



ΔΙΕΘΝΕΣ
ΠΑΝΕΠΙΣΤΗΜΙΟ
ΕΛΛΑΔΟΣ

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ
ΚΑΙ ΗΛΕΚΤΡΟΝΙΚΩΝ ΣΥΣΤΗΜΑΤΩΝ

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

"Έλεγχος ποιότητας νερού σε δεξαμενή ύδρευσης"



ΦΟΙΤΗΤΗΣ: ΚΑΡΑΜΠΙΔΗΣ ΘΩΜΑΣ

ΥΠΕΥΘΥΝΟΣ ΚΑΘΗΓΗΤΗΣ: ΓΙΑΚΟΥΜΗΣ ΑΓΓΕΛΟΣ

ΘΕΣΣΑΛΟΝΙΚΗ ΧΕΙΜΕΡΙΝΟ ΕΞΑΜΗΝΟ 2023-2024

Τίτλος Δ.Ε. : Έλεγχος ποιότητας νερού σε δεξαμενή ύδρευσης

Κωδικός Δ.Ε. : 23104

Όνοματεπώνυμο φοιτητή : Καραμπίδης Θωμάς

Όνοματεπώνυμο εισηγητή : Γιακουμής Άγγελος

Ημερομηνία ανάληψης Δ.Ε. : 31-01-2023

Ημερομηνία περάτωσης Δ.Ε. : 24-01-2024

Βεβαιώνω ότι είμαι ο συγγραφέας αυτής της εργασίας και ότι κάθε βοήθεια την οποία είχα για την προετοιμασία της είναι πλήρως αναγνωρισμένη και αναφέρεται στην εργασία. Επίσης, έχω καταγράψει τις όποιες πηγές από τις οποίες έκανα χρήση δεδομένων, ιδεών, εικόνων και κειμένου, είτε αυτές αναφέρονται ακριβώς είτε παραφρασμένες. Επιπλέον, βεβαιώνω ότι αυτή η εργασία προετοιμάστηκε από εμένα προσωπικά, ειδικά ως διπλωματική εργασία, στο Τμήμα Μηχανικών Πληροφορικής και Ηλεκτρονικών Συστημάτων του ΔΙ.ΠΑ.Ε.

Η παρούσα εργασία αποτελεί πνευματική ιδιοκτησία του φοιτητή Καραμπίδη Θωμά που την εκπόνησε. Στο πλαίσιο της πολιτικής ανοικτής πρόσβασης, ο συγγραφέας/δημιουργός εκχωρεί στο Διεθνές Πανεπιστήμιο της Ελλάδος άδεια χρήσης του δικαιώματος αναπαραγωγής, δανεισμού, παρουσίασης στο κοινό και ψηφιακής διάχυσης της εργασίας διεθνώς, σε ηλεκτρονική μορφή και σε οποιοδήποτε μέσο, για διδακτικούς και ερευνητικούς σκοπούς, άνευ ανταλλάγματος. Η ανοικτή πρόσβαση στο πλήρες κείμενο της εργασίας, δεν σημαίνει καθ' οιονδήποτε τρόπο παραχώρηση δικαιωμάτων διανοητικής ιδιοκτησίας του συγγραφέα/δημιουργού, ούτε επιτρέπει την αναπαραγωγή, αναδημοσίευση, αντιγραφή, πώληση, εμπορική χρήση, διανομή, έκδοση, μεταφόρτωση (downloading), ανάρτηση (uploading), μετάφραση, τροποποίηση με οποιονδήποτε τρόπο, τμηματικά ή περιληπτικά της εργασίας, χωρίς τη ρητή προηγούμενη έγγραφη συναίνεση του συγγραφέα/δημιουργού.

Η έγκριση της διπλωματικής εργασίας από το Τμήμα Μηχανικών Πληροφορικής και Ηλεκτρονικών Συστημάτων του Διεθνούς Πανεπιστημίου της Ελλάδος, δεν υποδηλώνει απαραίτητως και αποδοχή των απόψεων του συγγραφέα, εκ μέρους του Τμήματος.

Πρόλογος

Η εργασία αυτή επικεντρώνεται στη χρήση του Arduino και τριών αισθητήρων για τον έλεγχο της ποιότητας του νερού σε μια δεξαμενή ύδρευσης. Οι αισθητήρες που χρησιμοποιώ είναι ο DS18B20 για τη μέτρηση της θερμοκρασίας, ένας αισθητήρας TDS για τη μέτρηση της ηλεκτρικής αγωγιμότητας και ένας αισθητήρας pH για τη μέτρηση της οξύτητας του νερού.

Αρχικά, θα αναλύσω το γιατί χρησιμοποιούμε δεξαμενή ύδρευσης, πως ξεκίνησε σαν εγχείρημα και γιατί ξεκίνησε να χρησιμοποιείται για πόσιμο νερό.

Έπειτα, θα σας δείξω πως μπορούμε να ελέγχουμε την ποιότητα του νερού και τι χρειάζεται για να δείξουμε ότι το νερό είναι πόσιμο.

Στη συνέχεια, θα κάνω μια αναφορά στη πλακέτα που χρησιμοποιώ και μια αναδρομή στο πως δημιουργήθηκε, θα σας δείξω και τους αισθητήρες που θα χρησιμοποιηθούν για αυτή την εργασία καθώς επίσης και στο πρόγραμμα που θα χρησιμοποιήσω για τον προγραμματισμό της πλακέτας.

Εν συνεχεία, θα σας παρουσιάσω τη διαδικασία σύνδεσης των αισθητήρων στο Arduino και θα σας εξηγήσω λειτουργίες του κώδικα προγραμματισμού που χρησιμοποιείται για τη μέτρηση και την ανάγνωση των δεδομένων που συλλέγονται από τους αισθητήρες.

Τέλος, θα δούμε μαζί τα συμπεράσματα της εργασίας αυτής και τρόπους βελτίωσης που πιθανό να δέχεται η συγκεκριμένη πτυχιακή, καθώς επίσης και τη πηγές που χρησιμοποίησα.

Περίληψη

Η εργασία με θέμα “ Έλεγχος ποιότητας νερού σε δεξαμενή ύδρευσης” αποσκοπεί στην ανάπτυξη ενός συστήματος ελέγχου ποιότητας νερού που χρησιμοποιεί μικροελεγκτή Arduino και τρεις αισθητήρες: τον DS18B20 για τη μέτρηση της θερμοκρασίας, έναν αισθητήρα TDS για τη μέτρηση των ολικών διαλυμένων στερεών και έναν αισθητήρα pH για τη μέτρηση του επιπέδου pH. Ξεκινά με μια εισαγωγή στην πρόκληση του έλεγχου της ποιότητας του νερού σε δεξαμενές ύδρευσης και τη σημασία της διασφάλισης καθαρού και ασφαλούς πόσιμου νερού για την υγεία μας. Στη συνέχεια, παρουσιάζονται οι αισθητήρες DS18B20, TDS και pH, μαζί με τη λειτουργία και την αρχή λειτουργίας τους. Στο επόμενο μέρος της εργασίας, περιγράφω την σύνδεση των αισθητήρων με το μικροελεγκτή Arduino και τη διαδικασία μέτρησης των παραμέτρων θερμοκρασίας, TDS και pH. Αναλύω τον κώδικα προγραμματισμού που απαιτείται για την ανάγνωση των αισθητήρων και την επικοινωνία με το μικροελεγκτή. Έπειτα, παρουσιάζω την ανάλυση των μετρήσεων που πραγματοποίησα με την βοήθεια των αισθητήρων. Εξετάζω τις διάφορες τιμές θερμοκρασίας, TDS και pH που ανιχνεύθηκαν και συγκρίνω με προκαθορισμένα πρότυπα ποιότητας νερού για να αξιολογηθεί η κατάσταση του νερού στη δεξαμενή ύδρευσης. Τέλος, συνοψίζονται τα αποτελέσματα και αναφέρομαι σε πιθανές βελτιώσεις και μελλοντικές επεκτάσεις του συστήματος ελέγχου ποιότητας νερού με τη χρήση Arduino και των αισθητήρων DS18B20, TDS και pH. Επίσης, θα δούμε τις πιθανές εφαρμογές του συστήματος σε άλλα πεδία και οι προοπτικές για περαιτέρω έρευνα και ανάπτυξη στον τομέα του ελέγχου ποιότητας του νερού. Η εργασία αποτελεί μια πρακτική και καινοτόμο προσέγγιση για τον έλεγχο της ποιότητας του νερού σε δεξαμενές ύδρευσης, εκμεταλλευόμενη την τεχνολογία του Arduino και τη χρήση αισθητήρων, προσφέροντας έναν αποτελεσματικό και οικονομικό τρόπο παρακολούθησης και διασφάλισης της ποιότητας του πόσιμου νερού.

«Water quality control in a water supply tank»

«Karampidis Thomas»

Abstract

The project "Water quality control in a water supply tank" aims to develop a water quality control system using an Arduino microcontroller and three sensors: the DS18B20 to measure temperature, a TDS sensor to measure total dissolved solids and a pH sensor to measure the pH level. It begins with an introduction to the challenge of water quality control in water tanks and the importance of ensuring clean and safe drinking water for our health. The DS18B20, TDS and pH sensors are then introduced, along with their function and principle of operation. In the next part of the paper, I describe the connection of the sensors to the Arduino microcontroller and the process of measuring the temperature, TDS and pH parameters. I analyze the programming code required to read the sensors and communicate with the microcontroller. Then, I present the analysis of the measurements made using the sensors. I examine the various temperature, TDS and pH values detected and compare them with predefined water quality standards to evaluate the water condition in the water supply tank. Finally, the results are summarized and possible improvements and future extensions of the water quality control system using Arduino and the DS18B20, TDS and pH sensors are discussed. Also, the possible applications of the system in other fields and the prospects for further research and development in the field of water quality control are discussed. The thesis is a practical and innovative approach to water quality control in water supply tanks, taking advantage of Arduino technology and the use of sensors, offering a practical and innovative approach to water quality control.

ΠΕΡΙΕΧΟΜΕΝΑ

Πρόλογος

Περίληψη-Abstract

1. ΕΙΣΑΓΩΓΗ

1.1 Πόσιμο Νερό.	Σελ.9
1.2 Πόσιμο Νερό σε Δεξαμενή Ύδρευσης.	Σελ.10
1.3 Δεξαμενή Ύδρευσης.	Σελ.12
1.4 Ιστορική Αναδρομή.	Σελ.14

2. ΠΕΡΙΒΑΛΛΟΝ ΣΤΟ ΕΣΩΤΕΡΙΚΟ ΤΟΥ ΘΕΡΜΟΚΗΠΙΟΥ

2.1 Ολικά Διαλυμένα Στοιχεία(TDS).	Σελ.16
2.2 Οξύτητα(pH).	Σελ.18
2.3 Θερμοκρασία(C).	Σελ.20

3. Ο ΜΙΚΡΟΕΛΕΓΚΤΗΣ ARDUINO

3.1 Ο μικροελεγκτής Arduino και τα διάφορα είδη του .	Σελ.22
3.2 Μικροελεγκτής Arduino Uno.	Σελ.26
3.3 Χαρακτηριστικά της Arduino Uno .	Σελ.27
3.4 Εισόδοι/Εξόδοι του μικροελεγκτή.	Σελ.28
3.5 Τροφοδοσία.	Σελ.30
3.6 Ο μικροελεγκτής ATmega328.	Σελ.31
3.7 Το Προγραμματιστικό Περιβάλλον Arduino IDE.	Σελ.33
3.8 Σειριακή Θύρα επικοινωνίας.	Σελ.35

4. ΠΕΡΙΓΡΑΦΗ ΤΟΥ ΤΡΟΠΟΥ ΚΑΤΑΣΚΕΥΗΣ

4.1 Υλικά Διασύνδεσης	Σελ.36
-----------------------	--------

4.2 Αισθητήρες.	Σελ.37
4.3 Αισθητήρας Αγωγιμότητας(TDS).	Σελ.37
4.4 Αισθητήρας Οξύτητας.(pH)	Σελ.38
4.5 Αισθητήρας Θερμοκρασίας.(C)	Σελ.39
4.6 Οθόνη LCD.	Σελ.40
4.7 Τροφοδοσία Αισθητήρων	Σελ.42
4.8 Βαθμονόμηση Αισθητήρων(Calibration)	Σελ.43
4.9 Διάγραμμα Ροής Προγράμματος.	Σελ.45
4.10 Block Διάγραμμα Κυκλώματος	Σελ.46
5. ΔΗΜΙΟΥΡΓΙΑ ΤΟΥ ΚΩΔΙΚΑ	
5.1 Βασικές Εντολές.	Σελ.47
5.2 Προγραμματισμός της Κατασκευής.	Σελ.48
5.3 Αποτελέσματα από την οθόνη LCD.	Σελ.53
6. ΣΥΜΠΕΡΑΣΜΑΤΑ-ΒΑΣΙΚΑ ΠΟΡΙΣΜΑΤΑ	
6.1 Συμπεράσματα.	Σελ.62
6.2 Βελτιώσεις.	Σελ.62
ΠΗΓΕΣ-ΒΙΒΛΙΟΓΡΑΦΙΑ	Σελ.62

Εισαγωγή

1.1 Πόσιμο Νερό

Με τον όρο πόσιμο νερό αναφερόμαστε σε νερό που είναι ασφαλές και κατάλληλο για κατανάλωση από τους ανθρώπους. Αποτελείται από νερό που πληροί τα πρότυπα και τις προδιαγραφές που ορίζονται από τις αρμόδιες υγειονομικές αρχές και οργανισμούς για την ασφάλεια του πόσιμου νερού. Για να θεωρηθεί νερό πόσιμο, πρέπει να πληροί διάφορες παραμέτρους και πρότυπα, συμπεριλαμβανομένων των ακόλουθων:

- Χαμηλή συγκέντρωση ή απουσία βλαβερών οργανικών και ανόργανων ουσιών.
- Απουσία μικροβίων, βακτηρίων, ιών και παρασίτων που μπορούν να προκαλέσουν ασθένειες.
- Κατάλληλη συγκέντρωση απαραίτητων ανόργανων στοιχείων, όπως ασβέστιο, μαγνήσιο και φώσφορο.
- Φυσικά και οργανοληπτικά χαρακτηριστικά που το καθιστούν ευχάριστο και αποδεκτό για κατανάλωση.



Εικόνα 1. Πόσιμο τρεχούμενο νερό.

1.2 Πόσιμο Νερό Σε Δεξαμενή Ύδρευσης

Το πόσιμο νερό σε δεξαμενή ύδρευσης είναι νερό που έχει αποθηκευτεί ήδη εντός μιας δεξαμενής ή δεξαμενών ύδρευσης και είναι προορισμένο για κατανάλωση από τους κατοίκους

ή χρήση σε καταναλωτικές δραστηριότητες. Το πόσιμο νερό προέρχεται συνήθως από δημόσια ή ιδιωτική ύδρευση και προσφέρει ασφαλή πηγή πόσιμου νερού για την κοινότητα. Προτού διοχετευθεί στη δεξαμενή ύδρευσης, το πόσιμο νερό υπόκειται σε διαδικασίες επεξεργασίας και καθαρισμού, όπως φιλτράρισμα, απολύμανση και πιθανώς εφαρμογή χημικών επεξεργασίας, για να διασφαλιστεί η ποιότητά του και η απομάκρυνση ακαθαρσιών, βακτηρίων και παθογόνων οργανισμών. Ο έλεγχος και η διασφάλιση της ποιότητας του πόσιμου νερού είναι σημαντικά για την προστασία της υγείας των ανθρώπων και την πρόληψη ασθενειών.[14]



Εικόνα 2. Δεξαμενή ύδρευσης με πόσιμο νερό στην Αμερική.

1.3 Δεξαμενή Ύδρευσης

Η δεξαμενή ύδρευσης είναι ένας αποθηκευτικός χώρος για τον εφοδιασμό με νερό, σχεδιασμένος να συγκεντρώνει, να αποθηκεύει και να διανέμει το νερό ύδρευσης σε μια κοινότητα ή σε έναν οικισμό. Συνήθως τοποθετούνται σε στρατηγικά σημεία του υδροδοτικού δικτύου για να εξασφαλιστεί η ισορροπία της παροχής νερού κατά τη διάρκεια της ημέρας. Αυτό επιτυγχάνεται με την αποθήκευση του πόσιμου νερού που παράγεται κατά τις περιόδους χαμηλής ζήτησης και τη διανομή του κατά τις περιόδους υψηλής ζήτησης, όπως τις πρωινές και απογευματινές ώρες.

Κατασκευάζονται συνήθως από ανθεκτικά υλικά όπως τσιμεντοειδή και χάλυβας, προσφέροντας αξιόπιστη προστασία από διαρροές και μόλυνση του υδάτινου αποθέματος. Επιπλέον, οι δεξαμενές συχνά εξοπλίζονται με συστήματα φιλτραρίσματος και επεξεργασίας νερού πριν από την αποθήκευση, για να διασφαλιστεί η ποιότητα του πόσιμου νερού.

Η συντήρηση και η παρακολούθηση των δεξαμενών ύδρευσης είναι απαραίτητες για να διασφαλιστεί η απρόσκοπτη λειτουργία τους. Πρέπει να πραγματοποιούνται τακτικοί έλεγχοι του επιπέδου του νερού, της πίεσης, της ποιότητας και της συντήρησης των δομικών στοιχείων. Επιπλέον, η καθαριότητα των δεξαμενών πρέπει να διατηρείται ώστε να αποφεύγονται οι ενδεχόμενες μολύνσεις του πόσιμου νερού.



Εικόνα 3. Παλιές Δεξαμενές Ύδρευσης.

1.4 Ιστορική Αναδρομή

Η ιστορία των δεξαμενών ύδρευσης χρονολογείται αιώνες πριν, αρχίζοντας από την αρχαιότητα.

Στην αρχαία Ρώμη, για παράδειγμα, κατασκευάζονταν εντυπωσιακές δεξαμενές ύδρευσης, γνωστές ως υδραγωγεία. Αυτά τα υδραγωγεία ήταν εκτεταμένα δίκτυα υδραγωγείων που συλλέγανε και διανέμονταν το νερό σε πολλές περιοχές της πόλης. Οι δεξαμενές ύδρευσης σε αυτές τις πόλεις χρησιμοποιούνταν για να αποθηκεύσουν το νερό από πηγές, ρυάκια ή υπόγειες δεξαμενές και να το διανέμουν στους κατοίκους της πόλης μέσω σωληνώσεων και αγωγών.

Κατά τη διάρκεια του Μεσαίωνα, οι πόλεις της Ευρώπης συνέχισαν να χρησιμοποιούν δεξαμενές ύδρευσης για την αντιμετώπιση της ανεπάρκειας νερού. Στα παλάτια και τα κάστρα, χρησιμοποιούνταν μεγάλες δεξαμενές ύδρευσης για να παρέχουν νερό σε επισκέπτες και κατοίκους. Κατά τη διάρκεια της Βιομηχανικής Επανάστασης, οι δεξαμενές ύδρευσης εξελίχθηκαν περαιτέρω με τη χρήση σιδηροσκελετών και υπόγειων αποθηκευτικών χώρων.

Στις μέρες μας, οι δεξαμενές ύδρευσης εξακολουθούν να χρησιμοποιούνται παγκοσμίως για την αποθήκευση και τη διανομή του νερού. Σχεδιάζονται και κατασκευάζονται με διάφορα υλικά, όπως σκυρόδεμα, μέταλλο ή πλαστικό, και μπορούν να έχουν διάφορους τύπους, όπως επιφανειακές δεξαμενές, υπόγειες δεξαμενές ή ακόμα και φορητές δεξαμενές για ειδικές ανάγκες.

Οι δεξαμενές ύδρευσης έχουν αποδειχθεί αναπόσπαστο μέρος των υδρολογικών συστημάτων πόλεων και έχουν συμβάλει σημαντικά στην ασφάλεια και την υγιεινή του πόσιμου νερού που παρέχεται στους ανθρώπους. Η συνεχής βελτίωση της τεχνολογίας και η εφαρμογή προηγμένων συστημάτων διαχείρισης του ύδατος συνεχίζουν να εξελίσσουν τον τρόπο με τον οποίο χρησιμοποιούνται οι δεξαμενές ύδρευσης σε σύγχρονες κοινότητες. [13]



Εικόνα 4. Αρχαίο Ρωμαϊκό Υδραγωγείο

2. Έλεγχος ποιότητας νερού

2.1 Ολικά Διαλυμένα Στοιχεία (TDS)

Το Total Dissolved Solids (TDS) ή αλλιώς ολικά διαλυμένα στερεά αναφέρεται στο συνολικό περιεχόμενο των ανόργανων και οργανικών ενώσεων που βρίσκονται διαλυμένες σε ένα υγρό, είτε σε μοριακή, ιονισμένη ή μικροκοκκώδη μορφή. Η μέτρηση των ολικών διαλυμένων στερεών γίνεται κυρίως σε συστήματα γλυκού νερού, καθώς η αλατότητα περιλαμβάνει ορισμένα από τα ιόντα που συνθέτουν τα TDS. Οι μετρήσεις αυτές χρησιμοποιούνται για να αξιολογηθεί η ποιότητα του νερού σε ρυάκια, ποταμούς και λίμνες. Αν και τα TDS δεν θεωρούνται γενικά ως κύριος παράγοντας ρύπανσης (δηλαδή δεν συνδέονται απευθείας με επιπτώσεις στην υγεία), χρησιμοποιούνται ως ένδειξη για τα αισθητικά χαρακτηριστικά του πόσιμου νερού και ως συγκεντρωτικός δείκτης της παρουσίας διάφορων χημικών ρύπων.

Τα ολικά διαλυμένα στερεά περιλαμβάνουν συνήθως χημικά συστατικά όπως ασβέστιο, φωσφορικά άλατα, νιτρικά άλατα, νάτριο, κάλιο και χλώριο. Αυτά τα συστατικά προέρχονται από την απορροή θρεπτικών συστατικών, τη γενική ροή από υδάτινες πηγές και τη χρήση αλατιού για το ξεπάγωμα των δρόμων σε περιοχές με χιονόπτωση. Τα χημικά αυτά μπορεί να είναι κατιόντα, ανιόντα, μόρια ή συσσωματώματα με μέγεθος μορίων στην τάξη των χιλίων ή ακόμα και λιγότερων. Ορισμένα από αυτά τα ολικά διαλυμένα στερεά προέρχονται από τη διάβρωση και διάλυση των βράχων και του εδάφους.

Η μέτρηση του TDS είναι σημαντική σε διάφορους τομείς, όπως η υδρολογία, η επεξεργασία νερού, η γεωργία και η υγεία. Επιπλέον, η παρακολούθηση του TDS μπορεί να βοηθήσει στον έλεγχο της ποιότητας του πόσιμου νερού και να παρέχει πληροφορίες σχετικά με την παρουσία χημικών ρυπαντών.[6][11][12]



Εικόνα 5. Είδος μετρητή TDS

2.2 Οξύτητα(pH)

Το pH είναι ένας αριθμητικός δείκτης που χρησιμοποιείται για να μετρήσει την οξύτητα ή την αλκαλικότητα μιας διάλυσης. Το "pH" αναπαριστά τη συγκέντρωση των ιοντικών υδρονίων (H^+) σε μια διάλυση. Ο αριθμητικός δείκτης του pH κυμαίνεται από 0 έως 14.

Όταν η τιμή του pH είναι μικρότερη από 7, η διάλυση θεωρείται οξεία, πράγμα που σημαίνει ότι έχει υψηλή συγκέντρωση ιοντικών υδρονίων. Όταν η τιμή του pH είναι μεγαλύτερη από 7, η διάλυση θεωρείται αλκαλική, πράγμα που σημαίνει ότι έχει υψηλή συγκέντρωση υδροξειδίων (OH^-). Όταν η τιμή του pH είναι 7, η διάλυση θεωρείται ουδέτερη, πράγμα που σημαίνει ότι η συγκέντρωση των ιοντικών υδρονίων και των υδροξειδίων είναι ισορροπημένη.

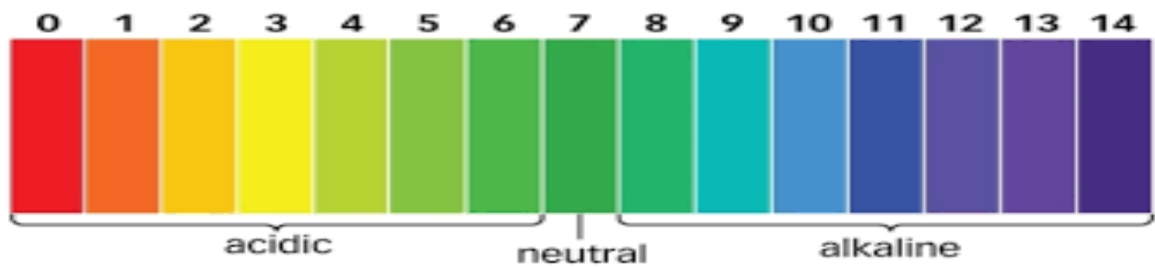
Ο δείκτης pH χρησιμοποιείται ευρέως σε πολλούς τομείς, όπως η χημεία, η βιολογία, η ιατρική, η γεωλογία και η τροφοδοσία, για να μετρηθεί η οξύτητα ή η αλκαλικότητα μιας ουσίας ή περιβάλλοντος.[18]



Εικόνα 6. Μετρητής pH



Εικόνα 7. Χαρτί μέτρησης pH



Εικόνα 8.Χρωματικός κώδικας pH

2.3 Θερμοκρασία(°C)

Η θερμοκρασία αναφέρεται στη μέτρηση της θερμότητας ή της θερμικής κατάστασης ενός αντικειμένου ή ενός συστήματος. Αντιπροσωπεύει τον βαθμό θερμότητας που υπάρχει σε ένα συγκεκριμένο αντικείμενο ή περιβάλλον. Μετράται συνήθως σε μονάδες όπως βαθμοί Κελσίου (°C) ή βαθμοί Φαρενάιτ (°F).

Όσον αφορά τη θερμοκρασία του νερού, αναφέρεται στην θερμοκρασία της υγρής κατάστασης του νερού. Η θερμοκρασία του νερού μπορεί να ποικίλει ανάλογα με τις συνθήκες και το περιβάλλον. Η κανονική θερμοκρασία του νερού κατά την πίεση ατμόσφαιρας είναι 100°C για τη βρασμένη κατάσταση και 0°C για την παγωμένη κατάσταση. Ανάμεσα σε αυτές τις δύο θερμοκρασίες, το νερό είναι σε υγρή κατάσταση.

Η θερμοκρασία του νερού επηρεάζει σημαντικά την ποιότητά του. Υψηλές θερμοκρασίες μπορεί να προκαλέσουν αλλαγές στη χημική σύσταση του νερού, όπως αύξηση της διάλυσης αερίων όπως το διοξείδιο του άνθρακα, μείωση της διαλυτότητας ανόργανων αλάτων και αλλαγές στην παρουσία οργανικών ουσιών. Αυτές οι αλλαγές μπορούν να επηρεάσουν την καταλληλότητα του νερού για κατανάλωση, γεωργία, βιολογικά οικοσυστήματα και άλλες χρήσεις. Επίσης, υψηλές θερμοκρασίες μπορούν να προωθήσουν την ανάπτυξη μικροβίων και επιδοτήσουν τη διάδοση ασθενειών στο νερό. Ως εκ τούτου, η θερμοκρασία του νερού είναι ένα σημαντικό παράγοντα για την ποιότητα και την καταλληλότητα του νερού για διάφορες χρήσεις.[17]



Εικόνα 9. Διάφορα Είδη Θερμόμετρων

3. ARDUINO



Εικόνα10. Arduino Logo

3.1 Ο μικροελεγκτής Arduino και τα διάφορα είδη του

Το Arduino αποτελεί μια πλακέτα ανοιχτού κώδικα με έναν ενσωματωμένο ελεγκτή, ο οποίος μπορεί να προγραμματιστεί με τη γλώσσα προγραμματισμού Wiring. Πρόκειται για μια γλώσσα παρόμοια με τη C++, περιλαμβάνοντας βιβλιοθήκες που είναι επίσης βασισμένες στη C++. Το Arduino διαθέτει επίσης αναλογικές και ψηφιακές εισόδους και εξόδους, επιτρέποντας τη σύνδεση διάφορων αισθητήρων. Με τη χρήση του καλωδίου προγραμματισμού, ο κώδικας μπορεί να μεταφορτωθεί στην πλακέτα μέσω υπολογιστή.

Η πλακέτα του Arduino ανοίγει τον δρόμο για τη δημιουργία διάφορων ηλεκτρονικών έργων. Συνδέοντάς τη με τον προσωπικό υπολογιστή του, ο χρήστης μπορεί να την προγραμματίσει χρησιμοποιώντας εργαλεία και προγράμματα όπως τα Processing, Max/MSP, Pure Data και SuperCollider. Οι περισσότερες εκδόσεις του Arduino είναι προσυναρμολογημένες, αλλά υπάρχουν διαθέσιμες πληροφορίες και διαγράμματα για όσους επιθυμούν να κατασκευάσουν την πλακέτα μόνοι τους.

Από το 2005, όπου και ξεκίνησε να σχεδιάζεται και να κατασκευάζεται το πρώτο Arduino, έχουν βγει σε κυκλοφορία συνολικά 17 εκδόσεις του. Κάποιες από αυτές οι οποίες είναι και οι πιο διαδεδομένες και χρησιμοποιημένες από τους χρήστες ανά τον κόσμο είναι οι παρακάτω:

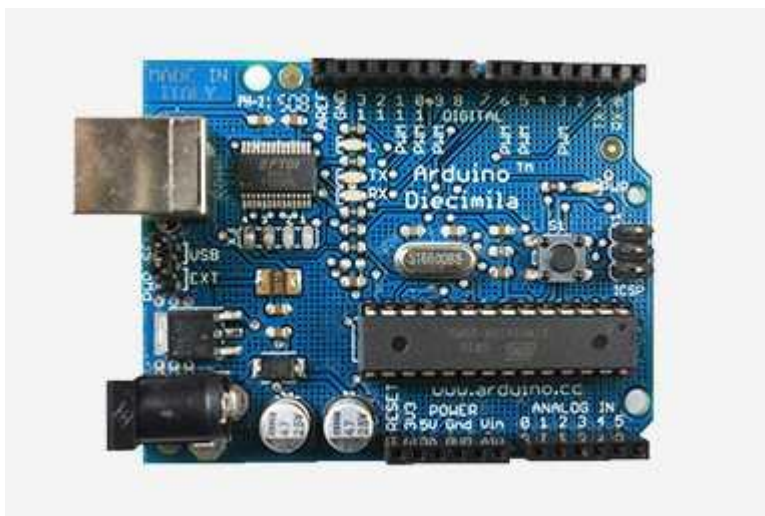
- Το Arduino Mini, μία έκδοση μινιατούρας του Arduino χρησιμοποιώντας τεχνολογία surface-mounted ATmega168.



- Το Arduino Mega, χρησιμοποιώντας τεχνολογία surface-mounted ATmega1280 για περαιτέρω I/O και μνήμη



- Το Arduino Diecimila, με ένα USB interface και χρησιμοποιεί τεχνολογία ATmega168 σε ένα DIP28 πακέτο.



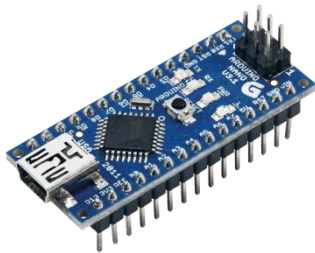
- Το Arduino Duemilanove (“2009”), χρησιμοποιεί τεχνολογία ATmega168 (ATmega328 για την καινούργια έκδοση) και τροφοδοτείται μέσω ενέργειας USB/DC, αυτόματα εναλλασσόμενης.



- Το Arduino Leonardo, με ένα ATmega32U4 chip που εξαλείφει την ανάγκη για συνδεσιμότητα μέσω USB και μπορεί να χρησιμοποιηθεί ως ψηφιακό πληκτρολόγιο ή ποντίκι. Κυκλοφόρησε στο Maker Faire Bay Area το 2012.



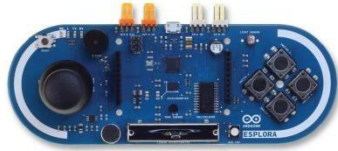
- Το Arduino Nano, ένα ακόμα πιο μικρό, USB τροφοδοτούμενη εκδοχή του Arduino χρησιμοποιώντας τεχνολογία surface-mounted ATmega168 (ATmega328 για την νεότερη έκδοση).



- Το Arduino Uno, χρησιμοποιώντας την ίδια τεχνολογία ATmega328



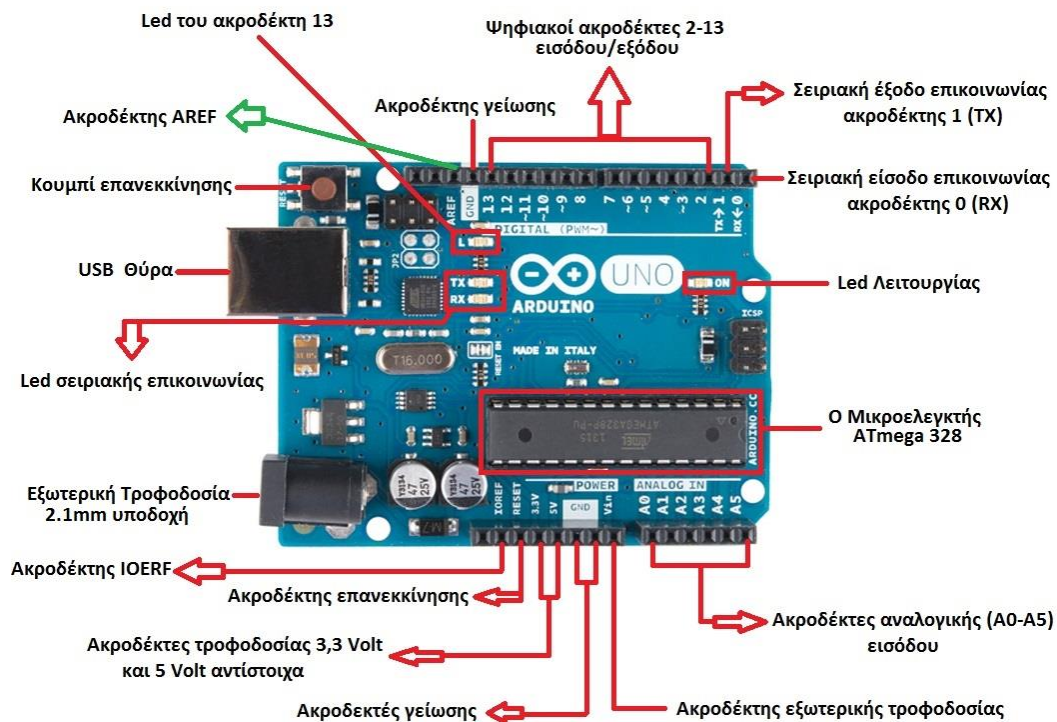
- Το Arduino Esplora, με εμφάνιση που παραπέμπει σε χειριστήριο κονσόλας βιντεοπαιχνιδιών με joystick και ενσωματωμένους αισθητήρες για ήχο, φως, θερμοκρασία και επιτάχυνση.



- Το Arduino Due είναι ένα μικροχειριστήριο board βασισμένο στην τεχνολογία Atmel SAM3X8E ARM Cortex-M3 CPU. Είναι το πρώτο board της Arduino βασισμένη σε επεξεργαστή 32-bit ARM microcontroller.[19]



3.2 Ο μικροελεγκτής Arduino Uno



Εικόνα 11.Arduino Uno

ΕΠΕΞΗΓΗΣΗ ΤΗΣ ΠΛΑΚΕΤΑΣ

Σε αυτήν την ενότητα θα αναλύσουμε το μοντέλο arduino που επιλέχθηκε για την υλοποίηση της κατασκευής.

Η επιλογή μας είναι το μοντέλο Arduino Uno το οποίο αποτελεί την βασική πλακέτα της τεχνολογίας Arduino.

Το Arduino Uno Rev3 βασίζεται στον μικροελεγκτή ATmega328 της Atmel. Είναι μια ολοκληρωμένη πλακέτα που περιέχει ότι χρειάζεται για να μπορεί να προγραμματιστεί και να λειτουργήσει συνδεδεμένης με ένα απλό καλώδιο USB στον υπολογιστή σας ή με ένα τροφοδοτικό στην πρίζα, επίσης μπορεί να λειτουργήσει με απλή μπαταρία.

Αναλυτικά η πλακέτα διαθέτει 14 ψηφιακές εισόδους ή εξόδους (6 από αυτές μπορεί να χρησιμοποιηθούν σαν PWM εξόδους), 6 αναλογικές εισόδους, 1 θύρα USB (τύπου B) για τον

προγραμματισμό και την τροφοδοσία της πλακέτας, 1 είσοδο τροφοδοσίας που μπορεί να χρησιμοποιηθεί εναλλακτικά για τροφοδοσία από τροφοδοτικό πρίζας ή από απλή μπαταρία, 1 υποδοχή ICSP και τέλος κουμπί για το reset της πλακέτας.

Ο μικροελεγκτής είναι συγχρονισμένος στους 16 megahertz (Crystal 16MHz). Αυτή η έκδοση του Arduino Uno είναι με μικροελεγκτή αποσπώμενο (DIP Version) και έχετε με αυτό την δυνατότητα να πειραματιστείτε άφοβα όσες φορές θέλετε, αν θα καεί ο ελεγκτής, διότι μπορείτε να τον αντικαταστήσετε με έναν καινούριο με μικρό κόστος.

Το Arduino Uno λειτουργεί με τροφοδοσία 5V DC από την είσοδο του USB ή με 7V~12V DC από την είσοδο της τροφοδοσίας.[19]

3.3 Χαρακτηριστικά Arduino Uno Rev3

- 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 [20]

3.4 Εισόδου/Εξόδου του μικροελεγκτή

Όπως προαναφέραμε η πλακέτα μας διαθέτει 14 ψηφιακές εισόδους ή εξόδους, 1 θύρα USB, 1 είσοδο για την τροφοδοσία, 1 υποδοχή ICSP και το κουμπί reset το οποίο κάνει επανεκκίνηση την πλακέτα.

Πιο αναλυτικά η λειτουργία των εισόδων και των εξόδων της πλακέτας:

- **Ψηφιακή είσοδος/έξοδος**

Οι 14 ακροδέκτες του Arduino έχουν τη δυνατότητα να λειτουργούν είτε ως ψηφιακές εισόδους είτε ως ψηφιακές εξόδους. Αυτό σημαίνει ότι μπορούν να παράγουν τάση εξόδου με τιμή είτε 0 είτε 5V, ενώ ταυτόχρονα μπορούν να διαβάσουν τάση εισόδου από 0V έως 5V. Για να χρησιμοποιήσουμε έναν από αυτούς τους ακροδέκτες ως είσοδο, χρησιμοποιούμε τη συνάρτηση `digitalRead(Pin)`, όπου το όρισμα `Pin` αναφέρεται στον αριθμό της θύρας που θέλουμε να πάρουμε είσοδο. Η συνάρτηση επιστρέφει την τιμή εισόδου, ενώ η τάση μπορεί να είναι είτε 0V είτε 5V.

LOW : όταν λάβει τάση 0 V στην είσοδο (pin)

HIGH : όταν λάβει τάση 5 V στην είσοδο (pin)

- **Αναλογική είσοδος**

Το Arduino διαθέτει 6 αναλογικές εισόδους, που αναγνωρίζονται με τα σύμβολα A0, A1, A2, A3, A4, και A5. Είναι δυνατόν να συνδεθεί ένα αναλογικό εξάρτημα σε οποιαδήποτε από αυτές τις θύρες και να διαβαστεί ως αναλογική είσοδος. Αυτή η λειτουργία επιτυγχάνεται χρησιμοποιώντας τη συνάρτηση `analogRead(Pin)`, όπου το όρισμα `Pin` αναφέρεται στον αριθμό της θύρας που θα χρησιμοποιηθεί ως αναλογική είσοδος.

- **Αναλογική έξοδος**

Ορισμένοι από τους 14 ακροδέκτες του Arduino διαθέτουν τη δυνατότητα PWM (Παλμοκωδικής Διαμόρφωσης Πλάτους), που σημαίνει ότι μπορούν να προσομοιώσουν αναλογική έξοδο μέσω παλμοκωδικής διαμόρφωσης. Με αυτόν τον τρόπο, μπορούμε να προσομοιώσουμε τιμές από το 0 έως το 255, αντιπροσωπεύοντας το διάστημα από 0 έως 5V. Για να επιτύχουμε αυτήν τη λειτουργία, χρησιμοποιούμε τη συνάρτηση `analogWrite(Pin, Value)`, όπου το όρισμα `Pin` αναφέρεται στον αριθμό της θύρας για την οποία θα ρυθμίσουμε το ρεύμα εξόδου, ενώ η τάση εξόδου περιορίζεται στο εύρος από 0V έως 5V.

Διπλή λειτουργία

Τέλος μερικοί από τους παραπάνω ακροδέκτες έχουν διπλή λειτουργία. Οπότε θα αναφερθούμε σε κάποια από αυτά:

- Οι ακροδέκτες 0 και 1 του Arduino δρουν ως διασύνδεση RX (Λήψη) και TX (Αποστολή) της σειριακής θύρας όταν το πρόγραμμά σας ενεργοποιεί τη σειριακή λειτουργία. Συνεπώς, όταν το πρόγραμμα σας αποστέλλει δεδομένα στη σειριακή θύρα, αυτά προωθούνται και στη θύρα USB μέσω του ελεγκτή Serial-Over-USB, αλλά επίσης και στον ακροδέκτη 0 για να μπορεί ενδεχομένως να τα διαβάσει μια άλλη συσκευή. Αυτό ενδέχεται να σημαίνει ότι, όταν ενεργοποιείται το σειριακό διασύνδεση, χάνονται δυνατότητες χρήσης δύο ψηφιακών ακροδεκτών ως είσοδους ή εξόδους.
- Οι ακροδέκτες 2 και 3 του Arduino διαθέτουν δυνατότητα εξωτερικών διακοπών (interrupts), γνωστοί ως interrupt 0 και interrupt 1 αντίστοιχα. Αυτό σημαίνει ότι μπορούμε να προγραμματίσουμε το πρόγραμμά μας έτσι, ώστε αυτοί οι ακροδέκτες να λειτουργούν αποκλειστικά ως ψηφιακές είσοδοι. Όταν συμβαίνουν συγκεκριμένες αλλαγές σε αυτές τις εισόδους, η κανονική ροή του προγράμματος διακόπτεται, και εκτελείται μια συγκεκριμένη συνάρτηση που έχουμε ορίσει.
- Οι ακροδέκτες 3, 5, 6, 9, 10 και 11 του Arduino διαθέτουν τη δυνατότητα να λειτουργούν ως ψευδοαναλογικές εξοδοι χρησιμοποιώντας το σύστημα PWM (Διαμόρφωση Εύρους Παλμών). Αυτό το σύστημα, που χρησιμοποιείται και από τις μητρικές των υπολογιστών για τον έλεγχο των ταχυτήτων ανεμιστήρων, επιτρέπει τον πλήρη έλεγχο της φωτεινότητας μιας συσκευής όπως ένα LED. Συνδέοντας, για παράδειγμα, ένα LED σε έναν από αυτούς τους ακροδέκτες, μπορούμε να ρυθμίσουμε τη φωτεινότητά του με ανάλυση 8 bit, αντί να έχουμε μόνο τη δυνατότητα ενεργοποίησης και απενεργοποίησής του, όπως συμβαίνει με τις υπόλοιπες ψηφιακές εξόδους.

Είναι σημαντικό να κατανοήσουμε ότι το σύστημα PWM (Διαμόρφωση Εύρους Παλμών) δεν λειτουργεί πραγματικά ως αναλογικό σύστημα. Όταν θέτουμε στην έξοδο μια τιμή, για παράδειγμα, 127, αυτό δεν σημαίνει ότι η έξοδος θα παράγει σταθερά 2.5V αντί της κανονικής τιμής των 5V. Αντίθετα, σημαίνει ότι θα παράγει έναν παλμό που εναλλάσσεται με μεγάλη συχνότητα, και ο αναλογικός αντίκτυπος προκύπτει από τον λόγο του χρόνου που η έξοδος βρίσκεται στο επίπεδο 5V προς τον συνολικό χρόνο του κύκλου.[9]

3.5 Τροφοδοσία

Για την πλακέτα Arduino, υπάρχουν δύο διαθέσιμοι τρόποι εφοδιασμού. Συγκεκριμένα, είναι εφικτό να τροφοδοτηθεί είτε μέσω της USB θύρας του υπολογιστή, είτε με ανεξάρτητη παροχή ενέργειας από μπαταρία. Η μονάδα παρέχει σταθερή τάση 5V στις εξόδους της. Σε περίπτωση παροχής ενέργειας από εξωτερική πηγή, απαιτείται τροφοδοσία από εξωτερικό βύσμα, και η εν λόγω εξωτερική τροφοδοσία πρέπει να κυμαίνεται από 7V έως 12V, ώστε η πλακέτα να λειτουργεί και να παρέχει σταθερά 5V στην έξοδο.

Η παροχή ρεύματος μπορεί να συνδεθεί απευθείας στους ακροδέκτες που είναι ειδικά σχεδιασμένοι για αυτό το σκοπό: το (+) συνδέεται στον ακροδέκτη VCC-IN, ενώ το (-) στον ακροδέκτη Gnd (γείωση) που βρίσκεται δίπλα. Στην περίπτωση που η μονάδα είναι μόνιμα συνδεδεμένη με τη θύρα USB, λειτουργεί απροβλημάτιστα με τα 5V που παρέχει η USB θύρα.

Σε αυτήν την περίπτωση θα χρησιμοποιήσουμε ρυθμιζόμενους αντάπτορες για την τροφοδοσία του κυκλώματος. Οι αντάπτορες θα συνδεθούν σε πολύμπριζο και στην συνέχεια στην πρίζα του τείχους. Το εύρος του αντάπτορα είναι από 3V έως 12V.



Εικόνα 12. Τροφοδοσία Arduino

Δίπλα από τα pin αναλογικής εισόδου, υπάρχει μια ακόμα συστοιχία από 6 pin με την σήμανση POWER.

Η λειτουργία του καθενός έχει ως εξής:

- Το πρώτο, με την ένδειξη RESET, όταν γειωθεί έχει ως αποτέλεσμα την επανεκκίνηση του Arduino.

- Το δεύτερο, με την ένδειξη 3.3V, μπορεί να τροφοδοτήσει τα εξαρτήματά σας με τάση 3.3V. Η τάση αυτή δεν προέρχεται από την εξωτερική τροφοδοσία αλλά παράγεται από τον ελεγκτή Serial_ over_USB και έτσι η μέγιστη ένταση που μπορεί να παρέχει είναι μόλις 50mA.
- Το τρίτο, με την ένδειξη 5V, μπορεί να τροφοδοτήσει τα εξαρτήματά με τάση 5V. Ανάλογα με τον τρόπο τροφοδοσίας του ίδιου του Arduino, η τάση αυτή προέρχεται είτε άμεσα από την θύρα USB, είτε από την εξωτερική τροφοδοσία αφού αυτή περάσει από ένα ρυθμιστή τάσης για να την «φέρει» στα 5V.
- Το τέταρτο και το πέμπτο pin, με την ένδειξη GND, είναι φυσικά γειώσεις.
- Το έκτο και τελευταίο pin, με την ένδειξη Vin έχει διπλό ρόλο. Σε συνδυασμό με το pin γείωσης δίπλα του, μπορεί να λειτουργήσει ως μέθοδος εξωτερικής τροφοδοσίας του Arduino, στην περίπτωση που δεν σας βολεύει να χρησιμοποιήσετε την υποδοχή του φινιρ των 2.1mm. Αν όμως έχετε ήδη συνδεδεμένη εξωτερική τροφοδοσία μέσω του φινιρ, μπορείτε να χρησιμοποιήσετε αυτό το pin για να τροφοδοτήσετε εξαρτήματα με την πλήρη τάση της εξωτερικής τροφοδοσίας (7-12V),πρίν αυτή περάσει από τον ρυθμιστή τάση όπως γίνεται με το pin των 5V.

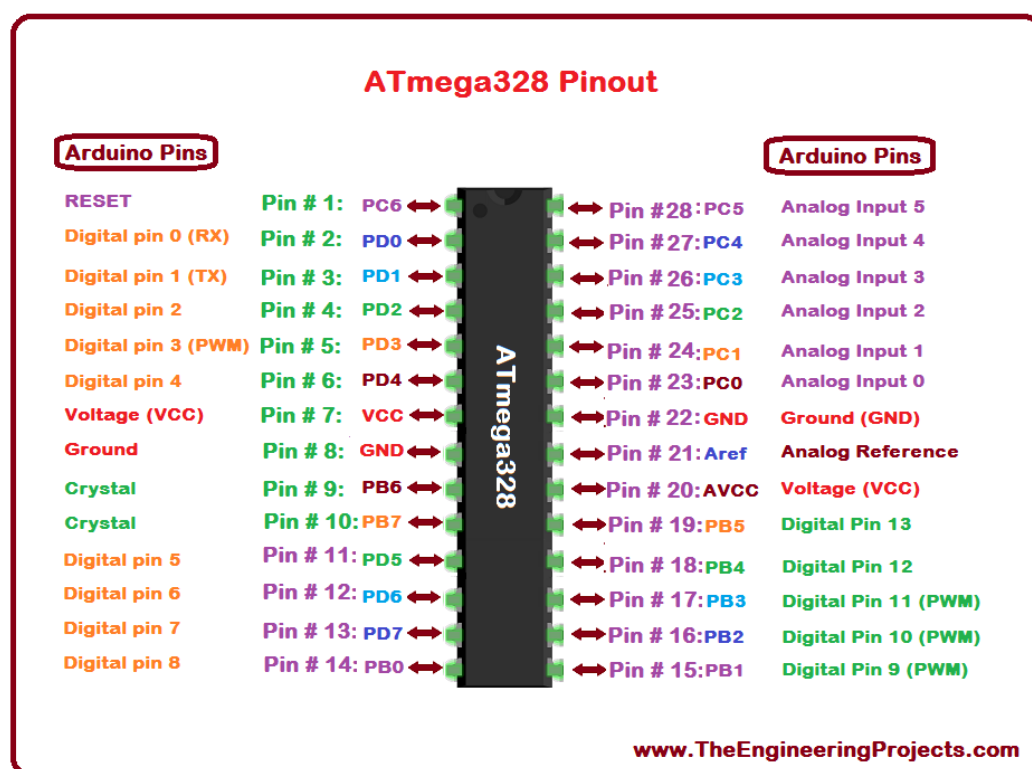
3.6 Μικροελεγκτής ATmega328

Όπως έχει γίνει αναφορά παραπάνω, η πλατφόρμα ARDUINO UNO χρησιμοποιεί τον μικροελεγκτή ATmega328. Ο μικροελεγκτής αυτός ανήκει στην κατηγορία των ‘Low Power Atmel AVR 8-bit Microcontroller’s. Είναι των 8 bit και βασίζεται στην αρχιτεκτονική RISC (Reduced Instruction Set Computing).

Ο πυρήνας του ATmega328 αποτελείται από την CPU, την μνήμη SRAM, την μνήμη EEPROM, την FLUSH και σύνολο περιφερειακών μονάδων.

Ο μικροελεγκτής ATmega328 μπορεί να υποστηρίξει τρία είδη σειριακής επικοινωνίας, την TWI (Two Wire Serial Interface) την USART (Universal Synchronous & Asynchronous Receiver & Transmitter) και την SPI (Serial Peripheral Interface). Κάθε μία αξιοποιείται σε διαφορετικές εφαρμογές ανάλογα με τις εκάστοτε απαιτήσεις και τα χαρακτηριστικά.

Στο παρακάτω σχήμα φαίνεται η δομή λειτουργίας του μικροελεγκτή ATmega328:



Εικόνα 13. Atmega328

Όπως προαναφέρθηκε, ο μικροελεγκτής Atmel 8-bit AVR RISC ενσωματώνει μια μνήμη flash ISP 32 KB με δυνατότητες ανάγνωσης κατά την εγγραφή, 1 KB EEPROM, 2 KB SRAM, 23 γραμμές I/O γενικής χρήσης, 32 καταχωρητές εργασίας γενικής χρήσης, καθώς και 3 ευέλικτους χρονοδιακόπτες/μετρητές με λειτουργίες σύγκρισης. Διαθέτει επίσης εσωτερικές και εξωτερικές διακοπές, σειριακό προγραμματιζόμενο USART, σειριακή διεπαφή 2 συρμάτων προσανατολισμένη στα byte, σειριακή θύρα SPI, μετατροπέα A/D 10-bit 6 καναλιών (8 κανάλια σε πακέτα TQFP και QFN/MLF), προγραμματιζόμενο χρονοδιακόπτη παρακολούθησης με εσωτερικό ταλαντωτή και 5 λειτουργίες εξοικονόμησης ενέργειας που μπορούν να επιλεγούν από το λογισμικό. Η συσκευή λειτουργεί μεταξύ 1,8 και 5,5 Volts. Η συσκευή επιτυγχάνει παροχή που προσεγγίζει το 1 MIPS/MHz.

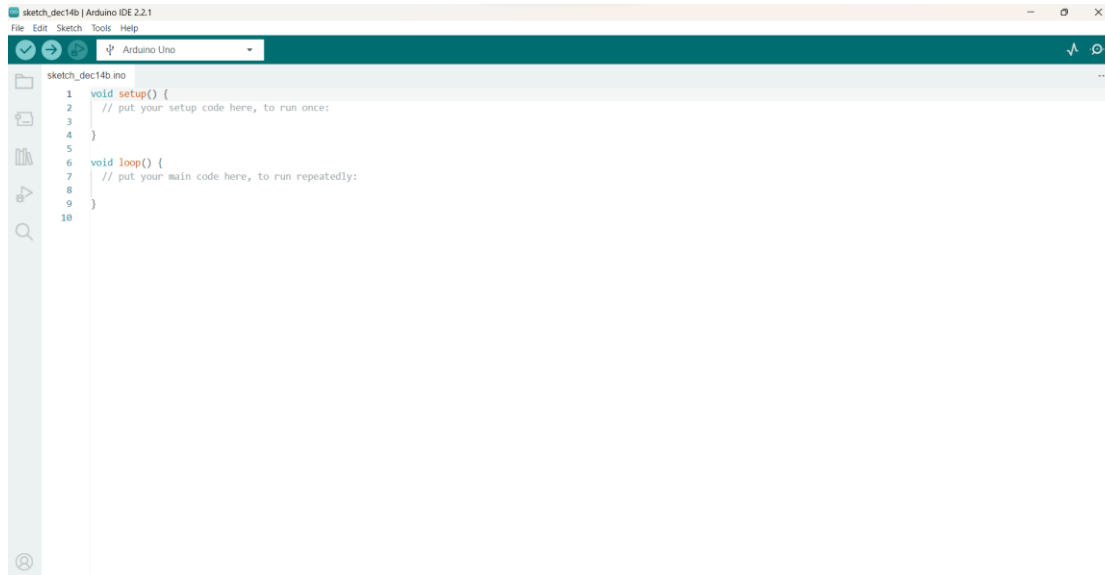
3.7 Το προγραμματιστικό περιβάλλον Arduino IDE

Το ολοκληρωμένο περιβάλλον προγραμματισμού (Arduino IDE: Integrated Development Environment) είναι μία εφαρμογή γραμμένη σε γλώσσα προγραμματισμού Java. Έχει σχεδιαστεί έτσι ώστε να εισαγάγει στον προγραμματισμό τους νέους και πολλούς άλλους που δεν είναι εξοικειωμένοι με την ανάπτυξη λογισμικού αλλά και την χρήση τέτοιου είδους εφαρμογών.

Μπορεί να λειτουργήσει σε διάφορες πλατφόρμες και προέρχεται από το IDE για την γλώσσα προγραμματισμού ανοιχτού κώδικα Processing και το σχέδιο Wiring. Το πρόγραμμα περιλαμβάνει ένα πλαίσιο επεξεργασίας κειμένου για τη συγγραφή κώδικα, μια κονσόλα μηνυμάτων, ένα μενού, μια γραμμή εργαλείων με κουμπιά για γρήγορη πρόσβαση στις βασικές λειτουργίες, καθώς και μια σειρά από μενού.

Ένα ολοκληρωμένο πρόγραμμα στο περιβάλλον Arduino ονομάζεται "sketch". Αυτό το πρόγραμμα είναι γραμμένο σε γλώσσα προγραμματισμού C ή C++, με το IDE να περιλαμβάνει τη βιβλιοθήκη Wiring. Αυτή η ενσωματωμένη βιβλιοθήκη βελτιώνει την απόδοση των λειτουργιών εισόδου/εξόδου. Επιπλέον, παρέχει βασικές λειτουργίες όπως αντιγραφή/επικόλληση και αναζήτηση κειμένου, προσφέροντας έτσι ένα περιβάλλον ευέλικτο και εύκολο στη χρήση για τους προγραμματιστές.

Η κονσόλα μηνυμάτων στο περιβάλλον Arduino εμφανίζει την έξοδο κειμένου, συμπεριλαμβανομένων πλήρων μηνυμάτων λάθους και άλλων χρήσιμων πληροφοριών. Τα κουμπιά στη γραμμή εργαλείων παρέχουν επιλογές όπως έλεγχος, ανέβασμα προγραμμάτων, δημιουργία νέου sketch, άνοιγμα και αποθήκευση sketch, καθώς και άνοιγμα της σειριακής οθόνης όπου εκτυπώνονται τα αποτελέσματα του προγράμματος.



Εικόνα 14. Προγραμματιστικό περιβάλλον Arduino IDE

ΒΑΣΙΚΕΣ ΛΕΙΤΟΥΡΓΙΕΣ ΤΟΥ IDE:

- 1) Εγκρίνει τον κώδικα. Ελέγχει τον κώδικα για τυχόν λάθη στην σύνταξη αυτού.
- 2) Φορτώνει το πρόγραμμα στην πλακέτα που έχουμε συνδέσει.
- 3) Φορτώνει έναν υπάρχον έτοιμο κώδικα .
- 4) Αποθηκεύει τον κώδικα που μόλις έχουμε γράψει.
- 5) Η κονσόλα κειμένου εμφανίζει τα μηνύματα από τυχόν σφάλματα.

setup (): μία συνάρτηση εκτελείται μία φορά κατά την έναρξη ενός προγράμματος η οποία μπορεί να αρχικοποιεί τις ρυθμίσεις.

loop (): μία συνάρτηση η οποία καλείται συνεχώς μέχρι να απενεργοποιηθεί η πλακέτα.

3.8 Σειριακή Θύρα Επικοινωνίας

Η σύνδεση με την πλακέτα του Arduino γίνεται μέσω μιας σειριακής θύρας για τη φόρτωση των προγραμμάτων και για να επικοινωνούν μεταξύ τους.

Για το σκοπό αυτό χρησιμοποιείται η σύνδεση με καλώδιο USB. Για ενεργοποίηση της σειριακής θύρας επικοινωνίας αρκεί να δοθεί στη διαδικασία `setup()` η εντολή `Serial.begin(BaudRate)`, όπου το `BaudRate` εκφράζει το ρυθμό με τον οποίο θα μεταδίδονται τα bits.

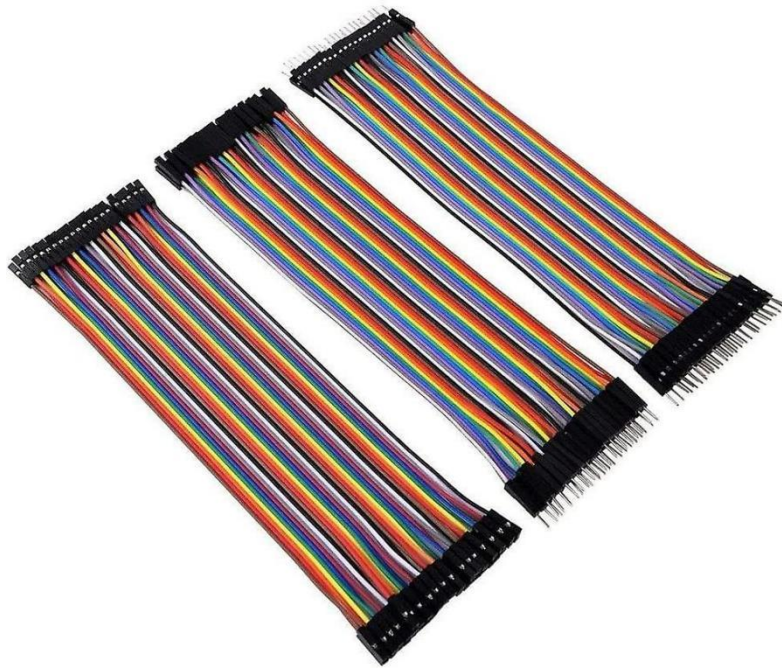
Η σειριακή θύρα μπορεί να χρησιμοποιηθεί σε εφαρμογές αμφίδρομης επικοινωνίας, προκειμένου να γίνεται αποστολή ή λήψη δεδομένων. Αυτό σημαίνει ότι ο χρήστης του μικροελεγκτή μπορεί να διαβάζει τις τιμές που παράγουν οι αισθητήρες και να παρακολουθεί τις τιμές των μεταβλητών μέσω της οθόνης σειριακής επικοινωνίας.

Μια χρήσιμη εντολή για αυτόν τον σκοπό είναι η `print()`, η οποία εκτυπώνει ένα μήνυμα ή τιμές. Επίσης, η `println()` λειτουργεί ακριβώς το ίδιο, αλλά προσθέτει αλλαγή γραμμής μετά από κάθε εκτύπωση. Όταν η σύνδεση του Arduino γίνεται μέσω USB στον υπολογιστή, η σειριακή οθόνη ενεργοποιείται μέσω του εικονιδίου "Σειριακή Οθόνη" στην πάνω δεξιά γωνία. Στο παράθυρο που ανοίγει, εμφανίζονται όλα τα μηνύματα που στέλνονται από τον κώδικα που έχει φορτωθεί και τρέχει στην πλακέτα. [1][2][3][15]

4. Περιγραφή του τρόπου κατασκευής

4.1 Υλικά Διασύνδεσης

Ο τρόπος με τον οποίο θα συνδέσουμε τα υλικά είναι με βραχυκυκλωτήρες ή jumper cables. Στην εργασία αυτή χρησιμοποιήσαμε jumper cables με αρσενικό και θηλυκό ακροδέκτη καθώς δεν χρειάστηκε να συνδέσουμε κάποιο εξάρτημα πάνω σε κάποιο ράστερ.



Εικόνα 15. Καλώδια με αρσενικό και θηλυκό ακροδέκτη.

4.2 Αισθητήρες

Στην ενότητα αυτή θα παρουσιάσουμε τον τρόπο με τον οποίο το Arduino συνδέεται ξεχωριστά με τον κάθε αισθητήρα αλλά και τον τρόπο με τον οποίο δέχεται τα ερεθίσματα μέσω των αισθητήρων που είναι συνδεδεμένοι επάνω στην πλακέτα και τα αποτελέσματα της λειτουργίας τους.

Οι αισθητήρες που επιλέξαμε να χρησιμοποιήσουμε για να ικανοποιούν τις ανάγκες της εργασίας αυτής είναι αισθητήρες: θερμοκρασίας, οξύμετρίας και αγωγιμότητας(TDS).

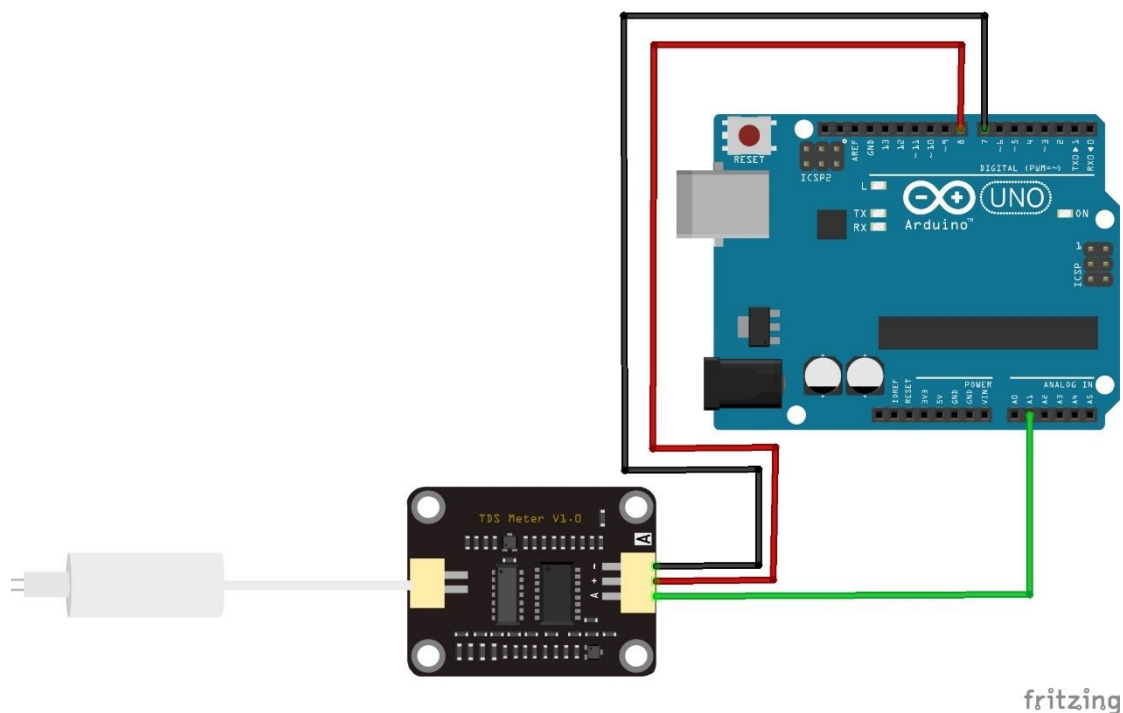
Όλοι οι αισθητήρες είναι εύκολοι στην σύνδεση τους με το Arduino και το μόνο που πρέπει να γνωρίζουμε πριν τις συνδέσουμε είναι η θέση που πρέπει να πάρει σύμφωνα με αυτό που θέλουμε να μετρήσουμε. Χρειαζόμαστε σε κάποιους από αυτούς τα jumper cables που έχουμε προαναφέρει.

4.3 Αισθητήρας Αγωγιμότητας(TDS)

Ο αισθητήρας TDS ανιχνεύει τα διαλυμένα μέταλλα που βρίσκονται μέσα στο νερό για να μας βγάλει σαν αποτέλεσμα πόσο καθαρό είναι το νερό. Μετράει πόσα χιλιογραμμάρια στερεών μετάλλων είναι διαλυμένα μέσα σε ένα λίτρο νερού. Γενικότερα, όσο μεγαλύτερος ο αριθμός που βγάζει σαν μέτρηση, τόσα περισσότερα στοιχεία έχει το νερό και αυτό δείχνει ότι το νερό δεν είναι καθαρό. Για να πάρουμε τη μέτρηση μέσα στο νερό αυτό που γίνεται είναι ότι μετράμε όλα τα διαλυμένα στοιχεία που υπάρχουν στο νερό με μονάδα μέτρησης PPM ή αλλιώς χιλιογραμμάρια ανά λίτρο (mg/l). Το ηλεκτρόδιο του αισθητήρα μπορεί να μετρήσει αγωγή μετρίων, βαρέων μετρίων και ιόντων του νερού.

Η τάση που ενεργοποιεί τον αισθητήρα είναι από 3.3V έως 5.5V και μπορεί να μετρήσει μέχρι και 1000ppm. Η ακρίβεια του αισθητήρα είναι $\pm 10\%$ F.S. (25 °C).[8]

Η σύνδεσή του γίνεται με τον παρακάτω τρόπο:

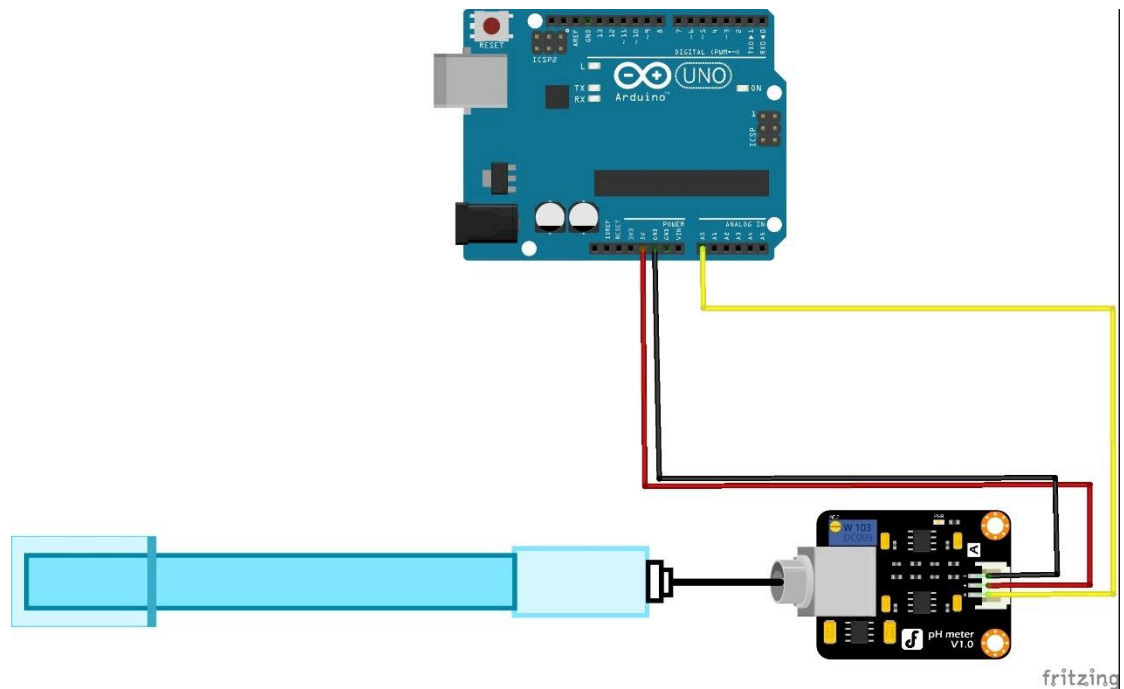


Εικόνα 16. Σύνδεση αισθητήρα TDS

4.4 Αισθητήρας Οξύτητας(PH)

Ο αισθητήρας οξύτητας pH που χρησιμοποιώ για να μετρήσω το pH του νερού μπορεί να ανιχνεύσει pH από 0 έως και 14 στη βαθμίδα μέτρησης του pH. Είναι ιδανικός για να χρησιμοποιείται με μικροεπεξεργαστές που είναι πάνω στις πλακέτες Arduino. Έχει ένα LED το οποίο δείχνει την παρουσία τάσης και ενεργοποίησης του αισθητήρα καθώς συνδέεται με το Arduino, ένα βύσμα BNC και ένα pH ηλεκτρόδιο σαν σένσορα. Ενεργοποιείται με τάση ίσης των 5V DC και η καλύτερη θερμοκρασία στην οποία μπορεί να έχει τα καλύτερα αποτελέσματα είναι η θερμοκρασία δωματίου 25 βαθμών κελσίου.

Η σύνδεσή του στο κύκλωμα γίνεται με τον παρακάτω τρόπο:



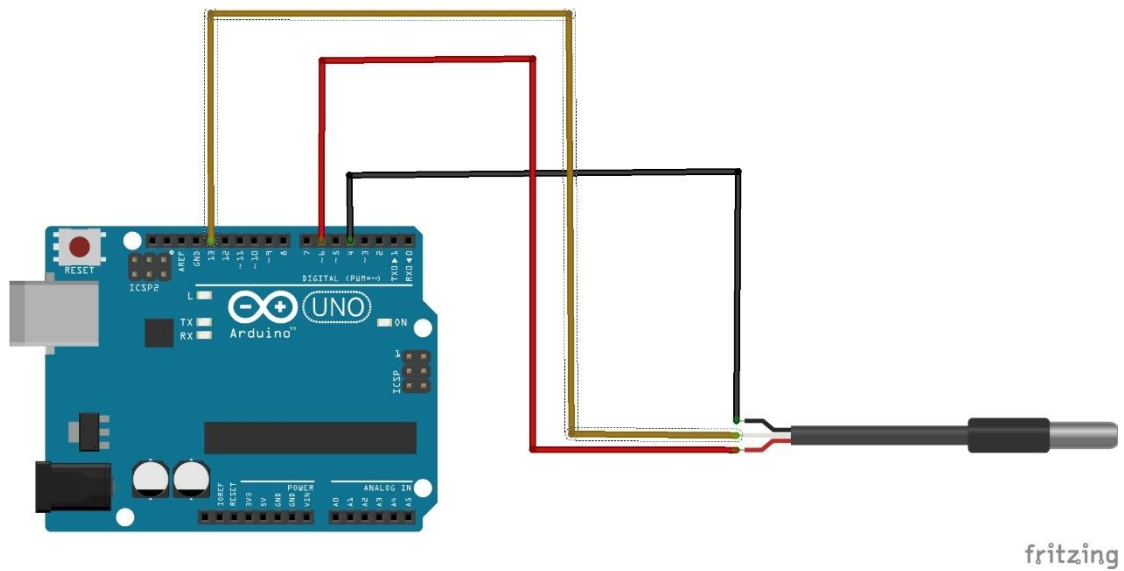
Εικόνα 17. Σύνδεση αισθητήρα pH

4.5 Αισθητήρας Θερμοκρασίας(C)

Ο αισθητήρας που χρησιμοποιώ για να μετρήσω την θερμοκρασία του νερού ονομάζεται DS18B20. Είναι ένας κλασικός αισθητήρας θερμοκρασίας που χρησιμοποιείται σε πολλά πειράματα και project. Είναι αδιάβροχος και μπορεί να χρησιμοποιηθεί σε διάφορες συνθήκες όπως ανίχνευση θερμοκρασίας νερού σε δεξαμενή όπως στη πτυχιακή μου εργασία, μέτρηση θερμοκρασίας χώματος κλπ.

Ο συγκεκριμένος αισθητήρας θερμοκρασίας χρειάζεται τροφοδοσία 3~5.5V , ανιχνεύει θερμοκρασίες μεταξύ -55 και 125 βαθμών κελσίου και είναι συμβατός με raspberry pi και πλακέτες Arduino.

Η σύνδεσή του στο κύκλωμα γίνεται με τον παρακάτω τρόπο:

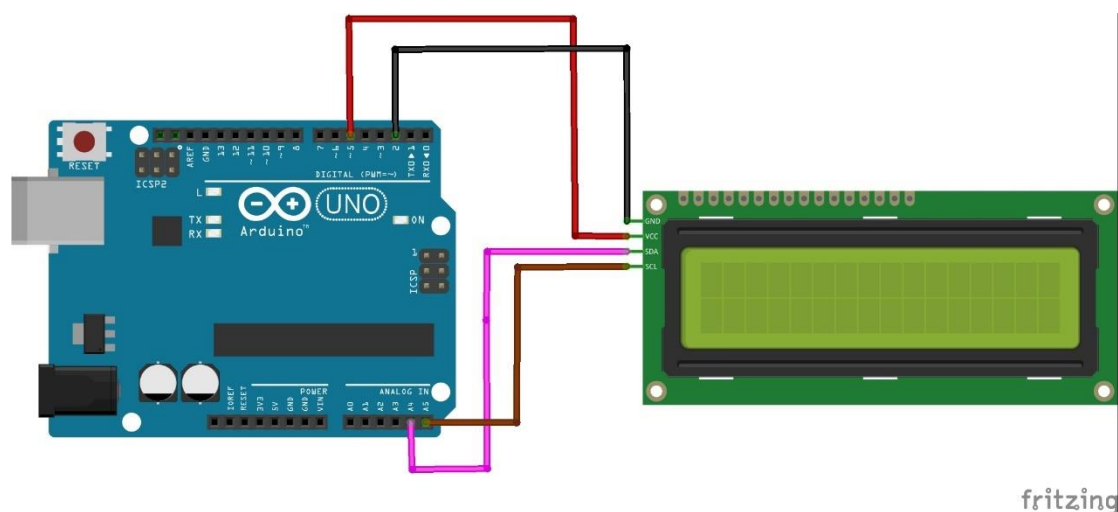


Εικόνα 18. Σύνδεση αισθητήρα θερμοκρασίας

4.6 Οθόνη LCD

Μία οθόνη υγρών κρυστάλλων είναι ο συνδυασμός δύο φίλτρων πόλωσης και μίας διάταξης υγρών κρυστάλλων. Ένας υγρός κρύσταλλος είναι μία ελεγχόμενη από ηλεκτρικό πεδίο διάταξη, η οποία μπορεί να αλλάζει ή να μη αλλάζει την πόλωση του φωτός που περνά μέσα απ' αυτό. Επειδή η διάταξη αυτή δεν παράγει μόνη της φως, χρησιμοποιείται ανάκλαση φωτισμού (backlight) που παράγεται από λαμπτήρες φθορισμού και κατευθύνεται προς τους υγρούς κρυστάλλους.

Η σύνδεση του γίνεται με τον παρακάτω τρόπο:

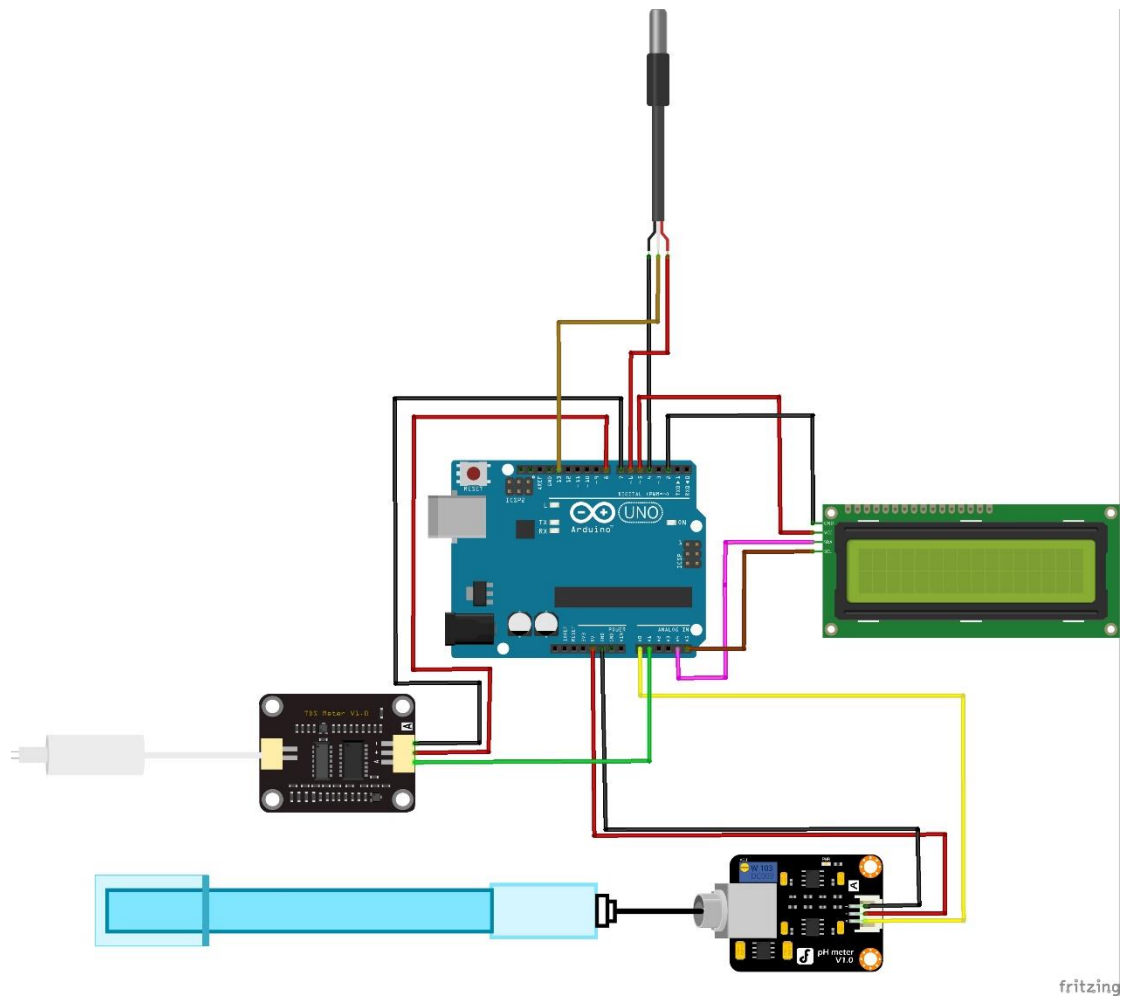


Εικόνα 19. Σύνδεση οθόνης LCD

Η οθόνη LCD (οθόνη υγρών κρυστάλλων) είναι μια ηλεκτρονική μονάδα ενδείξεων και βρίσκει ένα ευρύ φάσμα εφαρμογών. Μια οθόνη 16x2 LCD(σημαίνει ότι μπορεί να εμφανίσει 16 χαρακτήρες ανά γραμμή και υπάρχουν 2 τέτοιες γραμμές.) είναι μια πολύ βασική μονάδα και χρησιμοποιείται πολύ συχνά σε διάφορες συσκευές και κυκλώματα.

Οι οθόνες LCD είναι οικονομικές, εύκολα προγραμματιζόμενες και δεν έχουν περιορισμούς στην εμφάνιση ειδικών χαρακτήρων. Αυτή η οθόνη LCD έχει δύο καταχωρητές, την εντολή και τα δεδομένα. Ο κατάλογος εντολών αποθηκεύει τις εντολές που δίνονται στην οθόνη LCD. Μια εντολή δίνεται στην οθόνη LCD για να κάνει μια προκαθορισμένη εργασία όπως την προετοιμασία της, την εκκαθάριση της οθόνης, τη ρύθμιση της θέσης του δρομέα, τον έλεγχο της εμφάνισης κλπ. Ο κατάλογος δεδομένων αποθηκεύει τα δεδομένα που θα εμφανίζονται στην οθόνη LCD.

Τα SCL και SDA. Η γραμμή SCL είναι η γραμμή ρολογιού, ενώ η SDA είναι η γραμμή δεδομένων. Οι γραμμές αυτές συνδέονται σε όλες τις συσκευές, που υπάρχουν πάνω στο διάλυτο I²C. Προφανώς εκτός από τα παραπάνω καλώδια που μεταφέρουν δεδομένα, απαιτείται και ένα τρίτο καλώδιο το οποίο είναι η γείωση (GND) ή 0 V.



Εικόνα20. Κύκλωμα πτυχιακής εργασίας

4.7 Τροφοδοσία Αισθητήρων

Η τροφοδοσία των τριών αισθητήρων καθώς και της οθόνης LCD γίνεται εξ ολοκλήρου από το Arduino Uno R3. Ο κυριότερος λόγος είναι για να κάνουμε τη συσκευή όσο πιο αυτόνομη γίνεται. Εφόσον το Arduino μπορεί να δώσει έξοδο τάση 5V έπειτα από τον προγραμματισμό συγκεκριμένων θυρών PWM , το εκμεταλευόμαστε προκειμένου να μην χρειάζεται έξτρα εξωτερική τροφοδοσία. Τέτοιου είδους συσκευές συνήθως είναι φορητές και η τροφοδοσία της γίνεται με τη χρήση μπαταριών.

4.8 Βαθμονόμηση Αισθητήρων(Calibration)

Οι αισθητήρες του κυκλώματος TDS και pH , χρειάζονται μια βαθμονόμηση πριν ξεκινήσουμε να παίρνουμε μετρήσεις προκειμένου να έχουμε όσο το δυνατόν καλύτερα αποτελέσματα μπορούμε.

Ο αισθητήρας TDS προκειμένου να πάρει τη σωστή μέτρηση χρειάζεται προηγουμένως μια βαθμονόμηση. Ο τρόπος με τον οποίο την πραγματοποιούμε είναι εκτελώντας έναν κώδικα μέσω του Arduino IDE δίνοντας μια γνωστή μέτρηση στον αισθητήρα για να τον βαθμονομήσουμε. Έπειτα , τρέχουμε τον κώδικα προγράμματος κανονικά.

Για να βαθμονομήσουμε τον αισθητήρα του pH πρέπει να βραχυκυκλώσουμε το εξωτερικό μέρος του κονέκτορα BNC με το κέντρο αυτού. Αυτό μας δίνει έξοδο 2.5V στο pin Po Analog. Εκτελώντας τον κώδικα με τον οποίο ο αισθητήρας βαθμονομείται θα πρέπει να δούμε στην έξοδό του την τιμή 2.5 V . Αν δεν την δούμε , γυρνάμε το trimmer μέχρι αυτή η τιμή να εμφανιστεί. Η τιμή αυτή ακριβώς είναι η μέση τιμή του pH από τις τιμές που μπορεί να έχει(0-14) δηλαδή ο αριθμός 7.[4][5][7][9][10]

Κώδικας TDS Calibration:

```
#include <EEPROM.h>
#include "GravityTDS.h"

#define TdsSensorPin A1
GravityTDS gravityTds;

float temperature = 25,tdsValue = 0;

void setup()
{
  Serial.begin(115200);
  gravityTds.setPin(TdsSensorPin);
  gravityTds.setAref(5.0); //reference voltage on ADC, default 5.0V on Arduino UNO
  gravityTds.setAdcRange(1024); //1024 for 10bit ADC;4096 for 12bit ADC
  gravityTds.begin(); //initialization
}

void loop()
{
  //temperature = readTemperature(); //add your temperature sensor and read it
  gravityTds.setTemperature(temperature); // set the temperature and execute temperature
  compensation
  gravityTds.update(); //sample and calculate
  tdsValue = gravityTds.getTdsValue(); // then get the value
  Serial.print(tdsValue,0);
  Serial.println("ppm");
  delay(1000);
}
```

Στην γραμμή εντολών του Arduino IDE πληκτρολογούμε: enter , για να μπούμε στην διαδικασία βαθμονόμησης

Έπειτα , πληκτρολογούμε cal:TDSvalue , όπου TDSValue τιμή ενός γνωστού διαλύματος.

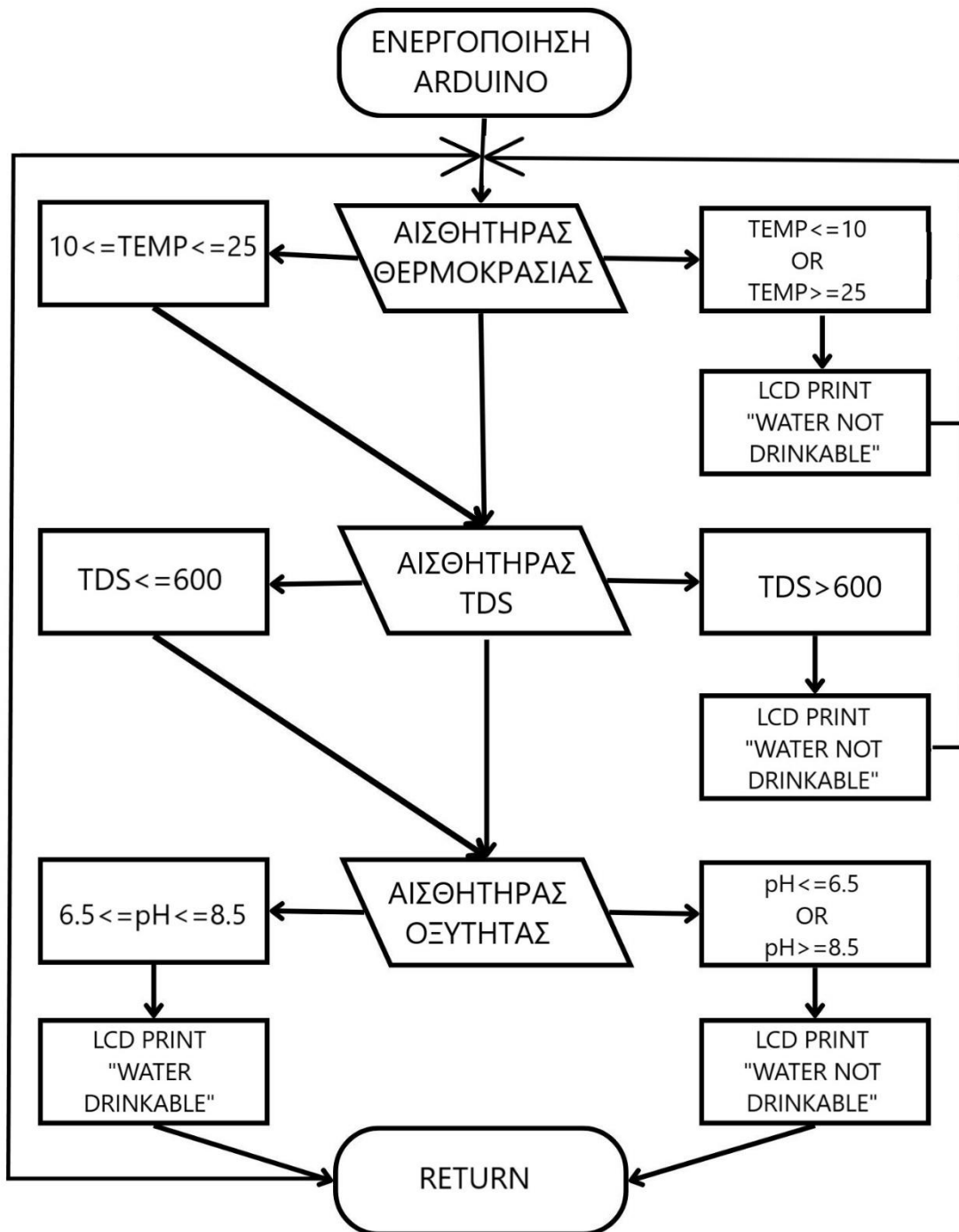
Στην συνέχεια πληκτρολογούμε exit και προχωράμε στον κώδικα του κυκλώματός μας.

Κώδικας pH Calibration:

```
int pH_Value;
float Voltage;

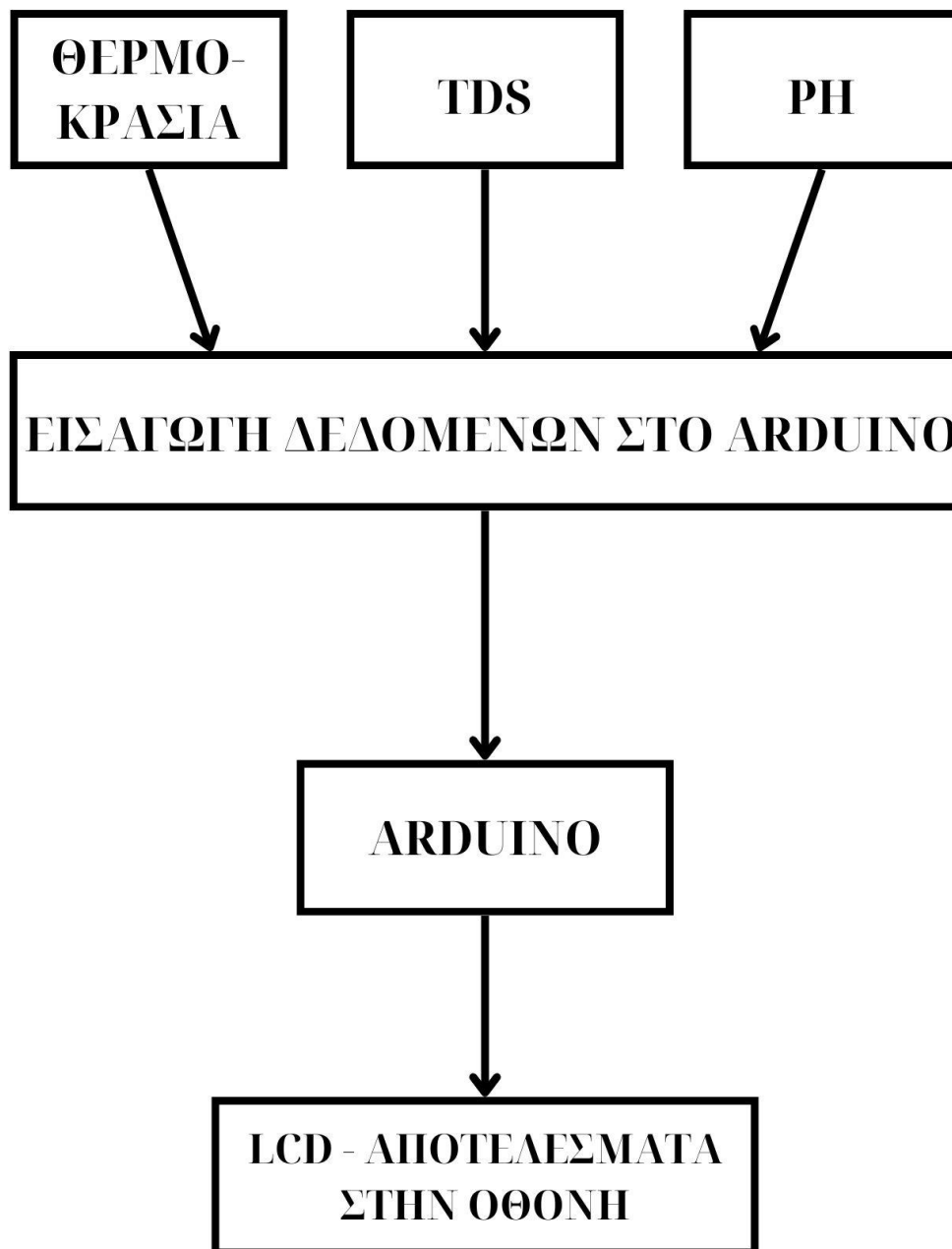
void setup()
{
  Serial.begin(9600);
  pinMode(pH_Value, INPUT);
}
void loop()
{
  pH_Value = analogRead(A0);
  Voltage = pH_Value * (5.0 / 1023.0);
  Serial.println(Voltage);
  delay(500);
}
```

4.9 Διάγραμμα Ροής Προγράμματος



Εικόνα 21. Διάγραμμα ροής προγράμματος

4.10 Block Διάγραμμα Κυκλώματος



Εικόνα 22. Διάγραμμα ροής προγράμματος

5. ΔΗΜΙΟΥΡΓΙΑ ΤΟΥ ΚΩΔΙΚΑ

5.1 ΒΑΣΙΚΕΣ ΕΝΤΟΛΕΣ

Οι βασικές εντολές που χρησιμοποιούνται είναι οι εξής:

- **pinMode(...)**; :Ρυθμίζει την ακίδα(pin)να συμπεριφέρεται σαν είσοδος/έξοδος.
- **analogRead(...)**; :Διαβάζει μια τιμή από την ακίδα που θα δηλώσουμε.
- **If(...)**; :Δομή ελέγχου μιας συνθήκης.
- **If(...)...else** :Δομή ελέγχου πολλαπλών συνθηκών.
- **delay(...)**; :Είναι ο χρόνος καθυστέρησης μέχρι να επαναληφθεί το περιεχομενο της `void loop(){ }`
- **lcd.print(...)** : Εκτύπωση αποτελεσμάτων του κώδικα

5.2 Ο ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΣ ΤΗΣ ΚΑΤΑΣΚΕΥΗΣ

Παρακάτω θα αναφερθούμε στον προγραμματισμό του κυκλώματος και η επεξήγηση ορισμένων εντολών στη μορφή που έχουν τα σχόλια (//) στο προγραμματιστικό περιβάλλον IDE.[3][16]

```
#include <EEPROM.h> // Δήλωση βιβλιοθηκών που θα χρησιμοποιηθούν στο πρόγραμμα.
```

```
#include "GravityTDS.h"
```

```
#include <LiquidCrystal_I2C.h>
```

```
#include <OneWire.h>
```

```
#include <DallasTemperature.h>
```

```
#include <Wire.h>
```

```
#define VCC2 5 //Δήλωση για χρήση έξτρα θυρών για τροφοδοσία των αισθητήρων.
```

```
#define GND2 2
```

```
#define VCC3 6
```

```
#define GND3 4
```

```
#define VCC4 8
```

```
#define GND4 7
```

```
OneWire oneWire(13); // Δήλωση μεταβλητών
```

```
int phval = 0;
```

```
unsigned long int avgValue;
```

```
int buf[10], temp;
```

```
LiquidCrystal_I2C lcd(0x3F,16,2);
```

```
DallasTemperature sensors(&oneWire);
```

```
GravityTDS gravityTds;
```

```
float Tdsvalue=0 ;
```

```
float voltage ;
```

```
float pH_value ;
```

```
void setup()
```

```
{
```

```
pinMode(A0, INPUT); //δήλωση PIN για χρήση του σαν είσοδος
```

```
pinMode(A1,INPUT); //δήλωση PIN για χρήση του σαν είσοδος
```

```
Serial.begin(9600);
```

```
sensors.begin();
```

```
pinMode(VCC2,OUTPUT);//Δήλωση της θύρας σαν έξοδο
```

```
digitalWrite(VCC2, HIGH);// Δήλωση της πόρτας σαν HIGH για να πάρει έξοδο 5V
```

```
pinMode(VCC3,OUTPUT);//
```

```
digitalWrite(VCC3, HIGH);//
```

```
pinMode(VCC4,OUTPUT);//
```

```
digitalWrite(VCC4, HIGH);//
```

```
pinMode(GND2,OUTPUT);// Δήλωση της θύρας σαν έξοδο
```

```
digitalWrite(GND2, LOW);// Δήλωση της πόρτας σαν LOW για να λειτουργεί σαν γείωση
```

```
pinMode(GND3,OUTPUT);//
```

```

digitalWrite(GND3, LOW);//

pinMode(GND4,OUTPUT);//
digitalWrite(GND4, LOW);//

lcd.init();
lcd.backlight();
gravityTds.setPin(A1); // εντολές βιβλιοθήκης για μέτρηση TDS
gravityTds.setAref(5.0);
gravityTds.setAdcRange(1024);
gravityTds.begin();
}

void loop()
{
sensors.requestTemperatures(); // εντολές για μέτρηση θερμοκρασίας
float temp_c=sensors.getTempCByIndex(0);
gravityTds.setTemperature(temp_c);
gravityTds.update();
Tdsvalue = gravityTds.getTdsValue(); // η τελική μέτρηση για την τιμή του TDS
for(int i=0;i<10;i++){
buf[i]=analogRead(A0); // μέτρηση για 10 τιμές για το Μ.Ο. αναλογικής μέτρησης pH
delay(100);
}
for(int i=0;i<9;i++){//διαδικασία για να κρατήσουμε το Μ.Ο. των αναλογικών μετρήσεων pH
for(int j=i+1;j<10;j++){
if(buf[i]>buf[j]){
temp=buf[i];
buf[i]=buf[j];
buf[j]=temp;
}
}
}
}

```

```

    }
}
avgValue=0;
for(int i=2;i<8;i++)avgValue+=buf[i];
float pHValue=(float)avgValue*5.0/1024/6; //υπολογισμός μ.ο. pH αναλογικών μετρήσεων
float Po= 7 + ((2.5 - pHValue) / 0.13) ; // μετατροπή σε μονάδα pH
lcd.setCursor(0,0); // τοποθέτηση κέρσορα της οθόνης στην πρώτη σειρά πρώτο ψηφίο
lcd.print("Temp:");//εκτύπωση της λέξης Temp:
lcd.print(temp_c);// εκτύπωση μέτρησης θερμοκρασίας
lcd.print("C");// εκτύπωση του γράμματος C για να απεικονίσει τους βαθμούς κελσίου
lcd.setCursor(0,1);// τοποθέτηση κέρσορα της οθόνης στην δεύτερη σειρά πρώτο ψηφίο
lcd.print("TDS:");// εκτύπωση της λέξης TDS:
lcd.print(Tdsvalue); // εκτύπωση μέτρησης TDS
lcd.print("ppm");//εκτύπωση των γραμμάτων ppm
delay(3000);// καθυστέρηση για 3 δευτερόλεπτα
lcd.clear();//καθαρισμός περιεχομένου οθόνης LCD
lcd.setCursor(0,0);// τοποθέτηση κέρσορα της οθόνης στην πρώτη σειρά πρώτο ψηφίο
lcd.print("pH:");// εκτύπωση της λέξης pH
lcd.print(Po, 2);// εκτύπωση μέτρησης pH με απεικόνιση 2 αριθμών μετά την υποδιαστολή
lcd.print(" pH");// εκτύπωση της μονάδας μέτρησης pH
if ((temp_c<=25) && (temp_c>=10)) // έλεγχος τιμών θερμοκρασίας
{
    if(Tdsvalue<=600)// έλεγχος τιμής TDS
    {
        if((Po<=8.5) && (Po>=6.5)) { //έλεγχος τιμής pH
            lcd.setCursor(0,1); τοποθέτηση κέρσορα της οθόνης στην δεύτερη σειρά πρώτο
ψηφίο
            lcd.print("Water drinkable");// εκτύπωση ότι το νερό είναι πόσιμο
        } else {
            lcd.setCursor(0,1);// τοποθέτηση κέρσορα της οθόνης στην δεύτερη σειρά πρώτο
ψηφίο
            lcd.print("Water not drinkable");// εκτύπωση ότι το νερό δεν είναι πόσιμο
        }
    }
}

```

```

}else if (Tdsvalue>600) // έλεγχος τιμής TDS

{
    lcd.setCursor(0,1 ); τοποθέτηση κέρσορα της οθόνης στην δεύτερη σειρά πρώτο
ψηφίο
    lcd.print("Water not drinkable");// εκτύπωση ότι το νερό δεν είναι πόσιμο
}
}
else if((temp_c<10) || (temp_c>25)) { // έλεγχος τιμών θερμοκρασίας
    lcd.setCursor(0,1); τοποθέτηση κέρσορα της οθόνης στην δεύτερη σειρά πρώτο
ψηφίο
    lcd.print("Water not drinkable");// εκτύπωση ότι το νερό είναι πόσιμο
}

delay(3000);// καθυστέρηση για 3 δευτερόλεπτα
lcd.clear();//καθαρισμός περιεχομένου οθόνης LCD
}

```

5.3 ΑΠΟΤΕΛΕΣΜΑΤΑ ΑΠΟ ΤΗΝ ΟΘΟΝΗ LCD



Εικόνα 23. Εμφάνιση θερμοκρασίας $>25^{\circ}\text{C}$ οπότε ο κώδικας θα μας εμφανίσει ότι το νερό δεν είναι πόσιμο.



Εικόνα 24. Εμφάνιση ότι το νερό δεν είναι πόσιμο σωστά ,σύμφωνα με τον έλεγχο του κώδικα, παρότι οι άλλες 2 μετρήσεις είναι μέσα στα επιτρεπτά όρια.



Εικόνα 25. Εμφάνιση 2 μετρήσεων στα επιτρεπτά όρια.



Εικόνα 26. Επιτρεπτό και το όριο της οξύτητας και εμφάνιση ότι το νερό είναι πόσιμο.



Εικόνα 27. Μέτρηση νερού βρύσης αφού περάσει από φίλτρο.



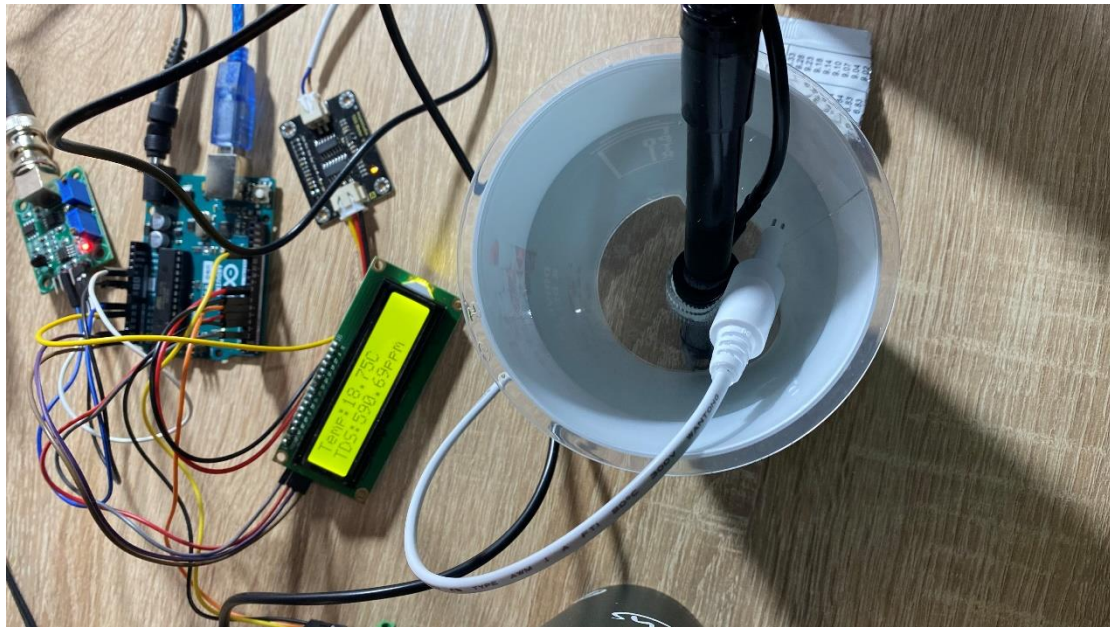
Εικόνα 28. Εμφάνιση ότι το νερό είναι πόσιμο αφού όλες οι μετρήσεις μας είναι εντός επιτρεπτών ορίων.



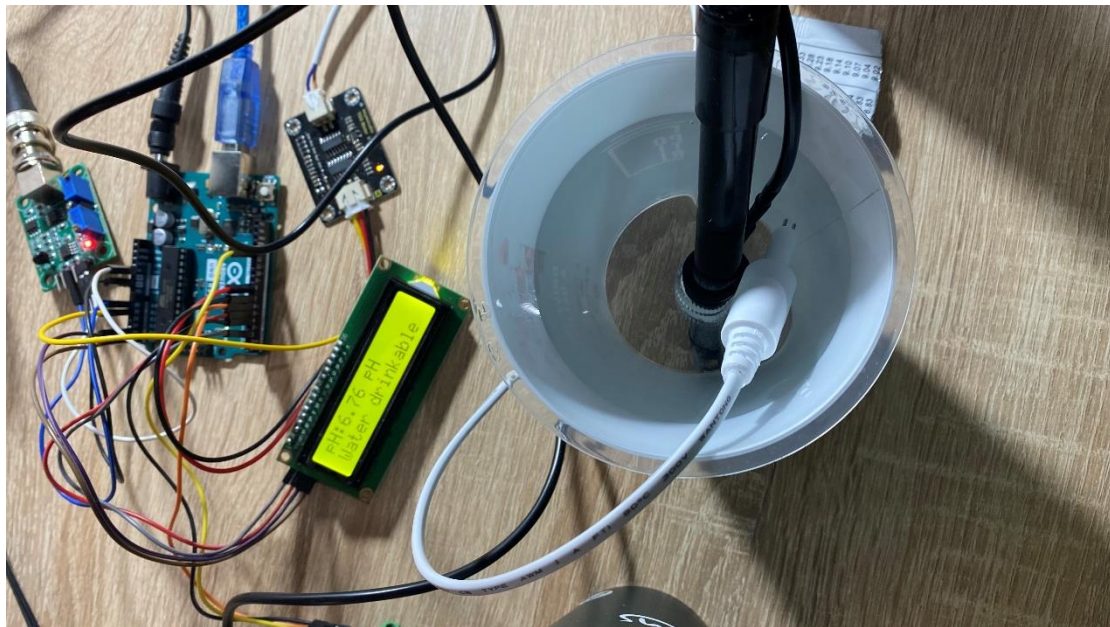
Εικόνα 29. Μέτρηση ποιότητας νερού διαλύματος γνωστού 4pH στους 25C.



Εικόνα 30. Σωστά μας εμφανίζει το αποτέλεσμα ότι το νερό δεν είναι πόσιμο καθώς η οξύτητα του νερού είναι εκτός ορίων.



Εικόνα 31. Μέτρηση ποιότητας νερού έχοντας τοποθετήσει μέσα διάλυμα οξύτητας 6,86pH σε θερμοκρασία 25C.



Εικόνα 32. Το αποτέλεσμα που παίρνουμε είναι σωστό καθώς οι μετρήσεις που έχουμε είναι μέσα στα επιτρεπτά όρια.

6.ΣΥΜΠΕΡΑΣΜΑΤΑ-ΒΑΣΙΚΑ ΠΟΡΙΣΜΑΤΑ

6.1 ΣΥΜΠΕΡΑΣΜΑΤΑ

Μετά από πολλές ώρες πάνω απο έναν υπολογιστή και αρκετού διαβάσματος έχω καταλάβει ότι το Arduino έχει πάρα πολλές εφαρμογές και τρομερές δυνατότητες.Οι αυτοματισμοί όμως είναι το μέλλον οπότε ήταν πολύ ενδιαφέρον να ασχοληθώ με την συγκεκριμένη πτυχιακή. Επίσης, ένα άλλο συμπέρασμα στο οποίο κατέληξα είναι ότι για να έχει κάποιος επαγγελματικό αποτέλεσμα στις μετρήσεις του και μεγάλη ακρίβεια, θα πρέπει να έχει στην κατοχή του ακριβούς αισθητήρες για να έχει τις δικές του βιβλιοθήκες ο κατασκευαστής και να έχει μεγάλη ακρίβεια στις μετρήσεις που θα πάρει.

6.2 ΒΕΛΤΙΩΣΕΙΣ

Θα μπορούσα να κάνω την απεικόνιση ακόμα πιο ωραία,δηλαδή όλα τα υλικά να ήταν πάνω σε μια motherboard αλλά σαν πρώτη φορά είμαι αρκετά ευχαριστημένος.Επίσης θα μπορούσα να ελέγγω κάποιες λειτουργίες απομακρυσμένα απο το κινητό.Επίσης σαν τελευταίο στάδιο με την βοήθεια κάποιων εξτρά εργαλείων και αρκετά παραπάνω κώδικα να μπορεί ο χρήστης να δίνει παραπάνω πληροφορίες για το υγρό που θα παίρνει μέτρηση και να έχει εξίσου αποτελέσματα. Αυτό προϋποθέτει να έχω αγοράσει ακριβούς αισθητήρες , ακόμα και μεγαλύτερη τιμή από ολόκληρο το υπάρχον κύκλωμα ο κάθε αισθητήρας, πράγμα το οποίο θα ήταν ασύμφορο σαν κόστος για την εργασία μου.

ΠΗΓΕΣ-ΒΙΒΛΙΟΓΡΑΦΙΑ

[1] Introduction to Arduino: A piece of cake, Alan G. Smith , CreateSpace Independent Publishing Platform

- [2] Getting Started With Arduino , Massimo Banzi-Michael Shiloh , Maker Media
- [3] <https://www.arduino.cc/reference/en/>
- [4] <https://robu.in/the-ph-meter-by-using-arduino-detail-guide/>
- [5] <https://www.electronicclinic.com/ph-meter-arduino-ph-meter-calibration-diymore-ph-sensor-arduino-code/>
- [6]https://projecthub.arduino.cc/aman_a_shastri/iot-based-water-quality-sensor-28be4a
- [7] https://www.electronicclinic.com/ph-meter-arduino-ph-meter-calibration-diymore-ph-sensor-arduino-code/#pH_Sensor_Calibration
- [8]<https://how2electronics.com/tds-sensor-arduino-interfacing-water-quality-monitoring/>
- [9]https://wiki.dfrobot.com/Gravity__Analog_pH_Sensor_Meter_Kit_V2_SKU_SEN0161-V2
- [10] <https://randomnerdtutorials.com/arduino-tds-water-quality-sensor/>
- [11] <https://quenchwater.com/blog/total-dissolved-solids/>
- [12]https://el.wikipedia.org/wiki/%CE%9F%CE%BB%CE%B9%CE%BA%CE%AC_%CE%B4%CE%B9%CE%B1%CE%BB%CF%85%CE%BC%CE%AD%CE%BD%CE%B1_%CF%83%CF%84%CE%B5%CF%81%CE%B5%CE%AC
- [13]<https://el.wikipedia.org/wiki/%CE%A5%CE%B4%CF%81%CE%B1%CE%B3%CF%89%CE%B3%CE%B5%CE%AF%CE%BF>
- [14] <https://www.eydap.gr/SocialResponsibility/HistoryRecord/HistoricalTrackBack/>

[15] <https://el.wikipedia.org/wiki/Arduino>

[16] <https://www.circuitschools.com/tds-meter-with-tds-sensor-and-arduino-for-water-quality-monitoring/>

[17] <https://el.wikipedia.org/wiki/%CE%98%CE%B5%CF%81%CE%BC%CE%BF%CE%BA%CF%81%CE%B1%CF%83%CE%AF%CE%B1>

[18] <https://el.wikipedia.org/wiki/PH>

[19] <https://el.wikipedia.org/wiki/Arduino#>

[20] <https://docs.arduino.cc/hardware/uno-rev3/#tech-specs>