



ΔΙΕΘΝΕΣ
ΠΑΝΕΠΙΣΤΗΜΙΟ
ΕΛΛΑΔΟΣ

ΔΙΕΘΝΕΣ ΠΑΝΕΠΙΣΤΗΜΙΟ ΤΗΣ ΕΛΛΑΔΟΣ
ΣΧΟΛΗ ΜΗΧΑΝΙΚΩΝ
ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ, ΥΠΟΛΟΓΙΣΤΩΝ ΚΑΙ
ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ

**Υλοποίηση σταθμού ελέγχου ποιότητας αέρα με τη χρήση
Arduino**

Πτυχιακή Εργασία του
Γεωργίου Κυριακόπουλου 3971

Επιβλέπων: Ι. Καλόμοιρος, Καθηγητής

ΣΕΡΡΕΣ, ΙΟΥΝΙΟΣ 2022

Ευχαριστίες

Θα ήθελα να ευχαριστήσω τον επιβλέποντα καθηγητή μου κ. Καλόμοιρο Ιωάννη για την καθοδήγηση που μου παρείχε σε όλη τη διάρκεια της εκπόνησης της Πτυχιακής μου εργασίας.

Επίσης ευχαριστώ και την οικογένεια μου για τη στήριξη που μου παρείχε καθ' όλη τη διάρκεια των σπουδών μου.

Υπεύθυνη Δήλωση: Βεβαιώνω ότι είμαι συγγραφέας αυτής της πτυχιακής εργασίας και ότι κάθε βοήθεια την οποία είχα για την προετοιμασία της, είναι πλήρως αναγνωρισμένη και αναφέρεται στην πτυχιακή εργασία. Επίσης έχω αναφέρει τις όποιες πηγές από τις οποίες έκανα χρήση δεδομένων, ιδεών ή λέξεων, είτε αυτές αναφέρονται ακριβώς είτε παραφρασμένες. Επίσης, βεβαιώνω ότι αυτή η πτυχιακή εργασία προετοιμάστηκε από εμένα προσωπικά ειδικά για τις απαιτήσεις του προγράμματος σπουδών του Τμήματος Μηχανικών Πληροφορικής, Υπολογιστών και Τηλεπικοινωνιών του Διεθνούς Πανεπιστημίου της Ελλάδας.

Περίληψη

Η παρούσα πτυχιακή εργασία αφορά το σχεδιασμό και την ανάπτυξη ενός σταθμού που ελέγχει την ποιότητα αέρα σε μια περιοχή και μπορεί να μετρήσει διάφορα φυσικά μεγέθη ώστε ο χρήστης να μπορεί να εξάγει ένα συμπέρασμα για το περιβάλλον που είναι τοποθετημένος ο σταθμός. Ο σταθμός αυτός αποτελείται από ένα αισθητήρα βαρομετρικής πίεσης μεγάλης ακρίβειας που μετράει και θερμοκρασία, ένα αισθητήρα που μετράει υγρασία και θερμοκρασία και έναν ελεγκτή ποιότητας αέρα που μετράει ένα ευρύ φάσμα επιβλαβών αερίων, όπως μονοξείδιο του άνθρακα, αλκοόλ, ακετόνη, διαλυτικό, φορμαλδεΰδη. Τα παραπάνω εξαρτήματα συνδέονται σε μία προγραμματιζόμενη πλακέτα Arduino Uno όπου και εκτελείται το πρόγραμμα ελέγχου του σταθμού.

Λέξεις κλειδιά

Arduino, BMP388, DTH11, Grove - Air Quality Sensor v1.3, σταθμός ελέγχου
ποιότητας αέρα, LoRaWAN , LoRa

Περιεχόμενα

| | |
|--|----|
| Ευχαριστίες | 2 |
| Περίληψη | 4 |
| Λέξεις κλειδιά | 5 |
| Εισαγωγή | 8 |
| 1. State of the art | 10 |
| 1.1 Μονάδα παρακολούθησης ποιότητας αέρα | 10 |
| 1.1.1 Τα μέρη που αποτελείται η μονάδα | 10 |
| 1.1.2 Περιγραφή αισθητήρων | 11 |
| 1.2 Xiaomi Air Purifier | 13 |
| 2. Μικροϋπολογιστής – αισθητήρες | 14 |
| 2.1.1 Arduino | 14 |
| 2.1.2 Πλεονεκτήματα του Arduino | 16 |
| 2.1.3 Μοντέλα Arduino | 17 |
| 2.1.4 Arduino Shields | 18 |
| 2.2.1 Arduino Uno Rev 3 | 19 |
| 2.2.2 Αναλυτική λειτουργία του Arduino | 22 |
| 2.2.2.1 Τροφοδοσία | 22 |
| 2.2.2.2 Μνήμη | 23 |
| 2.2.2.3 Ακροδέκτες | 24 |
| 2.2.2.4 Arduino IDE | 26 |
| 2.2.2.5 Γλώσσα προγραμματισμού | 27 |
| 2.3 Adafruit BMP388 | 28 |
| 2.4 DTH11 | 30 |
| 2.5 Grove - Air Quality Sensor v1.3 | 32 |
| 3 Κατασκευή και λειτουργία της εργασίας | 35 |

| | |
|---|----|
| 3.1 Κατασκευή | 35 |
| 3.1.1 Adafruit BMP388 | 35 |
| 3.1.2 DTH11 | 38 |
| 3.1.3 Grove - Air Quality Sensor v1.3 | 40 |
| 3.2 Κώδικας | 42 |
| 3.2.1 Κώδικας σταθμού ελέγχου ποιότητας αέρα | 42 |
| 3.2.2 Κώδικας εφαρμογής παρουσίασης των αποτελεσμάτων | 48 |
| 3.3 Ανάλυση της λειτουργίας του σταθμού | 50 |
| 4. Δίκτυα LoRa | 56 |
| 4.1 Εισαγωγή | 56 |
| 4.2 Τι είναι τα δίκτυα LoRa | 57 |
| 4.3 LoRaWAN | 58 |
| 4.4 Εύρος ζώνης έναντι απόστασης | 58 |
| 4.5 Πλεονεκτήματα του LoRaWAN | 59 |
| 4.6 Αρχιτεκτονική LoRaWAN | 60 |
| 4.6 Απαραίτητο υλικό για δημιουργία δικτύου LoRa | 61 |
| 4.6.1 Τερματική συσκευή | 61 |
| 4.6.2 Gateway | 62 |
| 5 Συμπεράσματα και επίλυση δυσκολιών | 64 |
| 5.1 Επίλυση δυσκολιών | 64 |
| 5.2 Συμπεράσματα | 64 |
| Βιβλιογραφία | 65 |

Εισαγωγή

Η κακή ποιότητα του αέρα που μπορεί να υπάρχει μέσα σε ένα χώρο πιθανόν να οδηγήσει σε πολλές αρνητικές επιπτώσεις στην υγεία μας. Όπως κόπωση, πονοκεφάλους, απώλεια συγκέντρωσης, αυξημένο καρδιακό ρυθμό κ.α. Η παρακολούθηση της ποιότητας του αέρα είναι πολύ σημαντική.

Στις μέρες μας, είναι ολοένα και μεγαλύτερη η ανάγκη να κάνουμε τη ζωή μας πιο αυτοματοποιημένη και να τη διευκολύνουμε όσο μπορούμε. Αυτή είναι μια σκέψη που κυριαρχεί στους περισσότερους ανθρώπους και κυρίως στους νέους λόγω της ενασχόλησής τους με την τεχνολογία από μικρή ηλικία. Με βάση αυτούς τους προβληματισμούς, ξεκίνησε η ιδέα για τη δημιουργία μίας καινοτόμου συσκευής με τη χρήση Arduino.

Η τελική ιδέα που υλοποιήθηκε για τη συγκεκριμένη εργασία είναι ένας σταθμός ποιότητας αέρα, δηλαδή ένα σταθμός που θα παίρνει σε τακτά χρονικά διαστήματα μετρήσεις και θα ενημερώνει για την ποιότητα του αέρα στο χώρο στον οποίο είναι εγκατεστημένος.

Ο σταθμός ελέγχου ποιότητας αέρα είναι ένας σταθμός που μπορεί να μετράει φυσικά μεγέθη μέσω αισθητήρων, όπως ατμοσφαιρική πίεση, θερμοκρασία, υγρασία, μονοξείδιο του άνθρακα, αλκοόλ, ακετόνη, διαλυτικό και φορμαλδεΐδη. Αυτό επιτυγχάνεται με την χρήση των παρακάτω αισθητήρων:

Για την μέτρηση της ατμοσφαιρική πίεση και της θερμοκρασίας γίνεται η χρήση ενός αισθητήρα βαρομετρικής πίεσης μεγάλης ακριβείας, του BMP388. Για τη μέτρηση της υγρασίας χρησιμοποιείται ο αισθητήρας DHT11. Ο συγκεκριμένος αισθητήρας εκτός από υγρασία μετράει και θερμοκρασία αλλά ο σταθμός χρησιμοποιεί την ένδειξη θερμοκρασία από τον αισθητήρα BMP388 για λόγους ακριβείας. Οι μετρήσεις για τα αέρια στοιχεία μονοξείδιο του άνθρακα, αλκοόλ, ακετόνη, διαλυτικό και φορμαλδεΐδη συλλέγονται με τη χρήση του αισθητήρα Grove - Air Quality Sensor v1.3. Λόγω του μηχανισμού μέτρησής του αυτός ο αισθητήρας δεν μπορεί να εξάγει συγκεκριμένα δεδομένα για να περιγράψει ποσοτικά τις συγκεντρώσεις των αερίων. Για αυτό το λόγο και στην έξοδό του παρουσιάζεται ένα συνολικό αποτέλεσμα των φυσικών στοιχείων που μετράει υποδηλώνοντας την κατάσταση που βρίσκεται ο αέρας εκείνη τη στιγμή.

Τέλος, όλοι αυτοί οι αισθητήρες οδηγούνται μέσω ενός μικροϋπολογιστή, του Arduino Uno Rev3, και έτσι οι μετρήσεις εμφανίζονται στο χρήστη

παρακολουθώντας τη σειριακή θύρα του μικροϋπολογιστή. Ο προγραμματισμός του Arduino γίνεται μέσω ηλεκτρονικού υπολογιστή, με τη βοήθεια του περιβάλλοντος ανάπτυξης εφαρμογών Arduino IDE.

Ένα μέρος της παρακάτω πτυχιακής εργασίας είναι η διερεύνηση ενός τρόπου επικοινωνίας του σταθμού με το διαδίκτυο χωρίς τη χρήση δικτύου GSM ή WIFI. Η μέθοδος που αξιοποιήθηκε είναι η χρήση των δικτύων LoRa, τα οποία θα παρουσιαστούν εκτενώς στη συνέχεια.

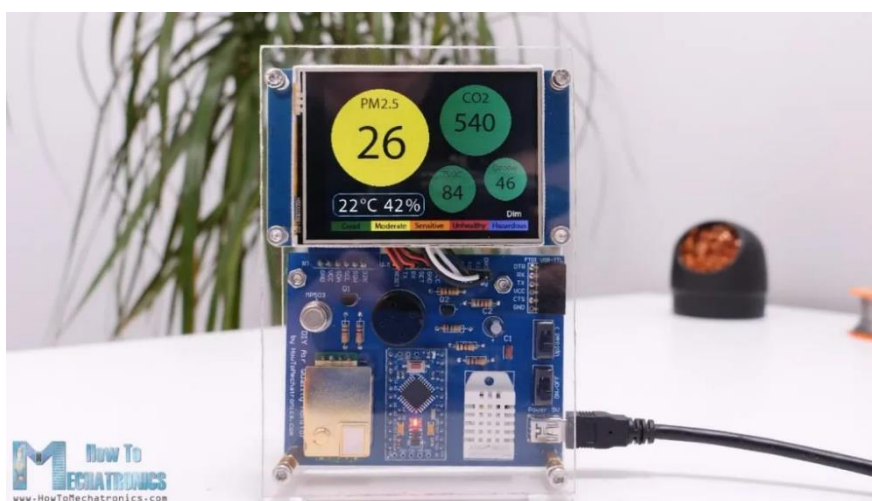
Τα δίκτυα LoRa είναι ένα πρωτόκολλο με το οποίο μεταδίδονται δεδομένα μικρά σε μέγεθος όπως τιμές από αισθητήρες σε απόσταση μεγαλύτερη των 10 χιλιομέτρων. Η αναφορά στα δίκτυα LoRa γίνεται με σκοπό την περαιτέρω ανάπτυξη της συγκεκριμένης εργασίας στο μέλλον χρησιμοποιώντας αυτά ως δίκτυο μεταφοράς δεδομένων με χαμηλό κόστος.

1. State of the art

Σε αυτό το κεφάλαιο θα παρουσιαστούν παρόμοια project που υπάρχουν στο διαδίκτυο αλλά και έτοιμες προτάσεις από την βιομηχανία.

1.1 Μονάδα παρακολούθησης ποιότητας αέρα

Πρόκειται για μια μονάδα που παρακολουθεί τον αέρα που υπάρχει σε ένα κλειστό χώρο και ειδοποιεί για την κατάστασή του με μια οθόνη η οποία εμφανίζει τις τιμές του λαμβάνουν οι αισθητήρες αλλά έχει και την δυνατότητα να παρουσιάζει απευθείας ένα διάγραμμα από τις τελευταίες μετρήσεις των αισθητήρων που διαθέτει.



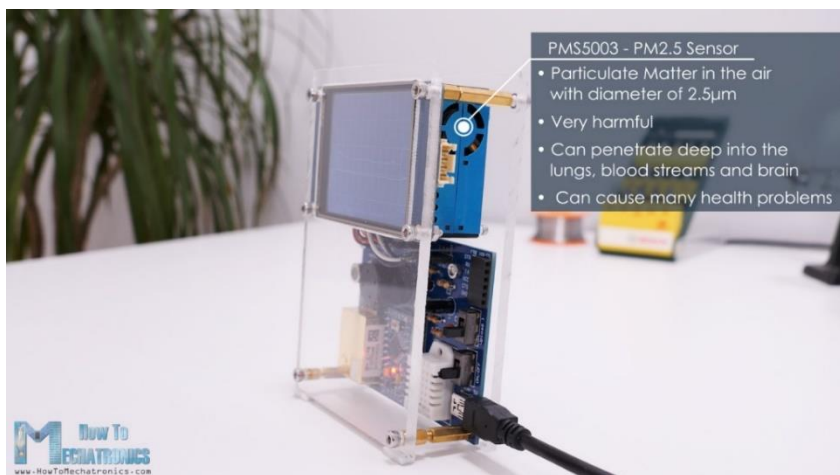
Εικόνα 1.1 Μονάδα παρακολούθησης ποιότητας αέρα

1.1.1 Τα μέρη που αποτελείται η μονάδα

Ο εγκέφαλος αυτού του πρότζεκτ είναι μια πλακέτα Arduino Pro Mini και μια οθόνη αφής Nextion 2,8". Δίνεται η δυνατότητα παρουσίασης όλων των μετρήσεων από όλους τους αισθητήρες σε πραγματικό χρόνο καθώς και προβολή των μετρήσεων από κάθε αισθητήρα μεμονωμένα, λαμβάνοντας τιμές από τις τελευταίες 24 ώρες. Υπάρχει επίσης η λειτουργία dimming, μέσω της οποίας είναι εφικτή η μείωση της φωτεινότητας της οθόνης.

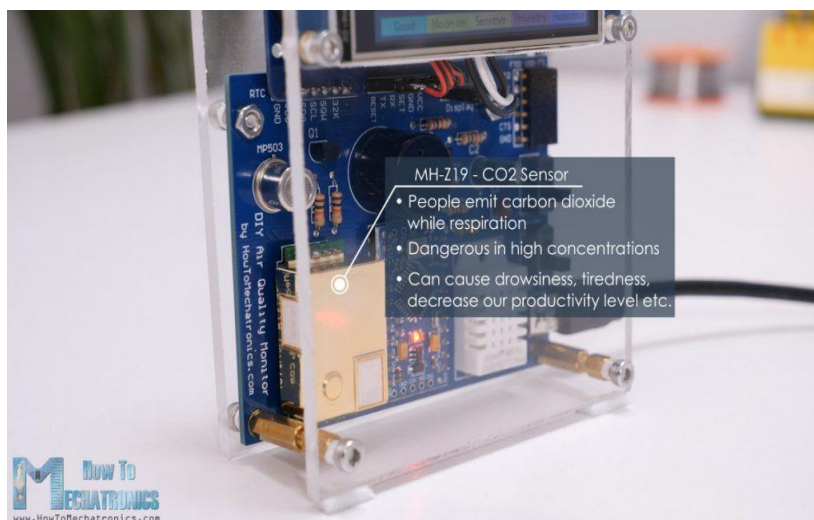
1.1.2 Περιγραφή αισθητήρων

Αυτή η συσκευή διαθέτει τέσσερα κύρια εξαρτήματα ή αισθητήρες ποιότητας αέρα. Χρησιμοποιείται ο αισθητήρας PMS5003 για τη μέτρηση σωματιδίων στον αέρα με διάμετρο περίπου 2,5 microns. Τα σωματίδια αυτά είναι η πιο επιβλαβής μορφή ατμοσφαιρικής ρύπανσης επειδή μπορούν να διεισδύσουν βαθιά στους πνεύμονες, και στον εγκέφαλο, προκαλώντας πολλά προβλήματα υγείας.



Εικόνα 1.2 Αισθητήρας PMS5003

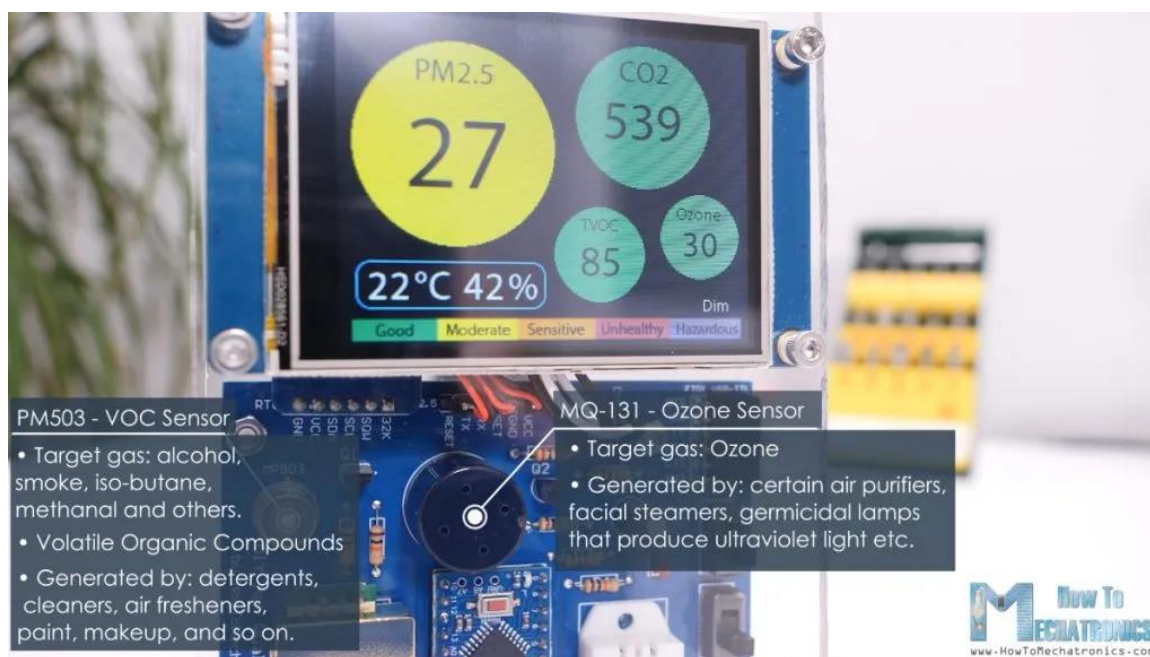
Στη συνέχεια, χρησιμοποιείται ο αισθητήρας MH-Z19 για τη μέτρηση του CO₂, δηλαδή του διοξειδίου του άνθρακα. Καθώς οι άνθρωποι εκπέμπουν διοξείδιο του άνθρακα κατά την αναπνοή, η συγκέντρωση CO₂ σε εσωτερικούς χώρους μπορεί εύκολα να αυξηθεί πολύ. Το CO₂ δεν είναι μόνο επικίνδυνο σε υψηλές συγκεντρώσεις, αλλά μπορεί επίσης να προκαλέσει υπνηλία, κούραση και μείωση του επιπέδου παραγωγικότητάς.



Εικόνα 1.3 Αισθητήρας MH-Z19

Για τη μέτρηση VOC και του όζοντος χρησιμοποιούμε τους αισθητήρες αερίου MP503 και MQ131. Αυτοί είναι αισθητήρες θερμαινόμενου οξειδίου μετάλλου και η αρχή λειτουργίας τους βασίζεται στην ανίχνευση αλλαγής αντίστασης με την παρουσία στοχευμένων αερίων.

Το αέριο που ανιχνεύει ο αισθητήρας MQ131 είναι το όζον, το οποίο μπορεί να δημιουργηθεί από προϊόντα όπως καθαριστές αέρα, ατμοκαθαριστές προσώπου, μικροβιοκτόνους λαμπτήρες που παράγουν υπεριώδες φως κ.α



Εικόνα 1.4 Αισθητήρες MQ131 και MP503

Από την άλλη πλευρά, ο αισθητήρας MP503 ανιχνεύει πολλαπλάσια αέρια, όπως αλκοόλ, καπνό, ισοβουτάνιο, μεθανάλη και άλλα. Ο όρος VOC αναφέρεται στις πτητικές οργανικές ενώσεις οι οποίες είναι οργανικές εκπομπές από προϊόντα που χρησιμοποιούμε σε καθημερινή βάση, όπως απορρυπαντικά πλυντηρίων ρούχων, καθαριστικά, αποσμητικά χώρου, χρώματα, μακιγιάζ κ.λπ. Οι πτητικές οργανικές ενώσεις μπορούν να προκαλέσουν πολλές αρνητικές επιπτώσεις στην υγεία, όπως πονοκεφάλους, ερεθισμούς στα μάτια, δερματικές αντιδράσεις, ζάλη κ.α.

1.2 Xiaomi Air Purifier

Η εταιρία Χίαομι έχει κατασκευάσει έναν «καθαριστή αέρα» που είναι εφοδιασμένος με διάφορους αισθητήρες όπως αισθητήρα θερμοκρασίας και υγρασίας, αισθητήρα ανίχνευσης σωματιδίων μεγέθους PM2.5, αισθητήρα φορμαλδεΐδης και άλλους. Εκτός από την ανίχνευση βλαβερών στοιχείων για τον ανθρώπινο οργανισμό η συγκεκριμένη συσκευή έχει τη δυνατότητα να καθαρίσει τον αέρα που υπάρχει σε ένα κλειστό χώρο με τη χρήση ειδικών φίλτρων. Επίσης δίνεται η δυνατότητα ελέγχου της συσκευής μέσω της χρήσης ειδικής εφαρμογής στο κινητό τηλέφωνο του χρήστη.



Εικόνα 1.5 Xiaomi Air Purifier

2. Μικροϋπολογιστής – αισθητήρες

Όπως αναφέρθηκε προηγουμένως, για την υλοποίηση του σταθμού ελέγχου ποιότητας αέρα χρησιμοποιήθηκαν ένας μικροϋπολογιστής Arduino και τρεις αισθητήρες.

2.1.1 Arduino

Το Arduino είναι μια εταιρεία υλικού και λογισμικού ανοιχτού κώδικα, κοινότητα έργων και χρηστών που σχεδιάζει και κατασκευάζει μικροελεγκτές μίας κάρτας και κιτ μικροελεγκτών για την κατασκευή ψηφιακών συσκευών και διαδραστικών αντικειμένων που μπορούν να ανιχνεύσουν και να ελέγξουν τόσο φυσικά όσο και ψηφιακά. Τα προϊόντα της έχουν άδεια χρήσης της GNU Lesser General Public License (LGPL) ή της Γενικής Δημόσιας Άδειας GNU (GPL), επιτρέποντας την κατασκευή αρχείων Arduino και τη διανομή λογισμικού από οποιονδήποτε. Οι πίνακες Arduino διατίθενται στο εμπόριο σε προ-συναρμολογημένη μορφή ή ως κιτ ετοιμοπαράδοτων (DIY).



Εικόνα 2.1 Λογότυπο του μικροϋπολογιστή Arduino

Οι πίνακες Arduino χρησιμοποιούν μια ποικιλία μικροεπεξεργαστών και ελεγκτών. Οι πίνακες είναι εφοδιασμένοι με σετ ψηφιακών και αναλογικών εισόδων / εξόδων (I / O) ακίδων που μπορούν να διασυνδεθούν με διάφορες πλακέτες επέκτασης ή πλάκες (θωράκια) και άλλα κυκλώματα. Οι κάρτες διαθέτουν σειριακές διεπαφές επικοινωνίας, συμπεριλαμβανομένου του Universal Serial Bus (USB) σε ορισμένα

μοντέλα, τα οποία χρησιμοποιούνται επίσης για τη φόρτωση προγραμμάτων από προσωπικούς υπολογιστές. Οι μικροελεγκτές προγραμματίζονται συνήθως χρησιμοποιώντας μια διάλεκτο χαρακτηριστικών από τις γλώσσες προγραμματισμού C και C++. Εκτός από τη χρήση παραδοσιακών αλυσίδων εργαλείων μεταγλωττιστή, το έργο Arduino παρέχει ένα ολοκληρωμένο περιβάλλον ανάπτυξης (IDE) με βάση το πρόγραμμα επεξεργασίας γλώσσας.

Το πρόγραμμα Arduino ξεκίνησε το 2005 ως πρόγραμμα για φοιτητές στο Ινστιτούτο Σχεδίασης της Αλληλεπίδρασης Ivrea στην Ivrea της Ιταλίας. Στόχο είχε να προσφέρει ένα χαμηλού κόστους και εύκολο μέσο για τους ερασιτέχνες και τους επαγγελματίες ώστε να δημιουργούν συσκευές που αλληλοεπιδρούν με το περιβάλλον τους χρησιμοποιώντας αισθητήρες και ενεργοποιητές. Κοινά παραδείγματα τέτοιων συσκευών που προορίζονται για αρχάριους χομπίστες περιλαμβάνουν απλά ρομπότ, θερμοστάτες και ανιχνευτές κίνησης.

Το όνομα Arduino προέρχεται από ένα μπαρ στην Ivrea της Ιταλίας, όπου συναντήθηκαν μερικοί από τους ιδρυτές του έργου. Το μπαρ πήρε το όνομά του από τον Arduin της Ivrea, ο οποίος ήταν ο σηματοδότης του Μαρτίου του Ivrea και του βασιλιά της Ιταλίας από το 1002 έως το 1014.

Το Arduino είναι υλικό ανοικτού κώδικα. Τα σχέδια αναφοράς υλικού διανέμονται υπό την άδεια Creative Commons Attribution Share-Alike 2.5 και είναι διαθέσιμα στην ιστοσελίδα του Arduino. Τα αρχεία διάταξης και παραγωγής για ορισμένες εκδόσεις του υλικού είναι επίσης διαθέσιμα.

Παρόλο που τα σχέδια υλικού και λογισμικού είναι ελεύθερα διαθέσιμα βάσει των αδειών copyleft, οι προγραμματιστές ζήτησαν από το όνομα Arduino να είναι αποκλειστικός στο επίσημο προϊόν και να μην χρησιμοποιείται για παράγωγα έργα χωρίς άδεια. Το επίσημο έγγραφο πολιτικής για τη χρήση του ονόματος Arduino τονίζει ότι το έργο είναι ανοικτό για να ενσωματώσει την εργασία των άλλων στο επίσημο προϊόν. Αρκετά προϊόντα συμβατά με το Arduino που κυκλοφορούν στο εμπόριο έχουν αποφύγει το όνομα του έργου χρησιμοποιώντας διάφορα ονόματα που λήγουν στο -duino.



Εικόνα 2.2 Μικροϋπολογιστής Arduino Uno

2.1.2 Πλεονεκτήματα του Arduino

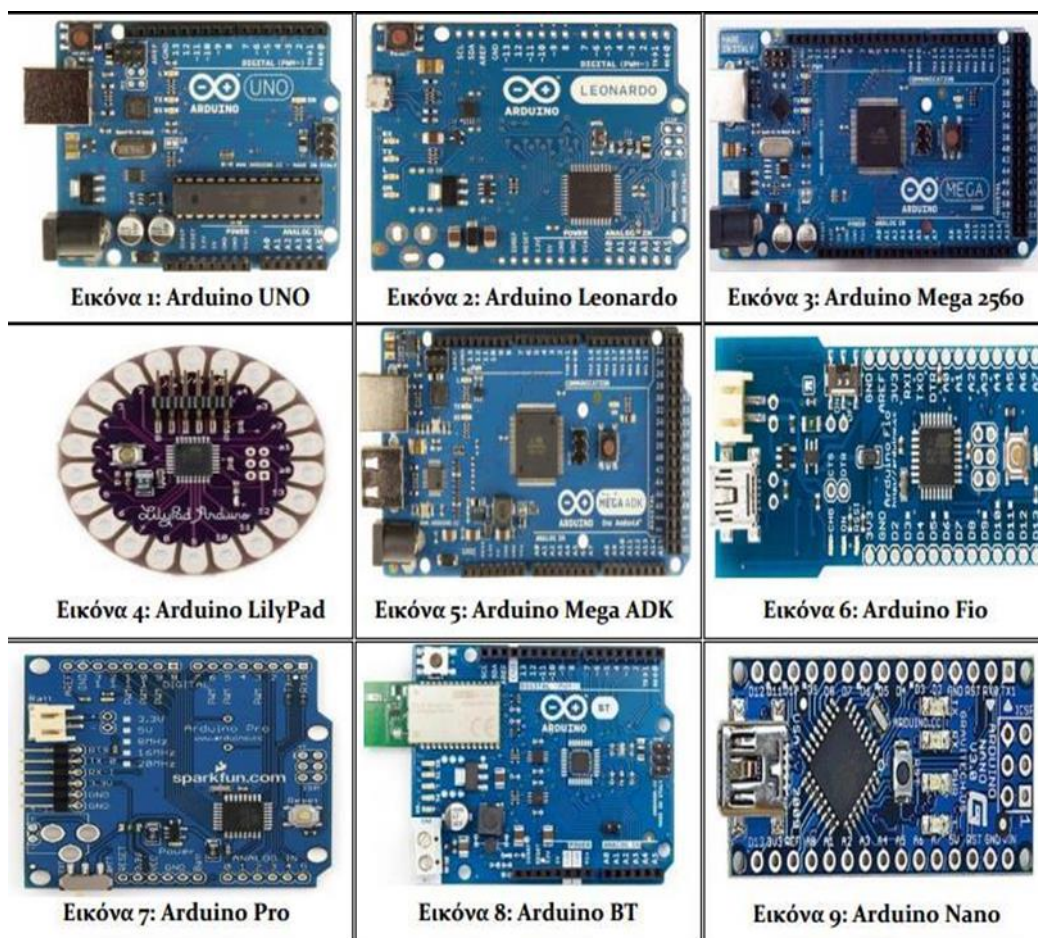
Το Arduino διαφέρει από τους άλλους μικροελεγκτές γιατί απλοποιεί τη διαδικασία να δουλεύει κάποιος με μικροελεγκτές. Προσφέρει πολλά πλεονεκτήματα σε σχέση με άλλους μικροελεγκτές, τα οποία σχετίζονται με τη χρήση του από δασκάλους, μαθητές, προγραμματιστές. Διαθέτει αναλυτικές οδηγίες για την κατασκευή μιας πλακέτας, όπως επίσης κυκλοφορούν προπαρασκευασμένες πλακέτες Arduino στο διαδίκτυο ή στα καταστήματα ηλεκτρονικών καλύπτοντας μεγάλο φάσμα λειτουργιών για την εξυπηρέτηση όλων. Είναι πολύ φθηνός σε σχέση με άλλους μικροελεγκτές, ακόμα και μία προπαρασκευασμένη πλακέτα είναι προσιτή οικονομικά. Οι μηχανικοί λογισμικού, ανέπτυξαν το περιβάλλον προγραμματισμού του Arduino για διάφορα Λειτουργικά Συστήματα όπως είναι τα Windows, Machinstoh OSX και τα Linux, ενώ τα περισσότερα προγράμματα ανάπτυξης μικροελεγκτών περιορίζονται στα Windows.

Το περιβάλλον προγραμματισμού ενός Arduino ενδείκνυται για αρχάριους, αλλά είναι ταυτόχρονα και ευέλικτο και για πιο προχωρημένους χρήστες. Είναι βασισμένο σε Java και περιέχει ένα πρακτικό περιβάλλον για τη συγγραφή των προγραμμάτων, με συντακτική χρωματική σήμανση, μερικές έτοιμες βιβλιοθήκες για προέκτασή του, τον compiler για τη μεταγλώττιση των sketch, μία σειριακή οθόνη (serial monitor) που παρακολουθεί τις επικοινωνίες της σειριακής (USB), αναλαμβάνει να στείλει αλφαριθμητικά στο Arduino μέσω αυτής και είναι ιδιαίτερα χρήσιμο για την αποσφαλμάτωση των sketch και την επιλογή για ανέβασμα των μεταγλωττισμένων sketch στο Arduino.

Η γλώσσα του Arduino βασίζεται στη γλώσσα Wiring, μια παραλλαγή C/C++ και μπορούν να χρησιμοποιηθούν ουσιαστικά οι ίδιες βασικές εντολές και συναρτήσεις, με την ίδια σύνταξη, τους ίδιους τύπων δεδομένων και τους ίδιους τελεστές όπως και στη C. Πέρα από αυτές όμως, οι άνθρωποι που θέλουν να ασχοληθούν περισσότερο με τους μικροελεγκτές μπορούν να μεταβούν από τον Arduino στην AVR-C που είναι για προγραμματισμό των Atmel μικροελεγκτών και η γλώσσα στην οποία βασίστηκε το λογισμικό του Arduino.

2.1.3 Μοντέλα Arduino






Τα μοντέλα Arduino που κυκλοφορούν στην αγορά, παρουσιάζονται στην παρακάτω εικόνα. Στη συγκεκριμένη μελέτη, επιλέχθηκε το μοντέλο Arduino UNO Rev 3 το οποίο καλύπτει όλες τις ανάγκες της εργασίας, καθώς τα pins που διαθέτει επαρκούν για όλες τις λειτουργίες.



Εικόνα 2.3 Μοντέλα Arduino

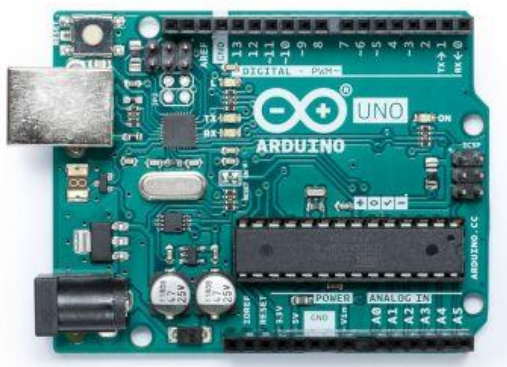
2.1.4 Arduino Shields

Τα Shields είναι τα εξαρτήματα που συνδέονται απευθείας με όλα τα pin του Arduino. Μερικά από αυτά φαίνονται στον παρακάτω πίνακα.

| | | |
|-------------------------|---|---|
| Arduino WiFi Shield |  | Το Arduino WiFi Shield συνδέει το Arduino στο διαδίκτυο ασύρματα. |
| Arduino Ethernet Shield |  | Το Arduino Ethernet Shield συνδέει το Arduino στο διαδίκτυο με ένα RJ45 καλώδιο. |
| Wireless SD Shield |  | Το Wireless SD Shield επιτρέπει σε μια πλακέτα Arduino να επικοινωνεί ασύρματα με μια ασύρματη μονάδα. Η μονάδα μπορεί να επικοινωνήσει έως και 100 πόδια σε εσωτερικούς χώρους ή σε εξωτερικούς χώρους ως 300 πόδια. Η μονάδα περιλαμβάνει μια θύρα υποδοχής SD. |
| Wireless Proto Shield |  | Το Wireless Proto Shield επιτρέπει στο Arduino να επικοινωνεί ασύρματα με μια ασύρματη μονάδα. Η μονάδα μπορεί να επικοινωνήσει έως και 100 πόδια σε εσωτερικούς χώρους ή σε εξωτερικούς χώρους ως 300 πόδια. Η μονάδα δεν περιλαμβάνει θύρα υποδοχής SD. |
| Arduino Motor Shield |  | Το Arduino Motor Shield επιτρέπει την οδήγηση δύο DC κινητήρων από την ίδια συσκευή, ελέγχοντας την ταχύτητα και την κατεύθυνση του καθενός ξεχωριστά. |

Εικόνα 2.4 Arduino Shields

2.2.1 Arduino Uno Rev 3



Εικόνα 2.5 Μικροελεκτήρας Arduino Uno Rev 3

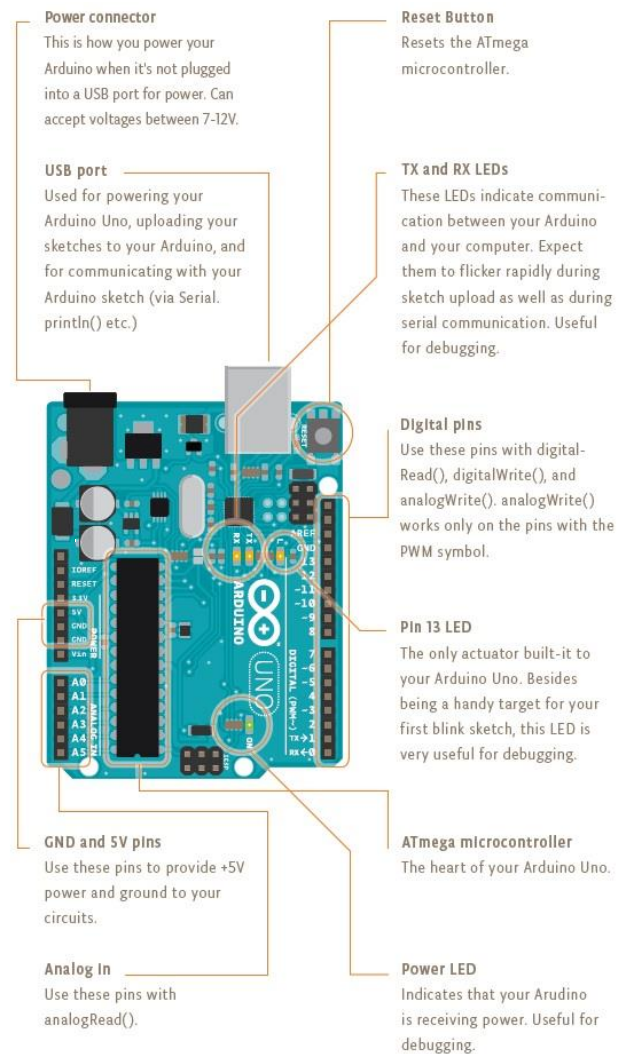
Όπως αναφέρθηκε προηγουμένως, στην συγκεκριμένη μελέτη χρησιμοποιήθηκε ο μικροελεκτήρας Arduino Uno Rev 3. Επιλέχθηκε η συγκεκριμένη έκδοση Arduino καθώς το μοντέλο Uno της οικογένειας Arduino είναι το πιο διαδεδομένο και είναι συμβατό με πλήθος αισθητήρων και επεκτάσεων. Η έκδοση 3 (Rev3) είναι η τελευταία που κυκλοφορεί και είναι πλήρως συμβατή με καινούργιες και παλιές πλακέτες επέκτασης. Ο Arduino Uno Rev3 βασίζεται στο μικροελεκτήρα ATmega328 της Atmel. Είναι μια ολοκληρωμένη πλακέτα που περιέχει ό,τι χρειάζεται για να μπορεί να προγραμματιστεί και να λειτουργήσει συνδέοντάς την με ένα απλό καλώδιο USB στον υπολογιστή ή με ένα τροφοδοτικό στην πρίζα. Βέβαια μπορεί να λειτουργήσει επίσης και με απλή μπαταρία.

Η πλακέτα διαθέτει δεκατέσσερις ψηφιακές εισόδους ή εξόδους, έξι αναλογικές εισόδους, μια θύρα USB (τύπου B) για τον προγραμματισμό και την τροφοδοσία της πλακέτας, μια είσοδο τροφοδοσίας που μπορεί να χρησιμοποιηθεί εναλλακτικά για τροφοδοσία από τροφοδοτικό πρίζας ή από απλή μπαταρία, μια υποδοχή ICSP και τέλος ένα κουμπί για το reset της πλακέτας. Ο μικροελεκτήρας είναι συγχρονισμένος στους δεκαέξι megahertz (16MHz).

Αυτή η έκδοση του Arduino Uno είναι με αποσπώμενο μικροελεγκτή (DIP Version) και έτσι αν καεί ο ελεγκτής υπάρχει η δυνατότητα να αντικατασταθεί με έναν καινούριο, με μικρό κόστος.

Η μνήμη Flash του Arduino Uno που αποθηκεύεται το πρόγραμμα (sketch) είναι 32KB, ικανή να δεχτεί τα περισσότερα απλά προγράμματα, αν και σε περίπτωση που δεν αρκεί, μπορεί να επιλεγθεί μία μεγαλύτερη πλακέτα Arduino, όπως η Arduino Mega 2560. Το Arduino Uno λειτουργεί με τροφοδοσία 5V DC από την είσοδο του USB ή με 9V/12V DC από την είσοδο της τροφοδοσίας χωρίς να υπάρχει κίνδυνος ηλεκτροπληξίας.

Το UNO στα Ιταλικά σημαίνει ENA και για αυτό το λόγο η εταιρεία Arduino έδωσε το όνομα της σε αυτή την πλακέτα που ήταν η πρώτη της οικογένειας Arduino που είχε επικοινωνία απευθείας με τον υπολογιστή μέσω θύρας USB σε συνεργασία με το πρόγραμμα Arduino 1.0 (IDE 1.0).



Εικόνα 2.6 Επεξήγηση της πλακέτας Arduino uno Rev 3 με την χρήση διαγράμματος

ΧΑΡΑΚΤΗΡΙΣΤΙΚΑ

Κατασκευαστής : Arduino

Part Number: A000066

Καθαρό Βάρος: 0.05kg

Χώρα Προέλευσης: Ιταλία

Breadboard Friendly

Μικροελεγκτής: ATmega328

Τάση λειτουργίας: 5V

Τάση εισόδου: 7-12V

Τάση εισόδου (limits): 6-20V

Ψηφιακά I/O Pins: 14

Αναλογικές εισόδους: 6

PWM εισόδους: 6

DC Ρεύμα ανά I/O Pin: 20 mA

DC Ρεύμα για 3.3V Pin: 50mA

Μνήμη Flash: 32 KB

Μνήμη SRAM: 2 KB (ATmega328)

Μνήμη EEPROM: 1 KB (ATmega328)

Ταχύτητα (Clock Speed): 16 MHz

ΕΠΙΠΛΕΟΝ ΧΑΡΑΚΤΗΡΙΣΤΙΚΑ

Αρχιτεκτονική ελεγκτή: AVR

Κωδικός προϊόντος:

A000066 (THT, DIP version)

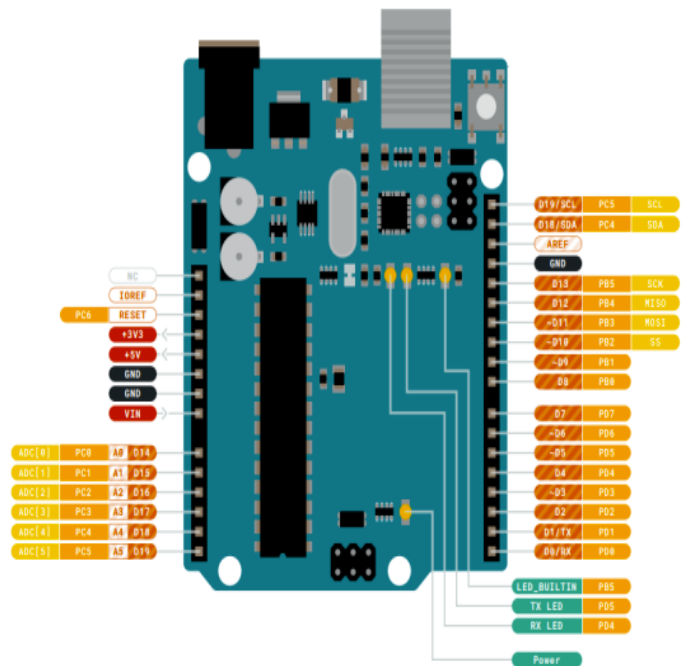
A000073 (SMD)

Διαστάσεις πλακέτας: 68.6mm x 53.4mm

Βάρος πλακέτας: 25g



2.7 Μικροελεγκτής Arduino Uno Rev 3



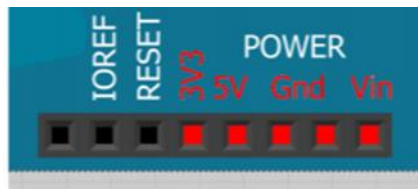
Εικόνα 2.8 Επεξήγηση των pin της πλακέτας Arduino uno Rev 3 με την χρήση διαγράμματος

2.2.2 Αναλυτική λειτουργία του Arduino

Στις παραγράφους που ακολουθούν θα περιγραφεί εκτενέστερα ο τρόπος λειτουργίας του μικροϋπολογιστή Arduino.

2.2.2.1 Τροφοδοσία

Το Arduino μπορεί να τροφοδοτηθεί με ρεύμα είτε από τον υπολογιστή μέσω της σύνδεσης USB, είτε από εξωτερική τροφοδοσία που παρέχεται μέσω μιας υποδοχής φινις των 2.1mm που βρίσκεται στην κάτω αριστερά γωνία. Για την αποφυγή προβλημάτων, η εξωτερική τροφοδοσία θα πρέπει να είναι από 7 ως 12V. Η εικόνα παρουσιάζει τις εισόδους και εξόδους τροφοδοσίας του Arduino UNO.



Εικόνα 2.9 Είσοδοι/Εξοδοι τροφοδοσίας

Οι ακροδέκτες τροφοδοσίας είναι οι ακόλουθοι:

- Vin: Η τάση εισόδου της πλακέτας όταν χρησιμοποιεί εξωτερική πηγή ενέργειας. Η τροφοδοσία τάσης γίνεται μέσω αυτού του ακροδέκτη.
- 5V: Η τάση που χρησιμοποιείται από τα διάφορα μέρη της πλακέτας και το μικροελεγκτή είναι 5V. Η τάση αυτή, την οποία δίνει αυτός ο ακροδέκτης, είναι είτε η τάση 5V που δίνει η σύνδεση με USB, είτε η ρυθμισμένη τάση που δίνεται μέσω του Vin.
- 3.3V: Η τάση αυτή παράγεται από το ολοκληρωμένο FTDI. Το όριο άντλησης ρεύματος είναι 50mA.
- GND: Είσοδοι γείωσης.

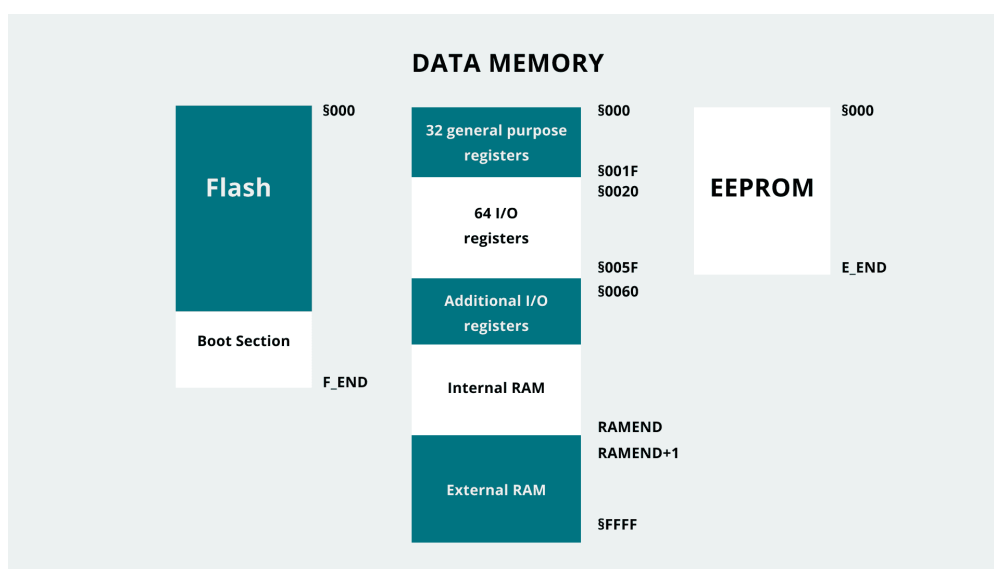
2.2.2.2 Μνήμη

Ο μικροεπεξεργαστής ATmega328 έχει τρεις ομάδες μνήμης. Διαθέτει flash memory, στην οποία αποθηκεύονται τα Arduino sketch, SRAM (static random access memory), στην οποία δημιουργείται το sketch και χρησιμοποιεί τις μεταβλητές όταν τρέχει, και EEPROM, η οποία χρησιμοποιείται από τους προγραμματιστές για την αποθήκευση μακροχρόνιων πληροφοριών.

2KB μνήμης SRAM: Η ωφέλιμη μνήμη που μπορούν να χρησιμοποιήσουν τα προγράμματα για να αποθηκεύουν μεταβλητές, πίνακες κ.λπ. Η μνήμη χάνει τα δεδομένα της όταν η παροχή ρεύματος στο Arduino σταματήσει ή πατηθεί το κουμπί επανεκκίνησης.

1KB μνήμης EEPROM: Μπορεί να χρησιμοποιηθεί για εγγραφή ή ανάγνωση δεδομένων από τα προγράμματα. Σε αντίθεση με την SRAM, δεν χάνει τα περιεχόμενά της με απώλεια τροφοδοσίας ή επανεκκίνησης.

32KB μνήμης Flash: 2 KB χρησιμοποιούνται από το firmware του Arduino που έχει εγκαταστήσει ήδη ο κατασκευαστής του. Το firmware είναι αναγκαίο για την εγκατάσταση προγραμμάτων στο μικροελεγκτή μέσω της θύρας USB. Τα υπόλοιπα 30KB της μνήμης Flash χρησιμοποιούνται για την αποθήκευση αυτών ακριβώς των προγραμμάτων, αφού πρώτα μεταγλωττιστούν στον υπολογιστή. Η μνήμη Flash, δεν χάνει τα περιεχόμενά της με απώλεια τροφοδοσίας ή επανεκκίνησης.



Εικόνα 2.10 Διάγραμμα μνήμης του μικροπολογιστή Arduino

2.2.2.3 Ακροδέκτες

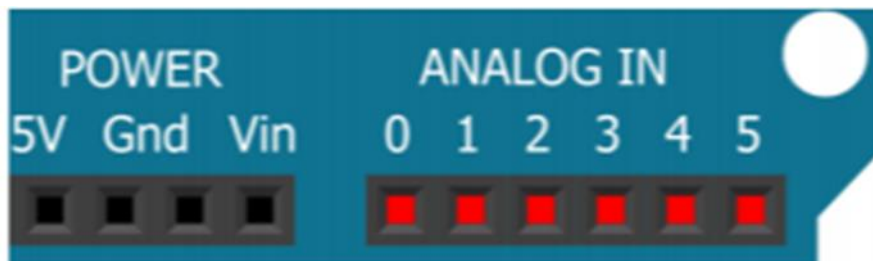


Εικόνα 2.11 Οι ψηφιακοί ακροδέκτες του Arduino

Ακροδέκτες 0 και 1: λειτουργούν ως RX και TX της σειριακής θύρας όταν το πρόγραμμά ενεργοποιεί τη σειριακή θύρα. Έτσι, όταν το πρόγραμμά στέλνει δεδομένα στη σειριακή θύρα, αυτά προωθούνται και στη θύρα USB μέσω του ελεγκτή Serial-Over-USB, αλλά και στον ακροδέκτη 0 για να τα διαβάσει ενδεχομένως μια άλλη συσκευή. Αυτό φυσικά σημαίνει ότι αν στο πρόγραμμα ενεργοποιηθεί το σειριακό interface, χάνει 2 ψηφιακές εισόδους/εξόδους η πλατφόρμα.

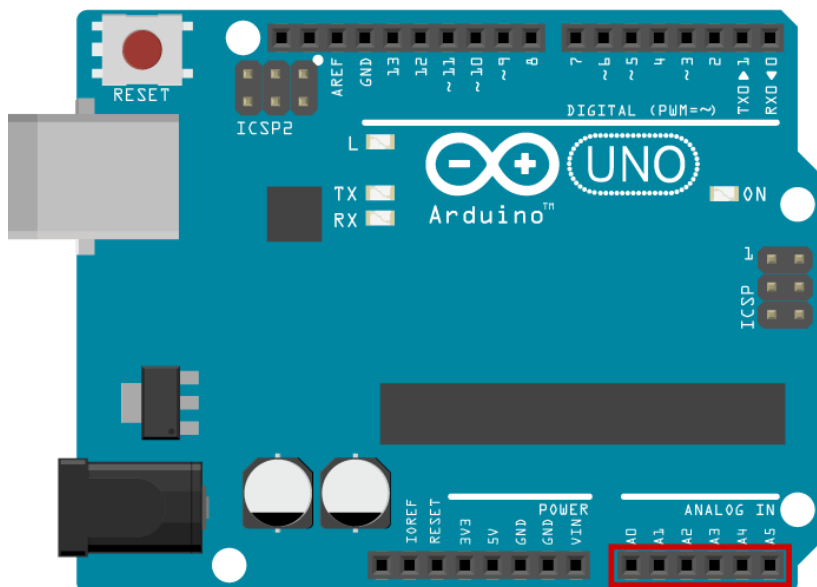
Ακροδέκτες 2 και 3: λειτουργούν και ως εξωτερικά interrupt (interrupt 0 και 1 αντίστοιχα). Ρυθμίζονται μέσα από το πρόγραμμά ώστε να λειτουργούν αποκλειστικά ως ψηφιακές εισοδοι στις οποίες όταν συμβαίνουν συγκεκριμένες αλλαγές, η κανονική ροή του προγράμματος σταματάει άμεσα και εκτελείται μια συγκεκριμένη συνάρτηση. Τα εξωτερικά interrupt είναι ιδιαίτερα χρήσιμα σε εφαρμογές που απαιτούν συγχρονισμό μεγάλης ακρίβειας.

Ακροδέκτες 3, 5, 6, 9, 10 και 11: μπορούν να λειτουργήσουν και ως ψευδόαναλογικές έξοδοι με το σύστημα PWM (Pulse Width Modulation).



Εικόνα 2.12 Αναλογικοί ακροδέκτες Arduino







Στην κάτω πλευρά του Arduino, με τη σήμανση ANALOG IN όπως φαίνεται και στην παραπάνω εικόνα υπάρχει μια ακόμη σειρά από 6 pin, αριθμημένα από το 0 ως το 5. Η τάση αναφοράς μπορεί να ρυθμιστεί με μια εντολή στο 1.1V (μεταξύ 2 και 5V) τροφοδοτώντας εξωτερικά με αυτή την τάση το pin με τη σήμανση AREF που βρίσκεται στην απέναντι πλευρά της πλακέτας. Έτσι, αν τροφοδοτηθεί ο ακροδέκτης AREF με 3.3V και στη συνέχεια διαβάσει κάποιον ακροδέκτη αναλογικής εισόδου στο οποίο εφαρμόζεται τάση 1.65V, το Arduino θα επιστρέψει την τιμή 512.



Εικόνα 2.13 Αναλογικοί ακροδέκτες Arduino

2.2.2.4 Arduino IDE

Το περιβάλλον ανάπτυξης Arduino περιέχει ένα πρόγραμμα επεξεργασίας κειμένου, για τη σύνταξη του κώδικα, μια περιοχή στην οποία εμφανίζονται μηνύματα, μία κονσόλα κειμένου και μια γραμμή εργαλείων υπό μορφή κουμπιών. Συνδέεται με το hardware μέρος του Arduino για να φορτώσει προγράμματα και να επικοινωνεί μαζί τους. Ο κώδικας που έχει γραφεί για το Arduino ονομάζεται sketch.

| | | |
|---|----------------|---|
|  | Verify | Ελέγχει για συντακτικά λάθη στον κώδικα. |
|  | Upload | Μεταγλωττίζει τον κώδικα και τον φορτώνει στο Arduino. |
|  | New | Δημιουργεί ένα νέο sketch. |
|  | Open | Παραθέτει ένα μενού με όλα τα sketch. Κάνοντας κλικ σε ένα από αυτά, θα ανοίξει αυτόματα στο τρέχον παράθυρο. |
|  | Save | Αποθηκεύει ένα sketch. |
|  | Serial Monitor | Ανοίγει την σειριακή οθόνη. |

Εικόνα 2.14 Εργαλεία υπό μορφή κουμπιών

Το Arduino IDE είναι βασισμένο σε Java και συγκεκριμένα παρέχει:

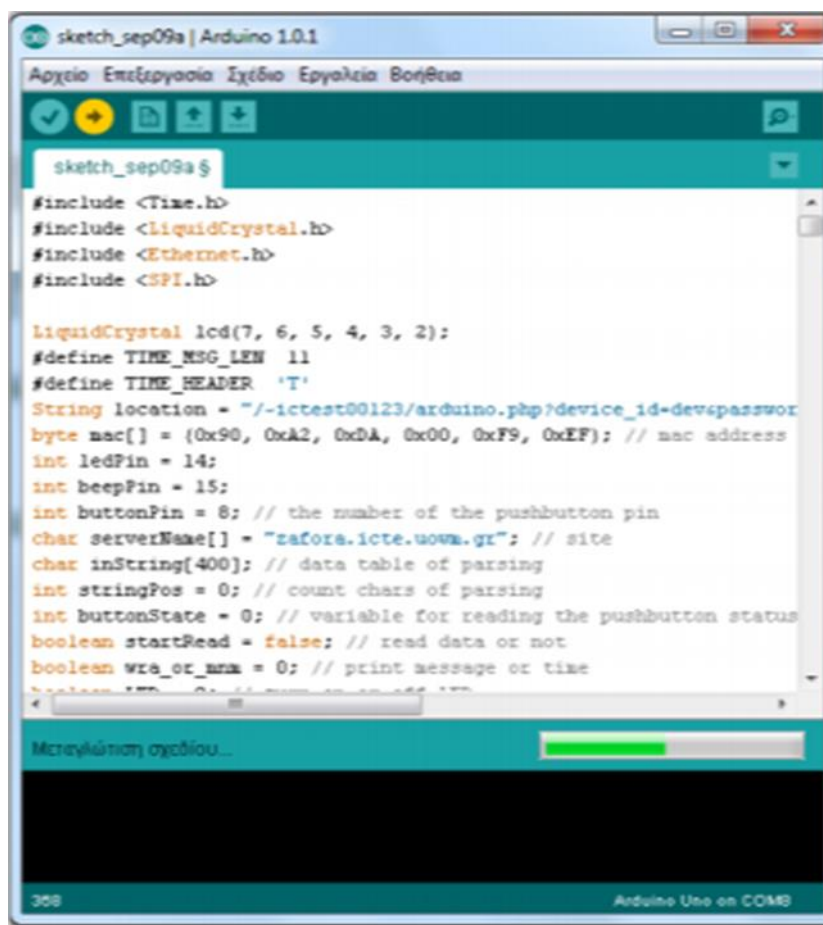
- Ένα πρακτικό περιβάλλον για τη συγγραφή των προγραμμάτων, με συντακτική χρωματική σήμανση.
- Μερικές έτοιμες βιβλιοθήκες για προέκταση της.
- Τον compiler για τη μεταγλώττιση των sketch.
- Μία σειριακή οθόνη (serial monitor) που παρακολουθεί τις επικοινωνίες της σειριακής (USB), αναλαμβάνει να στείλει αλφαριθμητικά στο Arduino μέσω αυτής και είναι ιδιαίτερα χρήσιμο για την αποσφαλμάτωση των sketch.
- Την επιλογή για ανέβασμα των μεταγλωττισμένων sketch στο Arduino.

2.2.2.5 Γλώσσα προγραμματισμού

Η γλώσσα του Arduino βασίζεται στη γλώσσα Wiring μια παραλλαγή C/C++ για μικροελεγκτές αρχιτεκτονικής AVR όπως ο ATmega, και υποστηρίζει όλες τις βασικές δομές της C καθώς και μερικά χαρακτηριστικά της C++. Για compiler χρησιμοποιείται ο AVR gcc και ως βασική βιβλιοθήκη C χρησιμοποιείται η AVR libc.

Λόγω της καταγωγής της από τη C, στη γλώσσα του Arduino, μπορούν να χρησιμοποιηθούν ουσιαστικά οι ίδιες βασικές εντολές και συναρτήσεις, με την ίδια σύνταξη, τους ίδιους τύπων δεδομένων και τους ίδιους τελεστές όπως και στη C. Πέρα από αυτές όμως, υπάρχουν κάποιες ειδικές εντολές, συναρτήσεις και σταθερές που βοηθούν για τη διαχείριση του ειδικού hardware του Arduino.

Τα προγράμματα του Arduino διαίρούνται σε τρία μέρη: δομή (structure), τιμές (values) και συναρτήσεις (functions).



```
sketch_sep09a | Arduino 1.0.1
Αρχείο Επιτήρησης Σχέδιο Εργαλεία Βοήθεια

sketch_sep09a.g
#include <Time.h>
#include <LiquidCrystal.h>
#include <Ethernet.h>
#include <SPI.h>

LiquidCrystal lcd(7, 6, 5, 4, 3, 2);
#define TIME_MSG_LEN 11
#define TIME_HEADER 'T'
String location = "/-icctest00123/arduino.php?device_id=dev&passwor
byte mac[] = {0x90, 0xA2, 0xDA, 0x00, 0xF9, 0xEF}; // mac address
int ledPin = 14;
int beepPin = 15;
int buttonPin = 8; // the number of the pushbutton pin
char serverName[] = "zafora.icte.uowm.gr"; // site
char inString[400]; // data table of parsing
int stringPos = 0; // count chars of parsing
int buttonState = 0; // variable for reading the pushbutton status
boolean startRead = false; // read data or not
boolean wrd_or_num = 0; // print message or time
boolean int_...

Μεταβίβαση σχεδίου...

308 Arduino Uno on COM8
```

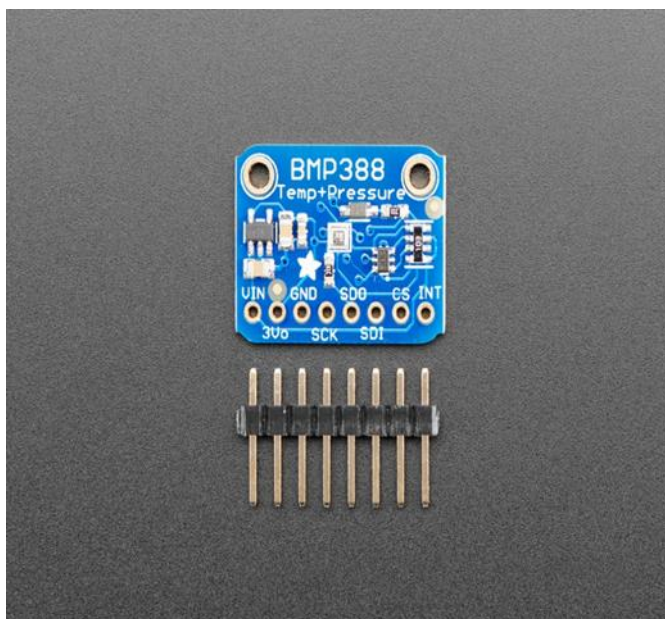
Εικόνα 2.15 Arduino IDE

2.3 Adafruit BMP388

Ο πρώτος αισθητήρας που χρησιμοποιήθηκε είναι ο Adafruit BMP388.

Ο Adafruit BMP388 είναι ένας αισθητήρας βαρομετρικής πίεσης μεγάλης ακριβείας που είναι κατασκευασμένος από την εταιρία Bosch. Η Bosch κατέχει ηγετική θέση στους αισθητήρες βαρομετρικής πίεσης, από το BMP085, BMP180 και BMP280, τώρα υπάρχει η επόμενη γενιά, ο αισθητήρας βαρομετρικής πίεσης ακριβείας Adafruit BMP388.

Ο αισθητήρας BMP388 έχει πολύ μεγάλη ακρίβεια, γεγονός που τον καθιστά εξαιρετικό για περιβαλλοντική ανίχνευση υψόμετρου με πολύ μεγάλη ακρίβεια. Ο αισθητήρας αυτός έχει σχετική ακρίβεια 8 Pascal, που μεταφράζεται σε περίπου $\pm 0,5$ μέτρα υψόμετρο. Λόγω της εξαιρετικά μεγάλης ακρίβειας που έχει ο αισθητήρας μπορεί να χρησιμοποιηθεί για drones και τετρακόπτερα, για να διατηρηθεί το υψόμετρο σταθερό. Με τον Adafruit BMP388 μπορεί επίσης να μετρηθεί η θερμοκρασία με ακρίβεια $\pm 0,5^{\circ}\text{C}$. Ο αισθητήρας μπορεί να χρησιμοποιηθεί με τάση 3 και 5 Volt. Μπορεί να χρησιμοποιηθεί ακόμη και σε διαμορφώσεις I2C αλλά και SPI. Ο BMP3xx είναι η επόμενη γενιά αισθητήρων της Bosch και είναι η αναβάθμιση του BMP280. Είναι αισθητήρας ο οποίος έχει πολύ χαμηλό θόρυβο και μπορεί να μετρήσει υψόμετρο έως και 0,1 m. Ο αισθητήρας επιφανειακής τοποθέτησης είναι κολλημένος σε PCB και διαθέτει ρυθμιστή 3,3 V και αλλαγή στάθμης, ώστε να μπορεί να χρησιμοποιηθεί με έναν λογικό μικροελεγκτή 3V ή 5V χωρίς ανησυχία.



Εικόνα 2.16 Αισθητήρας Adafruit BMP388

ΧΑΡΑΚΤΗΡΙΣΤΙΚΑ

Κατασκευαστής : Adafruit

Part Number: ADA3966

Καθαρό Βάρος: 0.001kg

Χώρα Προέλευσης: ΗΠΑ

Breadboard Friendly

Τύπος Αισθητήρα:

Βαρομετρικής Πίεσης

Υψομέτρου

Τυπική Τάση Εισόδου:

3VDC

3.3VDC

5VDC

Διασύνδεση: Ψηφιακή

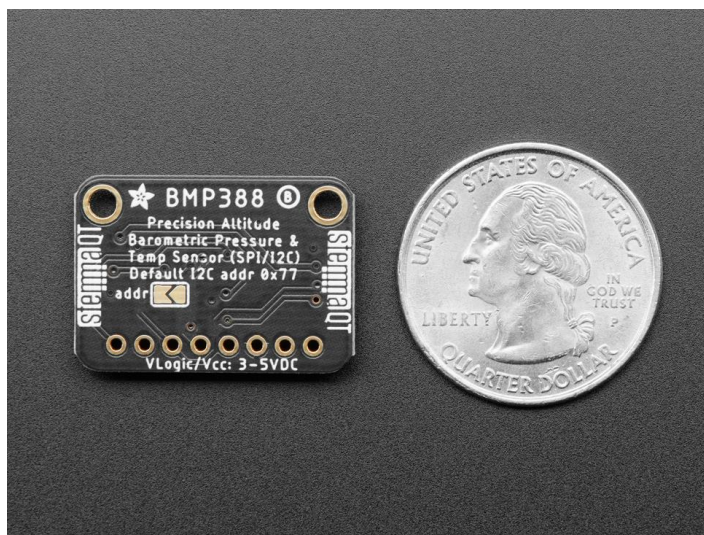
Πρωτόκολλο Επικοινωνίας: I2C ή

SPI

ΔΙΑΣΤΑΣΕΙΣ

21.6x16.6x3.0mm / 0.9"x0.7"x0.1"

Βάρος: 1.2g / 0.0oz



Εικόνα 2.17 Αισθητήρας Adafruit BMP388 σε αντιπαράθεση με ένα νόμισμα για να γίνει ευκολότερα κατανοητό το μέγεθος του

Στον παρακάτω πίνακα αποτυπώνετε η συνδεσμολογία που ακολούθησα.

| | |
|-----|--------|
| Vin | VCC 5V |
| GND | GND |
| SCL | A5 |
| SDA | A4 |

Πίνακας 2.1 Συνδεσμολογία αισθητήρα Adafruit BMP388

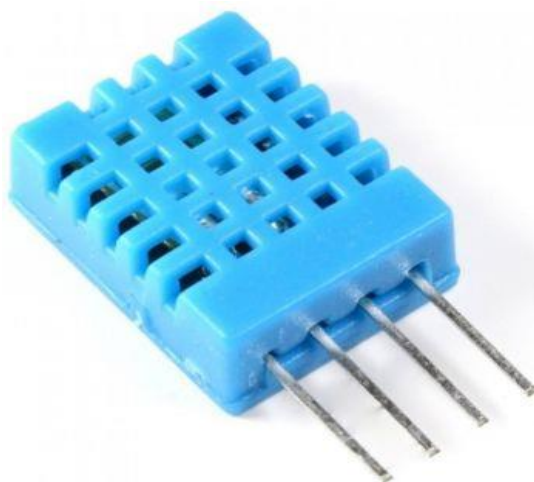
A5 και A4 είναι οι αναλογικές εισοδοι του μικροελεγκτή Arduino

2.4 DHT11

Ο DHT11 είναι ένας αισθητήρας υγρασίας θερμοκρασίας με βαθμονομημένη έξοδο ψηφιακού σήματος. Μπορεί να μετρήσει την θερμοκρασία μέσω μια αντίστασης και μέσω NTC. Τα αποτελέσματα συλλέγονται μέσω ενός μικροελεγκτή 8-bit προσφέροντας εξαιρετική ποιότητα και γρήγορη απόκριση. Κάθε αισθητήρας DHT11 είναι αυστηρά βαθμονομημένος αναφορικά με την υγρασία. Οι συντελεστές βαθμονόμησης αποθηκεύονται ως προγράμματα στη μνήμη OTP, τα οποία χρησιμοποιούνται για την ανίχνευση σήματος από τον αισθητήρα. Η σειριακή επαφή του αισθητήρα καθιστά την επικοινωνία του συστήματος γρήγορη και εύκολη. Το μικρό του μέγεθος, η χαμηλή κατανάλωση ενέργειας και η μετάδοση σήματος έως και 20 μέτρα τον καθιστούν την καλύτερη επιλογή για διάφορες εφαρμογές, συμπεριλαμβανομένων και των πιο απαιτητικών.

| | Εύρος Μέτρησης | Ακρίβεια (Υγρασία) | Ακρίβεια (Θερμοκρασία) |
|-------|-------------------|-----------------------|---------------------------|
| DHT11 | 20-90% RH 0-50 °C | ±5% RH | ±2°C |

Πίνακας 2.2 Εύρος μετρήσεων του αισθητήρα DHT11



Εικόνα 2.18 Αισθητήρας DHT11

Παρακάτω φαίνεται η συνδεσμολογία που ακολουθήθηκε:

| | |
|---|-----|
| V | 5V |
| S | A1 |
| G | GND |

Πίνακας 2.3 Συνδεσμολογία Ακροδεκτών

A1 είναι η αναλογική είσοδος του μικροελεγκτή Arduino

ΧΑΡΑΚΤΗΡΙΣΤΙΚΑ

Κατασκευαστής: OEM

Part Number: DHT11

Καθαρό Βάρος: 0.006kg

Χώρα Προέλευσης: Κίνα

Breadboard Friendly

Τύπος Αισθητήρα: Θερμοκρασίας – Υγρασίας

Τυπική Τάση Εισόδου:

3VDC

3.3VDC

5VDC

Ρεύμα Λειτουργίας: 2.5mA

Διασύνδεση: Ψηφιακή

Πρωτόκολλο Επικοινωνίας: Χωρίς

ΕΠΙΠΛΕΟΝ ΧΑΡΑΚΤΗΡΙΣΤΙΚΑ

Λειτουργεί με τάση από 3 V έως 5 VDC.

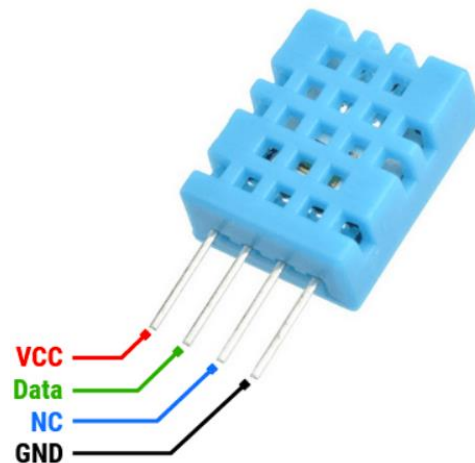
Χαμηλή κατανάλωση 2.5mA max

Ιδανικό για μέτρηση μεταξύ 20-80% (υγρασία) με ακρίβεια 5%

Ιδανικό για μέτρηση 0-50°C (θερμοκρασία) με ακρίβεια $\pm 2^{\circ}\text{C}$

Δειγματοληψία ανά ένα δευτερόλεπτο (1HZ)

Ψηφιακή έξοδος



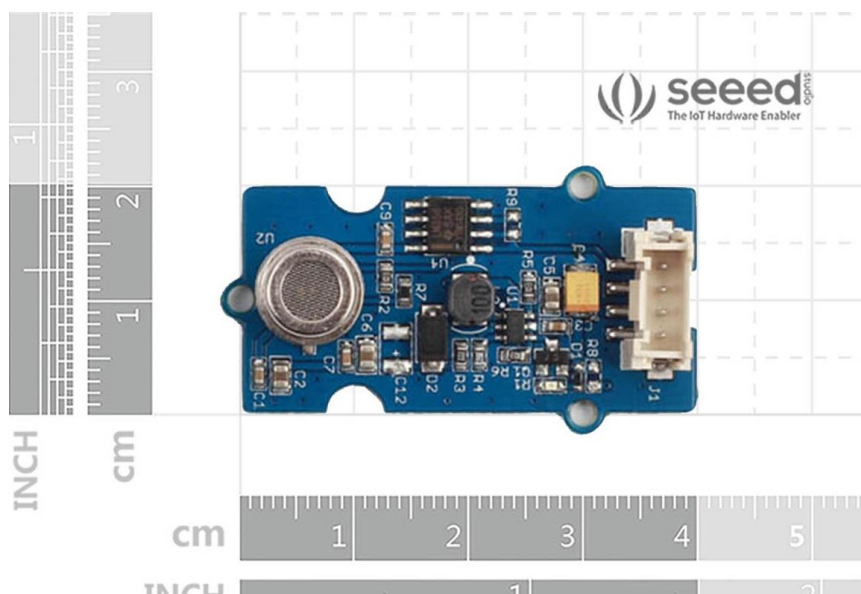
Εικόνα 2.19 Επεξήγηση ακροδεκτών του αισθητήρα DHT11

2.5 Grove - Air Quality Sensor v1.3

Αυτός ο αισθητήρας έχει σχεδιαστεί για τον έλεγχο ποιότητας αέρα σε εσωτερικό χώρο. Ανταποκρίνεται σε ένα ευρύ φάσμα επιβλαβών αερίων, όπως μονοξείδιο του άνθρακα, αλκοόλ, ακετόνη, διαλυτικό, φορμαλδεΐδη. Λόγω του μηχανισμού μέτρησής του, αυτός ο αισθητήρας δεν μπορεί να εξάγει συγκεκριμένα δεδομένα για να περιγράψει ποσοτικά τις συγκεντρώσεις των αερίων.

Λειτουργεί με 5V καθώς και με 3,3V, μπορεί να λειτουργήσει με Arduino και Raspberry Pi. Λόγω της μεγάλης σταθερότητάς του και της χαμηλής κατανάλωσης ενέργειας που έχει ο αισθητήρας, είναι μια καλή επιλογή για την παρακολούθηση της ποιότητας του αέρα σε οποιονδήποτε χώρο.

Ο συγκεκριμένος αισθητήρας αν εκτίθεται σε πολύ μολυσμένο αέρα για μεγάλο χρονικό διάστημα, μπορεί να εξασθενίσει σημαντικά την ευαισθησία του .



Εικόνα 2.20 μέτρηση του μεγέθους του αισθητήρα Grove - Air Quality Sensor v1.3 σε ίντσες και εκατοστά

| | |
|--------------------------------------|--|
| Air fresh | Υποδηλώνει καλή κατάσταση του αέρα |
| Low pollution | Υποδηλώνει ότι υπάρχει χαμηλή συγκέντρωση αερίων-στόχων |
| High pollution | Υποδηλώνει ότι υπάρχει μεγάλη συγκέντρωση αερίων-στόχων |
| High pollution “Force signal active” | Υποδηλώνει ότι υπάρχει πολύ μεγάλη συγκέντρωση αερίων-στόχων |

Πίνακας 2.4 Υπόμνημα ενδείξεων του αισθητήρα Grove - Air Quality Sensor v1.3



Εικόνα 2.21 Αισθητήρας Grove - Air Quality Sensor v1.3

Παρακάτω φαίνεται η συνδεσμολογία που ακολουθήθηκε:

| | |
|--------|-----------|
| Black | GND |
| Red | VCC +5V |
| Yellow | Signal A0 |

Πίνακας 2.5 Συνδεσμολογία Ακροδεκτών του αισθητήρα Grove - Air Quality Sensor v1.3

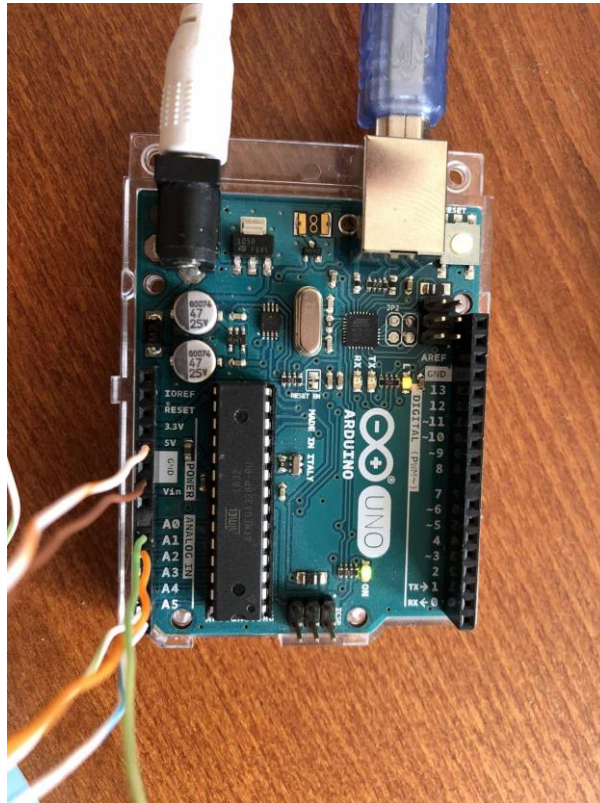
A0 είναι η αναλογική είσοδος του μικροελεγκτή Arduino

3 Κατασκευή και λειτουργία της εργασίας

Σε αυτό το κεφάλαιο θα γίνει μια περιγραφή για το πως κατασκευάστηκε ο σταθμός ελέγχου ποιότητας αέρα αλλά και το πως λειτουργεί.

3.1 Κατασκευή

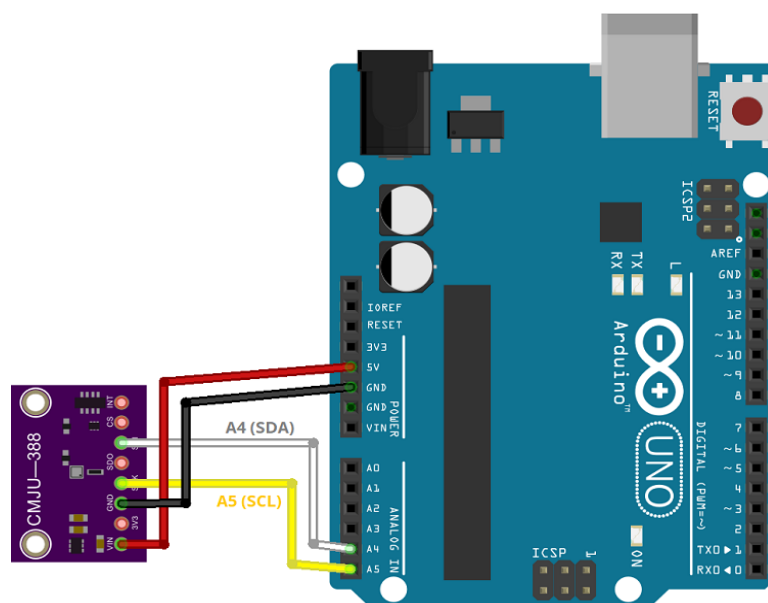
Η αρχή γίνεται με τον μικροϋπολογιστή Arduino. Το Arduino συνδέεται σε ανεξάρτητη τροφοδοσία 12V και μετά ξεκινά η σύνδεση των αισθητήρων πάνω στον μικροϋπολογιστή.



Εικόνα 3.1 Μικροϋπολογιστής Arduino κατά την σύνδεση των αισθητήρων

3.1.1 Adafruit BMP388

Ο πρώτος αισθητήρας που συνδέεται είναι ο Adafruit BMP388. Όπως αναφέρεται και παραπάνω ο Adafruit BMP388 είναι ένας αισθητήρας βαρομετρικής πίεσης μεγάλης ακριβείας που είναι κατασκευασμένος από την εταιρία Bosch. Στη συγκεκριμένη μελέτη χρησιμοποιήθηκε για τη μέτρηση της βαρομετρικής πίεσης καθώς και για τη μέτρηση της θερμοκρασίας.



Εικόνα 3.2 Διάγραμμα σύνδεσης του αισθητήρα Adafruit BMP388

Στον παρακάτω πίνακα φαίνεται ο τρόπος με τον οποίο συνδέεται ο Adafruit BMP388 με το Arduino.

| | |
|-----|--------|
| Vin | VCC 5V |
| GND | GND |
| SCL | A5 |
| SDA | A4 |

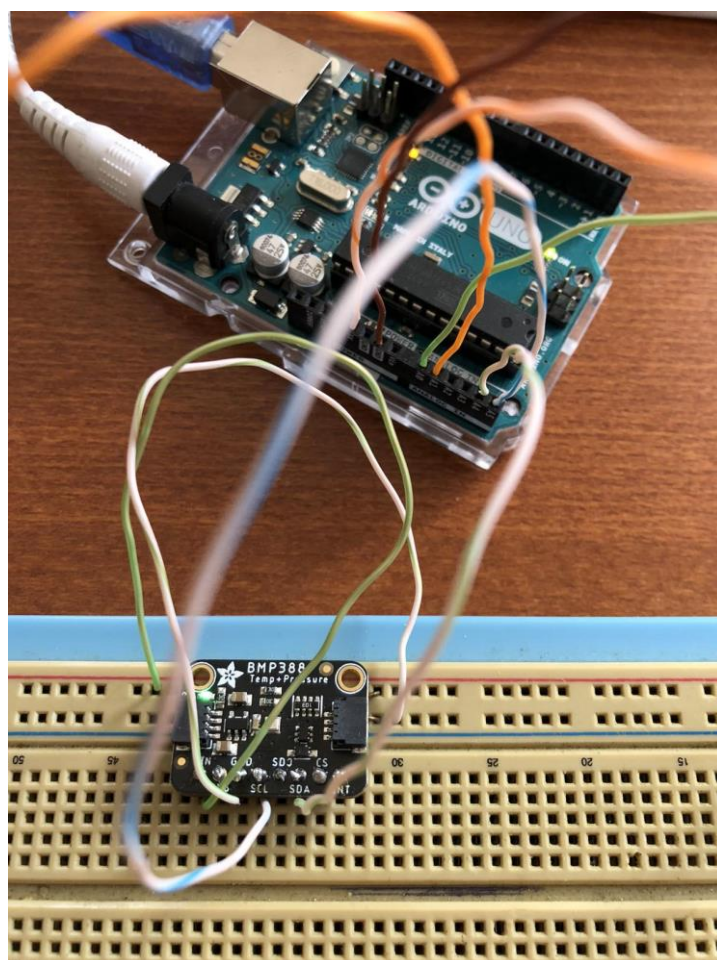
Πίνακας 3.1 Πίνακας συνδεσμολογίας ανάμεσα στο Arduino και τον αισθητήρα Adafruit BMP388.

Αριστερά φαίνονται οι ακροδέκτες του αισθητήρα Adafruit BMP388 και δεξιά οι ακροδέκτες του Arduino. Οι θέσεις A5 και A4 είναι οι αναλογικές εισοδοί του Arduino.

Υλοποίηση σταθμού ελέγχου ποιότητας αέρα με την χρήση Arduino



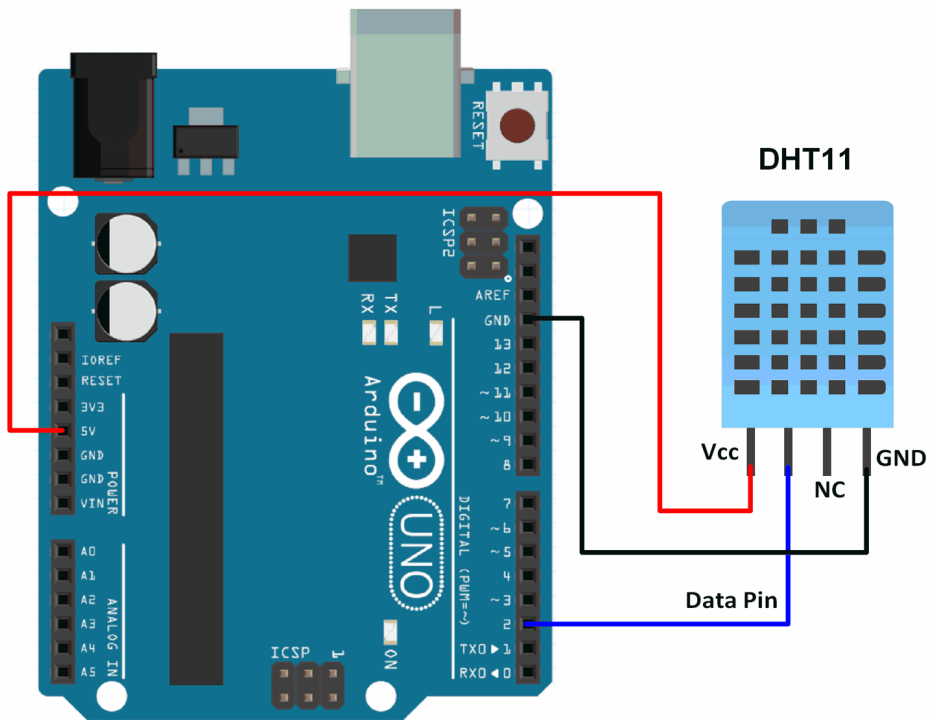
Εικόνα 3.3 Αισθητήρας Adafruit BMP388



Εικόνα 3.4 Αισθητήρας Adafruit BMP388 κατά την σύνδεση του στον μικροϋπολογιστή Arduino

3.1.2 DHT11

Ο επόμενος αισθητήρας είναι ο DHT11. Ο DHT11 είναι ένας αισθητήρας υγρασίας θερμοκρασίας. Από τα δυο φυσικά στοιχεία που μετράει ο συγκεκριμένος αισθητήρας προτιμήθηκε η συλλογή μετρήσεων μόνο για την υγρασία καθώς ο Adafruit BMP388 έχει μεγαλύτερη ακρίβεια στη μέτρηση θερμοκρασίας.



Εικόνα 3.5 Διάγραμμα σύνδεσης του αισθητήρα DHT11

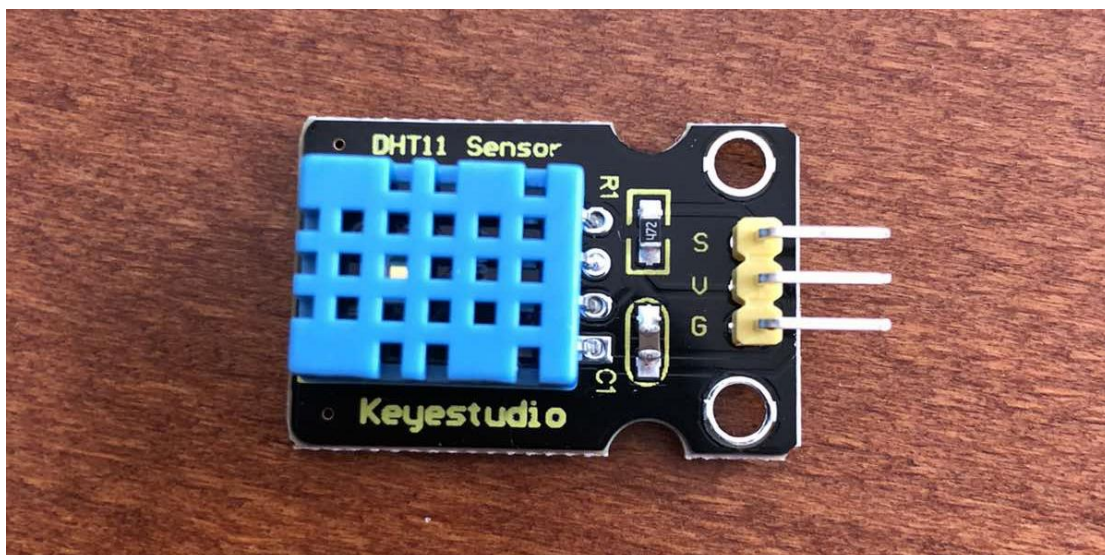
Στον παρακάτω πίνακα φαίνεται η συνδεσμολογία που ακολουθήθηκε για να συνδεθεί ο αισθητήρας με το Arduino .

| | |
|---|-----|
| 1 | 5V |
| 2 | A1 |
| 4 | GND |

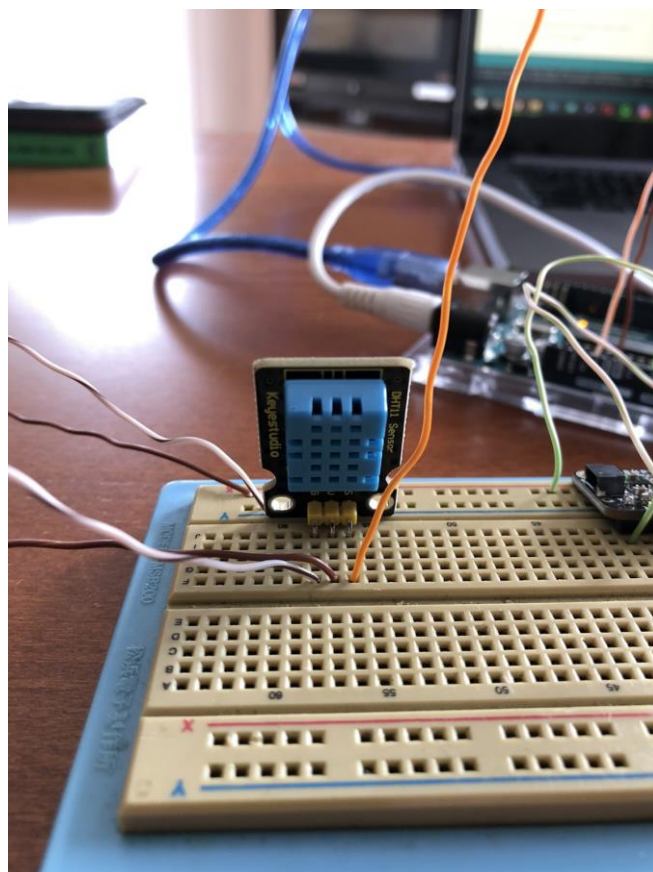
Πίνακας 3.2 Πίνακας συνδεσμολογίας ανάμεσα στο Arduino και τον αισθητήρα DHT11

Αριστερά φαίνονται οι ακροδέκτες του αισθητήρα DHT11 και δεξιά οι ακροδέκτες του Arduino. Η θέση A1 είναι η αναλογική είσοδος του Arduino.

Υλοποίηση σταθμού ελέγχου ποιότητας αέρα με την χρήση Arduino



Εικόνα 3.6 Αισθητήρας DHT11

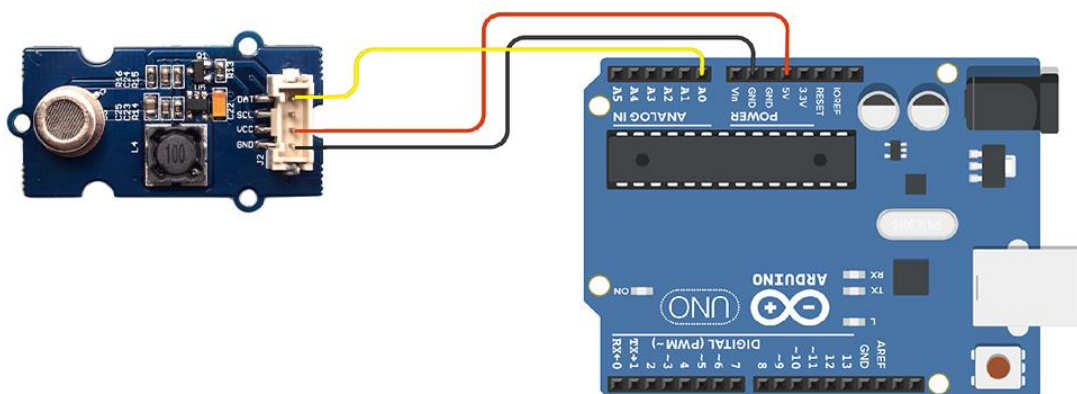


Εικόνα 3.7 Αισθητήρας DHT11 κατά τη σύνδεση του στον μικροεπεξεργαστή Arduino

3.1.3 Grove - Air Quality Sensor v1.3

Ο τελευταίος αισθητήρας που αξιοποιήθηκε για τη συγκεκριμένη μελέτη είναι ο Grove - Air Quality Sensor v1.3.

Αυτός ο αισθητήρας έχει σχεδιαστεί για έλεγχο ποιότητας αέρα εσωτερικού χώρου, και ανταποκρίνεται σε ένα ευρύ φάσμα επιβλαβών αερίων, όπως μονοξείδιο του άνθρακα, αλκοόλ, ακετόνη, διαλυτικό, φορμαλδεΰδη. Στον παρακάτω πίνακα φαίνεται ο τρόπος με τον οποίο έγινε η συνδεσμολογία του αισθητήρα.



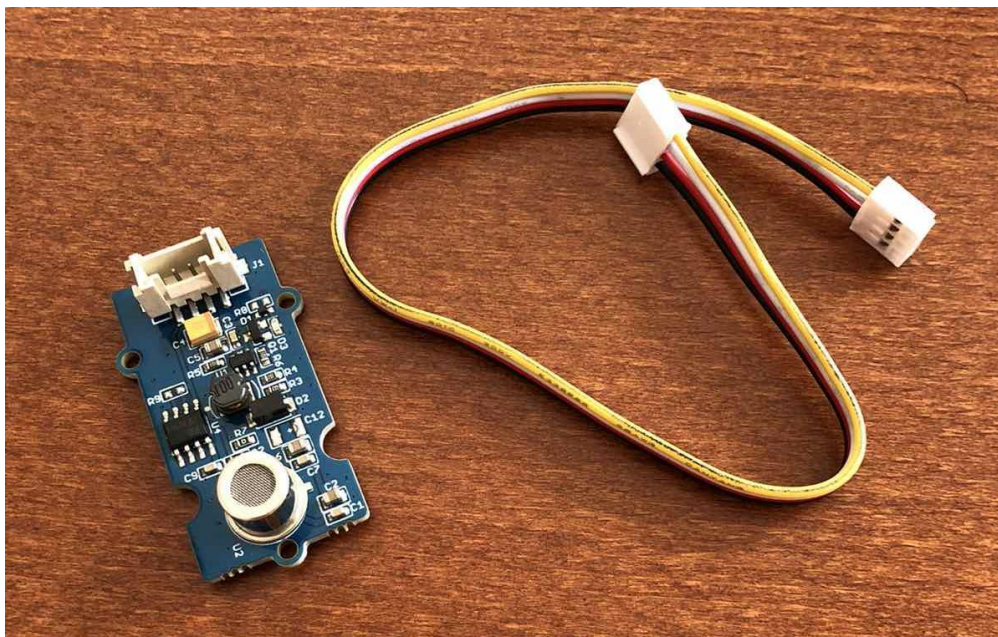
Εικόνα 3.8 Διάγραμμα σύνδεσης του αισθητήρα Grove - Air Quality Sensor v1.3

| | |
|--------|-----------|
| Black | GND |
| Red | VCC +5V |
| Yellow | Signal A0 |

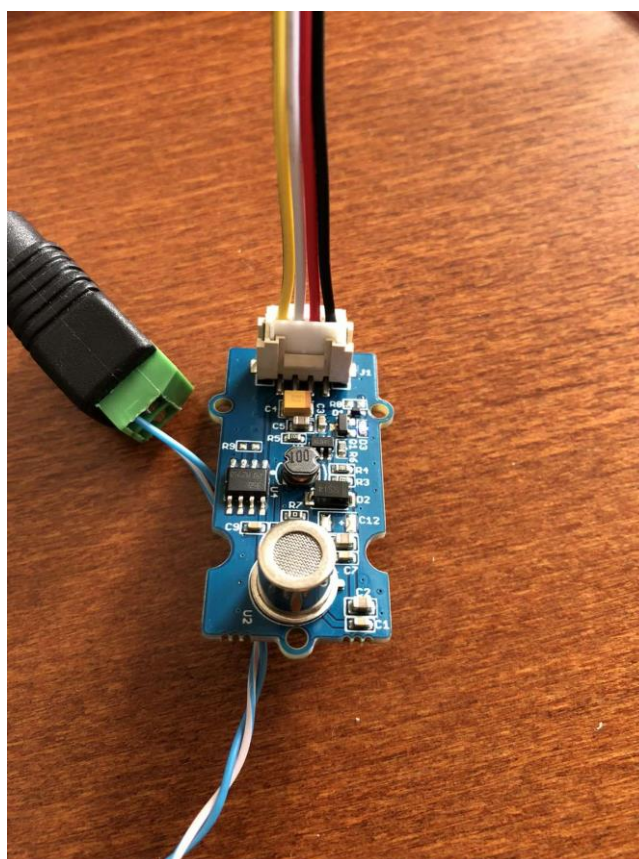
Πίνακας 3.3 Πίνακας συνδεσμολογίας ανάμεσα στο Arduino και τον αισθητήρα Grove - Air Quality Sensor v1.3.

Αριστερά φαίνονται οι ακροδέκτες του αισθητήρα Grove - Air Quality Sensor v1.3 και δεξιά οι ακροδέκτες του Arduino. Η θέση A0 είναι η αναλογική είσοδος του Arduino.

Υλοποίηση σταθμού ελέγχου ποιότητας αέρα με την χρήση Arduino



Εικόνα 3.9 Αισθητήρας Grove - Air Quality Sensor v1.3



Εικόνα 3.10 Αισθητήρας Grove - Air Quality Sensor v1.3 κατά την σύνδεση του στον μικροϋπολογιστή Arduino

3.2 Κώδικας

Ο κώδικας του σταθμού ελέγχου ποιότητας αέρα γράφτηκε στην εφαρμογή Arduino IDE, την εφαρμογή που χρειάζεται για να δημιουργηθεί οποιοδήποτε πρόγραμμα για έναν μικροϋπολογιστή Arduino. Για να μπορέσουν να χρησιμοποιηθούν οι παραπάνω αισθητήρες θα πρέπει να ενσωματωθούν στον κώδικα κάποιες βιβλιοθήκες.

Οι βιβλιοθήκες είναι οι εξής :

- Adafruit_Sensor.h
- Adafruit_BMP3XX.h
- AirQuality.h
- Arduino.h
- dht.h

Παρακάτω παρατίθεται ο κώδικας της πτυχιακής εργασίας καθώς και κάποια απαραίτητα σχόλια για την κατανόηση του κώδικα .

3.2.1 Κώδικας σταθμού ελέγχου ποιότητας αέρα

```
#include <Adafruit_Sensor.h>
#include "Adafruit_BMP3XX.h"
#include "AirQuality.h"
#include "Arduino.h"
#include <dht.h>

#define SEALEVELPRESSURE_HPA (1013.25)
#define dht_apin A1 // Analog Pin sensor is connected to

//Τιμές και κείμενο Υγρασίας
#define humidity_lowText "Πολύ Χαμηλή"
#define humidity_lowValue 30
#define humidity_goodText "Ιδανική"
#define humidity_goodValue 45
#define humidity_highText "Πολύ Υψηλή"
#define humidity_highValue 60
```

//Τιμές και κείμενο Θερμοκρασίας

```
#define temperature_veryLowText "Πολύ Ψυχρή"  
#define temperature_veryLowValue 0  
#define temperature_lowText "Ψυχρή"  
#define temperature_lowValue 10  
#define temperature_goodText "Ιδανική"  
#define temperature_goodValue 20  
#define temperature_highText "Θερμή"  
#define temperature_highValue 30  
#define temperature_veryHighText "Πολύ Θερμή"  
#define temperature_veryHighValue 40
```

```
dht DHT;  
Adafruit_BMP3XX bmp;  
AirQuality airqualitysensor;  
int current_quality =-1;
```

```
void setup() {
```

//Εγκατάσταση – αρχικοποίηση του αισθητήρα BMP388

```
Serial.begin(115200);  
while (!Serial);  
  
if (!bmp.begin_I2C()) { // hardware I2C mode, can pass in address & alt Wire  
  Serial.println("Δεν βρέθηκε ο BMP3 αισθητήρας, ελέγξτε την καλωδίωση!");  
  while (1);  
}
```

//Ρύθμιση υπερδειγματοληψίας και αρχικοποίησης φίλτρου

```
bmp.setTemperatureOversampling(BMP3_OVERSAMPLING_8X);
```

```
bmp.setPressureOversampling(BMP3_OVERSAMPLING_4X);  
bmp.setIIRFilterCoeff(BMP3_IIR_FILTER_COEFF_3);  
bmp.setOutputDataRate(BMP3_ODR_50_HZ);
```

//Τέλος εγκατάστασης – αρχικοποίησης του αισθητήρα BMP388

//Εγκατάσταση – αρχικοποίηση του αισθητήρα DHT11

```
Serial.begin(9600);  
delay(500); //Delay to let system boot  
//Serial.println("Αισθητήρας Υγρασίας & Θερμότητας DHT11\n\n");  
delay(1000);
```

//Τέλος εγκατάσταση – αρχικοποίηση του αισθητήρα DHT11

//Εγκατάσταση – αρχικοποίηση του αισθητήρα Air Quality Sensor setup

```
Serial.begin(9600);  
airqualitysensor.init(14);
```

//Τέλος εγκατάσταση – αρχικοποίηση του αισθητήρα Air Quality Sensor setup

```
}
```

//Τέλος εγκατάσταση – αρχικοποίηση

```
void loop() {
```

//Loop του αισθητήρα BMP388

```
if (!bmp.performReading()) {  
  Serial.println("Αδυναμία διαβάσματος BMP αισθητήρα :(");  
  return;  
}
```

//Έλεγχος Θερμοκρασίας του αισθητήρα BMP388 για πιθανές τιμές

```
if(bmp.temperature <= temperature_veryLowValue){
```

```
Serial.print("Θερμοκρασία ");
Serial.print(bmp.temperature);
Serial.print("C ");
Serial.println(temperature_veryLowText);
} else if(bmp.temperature >= temperature_veryLowValue && bmp.temperature <=
temperature_lowValue){
    Serial.print("Θερμοκρασία ");
    Serial.print(bmp.temperature);
    Serial.print("C ");
    Serial.println(temperature_lowText);
} else if(bmp.temperature >= temperature_lowValue && bmp.temperature <=
temperature_goodValue){
    Serial.print("Θερμοκρασία ");
    Serial.print(bmp.temperature);
    Serial.print("C ");
    Serial.println(temperature_goodText);
} else if(bmp.temperature >= temperature_goodValue && bmp.temperature <=
temperature_highValue){
    Serial.print("Θερμοκρασία ");
    Serial.print(bmp.temperature);
    Serial.print("C ");
    Serial.println(temperature_highText);
} else {
    Serial.print("Θερμοκρασία ");
    Serial.print(bmp.temperature);
    Serial.print("C ");
    Serial.println(temperature_veryHighText);
}

Serial.print("Ατμοσφαιρική Πίεση = ");
Serial.print(bmp.pressure / 100.0);
Serial.println("hPa");

Serial.print("Υψόμετρο = ");
```

```
Serial.print(bmp.readAltitude(SEALEVELPRESSURE_HPA));  
Serial.println("m");  
delay(2000);
```

//Τέλος του loop του αισθητήρα BMP388

//Loop του αισθητήρα DHT11

```
DHT.read11(dht_apin);
```

//Ελεγχος Υγρασίας του αισθητήρα DHT11 για πιθανές τιμές

```
if(DHT.humidity <= humidity_lowValue){  
    Serial.print("Υγρασία ");  
    Serial.print(DHT.humidity);  
    Serial.print("% ");  
    Serial.println(humidity_lowText);  
} else if(DHT.humidity >= humidity_lowValue && DHT.humidity <=  
humidity_goodValue){  
    Serial.print("Υγρασία ");  
    Serial.print(DHT.humidity);  
    Serial.print("% ");  
    Serial.println(humidity_goodText);  
} else {  
    Serial.print("Υγρασία ");  
    Serial.print(DHT.humidity);  
    Serial.print("% ");  
    Serial.println(humidity_highText);  
}
```

```
delay(2000);
```

// Περιμένετε 2 δευτερόλεπτα πριν αποκτήσει ξανά πρόσβαση στον αισθητήρα.

//Τέλος loop του αισθητήρα DHT11

// Τέλος loop του αισθητήρα Air Quality Sensor

```
current_quality=airqualitysensor.slope();
if (current_quality >= 0)// if a valid data returned.
{
  if (current_quality==0)
    Serial.println("Προσοχή! Εντοπίστηκε υψηλή ατμοσφαιρική μόλυνση!");
  else if (current_quality==1)
    Serial.println("Ατμοσφαιρική μόλυνση!");
  else if (current_quality==2)
    Serial.println("Χαμηλή ατμοσφαιρική μόλυνση!");
  else if (current_quality ==3)
    Serial.println("Καθαρός αέρας!");
  Serial.println();
}
```

// Τέλος loop του αισθητήρα Air Quality Sensor

}

// Τέλος loop

ISR(TIMER1_OVF_vect)

```
{
  if(airqualitysensor.counter==61)//set 2 seconds as a detected duty
  {
    airqualitysensor.last_vol=airqualitysensor.first_vol;
    airqualitysensor.first_vol=analogRead(A0);
    airqualitysensor.counter=0;
    airqualitysensor.timer_index=1;
    PORTB=PORTB^0x20;
  }
  else
  {
    airqualitysensor.counter++;
  }
}
```


3.2.2 Κώδικας εφαρμογής παρουσίασης των αποτελεσμάτων

Παρακάτω παρουσιάζεται ο κώδικας που γράφτηκε σε γλώσσα Python για την παρουσίαση των τιμών του που εξάγονται από τους αισθητήρες σε γραφήματα σε συνάρτηση με το χρόνο, καθώς και κάποια σχόλια για την κατανόηση του κώδικα.

```
# Ορίζουμε τις βιβλιοθήκες που θα χρησιμοποιήσουμε
import matplotlib.pyplot as plt
import numpy as np

def plot_measurement_values(values, measurement_type, unit):
    """Συνάρτηση που πλοτάρει τις τιμές που μετρήθηκαν ως προς τον χρόνο

    Δέχεται τις παραμέτρους
    values: Η λίστα των τιμών που μετρήθηκαν
    measurement_type: Τι μετρήθηκε
    unit: Μονάδα μέτρησης
    """
    # Χρόνος μεταξύ δυο διαδοχικών μετρήσεων σε δευτερόλεπτα
    time_between_measurements_in_seconds = 0.1

    dt = time_between_measurements_in_seconds

    # Δημιουργία τιμών για τον άξονα του χρόνου ξεκινώντας από το 0.0
    t = np.arange(0.0, dt * len(values), dt)

    # Δημιουργία των plot objects
    fig, ax = plt.subplots()

    ax.plot(t, values)

    # Ορίζουμε τον τίτλο του γραφήματος, του άξονα X και Y
    ax.set(xlabel='time (s)', ylabel=f'{measurement_type} ({unit})',
           title=f'{measurement_type} measurement')
    ax.grid()

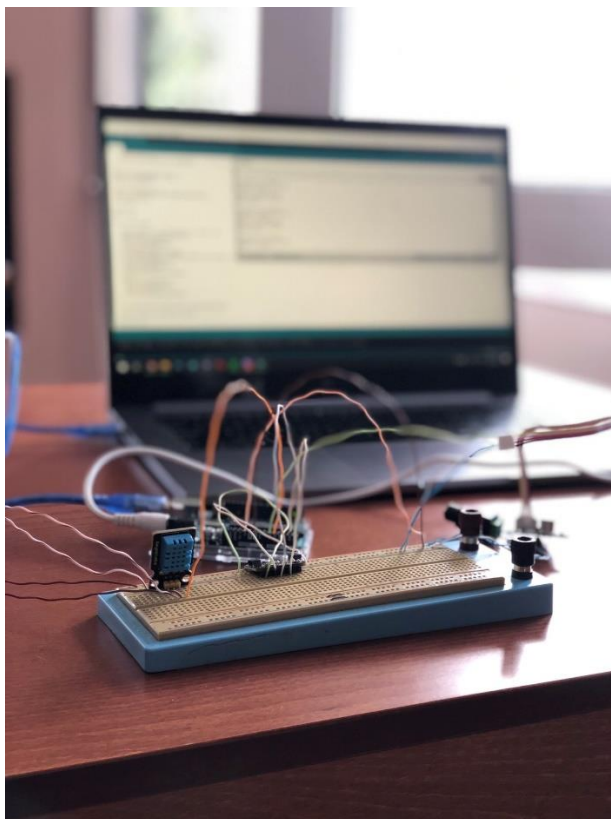
    # Εμφάνιση του γραφήματος
    plt.show()

# Διάβασμα αρχείου με τις μετρήσεις
with open('test2.txt', 'r') as f:
    s = f.read()
```

```
# Ορίζουμε μια δομή όπου θα αποθηκεύσουμε τις μετρήσεις μας
# Χρησιμοποιούμε την δομή dict με keys τα είδη των μετρήσεων (temperature,
pressure, altitude)
# και values ένα dict με τα δεδομένα των μετρήσεων
measurements = {
    'temperature': {
        'measurement_type': 'Temperature', # τι μετρήθηκε
        'values': [], # τιμές των μετρήσεων
        'unit': 'C' # μονάδα μέτρησης
    },
    'pressure': {
        'measurement_type': 'Pressure',
        'values': [],
        'unit': 'hPa'
    },
    'altitude': {
        'measurement_type': 'Altitude',
        'values': [],
        'unit': 'm'
    }
}
# Η μορφή του αρχείου είναι η εξής:
# τιμή1,τιμή2,τιμή3_τιμή4,τιμή5,τιμή6_τιμή7,τιμή8,τιμή9 ...
for line in s.split('_'): # κάνουμε split τις τιμές ανά 3-άδες
    try:
        # Μετατρέπουμε κάθε τριάδα τιμών από τύπο string σε τύπο float
        values = [float(x) for x in line.split(',')]
        # και τις αποθηκεύουμε στη δομή μας
        measurements['temperature']['values'].append(values[0])
        measurements['pressure']['values'].append(values[1])
        measurements['altitude']['values'].append(values[2])
    except ValueError:
        # Εάν δεν υπάρχει αριθμός αγνοείται όλο το περιεχόμενο της line
        pass
```

3.3 Ανάλυση της λειτουργίας του σταθμού

Η λειτουργία του σταθμού είναι σχετικά απλή. Αρχικά συνδέεται η τροφοδοσία στον σταθμό και έπειτα με τη χρήση ενός καλωδίου USB συνδέεται ο σταθμός με μία συσκευή η οποία τρέχει την εφαρμογή Arduino IDE. Αυτή η συσκευή θα μπορούσε να είναι ένας σταθερός ηλεκτρονικός υπολογιστής ή ένας φορητός ηλεκτρονικός υπολογιστής καθώς ακόμα και ένα κινητό τηλέφωνο Android. Έπειτα γίνεται εκκίνηση της εφαρμογής Arduino IDE και μέσα από την εφαρμογή ανοίγεται η σειριακή θύρα του Arduino. Μόλις γίνει το πρώτο boot του μικροϋπολογιστή ξεκινούν να εμφανίζονται στην οθόνη οι μετρήσεις των αισθητήρων.



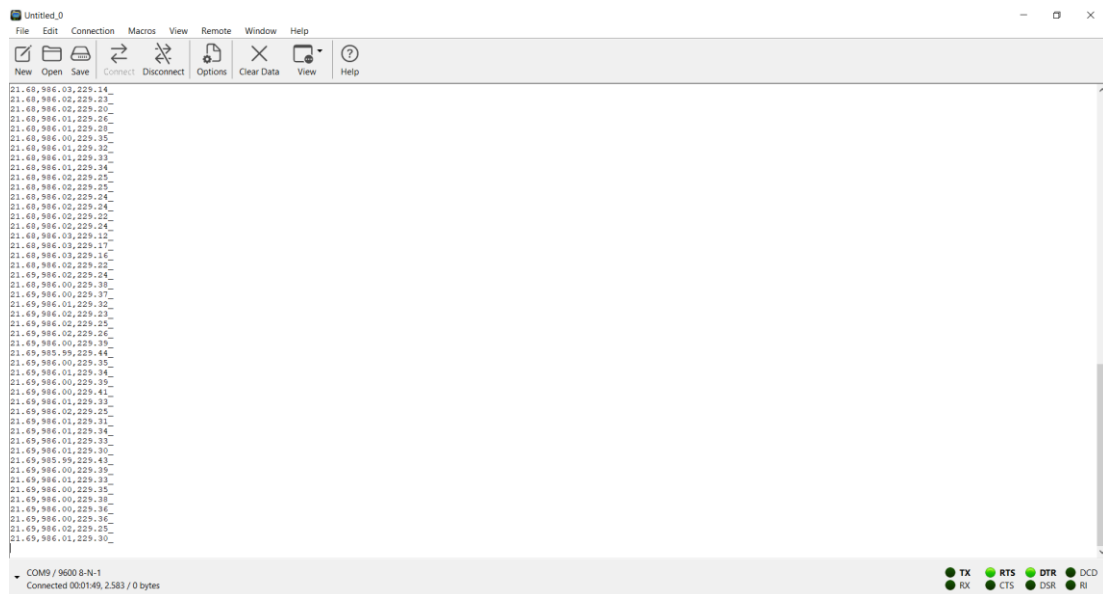
Εικόνα 3.11 Φωτογραφία του σταθμού κατά την λειτουργία του

```
COM8  
|  
|  
Υψόμετρο = 144.85m  
Υγρασία 0.00% Πολύ Χαμηλή  
sensor_value:25 Air fresh  
Καθαρός αέρας!  
  
Θερμοκρασία 27.17C Θερμή  
Ατμοσφαιρική Πίεση = 995.98hPa  
Υψόμετρο = 144.78m  
Υγρασία 0.00% Πολύ Χαμηλή  
sensor_value:26 Air fresh  
Καθαρός αέρας!  
  
Θερμοκρασία 27.18C Θερμή  
Ατμοσφαιρική Πίεση = 995.98hPa  
Υψόμετρο = 144.77m  
Υγρασία 0.00% Πολύ Χαμηλή  
sensor_value:27 Air fresh  
Καθαρός αέρας!
```

Εικόνα 3.12 Εμφάνιση των μετρήσεων που παίρνουν οι αισθητήρες παρακολουθώντας την σειριακή θύρα του Arduino

Για την παρουσίαση των αποτελεσμάτων όπως αναφέρθηκε και παραπάνω δημιουργήθηκε μια εφαρμογή σε γλώσσα Python. Για την εξαγωγή των δεδομένων σε ένα .txt αρχείο και την ενσωμάτωση στο πρόγραμμα χρησιμοποιήθηκε η εφαρμογή CoolTermWin.

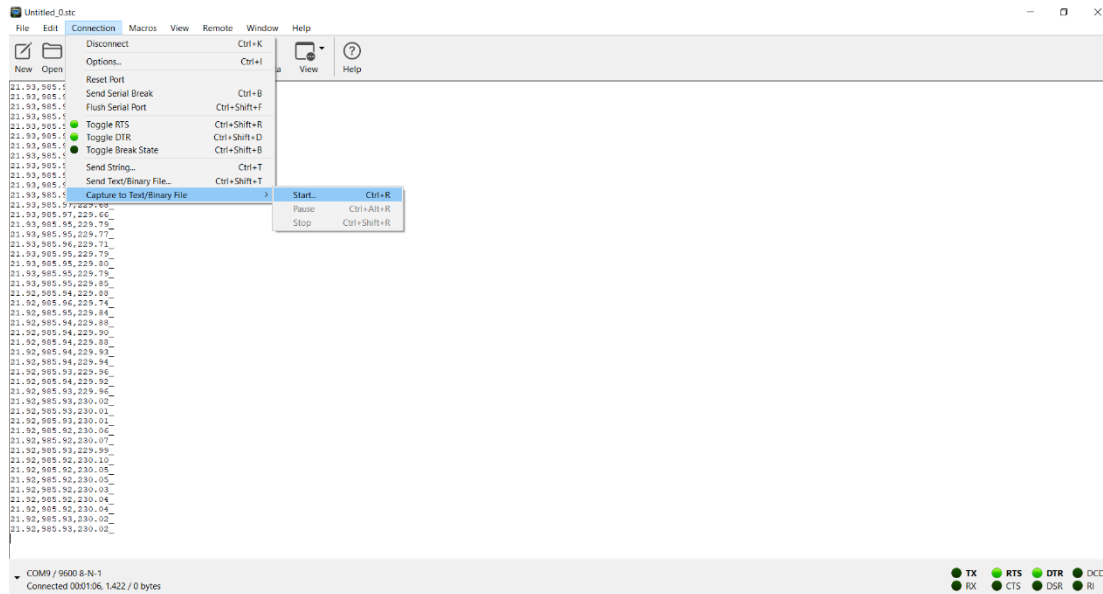
- Αρχικά τίθεται σε λειτουργία ο σταθμός.
- Μετά ο χρήστης ανοίγει την εφαρμογή CoolTerm και επιλέγει Connect ώστε να ξεκινήσει η εφαρμογή να παρακολουθεί τη σειριακή θύρα του υπολογιστή.



Εικόνα 3.13 Στιγμιότυπο από την εφαρμογή CoolTerm

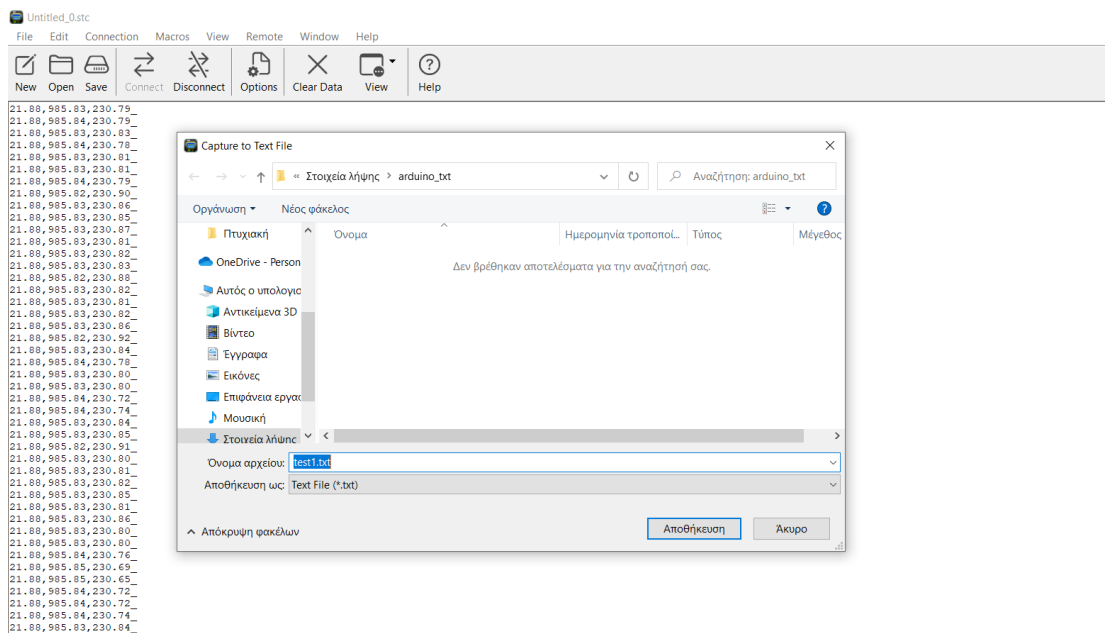
Υλοποίηση σταθμού ελέγχου ποιότητας αέρα με την χρήση Arduino

- Έπειτα ο χρήστης επιλέγει Start για να ξεκινήσει να καταγραφεί το πρόγραμμα.



Εικόνα 3.14 Στιγμιότυπο από την εφαρμογή CoolTerm

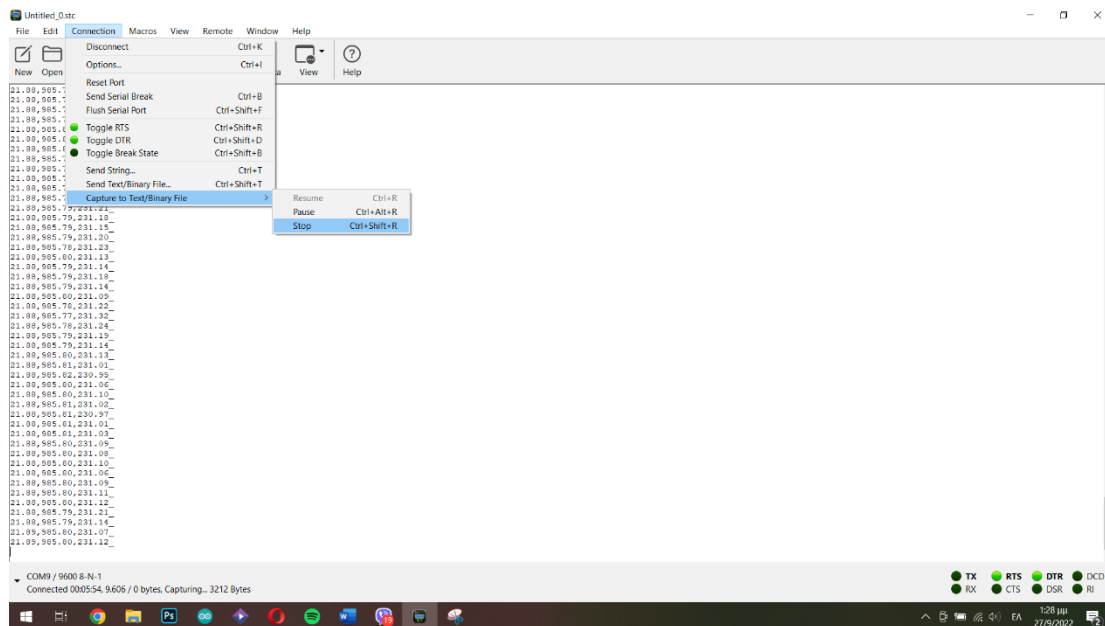
- Στην συνέχεια επιλέγει που θα αποθηκευτεί το αρχείο.



Εικόνα 3.15 Στιγμιότυπο από την εφαρμογή CoolTerm

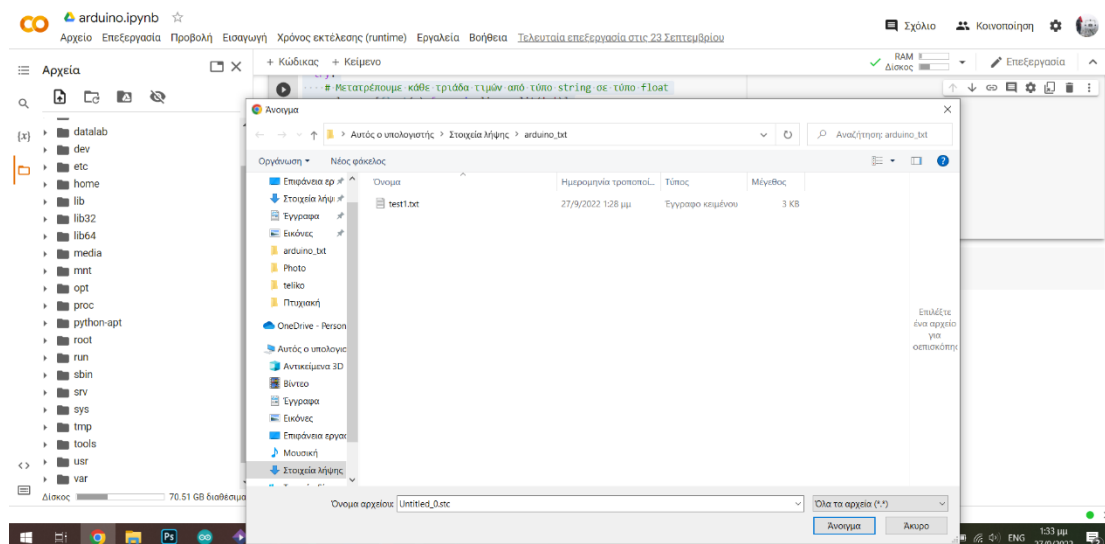
Υλοποίηση σταθμού ελέγχου ποιότητας αέρα με την χρήση Arduino

- Μετά επιλέγει Stop για να σταματήσει η καταγραφή των τιμών.



Εικόνα 3.16 Στιγμιότυπο από την εφαρμογή CoolTerm

- Στην συνέχεια ενσωματώνει το αρχείο με τις τιμές στην εφαρμογή που δημιουργήθηκε με τη χρήση της Python.

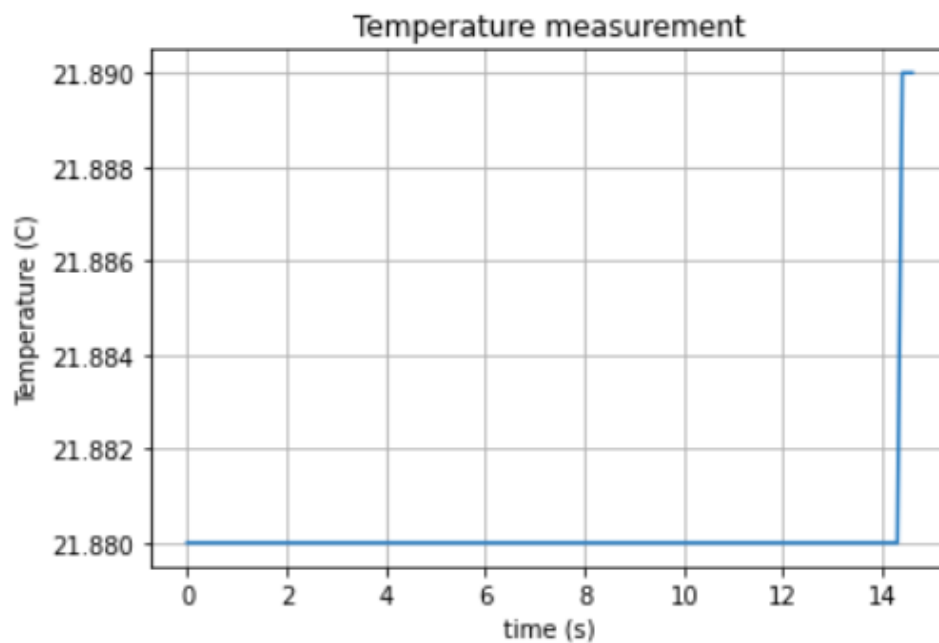


Εικόνα 3.17 Στιγμιότυπο από την εφαρμογή CoolTerm

- Στην συνέχεια ο χρήστης βάζει το όνομα του αρχείου στον κώδικα της εφαρμογής και την τρέχει.

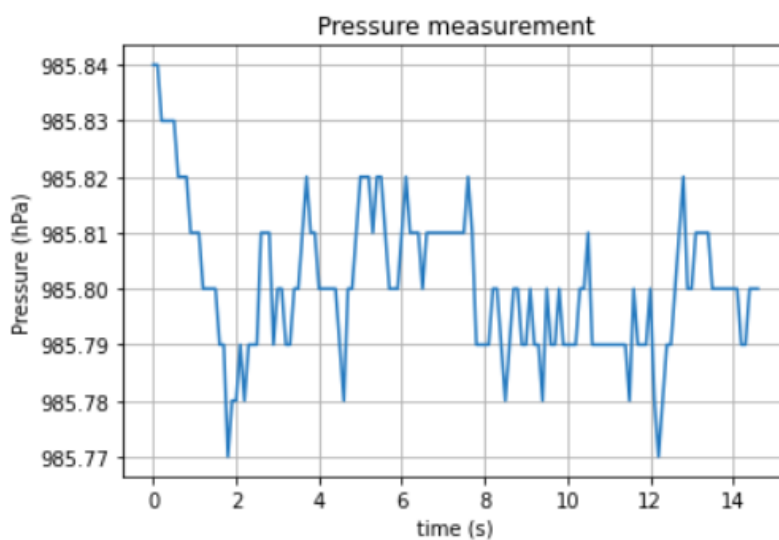
Παρακάτω εμφανίζονται τα γραφήματα από την εφαρμογή

```
[11] # Υπολογισμός γραφήματος για τη θερμοκρασία  
plot_measurement_values(**measurements['temperature'])
```

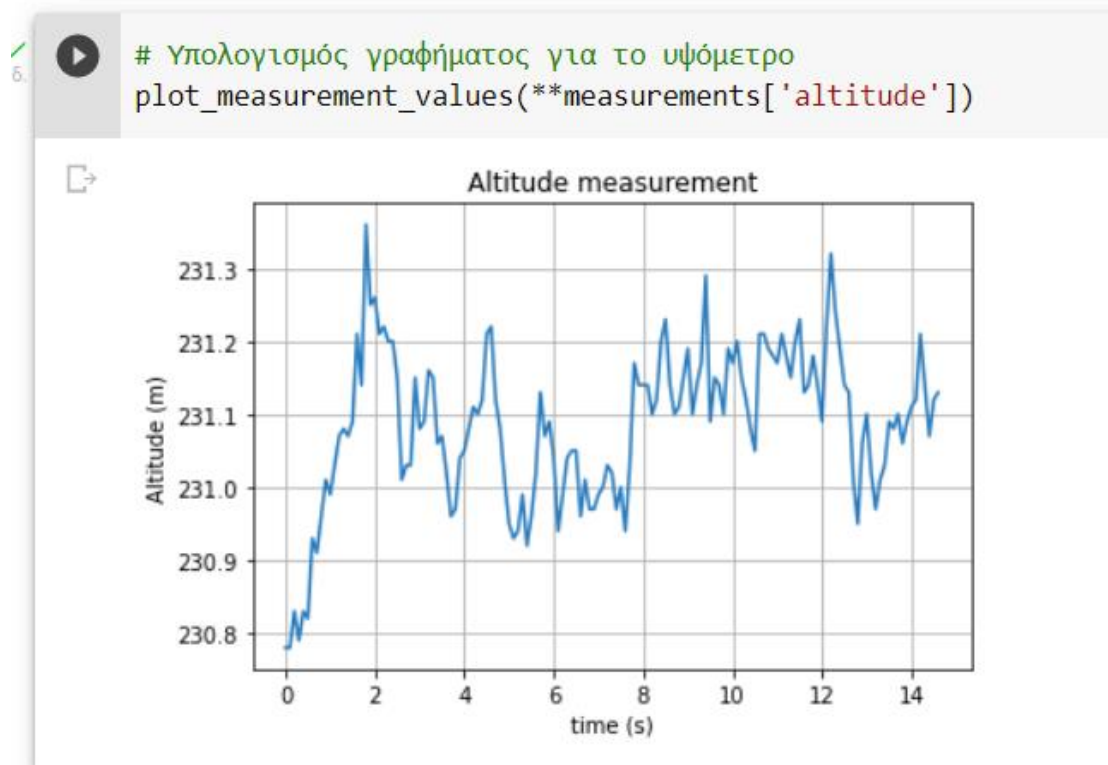


Εικόνα 3.18 Γράφημα θερμοκρασίας

```
[12] # Υπολογισμός γραφήματος για τη πίεση του αέρα  
plot_measurement_values(**measurements['pressure'])
```



Εικόνα 3.19 Γράφημα πίεσης του αέρα

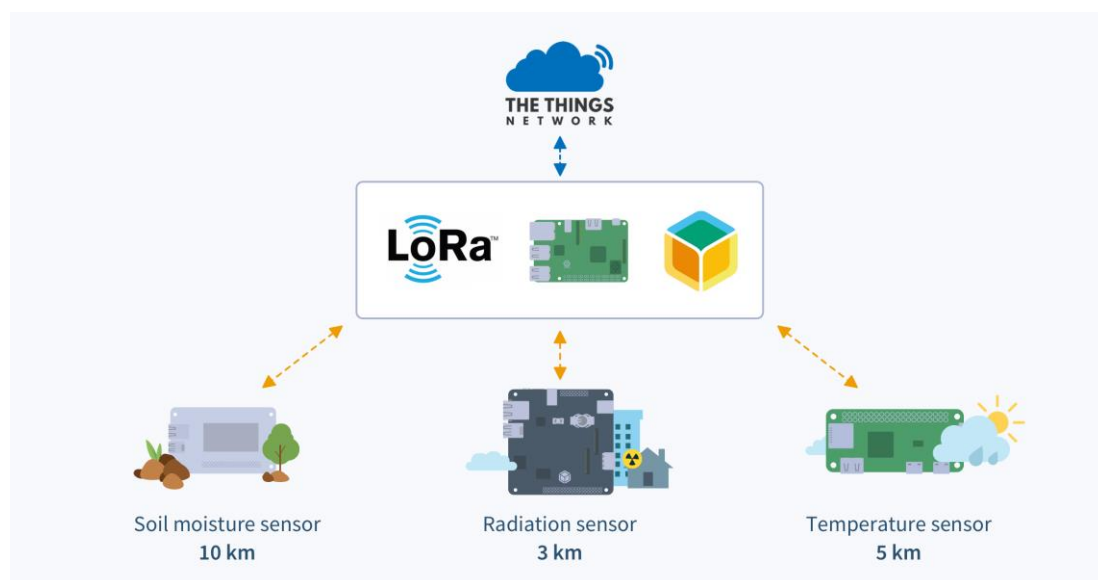


Εικόνα 3.20 Γράφημα υψομέτρου

αυξηθεί το κόστος με τη χρήση δικτύων WIFI ή GSM, θα ήταν η χρήση των δικτύων LoRa.

4.2 Τι είναι τα δίκτυα LoRa

Η τεχνολογία LoRa είναι ένα είδος νέου ασύρματου πρωτοκόλλου που έχει σχεδιαστεί ακριβώς για συνδεσιμότητα μεγάλης εμβέλειας και επικοινωνίες χαμηλής ισχύος. Τα αρχικά LoRa σημαίνουν Long Range Radio και απευθύνεται κυρίως στο Διαδίκτυο των πραγμάτων (internet of things), γιατί είναι ένα πρωτόκολλο το οποίο μπορεί να χρησιμοποιηθεί σε αισθητήρες λόγω της μεγάλης του εμβέλειας και των μικρών απαιτήσεων του σε ενέργεια. Το πρωτόκολλο Lora μπορεί να λειτουργήσει σε ελεύθερες συχνότητες και έτσι δεν χρειάζεται κάποια άδεια. Μερικές από τις συχνότητες που χρησιμοποιεί είναι οι εξής. 915 MHz, 868 MHz και 433 MHz. Βέβαια μπορεί επίσης να λειτουργήσει στα 2,4 GHz για να επιτύχει υψηλότερους ρυθμούς μετάδοσης δεδομένων με κόστος την εμβέλεια.



Εικόνα 4.2 Διάγραμμα σύνδεσης αισθητήρων με το δίκτυο LoRa

4.3 LoRaWAN

Το LoRaWAN είναι ένα πρωτόκολλο ελέγχου χτισμένο πάνω από τη διαμόρφωση LoRa. Είναι ένα επίπεδο λογισμικού που καθορίζει τον τρόπο με τον οποίο οι συσκευές χρησιμοποιούν το υλικό LoRa, για παράδειγμα πότε μια συσκευή μεταδίδει δεδομένα, και τη μορφή των μηνυμάτων που μεταδίδει.

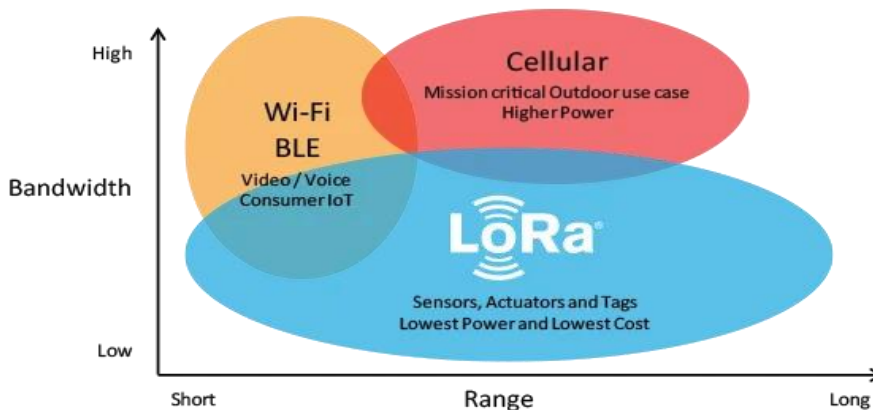
Το πρωτόκολλο LoRaWAN αναπτύχθηκε από τη εταιρία “LoRa Alliance”. Η πρώτη έκδοση LoRaWAN κυκλοφόρησε τον Ιανουάριο του 2015.

| Έκδοση | Ημερομηνία Κυκλοφορίας |
|--------|------------------------|
| 1.0 | Ιανουάριος 2015 |
| 1.01 | Φεβρουάριος 2016 |
| 1.02 | Ιούλιος 2016 |
| 1.1 | Οκτώβριος 2017 |
| 1.03 | Ιούλιος 2018 |
| 1.0.4 | Οκτώβριος 2020 |

Πίνακας 4.1 Ο παρακάτω πίνακας δείχνει το ιστορικό εκδόσεων LoRaWAN.

4.4 Εύρος ζώνης έναντι απόστασης

Το LoRaWAN είναι κατάλληλο για τη μετάδοση δεδομένων μικρού μεγέθους όπως δεδομένα από αισθητήρες σε μεγάλες αποστάσεις. Η διαμόρφωση LoRa παρέχει σημαντικά μεγαλύτερο εύρος απόστασης με χαμηλό εύρος ζώνης σε σχέση άλλες ανταγωνιστικές τεχνολογίες ασύρματης μετάδοσης δεδομένων. Το παρακάτω σχήμα δείχνει ορισμένες τεχνολογίες πρόσβασης που μπορούν να χρησιμοποιηθούν για ασύρματη μετάδοση δεδομένων και το αναμενόμενο εύρος μετάδοσης σε σχέση με το εύρος ζώνης.



Εικόνα 4.3 Διάγραμμα που παρουσιάζει την σχέση Bandwidth / Range

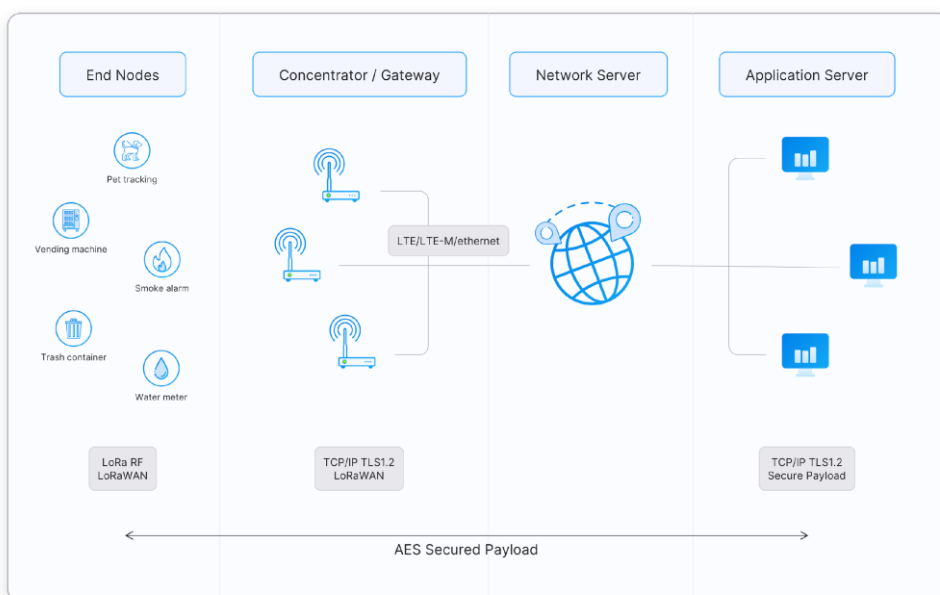
4.5 Πλεονεκτήματα του LoRaWAN

1. **Εξαιρετικά χαμηλή απαίτηση σε ενέργεια** - Οι συσκευές LoRaWAN είναι βελτιστοποιημένες για να λειτουργούν σε λειτουργία χαμηλής κατανάλωσης και μπορούν να διαρκέσουν έως και 10 χρόνια με μία μπαταρία σε σχήμα νομίσματος.
2. **Μεγάλη εμβέλεια** - Οι πύλες LoRaWAN μπορούν να μεταδίδουν και να λαμβάνουν σήματα σε απόσταση άνω των 10 χιλιομέτρων σε αγροτικές περιοχές και έως 3 χιλιομέτρων σε πυκνές αστικές περιοχές.
3. **Μεγάλη απόδοση σε εσωτερικούς χώρους** - Τα δίκτυα LoRaWAN μπορούν να παρέχουν πολύ καλή κάλυψη εσωτερικών χώρων και να καλύπτουν εύκολα πολυώροφα κτίρια.
4. **Φάσμα χωρίς άδεια χρήσης** - Δεν χρειάζεται να πληρωθούν ακριβά τέλη άδειας χρήσης φάσματος συχνοτήτων για να αναπτυχθεί ένα δίκτυο LoRaWAN.
5. **Γεωγραφική τοποθεσία** - Ένα δίκτυο LoRaWAN μπορεί να προσδιορίσει τη θέση των τελικών συσκευών χωρίς την ανάγκη GPS.
6. **Υψηλή χωρητικότητα** - Οι διακομιστές δικτύου LoRaWAN χειρίζονται εκατομμύρια μηνύματα από χιλιάδες πύλες.
7. **Δημόσιες και ιδιωτικές αναπτύξεις** - Είναι εύκολο να αναπτυχθούν δημόσια και ιδιωτικά δίκτυα LoRaWAN χρησιμοποιώντας το ίδιο υλικό (πύλες, τελικές συσκευές, κεραίες) και λογισμικό.
8. **Ασφάλεια** - Το LoRaWAN διασφαλίζει ασφαλή επικοινωνία μεταξύ της τελικής συσκευής και του διακομιστή εφαρμογών χρησιμοποιώντας κρυπτογράφηση AES-128.
9. **Ενημερώσεις υλικολογισμικού** – Είναι εύκολη η ενημέρωση των συσκευών από απόσταση.
10. **Χαμηλό κόστος** – Λόγω της φθηνής τιμής των τερματικών κόμβων και του λογισμικού ανοιχτού κώδικα.
11. **Πρόγραμμα πιστοποίησης** - Το πρόγραμμα πιστοποίησης LoRa Alliance πιστοποιεί τις τελικές συσκευές και παρέχει στους τελικούς χρήστες τη σιγουριά ότι οι συσκευές είναι αξιόπιστες και συμβατές με τις προδιαγραφές LoRaWAN.

12. **Ευκολία στη εύρεση συσκευών** - Το LoRaWAN διαθέτει ένα πολύ μεγάλο οικοσύστημα κατασκευαστών συσκευών, κατασκευαστών πυλών, κατασκευαστών κεραιών, παρόχων υπηρεσιών δικτύου και προγραμματιστών εφαρμογών.

4.6 Αρχιτεκτονική LoRaWAN

Τα δίκτυα LoRaWAN αναπτύσσονται με τοπολογία αστέρα. Ένα τυπικό δίκτυο LoRaWAN αποτελείται από τα ακόλουθα στοιχεία.



Εικόνα 4.4 Διάγραμμα σύνδεσης δικτύου LoRa WAN

1. **Τερματικές συσκευές** - αισθητήρες ή συσκευές στέλνουν ή λαμβάνουν μηνύματα ασύρματα με διαμόρφωση LoRa στα gateway.
2. **Gateway** - λαμβάνει μηνύματα από συσκευές ή αισθητήρες και τα προωθεί στο διακομιστή δικτύου.
3. **Διακομιστής δικτύου** - ένα κομμάτι λογισμικού που εκτελείται στον διακομιστή που διαχειρίζεται ολόκληρο το δίκτυο.
4. **Διακομιστές εφαρμογών** - ένα κομμάτι λογισμικού που εκτελείται στον διακομιστή που είναι υπεύθυνος για την ασφαλή επεξεργασία των δεδομένων της εφαρμογής.

5. **Join Server** - ένα κομμάτι λογισμικού που εκτελείται στο διακομιστή που επεξεργάζεται μηνύματα ή αιτήματα σύνδεσης που αποστέλλονται από τις συσκευές

Οι συσκευές επικοινωνούν με κοντινές πύλες και κάθε πύλη συνδέεται με τον διακομιστή δικτύου. Τα δίκτυα LoRaWAN χρησιμοποιούν πρωτόκολλο που βασίζεται στην ALOHA. Τα μηνύματα που αποστέλλονται από τελικές συσκευές ταξιδεύουν σε όλες τις πύλες εντός εμβέλειας. Αυτά τα μηνύματα λαμβάνονται από τον διακομιστή δικτύου. Εάν ο διακομιστής δικτύου έχει λάβει πολλά αντίγραφα του ίδιου μηνύματος, διατηρεί ένα μόνο αντίγραφο του μηνύματος και απορρίπτει άλλα. Αυτό είναι γνωστό ως διαγραφή μηνυμάτων.

4.6 Απαραίτητο υλικό για δημιουργία δικτύου LoRa

Για να δημιουργηθεί ένα δίκτυο LoRa πρακτικά χρειάζονται τερματικές συσκευές και ένα Gateway στο οποίο θα συνδέονται οι τερματικές συσκευές. Αυτό με τη σειρά του θα συνδέεται στο διαδίκτυο, έτσι κάθε τερματική συσκευή θα μπορεί να βγει στο διαδίκτυο με την χρήση των δικτύων LoRa.

4.6.1 Τερματική συσκευή

Για να μετατραπεί ο σταθμός σε τερματική συσκευή θα χρειαστεί να προστεθεί στο Arduino ένα module. Ένα module που μπορεί να χρησιμοποιηθεί είναι το LoRaWAN module 868 MHz.



Εικόνα 4.5 LoRaWAN module 868 MHz

Με το παραπάνω module θα υπάρχει η δυνατότητα σύνδεσης σε ένα δίκτυο LoRaWAN και μέσω διαδικτύου να ελέγχεται απομακρυσμένα ο σταθμός ή να είναι ορατές οι τιμές που λαμβάνει ο σταθμός από τους αισθητήρες που διαθέτει.

Το κόστος του συγκεκριμένου module είναι σχετικά μικρό και ανέρχεται στα 48,70 €.

Πηγή : <https://www.hellasdigital.gr/go-create/lora/lora-radio-module-868mhz-tel0115/?sl=en>

4.6.2 Gateway

Όπως αναφέρθηκε σε προηγούμενη παράγραφο, για να υπάρχει η δυνατότητα σύνδεσης της συσκευής με το διαδίκτυο με την χρήση των δικτύων LoRaWAN, θα πρέπει να υπάρχει ένα gateway LoRaWAN. Το συγκεκριμένο εξάρτημα κοστίζει αρκετά αν ληφθεί υπόψιν ότι υπάρχει η ανάγκη ενός gateway εξωτερικού χώρου για να εκμεταλλευθούν στο έπακρο οι δυνατότητες που παρέχουν τα LoRaWAN να επικοινωνούν μεταξύ τους σε αποστάσεις μεγαλύτερες των δέκα χιλιομέτρων. Στην συγκεκριμένη περίπτωση επιλέχθηκε η συσκευή Milesight UG67 Outdoor LoRaWAN Gateway.



Εικόνα 4.6 Milesight UG67 Outdoor LoRaWAN Gateway.

Το Milesight UG67 είναι μια στιβαρή εξωτερική πύλη (gateway) LoRaWAN 8 καναλιών. Χρησιμοποιεί το τσιπ SX1302 LoRa και την τετραπύρηνη CPU υψηλής απόδοσης, το UG67 υποστηρίζει σύνδεση με περισσότερους από 2000 κόμβους. Το UG67 διαθέτει οπτική επαφή έως 15 km και αδιάβροχη θήκη IP67, η οποία είναι ιδανική για έξυπνη γεωργία, έξυπνη μέτρηση και πολλές άλλες εφαρμογές σε εξωτερικούς χώρους.

Η τιμή του Milesight UG67 ανέρχεται στα 585,16 €

Πηγή : <https://market.thingspark.com/milesight-ug67-outdoor-lorawan-gateway.html>

5 Συμπεράσματα και επίλυση δυσκολιών

5.1 Επίλυση δυσκολιών

Ένα βασικό πρόβλημα που αντιμετωπίζει το πρότζεκτ είναι η μικρή μνήμη του Arduino που εμποδίζει τη συγγραφή ενός μεγαλύτερου κώδικα, ο οποίος πιθανώς θα εξασφάλιζε και περισσότερες αναβαθμίσεις στο τελικό προϊόν. Με κάποιες συγχωνεύσεις στον κώδικα αυτό πρόβλημα μπορεί να επιλυθεί.

Ένα δεύτερο πρόβλημα που υπάρχει είναι η έλλειψη gateway LoRaWAN στην περιοχή των Σερρών. Έτσι δεν είναι εφικτή η μετατροπή του σταθμού σε τερματικό κόμβο LoRaWAN και συνεπώς ο σταθμός δεν μπορεί να συνδεθεί με τα δίκτυα LoRa. Η λύση της αγοράς ενός gateway απορρίφθηκε λόγω του αυξημένου κόστους.

5.2 Συμπεράσματα

Από την πτυχιακή μπορεί να γίνει αντιληπτή η χρησιμότητα της τεχνολογίας γενικότερα αλλά και η ανάγκη εξέλιξης της τεχνολογίας στο κομμάτι των μικροεπεξεργαστών όπως είναι το Arduino. Πλέον μπορεί σχεδόν ο καθένας με λίγες γνώσεις και μικρά έξοδα να πειραματιστεί πάνω σε αυτόν τον τομέα και να κατασκευάσει κάτι το οποίο να μην είναι απαραίτητα έτοιμο για να βγει στην αγορά αλλά θα είναι το κίνητρο ώστε να αναπτύξει και να εμβαθύνει τις γνώσεις του. Συγκεκριμένα σε αυτή την πτυχιακή εργασία μελετήθηκαν 2 κλάδοι. Ο κλάδος των δικτύων αλλά και ο κλάδος της ηλεκτρονικής. Μέσω τέτοιων εργασιών η μάθηση γίνεται περισσότερο ευχάριστη αλλά και αποδοτική. Η απόκτηση πρακτικών γνώσεων και εμπειρίας ενισχύει τις δυνατότητες του φοιτητή.

Τέλος, κρίνοντας το τελικό αποτέλεσμα, το πρότζεκτ δεν μπορεί να θεωρηθεί ως ολοκληρωμένο. Υπάρχουν αρκετά στοιχεία τα οποία πρέπει να επιλυθούν και άλλα που πρέπει να βελτιωθούν.

Βιβλιογραφία

1. <https://el.wikipedia.org/wiki/Arduino>
2. <https://cdn-learn.adafruit.com/downloads/pdf/adafruit-bmp388.pdf?timestamp=1600143854>
3. <https://www.mouser.com/datasheet/2/758/DHT11-Technical-Data-Sheet-Translated-Version-1143054.pdf>
4. https://cdn-reichelt.de/documents/datenblatt/A300/101020078_01.pdf
5. <https://store-usa.arduino.cc/products/arduino-uno-rev3>
6. <https://www.thethingsnetwork.org/docs/lorawan/what-is-lorawan/>
7. <https://www.thethingsnetwork.org/docs/lorawan/architecture/>
8. <https://www.hackster.io/fomi-T/from-zero-to-build-your-own-lora-network-with-rak833-and-rpi-bc3ee1>
9. <https://howtomechatronics.com/projects/diy-air-quality-monitor-pm2-5-co2-voc-ozone-temp-hum-arduino-meter/>
10. <https://www.hellasdigital.gr/go-create/lora/lora-radio-module-868mhz-tel0115/?sl=en>
11. <https://market.thingpark.com/milesight-ug67-outdoor-lorawan-gateway.html>
12. <https://www.mi.com/gr/>