ ARDUINO**

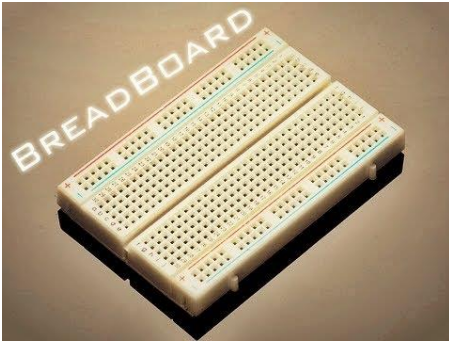
### **2.1 ΑΝΑΛΥΣΗ ΣΥΣΤΗΜΑΤΟΣ**

Σε αυτήν την εργασία θα ασχοληθούμε με ένα αποδοτικό έξυπνο οικιακό σύστημα. Αυτό το σύστημα έχει δύο βασικές ενότητες: τη μονάδα διασύνδεσης υλικού και τη μονάδα επικοινωνίας λογισμικού. Στην καρδιά αυτού του συστήματος είναι ο μικροελεγκτής Arduino UNO, ο οποίος είναι επίσης ικανός να λειτουργεί ως μικροελεγκτής και τη διασύνδεση για όλες τις μονάδες υλικού. Όλες οι επικοινωνίες και τα χειριστήρια σε αυτό το σύστημα περνούν από τον μικροελεγκτή. Το έξυπνο οικιακό σύστημα προσφέρει δυνατότητα παρακολούθησης του περιβάλλοντος χρησιμοποιώντας τους αισθητήρες θερμοκρασίας, υγρασίας.

Επίσης, προσφέρει λειτουργίες μεταγωγής για τον έλεγχο φωτισμού, ανεμιστήρων / κλιματιστικών και άλλων οικιακών συσκευών που είναι συνδεδεμένες στο σύστημα. Ένα άλλο χαρακτηριστικό αυτού του συστήματος είναι η ανίχνευση εισβολής που προσφέρει με τον αισθητήρα κίνησης.

## 2.2 ΠΕΡΙΓΡΑΦΗ ΥΛΙΚΩΝ

### Breadboard



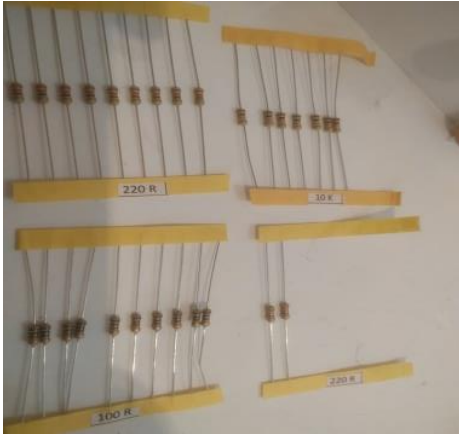
Το Breadboardείναι μια μονάδα για την κατασκευή προσωρινών κυκλωμάτων. Χρησιμοποιήθηκε για την προετοιμασία των αισθητήρων και την αλληλεπίδραση με τον κώδικα.

### LED (LightEmittingDiode)



Πρόκειται για έναν ημιαγωγό, από τον οποίο εκπέμπεται φως, όταν κατά την φορά της ορθής πόλωσης, του διοχετεύεται μια ηλεκτρική τάση.

## Αντιστάτης



Για τον έλεγχο ενός κυκλώματος σχετικά με τη ροή του ρεύματος χρησιμοποιείται ένα εξάρτημα που ονομάζεται αντιστάτης.

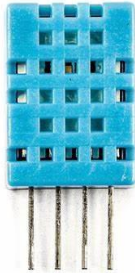
Από το υλικό που είναι κατασκευασμένος ένας αντιστάτης εξαρτάται η τιμή της ηλεκτρικής αντίστασης και αφορά τη δυσκολία που παρατηρείται κατά τη διέλευση του ρεύματος μέσα από αυτό.

## LDR Light-Dependent Resistor



Στον αισθητήρα φωτός επιστρέφεται μια τιμή που αντιπροσωπεύεται από την ένταση του φωτός όταν πέφτει πάνω στην επιφάνειά του. Πρόκειται για έναν αντιστάτη που η τιμή του μεταβάλλεται ανάλογα με τον αριθμό των φωτονίων που του διοχετεύονται. Η τιμή του είναι αντιστρόφως ανάλογη της τιμής της έντασης του φωτός.

## Αισθητήρας θερμοκρασίας και υγρασίας (DHT11 HUMIDITY SENSOR)



© Photo by CafeRobot

Αυτός ο αισθητήρας θερμοκρασίας και υγρασίας DHT11 διαθέτει αισθητήρα θερμοκρασίας και υγρασίας με ένα ψηφιακό σήμα το οποίο βαθμονομείται κατά την έξοδο. Η υψηλή αξιοπιστία και η μακροπρόθεσμη σταθερότητά του εξασφαλίζεται με τη χρήση λήψης ψηφιακού σήματος στο οποίο παρέχεται τεχνολογία ανίχνευσης θερμοκρασίας και υγρασίας. Για την καλύτερη ποιότητα, γρήγορη απόκριση, αντι-παρεμβολές, την ικανότητα της σχέσης κόστους αποτελεσματικότητας περιλαμβάνεται στον αισθητήρα, μέτρηση θερμοκρασίας NTC και μια μέτρηση υγρασίας αντιστατικού τύπου που συνδέεται με έναν μικροελεγκτή υψηλής απόδοσης 8 bit.

Η ποσότητα υδρατμού σε σχέση με το σημείο κορεσμού στον αέρα ονομάζεται σχετική υγρασία. Κατά τον κορεσμό ο υδρατμός αρχίζει να συσσωρεύεται και να συμπυκνώνεται στις επιφάνειες στις οποίες σχηματίζεται δροσιά. Με τη θερμοκρασία του αέρα μεταβάλλεται και το σημείο κορεσμού. Ο ψυχρός αέρας συγκρατεί λιγότερους υδρατμούς προτού κορεστεί ενώ ο θερμός αέρας θα κρατήσει πιο πολλούς υδρατμούς πριν κορεστεί.

Ο αισθητήρας ενδείκνυται για συστήματα περιβαλλοντικού ελέγχου, απομακρυσμένους μετεωρολογικούς σταθμούς, συστήματα παρακολούθησης σπιτιών ή κήπων.

## Ποτενσιόμετρο



Το ποτενσιόμετρο είναι αναλογικό ηλεκτρονικό εξάρτημα, με τρεις ακροδέκτες. Το ποτενσιόμετρο είναι ένας μεταβλητός αντιστάτης που χρησιμοποιείται συνήθως ως διαιρέτης τάσης, για τη μεταβολή δηλαδή του ηλεκτρικού δυναμικού. Μπορεί να λειτουργήσει και ως ροοστάτης αν χρησιμοποιηθούν μόνο οι δύο ακροδέκτες, ο μεσαίος και ένας από τους δύο άλλους ακροδέκτες του.

## UltrasonicSensor HC-SR04



Ο αισθητήρας υπερήχων HC-SR04 (ΗΠΑ) είναι ένας υπερηχητικός μετατροπέας που έρχεται με διεπαφή 4 ακίδων

που ονομάζεται Vcc, Trigger, Echo και Ground. Είναι πολύ χρήσιμο για την ακριβή μέτρηση απόστασης του αντικειμένου-στόχου και λειτουργεί κυρίως στα ηχητικά κύματα. Ο υπερηχητικός αισθητήρας HC-SR04 χρησιμοποιεί σόναρ για τον προσδιορισμό της απόστασης σε ένα αντικείμενο όπως οι νυχτερίδες ή τα δελφίνια. Προσφέρει εξαιρετική ανίχνευση εύρους χωρίς επαφή με υψηλή ακρίβεια και σταθερές ενδείξεις σε μια εύχρηστη συσκευασία. Από 2cm έως 400 cm (ή 1 "έως 13 πόδια).

## BUZZER



Χρησιμοποιούνται ευρέως σε χρονοδιακόπτες, συναγερμοί, ηλεκτρονικά παιχνίδια, υπολογιστές, τηλέφωνα και άλλα προϊόντα όπου απαιτείται η παραγωγή ήχου. Είναι δύο τύπων, ενεργός βομβητής και παθητικός βομβητής. Πριν βρεθούμε στη λεπτομέρεια, θα δούμε πρώτα τη διαφορά μεταξύ των δύο.

### Ενεργός βομβητής:

Ο ενεργός βομβητής θα παράγει μόνο ήχο όταν θα είναι ηλεκτροφόρος. Δημιουργεί ήχο μόνο σε μία συχνότητα. Αυτός ο βομβητής λειτουργεί με ακουστική συχνότητα 2 KHz.

### Παθητικοί βομβητές:

Οι παθητικοί βομβητές χρειάζονται ένα ηχητικό σήμα για να παράγουν έναν τόνο. Αυτό μπορεί να γίνει είτε δίδοντας ένα σήμα pwm στο βομβητή είτε ενεργοποιώντας το ON και το OFF σε διαφορετικές συχνότητες. Μπορεί να παράγει μια σειρά ηχητικών σημάτων ανάλογα με τη συχνότητα εισόδου. Μπορεί να παράγει τόνους συχνοτήτων μεταξύ 1,5 έως 2,5 KHz.

## MC-38 Magnetic Switch Sensor



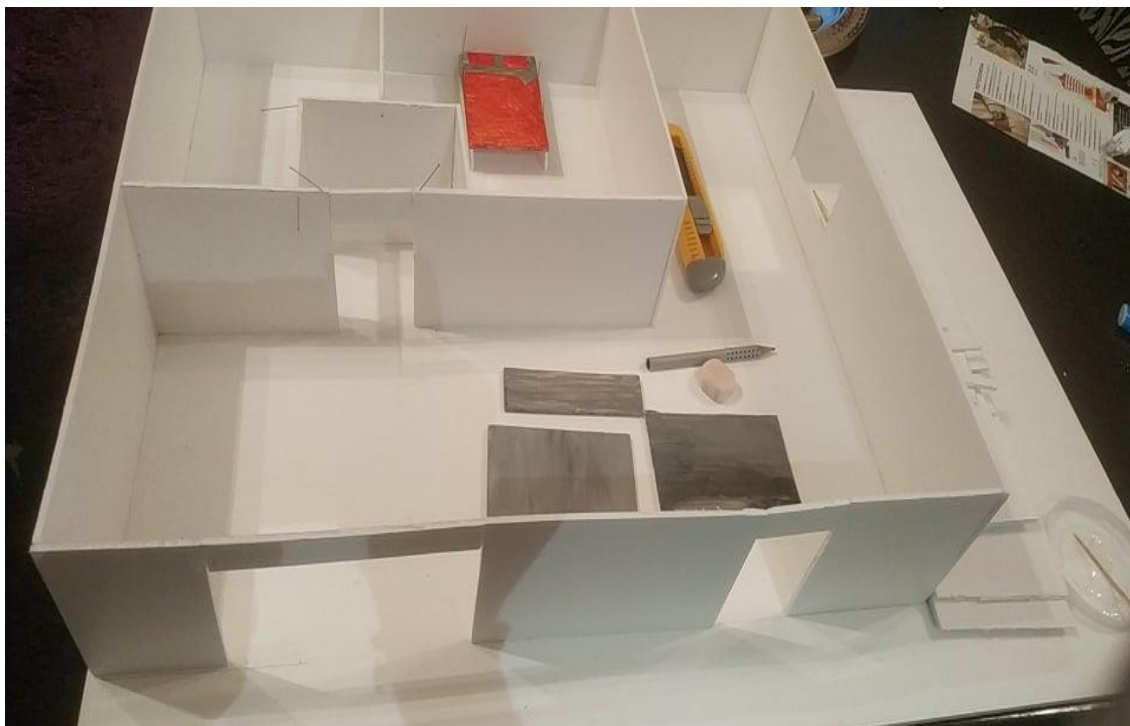
Το MC38 είναι μαγνητικός διακόπτης. Όταν ένας μαγνήτης (το ασύρματο μέρος του αισθητήρα) τον πλησιάζει, ο αισθητήρας λειτουργεί σαν διακόπτης και οι δύο ακίδες είναι συνδεδεμένες.

Αυτός ο αισθητήρας χρησιμοποιείται για την ανίχνευση κλειστών ή ανοιχτών θυρών, παραθύρων και συστημάτων συναγερμού.

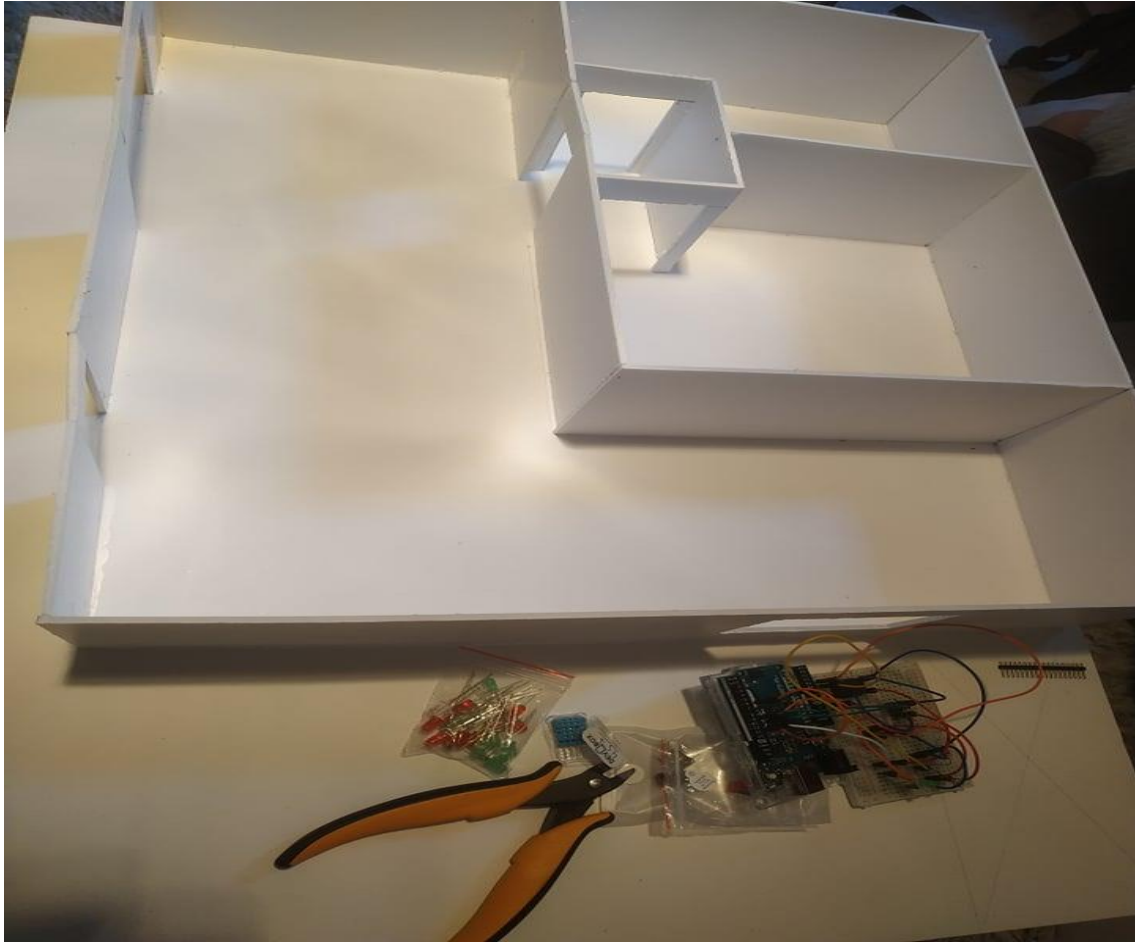
## 2.3 ΚΑΤΑΣΚΕΥΗ ΜΑΚΕΤΑΣ ΚΑΙ ΣΥΝΔΕΣΗ ΜΕ ΑΙΣΘΗΤΗΡΕΣ

Στην ενότητα αυτή θα συναρμολογήσουμε τα υλικά για το έξυπνο σπίτι καθώς και θα συνδέσουμε τους αισθητήρες που αναφέραμε σε προηγούμενη ενότητα με τον μικροελεγκτή Arduino UNO.

Αφού προμηθευτήκαμε τα υλικά για την δημιουργία της κατασκευής όπως: μακετόχαρτο, ειδική κόλλα για μακέτες, σιλικόνες και ειδικό πιστόλι σιλικόνης, χαρτοκόπτη, σχεδιάσαμε αρχικά το σχήμα του σπιτιού πάνω στο μακετόχαρτο και υστέρτα αρχίσαμε να κόβουμε και να κολλάμε τους τοίχους ώστε να δημιουργηθεί ένα πρότυπο σπιτιού σε μικρή κλίμακα όπως παρατηρούμε στις δύο παρακάτω εικόνες 1 και 2.



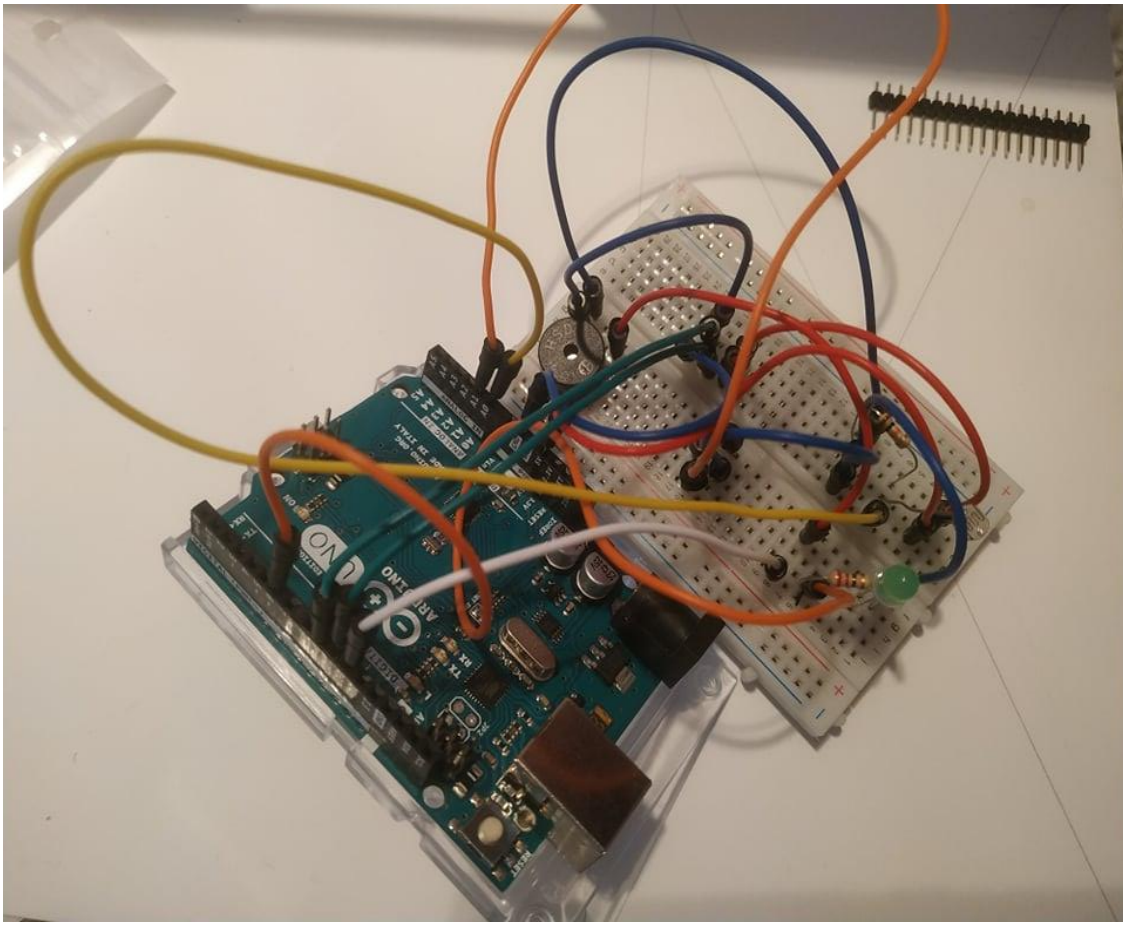
**Εικόνα 1**



**Εικόνα 2**

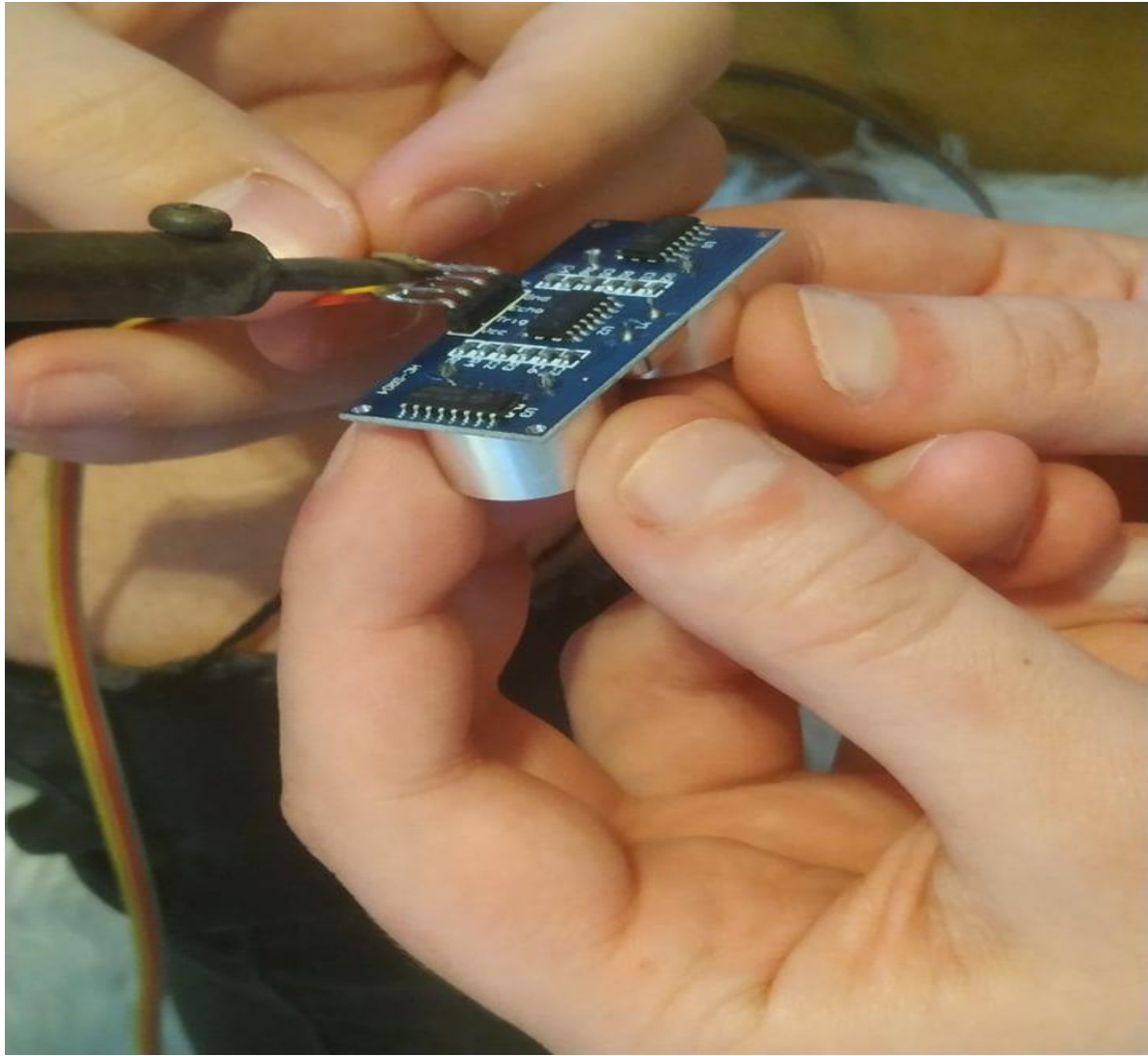


Οι δοκιμές για την εξοικείωσή μας με το λογισμικό του Arduino έγιναν αρχικά με τη χρήση μιας breadboard και με τον μικροελεγκτή Arduino UNO. Χρησιμοποιώντας μια Breadboard κατασκευάσαμε ένα προσωρινό κύκλωμα το οποίο θα μας βοηθήσει για την προετοιμασία των αισθητήρων και την αλληλεπίδρασή τους με τον κώδικα.



**Εικόνα 3**

Στην εικόνα 4 παρατηρούμε τη συγκόλληση των συρμάτων των καλωδίων με τον αισθητήρα υπερήχων. Για την καλύτερη συγκόλληση των καλωδίων με τους αισθητήρες χρησιμοποιήσαμε ένα εργαλείο υψηλής θερμότητας.



**Εικόνα 4**

Στην εικόνα 5 παρατηρούμε την τοποθέτηση ενός ποτενσιόμετρου πάνω στην μακέτα μας.



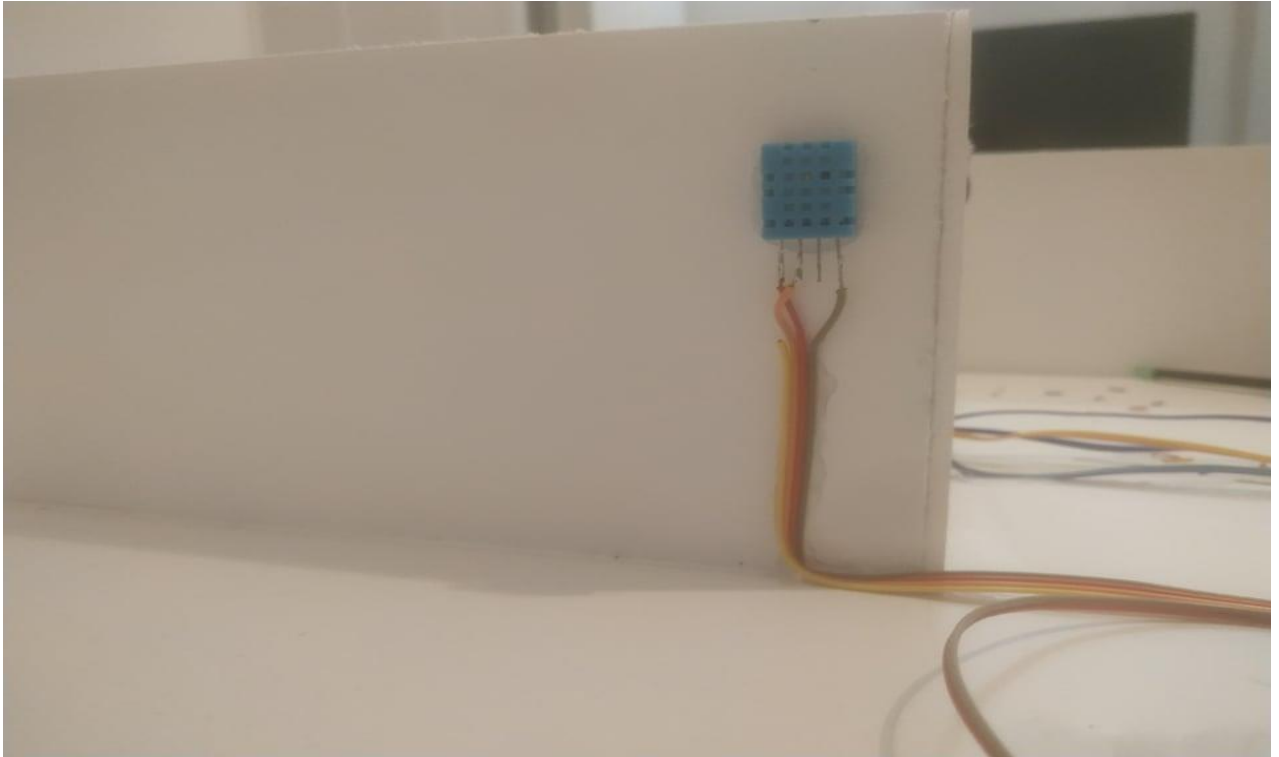
Εικόνα 5

Στην εικόνα 6 πραγματοποιείται η τοποθέτηση του αισθητήρα υπερήχων όπως και την κόλληση του καλωδίου περιμετρικά του σπιτιού μέχρι τον μικροελεγκτή μας.



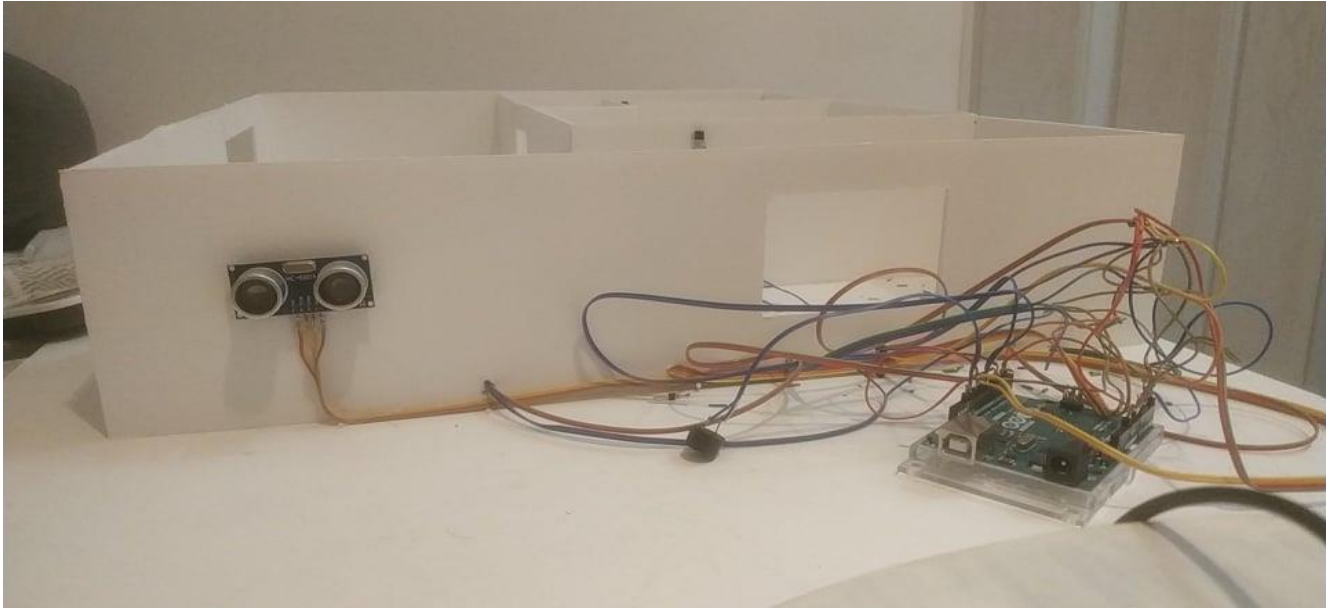
Εικόνα 6

Στην εικόνα 7 βλέπουμε τον αισθητήρα θερμοκρασίας και υγρασίας τον οποίο συνδέουμε χρησιμοποιώντας σιλικόνη για τη στήριξη τόσο των καλωδίων όσο και του αισθητήρα.

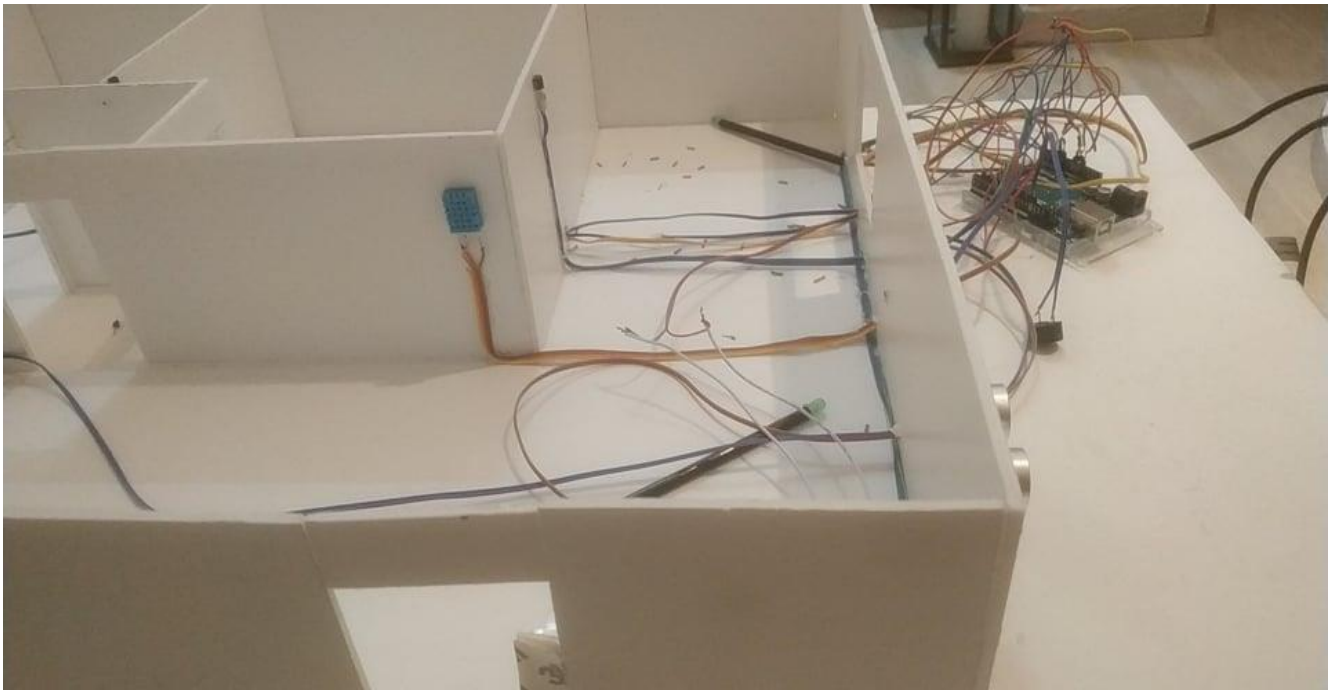


Εικόνα 7

Στις παραπάνω εικόνες 8 και 9 απεικονίζεται το εσωτερικό και το εξωτερικό του έξυπνου σπιτιού με όλους του αισθητήρες και τα Led που χρησιμοποιήθηκαν για την υλοποίηση της κατασκευής.

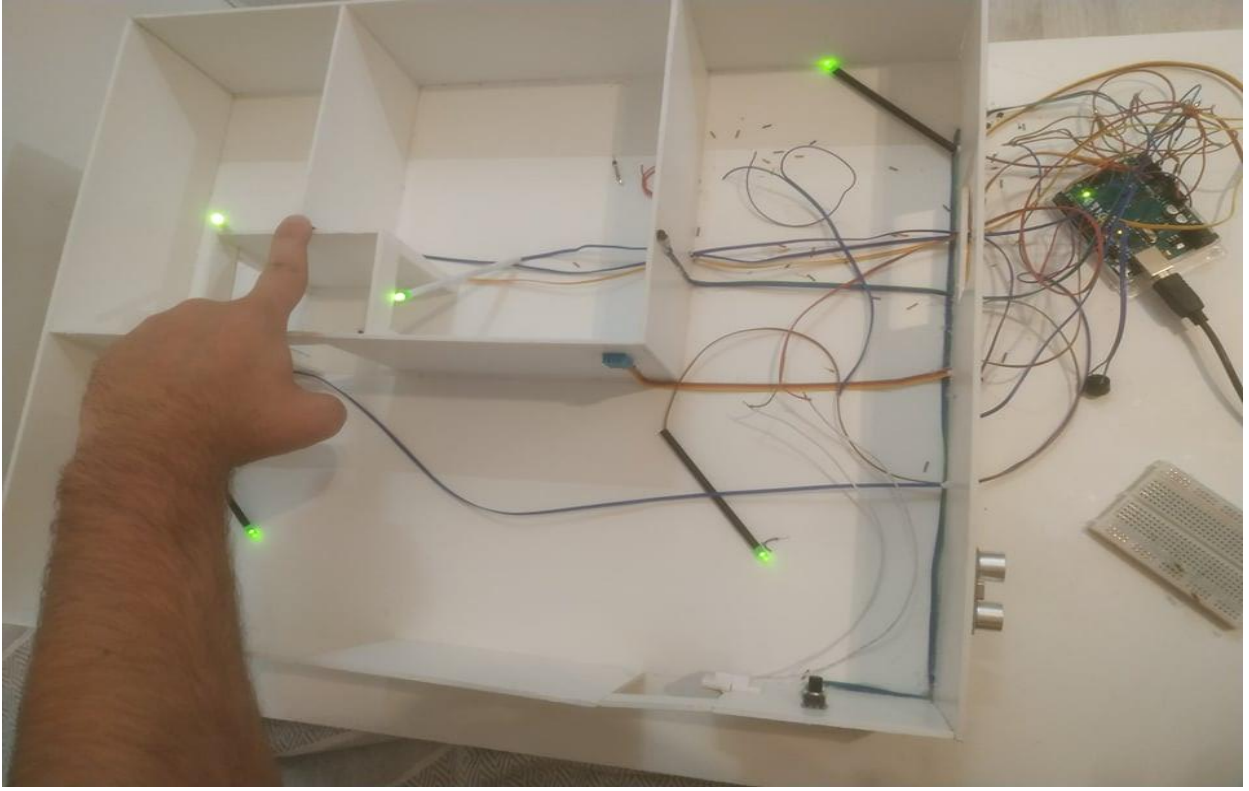


Εικόνα 8



Εικόνα 9

Στην εικόνα 10 παρατηρούμε την αλληλεπίδραση των αισθητήρων μετά τη σύνδεση και τη φόρτωση του κώδικα στην πλακέτα μας, πραγματοποιώντας δοκιμές λειτουργίας μετά τη σύνδεσή τους.



Εικόνα 10

Τέλος, αφού ολοκληρώσαμε τις δοκιμές και παρατηρήσαμε ότι όλοι οι αισθητήρες λειτουργούν σωστά προχωρήσαμε στην ολοκλήρωση της μακέτας όπως φαίνεται και στην εικόνα 11.

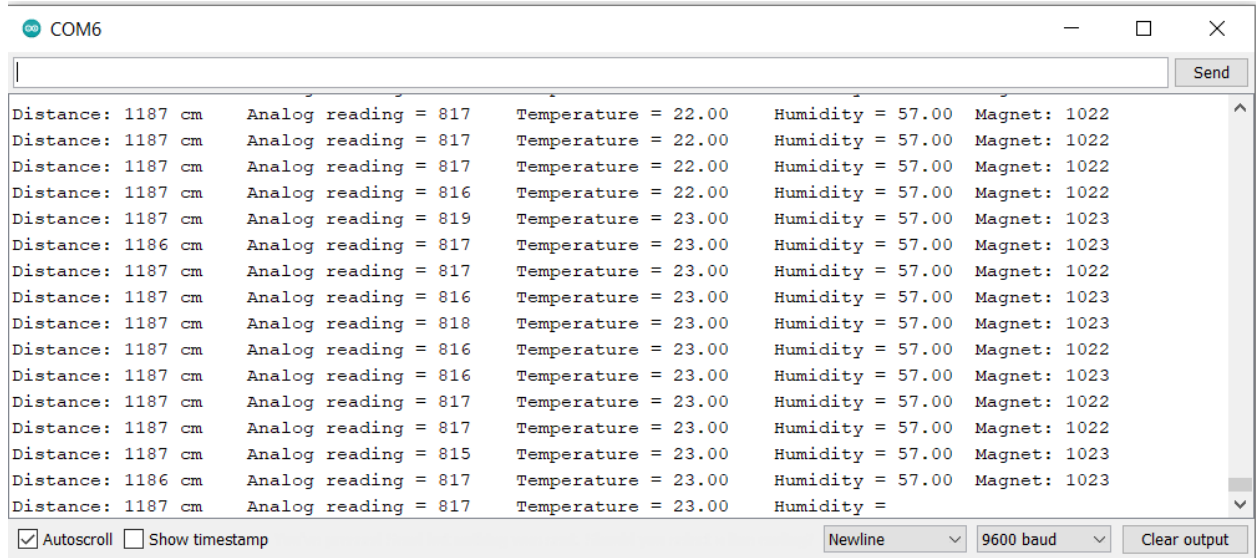


Εικόνα 11



## 3. OUTPUT & ΚΩΔΙΚΑΣ ΕΦΑΡΜΟΓΗΣ

### 3.1 output



```
COM6
Distance: 1187 cm   Analog reading = 817   Temperature = 22.00   Humidity = 57.00   Magnet: 1022
Distance: 1187 cm   Analog reading = 817   Temperature = 22.00   Humidity = 57.00   Magnet: 1022
Distance: 1187 cm   Analog reading = 817   Temperature = 22.00   Humidity = 57.00   Magnet: 1022
Distance: 1187 cm   Analog reading = 816   Temperature = 22.00   Humidity = 57.00   Magnet: 1022
Distance: 1187 cm   Analog reading = 819   Temperature = 23.00   Humidity = 57.00   Magnet: 1023
Distance: 1186 cm   Analog reading = 817   Temperature = 23.00   Humidity = 57.00   Magnet: 1023
Distance: 1187 cm   Analog reading = 817   Temperature = 23.00   Humidity = 57.00   Magnet: 1022
Distance: 1187 cm   Analog reading = 816   Temperature = 23.00   Humidity = 57.00   Magnet: 1023
Distance: 1187 cm   Analog reading = 818   Temperature = 23.00   Humidity = 57.00   Magnet: 1023
Distance: 1187 cm   Analog reading = 816   Temperature = 23.00   Humidity = 57.00   Magnet: 1022
Distance: 1187 cm   Analog reading = 816   Temperature = 23.00   Humidity = 57.00   Magnet: 1023
Distance: 1187 cm   Analog reading = 817   Temperature = 23.00   Humidity = 57.00   Magnet: 1022
Distance: 1187 cm   Analog reading = 817   Temperature = 23.00   Humidity = 57.00   Magnet: 1022
Distance: 1187 cm   Analog reading = 815   Temperature = 23.00   Humidity = 57.00   Magnet: 1023
Distance: 1186 cm   Analog reading = 817   Temperature = 23.00   Humidity = 57.00   Magnet: 1023
Distance: 1187 cm   Analog reading = 817   Temperature = 23.00   Humidity =
```

Serial monitor:

- Distance
- Analog reading
- Temperature
- Humidity
- Magnet

Ο παραπάνω πίνακας αφορά τα αποτελέσματα που εξάγονται σε πραγματικό χρόνο από όλους τους αισθητήρες ταυτόχρονα.

### 3.2 ΚΩΔΙΚΑΣ ΕΦΑΡΜΟΓΗΣ

```
// DHT11
#include <dht.h>
dht DHT;
#define DHT11_PIN 2
// END DHT11

// magnetic

int inPin = A5;
int magnet_VAL = 0; // variable to store the read value

// end magnetic

// Ultrasonic
#define echoPin 11 // attach pin D2 Arduino to pin Echo of HC-SR04
#define trigPin 10 //attach pin D3 Arduino to pin Trig of HC-SR04

// defines variables
long duration; // variable for the duration of sound wave travel
int distance; // variable for the distance measurement
// end ultasonic

//LIVING ROOM
```

```
int led_pin1 = 6;
int led_pin2 = 5;
int led_pin3 = 3;
int pot_pin = A0;
int buzzer = 13;
int output;
int led_value;
// END LIVING ROOM

// light sensor
int led_light1 = 8;
int led_light2 = 9;
// end light sensor

boolean flag = false;

void setup() {
  Serial.begin(9600);

  pinMode(inPin, INPUT); // sets the digital pin 7 as input

  // ultrasonic
  pinMode(trigPin, OUTPUT); // Sets the trigPin as an OUTPUT
  pinMode(echoPin, INPUT); // Sets the echoPin as an INPUT

  //end ultrasonic
```

```

//LIVING ROOM
pinMode(led_pin1, OUTPUT);
pinMode(led_pin2, OUTPUT);
pinMode(led_pin3, OUTPUT);
  // end living room
pinMode(buzzer, OUTPUT);
digitalWrite(buzzer,LOW);

// light sensor
pinMode(led_light1, OUTPUT);
pinMode(led_light2, OUTPUT);
// end light sensor

}
void loop() {

  ##### magnetic loop #####

magnet_VAL = analogRead(inPin); // read the input pin

  ##### end magnetic loop #####

##### ultrasonic #####
  // Clears the trigPin condition

```

```

digitalWrite(trigPin, LOW);
delayMicroseconds(2);
// Sets the trigPin HIGH (ACTIVE) for 10 microseconds
digitalWrite(trigPin, HIGH);
delayMicroseconds(10);
digitalWrite(trigPin, LOW);
// Reads the echoPin, returns the sound wave travel time in microseconds
duration = pulseIn(echoPin, HIGH);
// Calculating the distance
distance = duration * 0.034 / 2; // Speed of sound wave divided by 2 (go and back)
// Displays the distance on the Serial Monitor
Serial.print("Distance: ");
Serial.print(distance);
Serial.print(" cm");

```

```

##### ultrasonic #####

```

```

##### light sensor #####

```

```

int analogValuelight = analogRead(A1);
Serial.print(" Analog reading = ");
Serial.print(analogValuelight); // the raw analog reading

```

```

if(analogValuelight<= 100)
{
digitalWrite(led_light1,HIGH);
digitalWrite(led_light2,HIGH);

```

```

}
else
{
digitalWrite(led_light1,LOW);
digitalWrite(led_light2,LOW);
}

##### end light sensor #####

// dht 11

int chk = DHT.read11(DHT11_PIN);
Serial.print("  Temperature = ");
Serial.print(DHT.temperature);
Serial.print("  Humidity = ");
Serial.print(DHT.humidity);

float temp = DHT.temperature;
if(temp>= 24 || distance<=10 || magnet_VAL<=500 )
{
  flag = true;
  //digitalWrite(buzzer,HIGH);
}
else
{
  flag = false;
}

```

```

    //digitalWrite(buzzer,LOW);
}

// end dht11

if(flag == true)
{
digitalWrite(buzzer,HIGH);
}
else
{
digitalWrite(buzzer,LOW);
}

Serial.print(" Magnet: ");Serial.println(magnet_VAL);

// living room

output = analogRead(pot_pin); //Reading from potentiometer
//Mapping the Values between 0 to 255 because we can give output
//from 0 -255 using the analogwritefunction
led_value = map(output, 0, 1023, 0, 255);

analogWrite(led_pin1, led_value);
analogWrite(led_pin2, led_value);
analogWrite(led_pin3, led_value);
//delay(100);
}

```

#### 4.ΣΥΜΠΕΡΑΣΜΑΤΑ

Στην παρούσα πτυχιακή εργασία ασχοληθήκαμε με την κατασκευή μιας μακέτας ενός πρότυπου έξυπνου σπιτιού με τη χρήση μικροελεγκτή ARDUINO.

Παρατηρήσαμε τις δυνατότητες της τεχνολογίας χρησιμοποιώντας κάποιους αισθητήρες, ώστε να γίνεται η ζωή στο σπίτι πιο εύκολη και πιο ασφαλή.

Οι δυνατότητες που προσφέρονται από ένα τέτοιο σύστημα είναι πάρα πολλές και εξαρτώνται κάθε φορά από τις ανάγκες που προκύπτουν και τους αισθητήρες που μπορεί να χρησιμοποιηθούν για ένα έξυπνο σπίτι.



## ΕΥΧΑΡΙΣΤΙΕΣ

Θεωρούμε υποχρέωσή μας να ευχαριστήσουμε τον επιβλέποντα καθηγητή για την πολύτιμη καθοδήγησή του. Επιπρόσθετα, οφείλουμε να αφιερώσουμε την πτυχιακή μας εργασία στους γονείς μας που μας συμπαραστάθηκαν όλα τα χρόνια της φοίτησης μας στο Α.Τ.Ε.Ι ΠΑΤΡΩΝ.

## 5.ΒΙΒΛΙΟΓΡΑΦΙΑ

- [Arduino - Home - www.arduino.cc](http://www.arduino.cc)
- [Arduino Uno - Wikipedia - en.wikipedia.org/wiki/Arduino\\_Uno](http://en.wikipedia.org/wiki/Arduino_Uno)
- [ρομποτική με Arduino \(slideshare.net\) - www.slideshare.net/perlat-sp/perlat-kociaj-arduino](http://www.slideshare.net/perlat-sp/perlat-kociaj-arduino)
- [«Εφαρμογές Arduino στο σχολικό εργαστήριο για το περιβάλλον» - Εγχειρίδιο των Γεώργιου Γίδα, Ευάγγελου Μαντζάνα • Ανοικτή Βιβλιοθήκη \(openbook.gr\) - www.openbook.gr/efarmoges-arduino-sto-scholiko-ergastirio-gia-to-perivallon/](http://www.openbook.gr/efarmoges-arduino-sto-scholiko-ergastirio-gia-to-perivallon/)
- [Κατασκευάζω και Προγραμματίζω με τον μικροελεγκτή Arduino \(robotics-edu.gr\) - http://robotics-edu.gr/data/arduino/arduino\\_paliouras.pdf](http://robotics-edu.gr/data/arduino/arduino_paliouras.pdf)
- [Arduino - Devoxx4Kids - www.devoxx4kids.org/materials/workshops/internet-of-things/arduino/](http://www.devoxx4kids.org/materials/workshops/internet-of-things/arduino/)
- [ProgrammingArduino.pdf \(sch.gr\) - http://users.sch.gr/manpoul/docs/arduino/ProgrammingArduino.pdf](http://users.sch.gr/manpoul/docs/arduino/ProgrammingArduino.pdf)
- [S4A - s4a.cat/](http://s4a.cat/)
- [Arduino - YouTube - www.youtube.com/channel/UCfyfK0tzHZTpNFrc\\_NDKfTA](http://www.youtube.com/channel/UCfyfK0tzHZTpNFrc_NDKfTA)
- [14 smart home Projects - Arduino Project Hub - create.arduino.cc/projecthub/projects/tags/smart+home](http://create.arduino.cc/projecthub/projects/tags/smart+home)