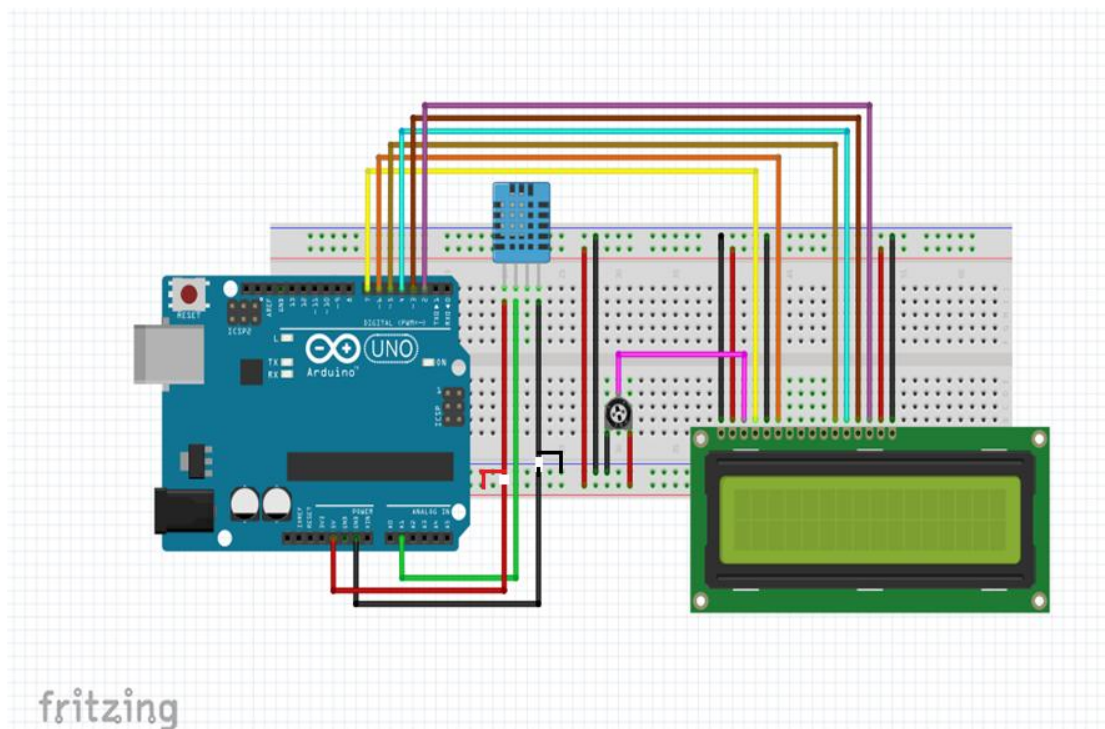




ΤΕΧΝΟΛΟΓΙΚΟ ΕΚΠΑΙΔΕΥΤΙΚΟ ΙΔΡΥΜΑ ΗΠΕΙΡΟΥ

ΣΧΟΛΗ ΜΗΧΑΝΙΚΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ

ΜΕΤΡΗΤΗΣ ΘΕΡΜΟΚΡΑΣΙΑΣ-ΥΓΡΑΣΙΑΣ ΜΕ ΤΟ ΜΙΚΡΟΕΛΕΓΚΤΗ ARDUINO



«ΜΕΤΡΗΤΗΣ ΘΕΡΜΟΚΡΑΣΙΑΣ-ΥΓΡΑΣΙΑΣ ΜΕ ΤΟ ΜΙΚΡΟΕΛΕΓΚΤΗ ARDUINO»

ΣΠΟΥΔΑΣΤΗΣ: Παναγιώτης Μανιφάβας

ΕΠΟΠΤΗΣ: Νικόλαος Αντωνιάδης

ΕΞΑΜΗΝΟ: 11ο **EMAIL:** manis13panos@gmail.com

ΔΗΛΩΣΗ ΠΝΕΥΜΑΤΙΚΗΣ ΙΔΙΟΚΤΗΣΙΑΣ

Η παρούσα εργασία αποτελεί προϊόν αποκλειστικά δικής μου προσπάθειας. Όλες οι πηγές που χρησιμοποιήθηκαν περιλαμβάνονται στη βιβλιογραφία και γίνεται ρητή αναφορά σε αυτές μέσα στο κείμενο όπου έχουν χρησιμοποιηθεί.

ΕΥΧΑΡΙΣΤΙΕΣ

Για την διεκπαιρέωση της παρούσας Πτυχιακής Εργασίας , θα ήθελα να ευχαριστήσω τον επιβλέποντα καθηγητή μου κ. Νικόλαο Αντωνιάδη για τις συμβουλές του και για τον πολύτιμο χρόνο που διέθεσε για την ολοκλήρωση της.

Τέλος θα ήθελα να ευχαριστήσω την οικογένεια μου και τους συγγενείς μου που με στήριξαν καθόλη τη διάρκεια των φοιτητικών μου χρόνων και ιδιαίτερα τους , Δημήτριο Ιωάννου και Χαρήκλεια Μανιφάβα , στους οποίους αφιερώνω και την εργασία αυτή.

ΠΕΡΙΛΗΨΗ

Το αντικείμενο της παρούσας εργασίας είναι η παρουσίαση του μικροελεγκτή **Arduino** και πιο συγκεκριμένα της έκδοσης **Uno**, ο προγραμματισμός θα γίνεται με τη γλώσσα προγραμματισμού **C** και τη **Wiring** μέσω του **ArduinoIDE** για τη δημιουργία ενός μετρητή θερμοκρασίας και υγρασίας του περιβάλλοντος και την εμφάνισή τους σε μια οθόνη υγρών κρυστάλλων. Επίσης χρησιμοποιήθηκε και ένα επιπλέον λογισμικό λεγόμενο και ως **Fritzing** μέσω του οποίου μπόρεσα να κάνω την αναπαράσταση των υλικών που χρησιμοποίησα για την εργασία αλλά και των αποτελεσμάτων της. Θα γίνει ανάλυση των υλικών και των προγραμμάτων που χρησιμοποιήθηκαν στην εργασία καθώς και παράθεση εικόνων για την καλύτερη κατανόηση τους.

ABSTRACT

The object of this work is the presentation of the microcontroller **Arduino** and more specifically the **Uno** version, programming is made with the **C** programming language and the **Wiring** via the **Arduino IDE** to create a temperature gauge and humidity of the environment and their appearance in a liquid crystal display. I also used as an additional a software called **Fritzing**, which with that I could make the representation of the materials used for the work and its results. I will be analyzing the materials and the programs I used to make this project and I will post pictures for the better understanding of what I did.

ΠΕΡΙΕΧΟΜΕΝΑ

ΠΕΡΙΛΗΨΗ

ABSTRACT

ΠΕΡΙΕΧΟΜΕΝΑ

ΚΕΦΑΛΑΙΟ 1: ARDUINO.....	8
1.1 Η πλακέτα Arduino.....	9
1.2 Μοντέλα Arduino.....	11
1.3 Arduino Shields.....	12
1.4 Arduino Uno.....	14
1.4.1 Ανάλυση της πλακέτας.....	15
1.4.2 Τεχνικά Χαρακτηριστικά.....	18
1.4.3 Flash Memory.....	19
1.4.4 SRAM memory.....	19
1.4.5 EEPROM memory.....	19
1.4.6 FTDI.....	20
1.5 Πλεονεκτήματα Arduino.....	20
ΚΕΦΑΛΑΙΟ 2: ΛΟΓΙΣΜΙΚΟ ARDUINO.....	21
2.1 Arduino IDE.....	22
2.2 Wiring.....	27
2.3 Fritzing.....	27
ΚΕΦΑΛΑΙΟ 3: ΒΙΒΛΙΟΘΗΚΕΣ.....	28
3.1 Βιβλιοθήκη DHT11 Sensor.....	29
3.2 Κώδικας Βιβλιοθήκης DHT11.....	32
3.3 Εντολές Βιβλιοθήκης LCD.....	33
ΚΕΦΑΛΑΙΟ 4: ΜΕΤΡΗΣΗ ΘΕΡΜΟΚΡΑΣΙΑΣ-ΥΓΡΑΣΙΑΣ.....	35
4.1 Τα υλικά της κατασκευής.....	36
4.2 Κώδικας Μετρητή.....	41
ΒΙΒΛΙΟΓΡΑΦΙΑ.....	43

ΚΕΓΑΛΑΙΟ 1^ο

ARDUINO

ΕΙΣΑΓΩΓΗ ΣΤΟΝ ΜΙΚΡΟΕΛΕΓΚΤΗ ARDUINO

Σε αυτό το κεφάλαιο θα μιλήσουμε για τον μικροελεγκτή Arduino και πιο συγκεκριμένα για τις εκδόσεις που υπάρχουν, τα έξτρα υλικά που μπορούν να χρησιμοποιηθούν από τον μικροεπεξεργαστή λεγόμενα και ως Shields και θα εμβαθύνουμε κυρίως στην έκδοση που θα χρησιμοποιήσω για την κατασκευή η οποία είναι η Uno αλλά και τα χαρακτηριστικά της.

1.1 Η πλακέτα arduino

Ο Arduino είναι ένα μικρό κύκλωμα που περιέχει ένα ολοκληρωμένο σύστημα υπολογιστή χρησιμοποιώντας ένα μικρό chip, ολοκληρωμένο κύκλωμα, που είναι ο μικροελεγκτής. Είναι πολύ φθηνότερος και πολύ χρήσιμος για την κατασκευή ηλεκτρονικών συσκευών. Η ρομποτική επιστήμη είναι μία από τις πολλές κατηγορίες στις οποίες το Arduino Uno διαπρέπει. Η ομάδα του Arduino έχει τοποθετήσει σε αυτήν την πλακέτα όλα τα απαραίτητα στοιχεία που απαιτούνται για τον μικροελεγκτή ώστε να μπορεί να λειτουργεί σωστά και να μπορεί να επικοινωνεί με τον υπολογιστή και άλλες σειριακές συσκευές. Οι πλατφόρμες Arduino κατασκευάζονται κυρίως από την εταιρία Smart Project στην Ιταλία. Το Arduino ξεκίνησε ως έργο προς ανάπτυξη το 2005 στην Ιταλία, στο Ινστιτούτο Αλληλεπίδρασης Σχεδίασης Inrea ώστε οι φοιτητές του Ινστιτούτου να αναπτύσσουν ενσωματωμένα συστήματα οικονομικά και αποδοτικά αξιοποιώντας τις δυνατότητες και τις ευκαιρίες που μπορεί να προσφέρει το ελεύθερο λογισμικό.

Επίσης, το Arduino θα λέγαμε ότι είναι ένα εργαλείο που μπορούμε να κατασκευάσουμε ένα υπολογιστικό σύστημα με την έννοια ότι αυτό θα ελέγχει συσκευές του φυσικού κόσμου, σε αντίθεση με τον κοινό H/Y. Βασίζεται σε ευέλικτο, εύκολο στη χρήση υλικό και λογισμικό, σε μια αναπτυξιακή πλακέτα που ενσωματώνει επάνω έναν μικροελεγκτή και συνδέεται με τον H/Y για να προγραμματιστεί μέσα από ένα απλό περιβάλλον ανάπτυξης (IDE). Με το Arduino δημιουργούνται συσκευές οι οποίες εξυπηρετούν διάφορους σκοπούς έχοντας την δυνατότητα να δέχονται ερεθίσματα από το περιβάλλον τους (μέσω των αισθητήρων (sensor)) και να αντιδρούν ανάλογα με το πως έχουν προγραμματιστεί κάτι που θα δούμε παρακάτω καθώς η εργασία βασίζεται στις μετρήσεις που παίρνει ο αισθητήρας.

Το Arduino βασίζεται σε τεχνολογίες ανοιχτού κώδικα (open source platform). Μπορεί να κατασκευαστεί από τον καθένα, μπορεί να ενσωματωθεί σε συσκευές ακόμα και για εμπορικούς σκοπούς και το σημαντικότερο είναι ότι υπάρχει μια

ολόκληρη κοινότητα που χρησιμοποιεί το Arduino σε κατασκευές άρα υπάρχει μεγάλος όγκος ελεύθερης πληροφορίας(forums). Γενικά, τα Projects στον εν λόγω Μικροελεγκτή μπορούν να είναι αυτόνομα σε επίπεδο hardware ή να επικοινωνούν με κάποιο software στον Η/Υ του προγραμματιστή. Έπειτα το Arduino χρησιμοποιεί ένα ειδικά προγραμματιζόμενο Atmega328 αντί του chip FTDI ώστε αυτό να επιτρέπει τόσο την πιο γρήγορη ταχύτητα μεταφοράς όσο και τη γρήγορη σειριακή επικοινωνία με τον Η/Υ.

Ο μικροεπεξεργαστής ενός Arduino συνήθως προγραμματίζεται εκ των προτέρων ώστε να παρέχει κάποιο φορτωτή εκκίνησης (BootLoader), ο οποίος είναι γραμμένος από τον κατασκευαστή. Ο φορτωτής εκκίνησης υπάρχει ώστε να απλοποιεί την διαδικασία της αποθήκευσης των προγραμμάτων στην Flash Memory του Arduino μέσω σειριακής USB θύρας.

Επιπλέον, η γλώσσα προγραμματισμού, οι διάφορες βιβλιοθήκες και το ολοκληρωμένο περιβάλλον ανάπτυξης που υπάρχουν για τον προγραμματισμό της πλατφόρμας Arduino αποτελούν ανοιχτό λογισμικό προσφέροντας έτσι ανεκτίμητη γνώση σε όλους καθώς και ευκολία στη χρήση.

1.2 Μοντέλα Arduino

Ο μικροελεγκτής Arduino έχει αρκετές εκδόσεις οι οποίες έχουν βγεί καθεμιά τους και για διαφορετική χρήση. Οι πιο βασικές εκδόσεις που υπάρχουν είναι :

- Arduino Uno
- Arduino Leonardo
- Arduino Mega 2560
- ArduinoLilyPad
- Arduino Mega ADK
- ArduinoFio
- Arduino Ethernet
- Arduino Pro
- Arduino BT
- Arduino Nano
- USB/Serial light adapter
- Arduino Mini
- Mini USB/Serial adapter
- Arduino Pro Mini

				
Arduino Uno	Arduino Leonardo	Arduino Mega 2560	Arduino LilyPad	Arduino Mega ADK
				
Arduino Fio	Arduino Ethernet	Arduino Pro	Arduino BT	Arduino Nano
				
USB/Serial Light Adapter	Arduino Mini	Mini USB/Serial Adapter	Arduino Pro Mini	

Εικόνα 1: Εκδόσεις Arduino

1.3 Arduino Shields

Arduino Shields είναι κάποιες επεκτάσεις από την αρχική πλακέτα όπου μας δίνουν επιπλέον δυνατότητες στον μικροεπεξεργαστή μας. Κάποιες από τις εκδόσεις του Arduino έχουν προεγκατεστημένες κάποιες από αυτές τις επεκτάσεις κάτι που τα κάνει ακόμα πιο εύχρηστα. Οι επεκτάσεις αυτές είναι εύκολο να τοποθετηθούν και είναι σχετικά φθηνές στην παραγωγή τους. Υπάρχουν πάρα πολλές επεκτάσεις, εγώ θα αναφέρω τις πιο βασικές, οι οποίες είναι οι εξής :

- **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 μέσω ενός καλωδίου RJ-45.

- **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 και 300 πόδια σε εξωτερικούς χώρους. Η μονάδα αυτή περιλαμβάνει επίσης μια θύρα υποδοχής SD.

- **Wireless Proto Shield**

Το Wireless Proto 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 και 300 πόδια σε εξωτερικούς χώρους.

- **Arduino Motor Shield**

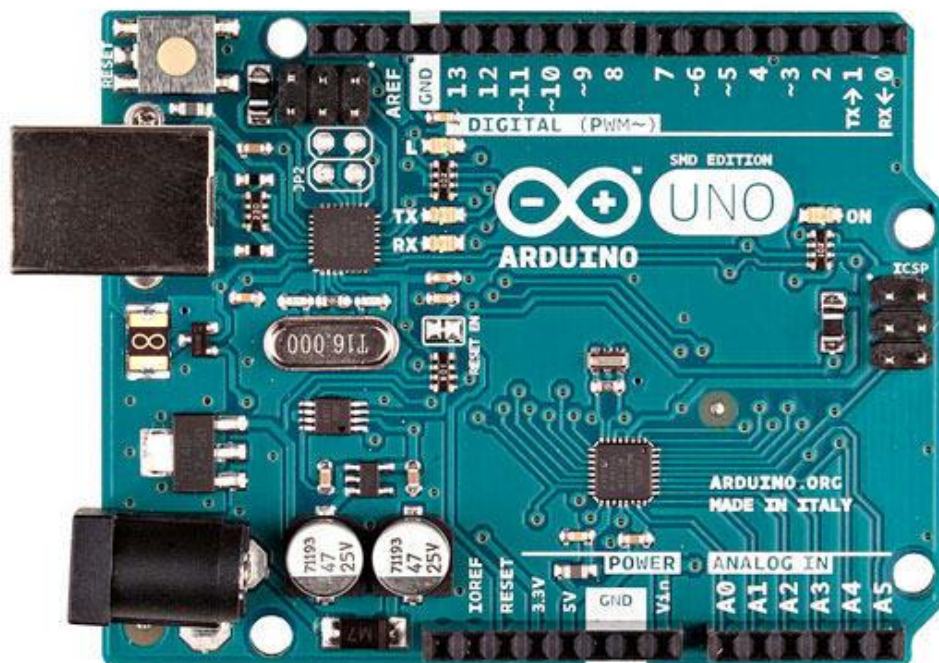
Το Arduino Motor Shield επιτρέπει την διαχείριση δυο DC κινητήρων από την ίδια συσκευή, ελέγχοντας την ταχύτητα και την κατεύθυνση του καθενός ξεχωριστά.

Arduino WiFi Shield	
Arduino Ethernet Shield	
Wireless SD Shield	
Wireless Proto Shield	
Arduino Motor Shield	

Εικόνα 2: Arduino Shields

1.4 Arduino Uno

Το Arduino Uno είναι μια πλακέτα μικροελεγκτή βασισμένη στο chip Atmega328p. Έχει 14 ψηφιακές εισόδους (6 από αυτές μπορούν να χρησιμοποιηθούν ως έξοδοι PWM), 6 αναλογικές εισόδους, ένα 16 MHz χαλαζία, μια σύνδεση USB, μια υποδοχή ρεύματος, μια κεφαλίδα ICSP και ένα κουμπί επαναφοράς. "Uno" σημαίνει ένα στα ιταλικά και επιλέχτηκε για να σηματοδοτήσει την έναρξη του Arduino λογισμικού (IDE) 1.0. Πλέον υπάρχουν νεότερες εκδόσεις του λογισμικού.



Εικόνα 3: ArduinoUno

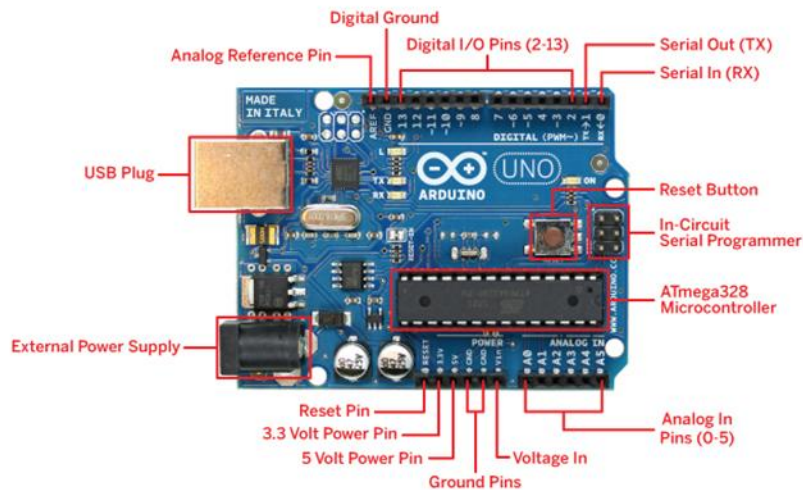
1.4.1 Ανάλυση της πλακέτας

Η πλακέτα arduino διαθέτει :

- 14 ψηφιακές I/O θύρες (εισόδου & εξόδου). Κυρίως με το πρόγραμμα που θα φορτωθεί στον μικροελεγκτή αυτές οι θύρες μπορούν να εργαστούν σαν είσοδοι ή εξοδοι I/O ψηφιακών σημάτων.
- Οι ψηφιακές θύρες 3, 5, 6, 9, 10, 11 μπορούν να λειτουργήσουν και ως ψευδοαναλογικές θύρες εξόδου με το σύστημα PWM (Pulse Width Modulation), δηλαδή το ίδιο σύστημα που διαθέτουν οι μητρικές των υπολογιστών για να ελέγχουν τις ταχύτητες των ανεμιστήρων και γενικότερα παλμικά σήματα. Το PWM παίρνει ένα εύρος τιμών από το 0 έως το 255. Δεν είναι πραγματικά αναλογικό σύστημα, έτσι θέτοντας στην έξοδο την τιμή 127, δεν σημαίνει ότι η έξοδος θα παρέχει 2.5V αντί της κανονικής τιμής των 5V, αλλά ότι θα δίνει έναν παλμό που η τάση του θα εναλλάσσεται με μεγάλη συχνότητα και για ίσα χρονικά διαστήματα μεταξύ των τιμών 0V και 5V με σκοπό η μέση τιμή να ισούται με 2,5V.
- Οι θύρες 0 και 1 χρησιμοποιούνται επίσης και για να λαμβάνουν (RX) και να μεταδίδουν (TX) TTL σειριακά δεδομένα. Έτσι, όταν για παράδειγμα το πρόγραμμα στέλνει δεδομένα σειριακά, τότε αυτά προωθούνται στην θύρα USB μέσω του ελεγκτή Serial-Over-USB (FTDI) όπως επίσης και στο pin 0 για να τα διαβάσει ενδεχομένως μία άλλη συσκευή (π.χ. ένα δεύτερο arduino στη δικιά του θύρα 1, η επικοινωνία είναι πολύ σημαντική). Αυτό φυσικά σημαίνει ότι αν στο πρόγραμμα ενεργοποιηθεί το σειριακό interface, καταλαμβάνονται δύο ψηφιακές θύρες εισόδου/εξόδου.
- Οι θύρες 2 και 3 λειτουργούν και ως εξωτερικά interrupt (interrupt 0 και 1 αντίστοιχα), δηλαδή σαν διακόπτες αν συμβεί κάτι που έχουμε θέσει εμείς. Με άλλα λόγια, μπορούν να ρυθμιστούν μέσα από το πρόγραμμα ώστε να λειτουργούν αποκλειστικά ως ψηφιακές είσοδοι στις οποίες όταν συμβαίνουν συγκεκριμένες αλλαγές τάσης, η κανονική ροή του προγράμματος να σταματάει άμεσα και να εκτελείται μία συγκεκριμένη συνάρτηση, ή αλλαγή στην ροή του ρεύματος. Τα εξωτερικά interrupt είναι ιδιαίτερα χρήσιμα σε εφαρμογές που απαιτούν συγχρονισμό μεγάλης ακρίβειας.
- 6 αναλογικές θύρες εισόδου αριθμημένες από το 0 έως το 5 μια προς μία. Το καθένα από αυτά λειτουργεί ως αναλογική είσοδος κάνοντας χρήση του ADC (Analog to Digital Converter). Για παράδειγμα, αν τροφοδοτηθεί ένα από αυτά τα pin με μία τάση η οποία μπορεί να κυμανθεί με ένα ποτενσιόμετρο ακριβείας από 0V ως μία τάση αναφοράς Vref (η οποία αν δεν γίνει κάποια αλλαγή είναι προρυθμισμένη στα 5V ή την τάση που θέλουμε να έχουμε εμείς), τότε μέσα από το πρόγραμμα μπορεί να «διαβαστεί» η τιμή της θύρας ως ένας ακέραιος αριθμός χωρητικότητας 10-bit, από

το 0 (όταν η τάση στο pin είναι 0V) μέχρι το 1023 (όταν η τάση στο pin είναι 5V). Η τάση αναφοράς V_{ref} μπορεί να ρυθμιστεί με μία εντολή όπως για παράδειγμα στα 1.1V. Ένας άλλος τρόπος όπου η τάση αναφοράς μπορεί να δηλωθεί από τον προγραμματιστή είναι τροφοδοτώντας με μία εξωτερική τάση αναφοράς τη θύρα με την σήμανση AREF που βρίσκεται στην απέναντι πλευρά της πλακέτας. Έτσι, αν τροφοδοτηθεί η θύρα AREF με 3.3V και στην συνέχεια εκτελεσθεί η εντολή να διαβαστεί κάποιο pin αναλογικής εισόδου στο οποίο εφαρμόζετε τάση 1.65V, το Arduino θα επιστρέψει την τιμή 512.

- Δίπλα από τις θύρες αναλογικής εισόδου, υπάρχει μία ακόμα συστοιχία από 6 pin με την σήμανση POWER. Η λειτουργία του καθενός pin έχει ως εξής:
- Το πρώτο, με την ένδειξη RESET, όταν γειωθεί (με οποιοδήποτε από τα 3 pin με την ένδειξη GND που υπάρχουν στο arduino) έχει ως αποτέλεσμα την επανεκκίνηση του arduino, υπάρχει βέβαια και κουμπί που κάνει αυτή την λειτουργία.
- Το δεύτερο με την ένδειξη 3.3V, μπορεί να τροφοδοτήσει διατάξεις, συσκευές ή αισθητήρες με τάση 3.3V και είναι σημαντική τάση. 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 για αυτό θέλει προσοχή αν αυτό που θέλουμε να τροφοδοτήσουμε θέλει περισσότερο πρέπει να γίνει με ξεχωριστή τάση.
- Η τρίτη θύρα με την ένδειξη 5V, μπορεί να χρησιμοποιηθεί και αυτή για την τροφοδότηση διαφόρων εξαρτημάτων, συσκευών ή αισθητήρων με τάση 5V(π.χ. άλλο Arduino). Ανάλογα με τον τρόπο τροφοδοσίας του ίδιου του Arduino, η τάση αυτή προέρχεται είτε άμεσα από την θύρα USB (που ούτως ή άλλως παρέχει τάση 5V), είτε από την εξωτερική τροφοδοσία αφού αυτή περάσει από ένα ρυθμιστή τάσης για να την «σταθεροποιήσει» στα 5V, η εξωτερική τροφοδοσία κυμένεται από 7v μέχρι 12v.
- Το τέταρτο και το πέμπτο pin με την ένδειξη GND είναι οι γειώσεις.
- Το έκτο και τελευταίο pin, με την ένδειξη Vin(τάση εισόδου) έχει διπλό ρόλο. Σε συνδυασμό με το pin γείωσης δίπλα του, μπορεί να λειτουργήσει ως μέθοδος εξωτερικής τροφοδοσίας του Arduino στην περίπτωση που δεν βολεύει να χρησιμοποιηθεί η υποδοχή του φισ των 2.1mm(JACK). Αν όμως υπάρχει ήδη συνδεδεμένη εξωτερική τροφοδοσία μέσω του φισ(jack), τότε μπορεί να χρησιμοποιηθεί αυτό το pin για να τροφοδοτήσει εξαρτήματα και συσκευές με την πλήρη τάση της εξωτερικής τροφοδοσίας (7~12V), πριν αυτή περάσει από τον ρυθμιστή τάσης όπως γίνεται με το pin των 5V.



Εικόνα 4: Ανάλυση της πλακέτας

- Η φόρτωση του sketch πραγματοποιείται μέσω μίας USB θύρας που διαθέτει η πλακέτα arduinouno. Έτσι οι πληροφορίες που προέρχονται από την USB θύρα του υπολογιστή εισέρχονται στην USB θύρα του arduinouno και στην συνέχεια οδηγούνται στο FDTI ολοκληρωμένο για να διαμορφωθούν σε μία κατάλληλη μορφή ώστε ο μικροελεγκτής να μπορέσει να τις διαβάσει επιτυχώς.

- Πάνω στην πλακέτα του arduinouno υπάρχει ένας μικροδιακόπτης (micro-switch) και 4 smd(επιφανειακής στήριξης) LED. Η λειτουργία του διακόπτη (που έχει την σήμανση RESET) και του ενός LED με την σήμανση POWER είναι προφανής στην πλακέτα. Τα δύο LED με τις σημάνσεις TX και RX, χρησιμοποιούνται ως ένδειξη λειτουργίας του σειριακού interface, καθώς ανάβουν όταν το arduino στέλνει ή λαμβάνει (αντίστοιχα) δεδομένα μέσω της USB θύρας. Τα LED αυτά ελέγχονται από τον ελεγκτή Serial-over-USB και συνεπώς δεν λειτουργούν όταν η σειριακή επικοινωνία γίνεται αποκλειστικά μέσω των ψηφιακών pin 0 και 1.

- Επίσης, υπάρχει το LED με τη σήμανση L. Η βασική λειτουργία του LED στην πλακέτα Arduino είναι για να αναβοσβήνει συνήθως για δοκιμαστικό σκοπό. Οι κατασκευαστές σκέφτηκαν να ενσωματώσουν ένα LED στην πλακέτα το οποίο το σύνδεσαν στη ψηφιακή θύρα 13 μέσω μιας αντίστασης. Έτσι ακόμα και αν δεν έχει συνδεθεί τίποτα πάνω στο φυσικό pin 13, αναθέτοντας του την τιμή HIGH μέσα από το πρόγραμμα, θα ανάψει το ενσωματωμένο LEDL επιτυχώς.

1.4.2 Τεχνικά Χαρακτηρηστικά

Microcontroller	ATmega328P
Operating Voltage	5V
Input Voltage (recommended)	7-12V
Input Voltage (limit)	6-20V
Digital I/O Pins	14 (of which 6 provide PWM output)
PWM Digital I/O Pins	6
Analog Input Pins	6
DC Current per I/O Pin	20 mA
DC Current for 3.3V Pin	50 mA
Flash Memory	32 KB (ATmega328P) of which 0.5 KB used by bootloader
SRAM	2 KB (ATmega328P)
EEPROM	1 KB (ATmega328P)
Clock Speed	16 MHz
LED_BUILTIN	13
Length	68.6 mm
Width	53.4 mm
Weight	25 g

Εικόνα 5: Τεχνικά χαρακτηριστικά ArduinoUno

Στην παραπάνω εικόνα παρατείνονται τα τεχνικά χαρακτηριστικά του ArduinoUnoόπου κάποια συγκεκριμένα θα τα αναλύσουμε παρακάτω.

1.4.3 FlashMemory

Η μνήμη flash έχει χωρητικότητα 32Kb, από τα οποία τα 2Kb χρησιμοποιούνται από το firmware του arduino που έχει εγκαταστήσει ήδη ο κατασκευαστής του στο χώρο μνήμης. Το firmware αυτό που στην ορολογία του arduino ονομάζεται bootloader είναι αναγκαίο για την εγκατάσταση των προγραμμάτων στον μικροελεγκτή μέσω της θύρας USB από το FTDI chip, χωρίς δηλαδή να χρειάζεται εξωτερικός hardware programmer. Τα υπόλοιπα 30Kb της μνήμης Flash χρησιμοποιούνται για την αποθήκευση αυτών ακριβώς των προγραμμάτων που θα επιχειρήσετε να αναπτύξετε από μόνοι σας αν ασχοληθείτε, αφού πρώτα μεταγλωττιστούν στον υπολογιστή. Η μνήμη Flash δεν χάνει τα περιεχόμενά της με την απώλεια της τροφοδοσίας ή κάνοντας reset το μικροελεγκτή αλλιώς θα είχαμε πρόβλημα στο να ξαναπρογραμματίζουμαι ξανά. Επίσης, ενώ η μνήμη Flash υπό κανονικές συνθήκες δεν προορίζεται για χρήση runtime, μέσα από τα προγράμματα λόγω της μικρής συνολικής μνήμης που είναι διαθέσιμη σε αυτά (2Kb SRAM + 1Kb EEPROM), έχει σχεδιαστεί μία βιβλιοθήκη από τις πολλές, που επιτρέπει την χρήση runtime στον χώρο που περισσεύει από την αποθήκευση των sketch (30Kb πλην το μέγεθος του προγράμματος σε μεταγλωττισμένη μορφή).

1.4.4 SRAMMemory

Η μνήμη SRAM (static random access memory) είναι η ωφέλιμη μνήμη που μπορούν να χρησιμοποιήσουν τα προγράμματα για να αποθηκεύουν μεταβλητές, πίνακες, ορίσματα, τιμές κ.λπ. κατά το runtime. Όπως και σε έναν υπολογιστή, αυτή η μνήμη χάνει τα δεδομένα της όταν η παροχή ρεύματος στο arduino σταματήσει ή αν γίνει reset, για αυτό λέγεται και στατική. Στο ATmega328 η SRAM μνήμη καταλαμβάνει χώρο 2048 bytes κατά την διάρκεια μίας κανονικής λειτουργίας και όλες οι μεταβλητές φορτώνονται σε αυτή καθ' όλη την διάρκεια της λειτουργίας του μικροελεγκτή.

1.4.5 EEPROMMemory

Το τελευταίο μέρος της μνήμης είναι η EEPROM και καταλαμβάνει 1024 bytes, αρκετά μικρή για μνήμη που χρησιμοποιείται μόνο για ανάγνωση (read-only). Η

EEPROM έχει όριο ζωής καθώς δε μπορεί να επαναπρογραμματιστεί για περισσότερες από 100.000 φορές. Είναι μία byte addressable μνήμη, γεγονός που καθιστά λίγο δυσκολότερο να τεθεί σε χρήση αφού απαιτείται ειδική βιβλιοθήκη ώστε να μπορέσει κάποιος να έχει πρόσβαση σε αυτή.

1.4.6 FTDI

Εκτός όμως από το AT Mega328 το arduino χρησιμοποιεί και ένα FDTI ολοκληρωμένο. Οι μικροελεγκτές AT Mega της AVR προγραμματίζονται χρησιμοποιώντας σειριακή επικοινωνία με τους υπολογιστές, έτσι το FDTI αναλαμβάνει την εργασία της μετατροπής της σειριακής θύρας σε USB για την καλύτερη συνδεσιμότητα.

1.5 Πλεονεκτήματα Arduino

- Οικονομική: Η πλατφόρμα Arduino αποτελεί οικονομική λύση διότι είναι φθηνότερη από άλλες αντάξιες της. Επιπλέον, είναι αρχιτεκτονικά ανοιχτή και μπορεί ο οποιοσδήποτε να την αναπτύξει από μόνος του, δηλαδή να την δημιουργήσει από την αρχή.
- Μεταφέρσιμη: Σε σχέση με τις υπάρχουσες πλατφόρμες στο εμπόριο η πλατφόρμα Arduino παρέχει πλήρη μεταφερσιμότητα με αποτέλεσμα να μπορεί να προγραμματιστεί στα περισσότερα λειτουργικά συστήματα και είναι και μικρή σε χώρο που την καθιστά εύκολη στη μεταφορά.
- Επεκτάσιμη: Το υλικό και το λογισμικό της πλατφόρμας Arduino είναι ανοιχτά και ελεύθερα για όλους, υλικό υπάρχει σε όλο το διαδίκτυο. Καθημερινά, χιλιάδες υποστηρικτές του ελεύθερου λογισμικού αναπτύσσουν διάφορες βιβλιοθήκες για την υποστήριξη της πλατφόρμας. Παράλληλα, τόσο η αρχιτεκτονική όσο και το υλικό της πλατφόρμας εξελίσσονται συνεχώς με τα χρόνια.

Σε αυτό το κεφάλαιο έγινε παραπομπή : [A],[C],[D],[1],[4],[7],[8]

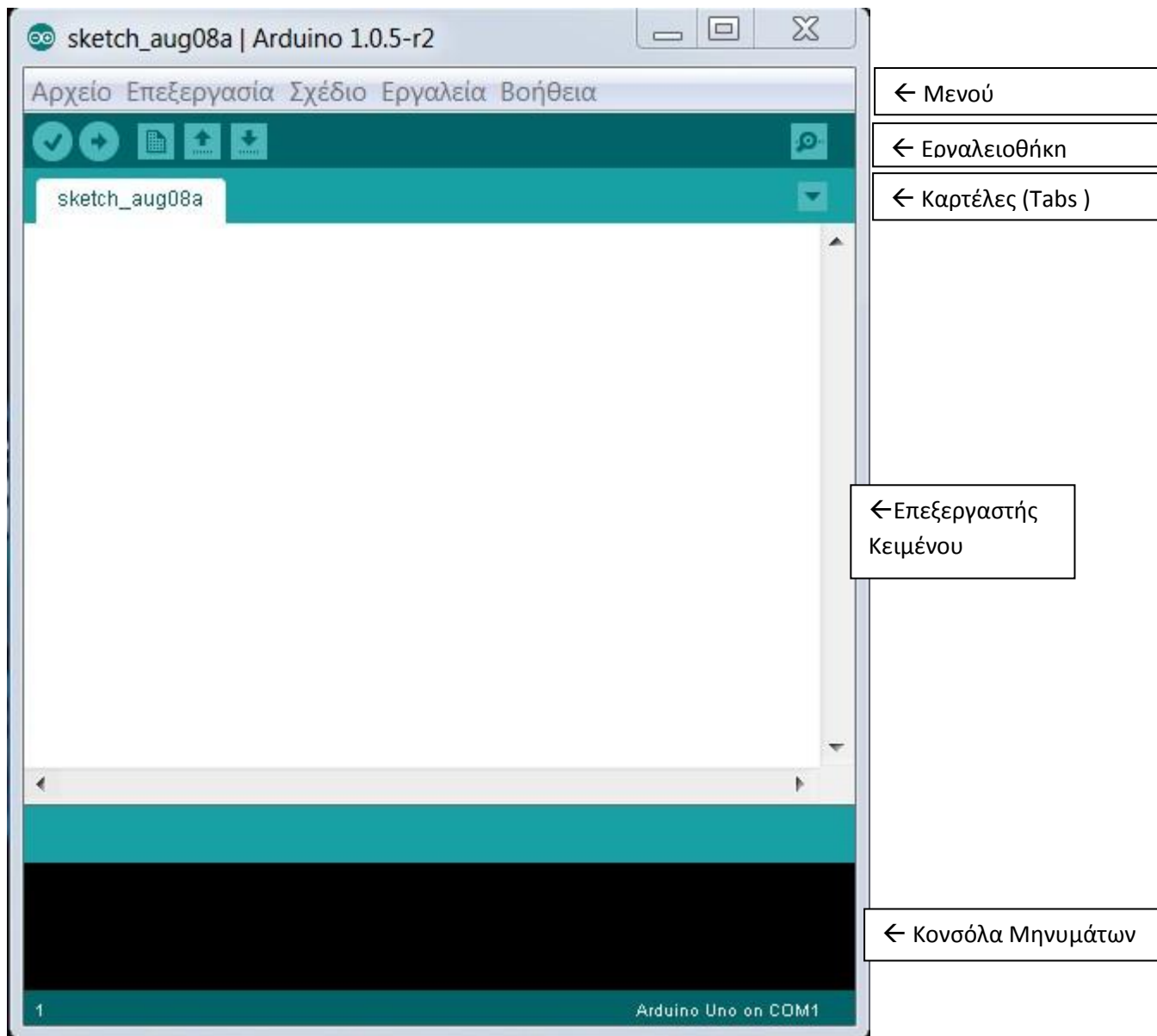
ΚΕΦΑΛΑΙΟ 2^ο

ΛΟΓΙΣΜΙΚΟ ARDUINO

2.1 ArduinoIDE

Το ολοκληρωμένο περιβάλλον ανάπτυξης (IDE) του Arduino είναι μία εφαρμογή γραμμένη σε Java, που λειτουργεί σε πολλές πλατφόρμες και προέρχεται από το IDE για τη γλώσσα προγραμματισμού Processing και το σχέδιο Wiring. Έχει σχεδιαστεί για να εισαγάγει τον προγραμματισμό σε όσους δεν είναι εξοικειωμένοι με την ανάπτυξη λογισμικού. Περιλαμβάνει ένα πρόγραμμα επεξεργασίας κώδικα με χαρακτηριστικά όπως είναι η επισήμανση σύνταξης και ο συνδυασμός αγκύλων και είναι επίσης σε θέση να μεταγλωττίζει και να φορτώνει προγράμματα στην πλακέτα με ένα μόνο κλικ. Τα Arduino προγράμματα είναι γραμμένα σε C ή C++. Το ArduinoIDE έρχεται με μια βιβλιοθήκη λογισμικού που ονομάζεται "Wiring". Το περιβάλλον ανάπτυξης Arduino περιέχει μια περιοχή επεξεργασίας κειμένου για τη συγγραφή κώδικα, μια περιοχή μηνυμάτων, ένα μενού, μια γραμμή εργαλείων με κουμπιά για κοινές λειτουργίες, καθώς και μια σειρά από μενού. Συνδέεται με το υλικό Arduino για τη φόρτωση προγραμμάτων και για να επικοινωνούν μεταξύ τους.

Ένα ολοκληρωμένο πρόγραμμα συνήθως ονομάζεται sketch. Αυτό το sketch είναι γραμμένο με το πρόγραμμα επεξεργασίας κειμένου. Έχει δυνατότητες για την αντιγραφή/επικόλληση και για την αναζήτηση/αντικατάσταση κειμένου. Η κονσόλα απεικονίζει την έξοδο του κειμένου από το περιβάλλον Arduino συμπεριλαμβάνοντας πλήρη μηνύματα λάθους και άλλες πληροφορίες. Τα κουμπιά της γραμμής εργαλείων επιτρέπουν τον έλεγχο και το ανέβασμα των προγραμμάτων, τη δημιουργία νέου sketch, το άνοιγμα και την αποθήκευση των sketch και άνοιγμα της σειριακής οθόνης.



Εικόνα 5: ArduinoIDE

Η εργαλειοθήκη του ArduinoIDE:



Verify/Compile (Ελεγχος/Μεταγλώττιση): Έλεγχος για λάθη στον κώδικα



Upload: Ανέβασμα του κώδικα στον μικροελεγκτή



New(Νέο): Δημιουργεί ένα νέο sketch



Open(Άνοιγμα): Παρουσιάζει ένα μενού με όλα τα sketch, κάνοντας κλικ σε ένα από αυτά θα ανοίξει μέσα στο τρέχον παράθυρο



Save(Αποθήκευση): Αποθηκεύει το sketch



Serial Monitor(Σειριακή οθόνη): Ανοίγει την σειριακή οθόνη ώστε να μπορούμε να δώσουμε δεδομένα από το πληκτρολόγιο

- Ένα τυπικό πρόγραμμα Arduino έχει την παρακάτω δομή:

```
//δήλωση μεταβλητών
```

```
void setup ()
```

```
{
```

```
//αρχικοποιήσεις
```

```
}
```

```
void loop ()
```

```
{
```

```
//Κώδικας
```

```
}
```

Υπάρχουν δυο ειδικές συναρτήσεις που είναι μέρος του κάθε sketch του Arduino οι οποίες είναι η setup() και η loop(). Η setup() καλείται μια φορά, όταν το sketch ξεκινά ή όποτε κάνει επαναφορά (reset) η πλατφόρμα Arduino. Κυρίως, σε αυτήν γίνονται οι αρχικοποιήσεις των μεταβλητών, η ρύθμιση της κατάστασης των ακίδων (pins) και η προετοιμασία των βιβλιοθηκών. Αντιθέτως, η συνάρτηση loop() καλείται ξανά και ξανά επιτρέποντας έτσι στο πρόγραμμα να ανταποκριθεί σε εξωτερικά ερεθίσματα. Και οι δυο συναρτήσεις πρέπει να περιλαμβάνονται στο sketch, ακόμα και αν δεν περιέχουν κάτι και να είναι κενές.

Κάποιες Συναρτήσεις:

Συναρτήσεις μετατροπής τύπων

- char(), byte()
- int(), word(), long()
- float(), double()

Συναρτήσεις εισόδου και εξόδου

- `pinMode()` (ορίζει μια επαφή ως είσοδο ή έξοδο)

Συναρτήσεις ψηφιακής εισόδου και εξόδου

- `digitalWrite()` (γράφει σε μία ψηφιακή επαφή έξοδο)
- `digitalRead()` (διαβάζει από μία ψηφιακή επαφή είσοδο)

Συναρτήσεις αναλογικής εισόδου και εξόδου

- `analogReference()` (ορίζει την τάση αναλογικής αναφοράς)
- `analogWrite()` (γράφει PWM σήματα σε μία επαφή έξοδο)
- `analogRead()` (διαβάζει από μία αναλογική επαφή είσοδο)

Προηγμένες συναρτήσεις εισόδου και εξόδου

- `tone()` (παράγει ένα τετραγωνικό σήμα ορισμένης συχνότητας)
- `noTone()` (διακόπτει την παραγωγή τετραγωνικών σημάτων)
- `shiftOut()` (ολισθαίνει τα ψηφία μιας τιμής σε μία επαφή έξοδο)
- `pulseIn()` (επιστρέφει την διάρκεια σε μs ενός παλμού HIGH ή LOW)

Συναρτήσεις χρόνου

- `millis()` (διάρκεια εκτέλεσης του προγράμματος σε ms)
- `micros()` (διάρκεια εκτέλεσης του προγράμματος σε μs)
- `delay()` (παύση προγράμματος - η διάρκεια δίδεται σε ms)
- `delayMicroseconds()` (παύση προγράμματος - η διάρκεια δίδεται σε μs)

Μαθηματικές και Τριγωνομετρικές συναρτήσεις

- `max()` (βρίσκει τον μεγαλύτερο ανάμεσα σε δύο αριθμούς)
- `min()` (βρίσκει τον μικρότερο ανάμεσα σε δύο αριθμούς)
- `abs()` (επιστρέφει την απόλυτη τιμή ενός αριθμού)
- `constrain()` (ελέγχει για υπερχείλιση ή υποχείλιση ορίων)
- `map()` (πραγματοποιεί γραμμικό μετασχηματισμό ορίων)
- `pow()` (επιστρέφει το αποτέλεσμα μίας δύναμης)
- `sqrt()` (επιστρέφει την ρίζα ενός αριθμού)
- `sin()` (υπολογίζει το ημίτονο ενός αριθμού)
- `cos()` (υπολογίζει το συνημίτονο ενός αριθμού)
- `tan()` (υπολογίζει την εφαπτομένη ενός αριθμού)

Συναρτήσεις γεννήτριας ψευδοτυχαίων αριθμών

- `random()` (δίδεται ένας νέος αριθμός από την γεννήτρια)
- `randomSeed()` (θέτει τον σπόρο της γεννήτριας παραγωγής)

Συναρτήσεις επεξεργασίας δυαδικών αριθμών

- `lowByte()` (επιστρέφει το δεξιότερο byte μίας μεταβλητής)
- `highByte()` (επιστρέφει το αριστερότερο byte μίας μεταβλητής)
- `bitRead()` (διαβάζει ένα συγκεκριμένο ψηφίο μίας μεταβλητής)
- `bitWrite()` (γράφει σε ένα συγκεκριμένο ψηφίο μιας μεταβλητής)
- `bitSet()` (γράφει την τιμή 1 σε κάποιο ψηφίο μίας μεταβλητής)
- `bitClear()` (γράφει την τιμή 0 σε κάποιο ψηφίο μιας μεταβλητής)

- bit() (υπολογίζει μία συγκεκριμένη δύναμη με βάση το 2)

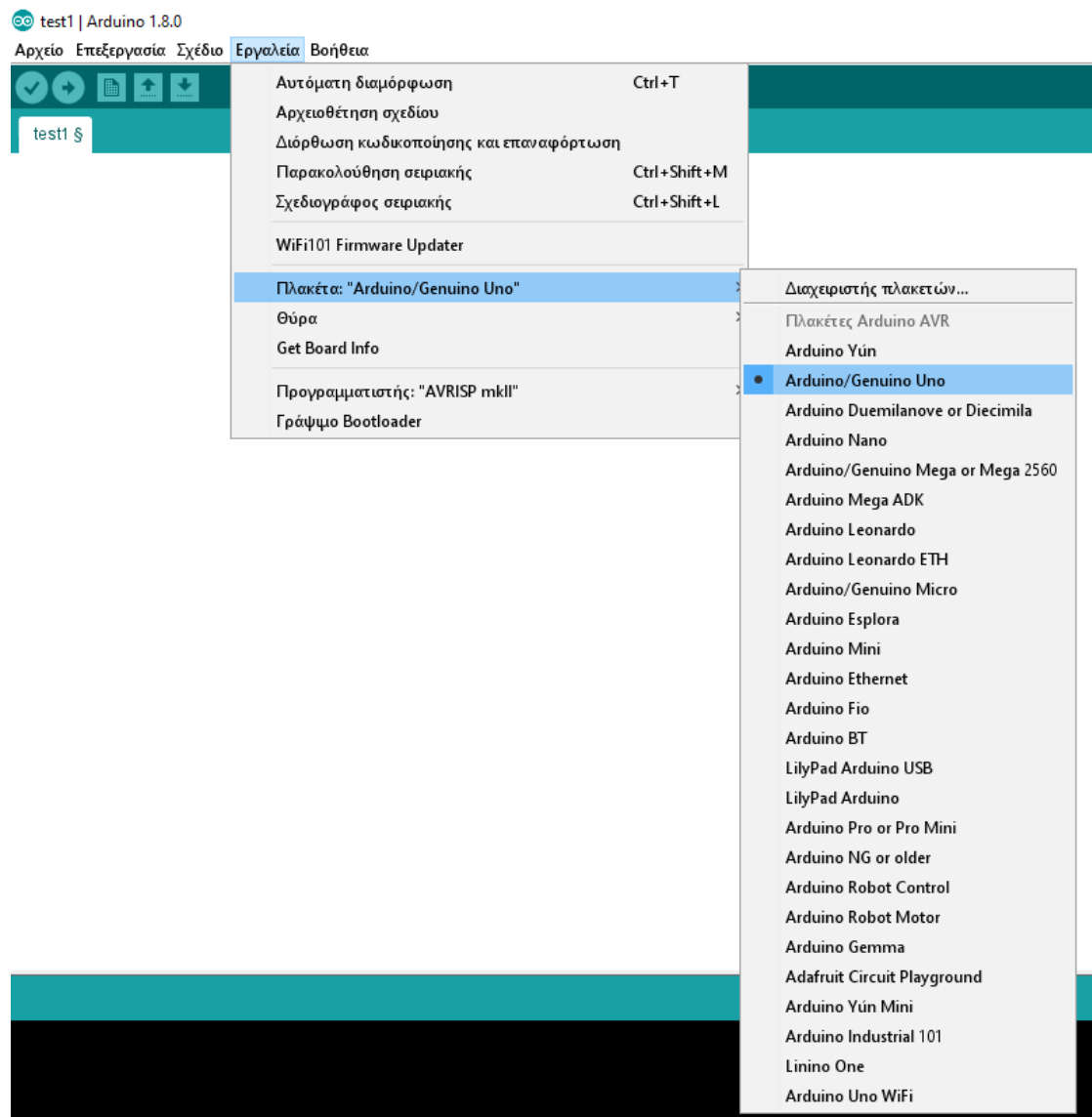
Συναρτήσεις χρήσης ρουτινών εξυπηρέτησης διακοπών

- attachInterrupt() (ενεργοποιεί μία ρουτίνα εξυπηρέτησης διακοπής)
- detachInterrupt() (απενεργοποιεί μία ρουτίνα εξυπηρέτησης διακοπής)

Συναρτήσεις ενεργοποίησης και απενεργοποίησης διακοπών

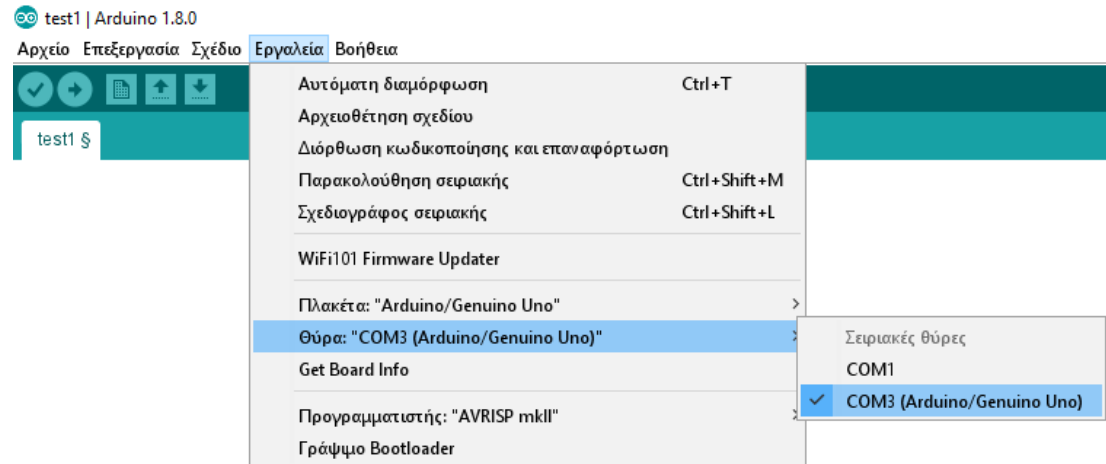
- interrupts() (ενεργοποιεί τα σήματα διακοπής)
- noInterrupts() (απενεργοποιεί τα σήματα διακοπής)

Επίσης πολύ σημαντικό είναι η επιλογή της πλακέτας που θα χρησιμοποιήσουμε να την δηλώσουμε στο πρόγραμμα όπως φαίνεται στην παρακάτω φωτογραφία



Εικόνα 6: Επιλογή πλακέτας

Ακόμα και η επιλογή της θύρας COM (σε εμένα συγκεκριμένα είναι η COM3) στην οποία επικοινωνεί ο Arduino με τον Η/Υ:



Εικόνα 7: επιλογή θύρας COM

2.2 Wiring

Wiring είναι μια ανοιχτού κώδικα πρωτότυπη πλατφόρμα ηλεκτρονικών η οποία απαρτίζεται από μια γλώσσα προγραμματισμού, ένα IDE περιβάλλον και ένα μικροεπεξεργαστή. Η εφαρμογή WiringIDE είναι γραμμένη σε Java η οποία προέρχεται από το IDE για τη γλώσσα προγραμματισμού Processing.

2.3 Fritzing

Η εφαρμογή Fritzing είναι ανοιχτού κώδικα και δημιουργήθηκε βάση της γλώσσας Processing και του μικροεπεξεργαστή Arduino για να βοηθήσει ερασιτέχνες στη δημιουργία ηλεκτρονικών κυκλωμάτων μέσω των οποίων να δημιουργήσουμε PCB , πλακέτες με αγωγίμες ράγες.

Σε αυτό το κεφάλαιο έγινε παραπομπή : [2],[9],[10],[11]

ΚΕΦΑΛΑΙΟ 3^ο

ΒΙΒΛΙΟΘΗΚΕΣ

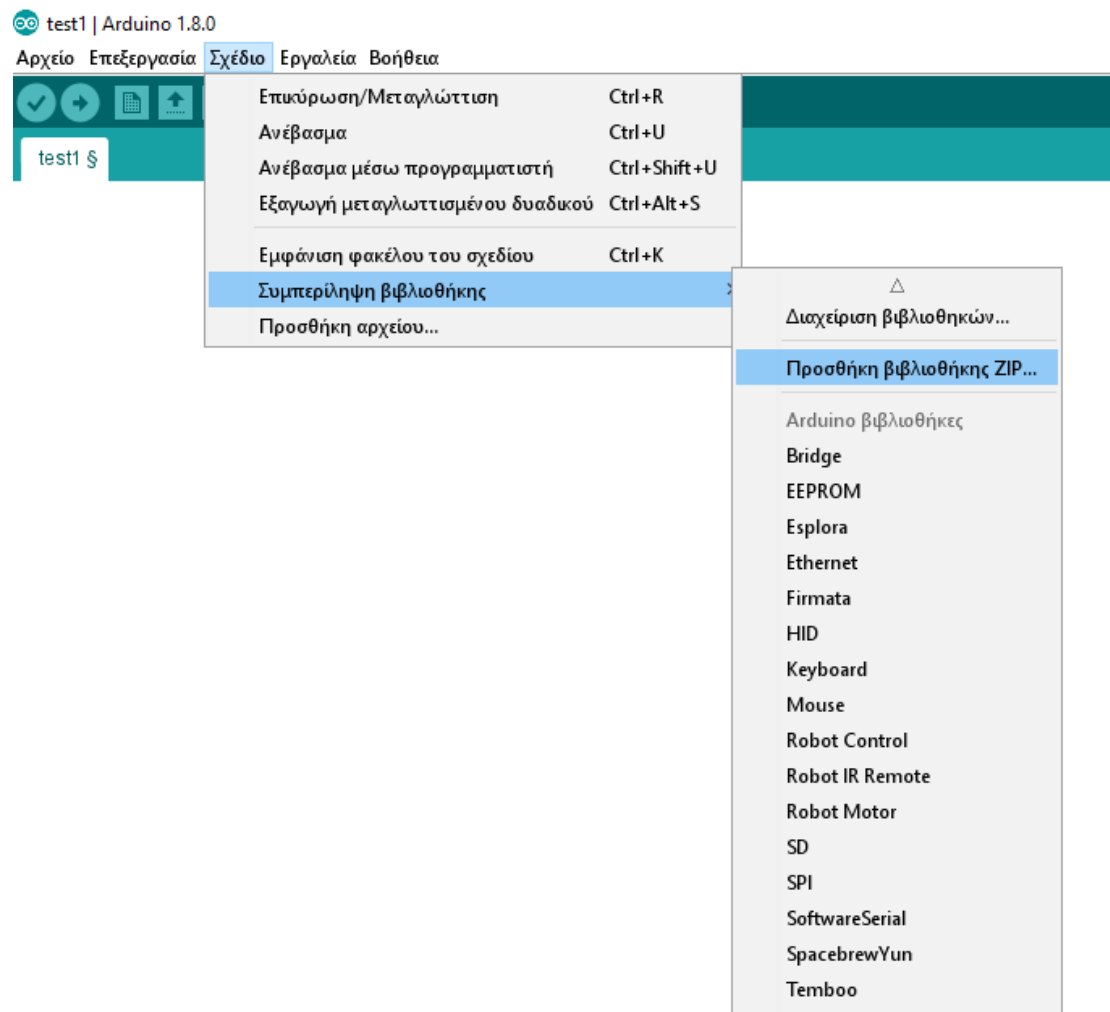
3.1 Βιβλιοθήκη DHT11 sensor

Το περιβάλλον Arduino μπορεί να επεκταθεί με τη χρήση βιβλιοθηκών, ακριβώς όπως οι περισσότερες πλατφόρμες προγραμματισμού. Οι κυριότερες βιβλιοθήκες είναι οι εξής:

- EEPROM
- Ethernet/ Ethernet 2
- Firmata
- GSM
- LiquidCrystal
- SD
- Servo
- SPI
- SoftwareSerial
- Stepper
- TFT
- WiFi
- Wire

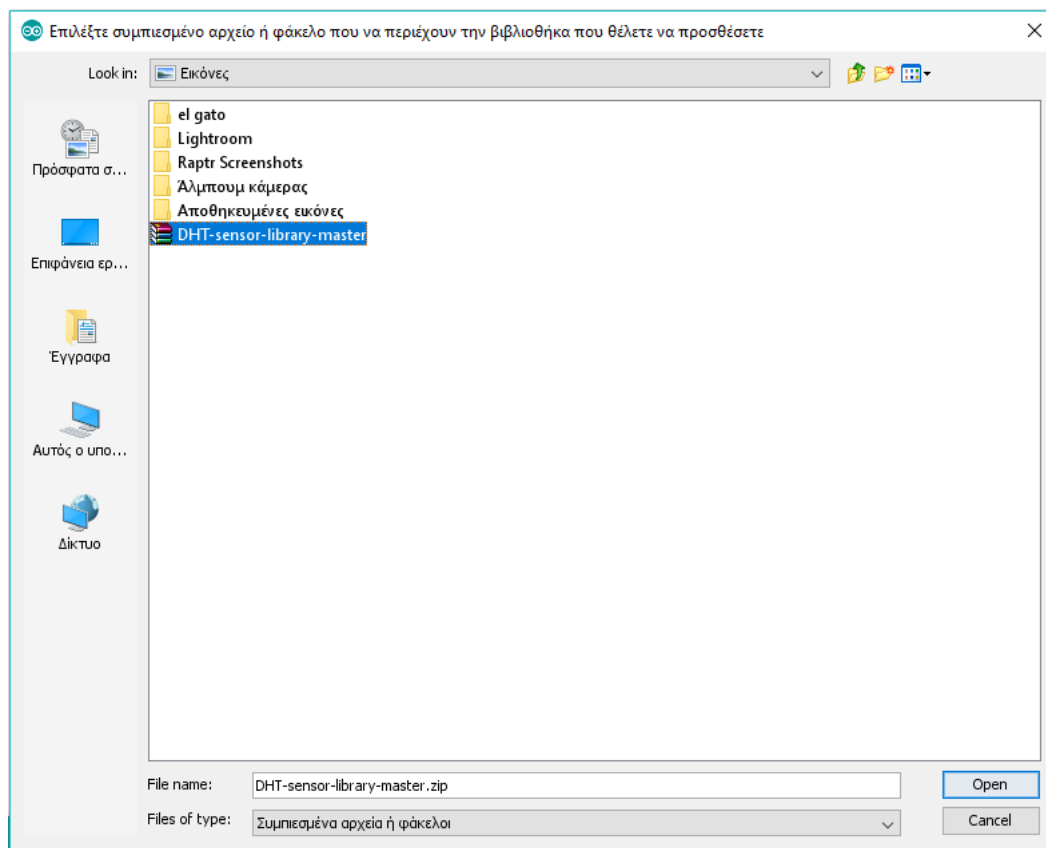
Οι δυνατότητες των βιβλιοθηκών είναι απεριόριστες καθώς ο καθένας μπορεί να φτιάξει δικές του βιβλιοθήκες για το Arduino. Λόγω του ότι θα χρησιμοποιήσουμε τον DHT11 sensor του οποίου δεν περιλαμβάνει την βιβλιοθήκη στις προεγκατεστημένες βιβλιοθήκες θα πρέπει να τον εγκαταστήσουμε μόνοι μας. Αρκεί να βρούμε την βιβλιοθήκη από το ίντερνετ σε μορφή .zip και έπειτα από το περιβάλλον του Arduino μπορούμε να την συμπεριλάβουμε με τις υπόλοιπες με τον εξής τρόπο.

- Αρχικά ανοίγουμε το Σχέδιο το οποίο βρίσκεται επάνω στην μπάρα, έπειτα Συμπερίληψη Βιβλιοθήκης και μετά Προσθήκη Βιβλιοθήκης ZIP όπως στην εικόνα παρακάτω.



Εικόνα 9: Εισαγωγή Βιβλιοθήκης (1/2)

- Προχωράμε στο επόμενο βήμα, βρίσκουμε την βιβλιοθήκη σε μορφή .zip που κατεβάσαμε και την επιλέγουμε και πατάμε Open.



Εικόνα 10: Εισαγωγή Βιβλιοθήκης (2/2)

3.2 Κωδικας Βιβλιοθηκης DHT11

```
#include <dht.h>

dht DHT;

#define DHT11_PIN 4

void loop()

{

// READ DATA

Serial.print("DHT11, \t");

chk = DHT.read11(DHT11_PIN);

switch (chk)

{

case DHTLIB_OK:

        Serial.print("OK,\t");

        break;

case DHTLIB_ERROR_CHECKSUM:

        Serial.print("Checksum error,\t");

        break;

case DHTLIB_ERROR_TIMEOUT:

        Serial.print("Time out error,\t");

        break;

default:

        Serial.print("Unknown error,\t");

        break;

}

// DISPLAY DATA

Serial.print(DHT.humidity,1);
```



```
Serial.print("\t");  
  
Serial.println(DHT.temperature,1);  
  
delay(1000);  
  
}
```

Το κομμάτι του κώδικα της βιβλιοθήκης είναι για την ανάγνωση δεδομένων από την θύρα που είναι ο SensorDHT11 αλλά και η εμφάνιση των δεδομένων αυτών στην σειριακή του Arduino.

3.3 Εντολές Βιβλιοθήκης LCD

Διαστάσεις οθόνης

```
lcd.begin(cols, rows) // στήλες και σειρές που έχει η LCD οθόνη μας
```

Θέση του Cursor

```
lcd.setCursor(0, 0); // πάνωαριστερά  
lcd.setCursor(15, 0); // πάνωδεξιά  
lcd.setCursor(0, 1); // κάτωαριστερά  
lcd.setCursor(15, 1); // κάτωδεξιά
```

Εμφάνιση Μηνύματος στην οθόνη

```
lcd.print(data)  
  
lcd.print(data, BASE)
```

Παράμετροι:

lcd: μεταβλητή του τύπου LiquidCrystal

data: δεδομέναγιαεμφάνιση (char, byte, int, long, or string)

BASE (προαιρετικό): η βάση στην οποία εμφανίζονται οι αριθμοί: BIN για δυαδικό (βάση 2), DEC για δεκαδικό (βάση 10), OCT για οκταδικό (βάση 8), HEX για δεκαεξαδικό (βάση 16).

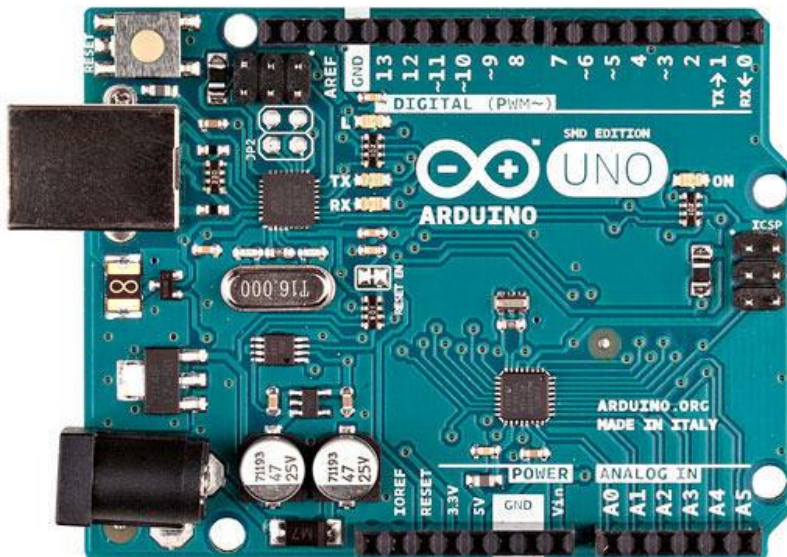
Σε αυτό το κεφάλαιο έγινε παραπομπή : [3],[6]

ΚΕΦΑΛΑΙΟ 4^ο
ΜΕΤΡΗΤΗΣ ΘΕΡΜΟΚΡΑΣΙΑΣ-ΥΓΡΑΣΙΑΣ

4.1 Τα υλικά της κατασκευής

- Arduino Uno
- LCD οθόνη (διάσταση 16x2)
- Sensor DHT11
- Ποντεσιόμετρο 10k ohm
- 1 διάτρητη πλακέτα

Arduino Uno



Εικόνα 11: Arduino Uno

Επέλεξα το Arduino Uno ως τον μικροελεγκτή για την δημιουργία της κατασκευής καθώς είναι από τους πιο πολύ χρησιμοποιημένους. Τα pins έχουν προαναφερθεί και πιο πάνω στην εργασία.

LCD οθόνη

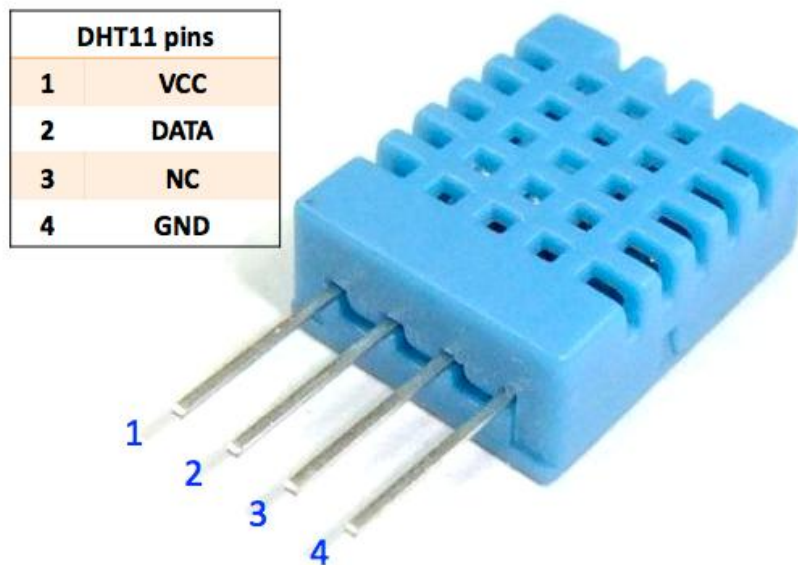


Εικόνα 12: LCD οθόνη

Επέλεξα να χρησιμοποιήσω μια οθόνη 16x2 καθώς είναι υπεραρκετή για την κατασκευή μας. Στην παραπάνω εικόνα βλέπουμε την αρίθμηση των pins της LCD. Χρησιμοποίησα τα εξής pins: 1,2,3,4,5,6,11,12,13,14,15,16 .

- Τα pin1,5,16 είναι η γείωση
- Το pin2,15 είναι η τάση (5V)
- Το pin3 είναι για τη ρύθμιση της αντίθεσης (μέσω του ποτεσιόμετρου)
- Το pin4,6,11,12,13,14 είναι για τη σύνδεση στις ψηφιακές θύρες του Arduino

Sensor DHT11



Εικόνα 13: Αισθητήρας DHT11

Ο αισθητήρας DHT11 είναι ικανός να παίρνει μετρήσεις θερμοκρασίας αλλά και υγρασίας. Παίρνει μετρήσεις θερμοκρασίας από 0-50C και υγρασίας από 20-80%.

- Το pin 1 είναι η τάση (5v)
- Το pin 2 είναι τα δεδομένα (εκεί συνδέουμε για τις μετρήσεις)
- Το pin 3 είναι null (κενό)
- Το pin 4 είναι η γείωση (GND)

Ποντεσιόμετρο 10k ohm

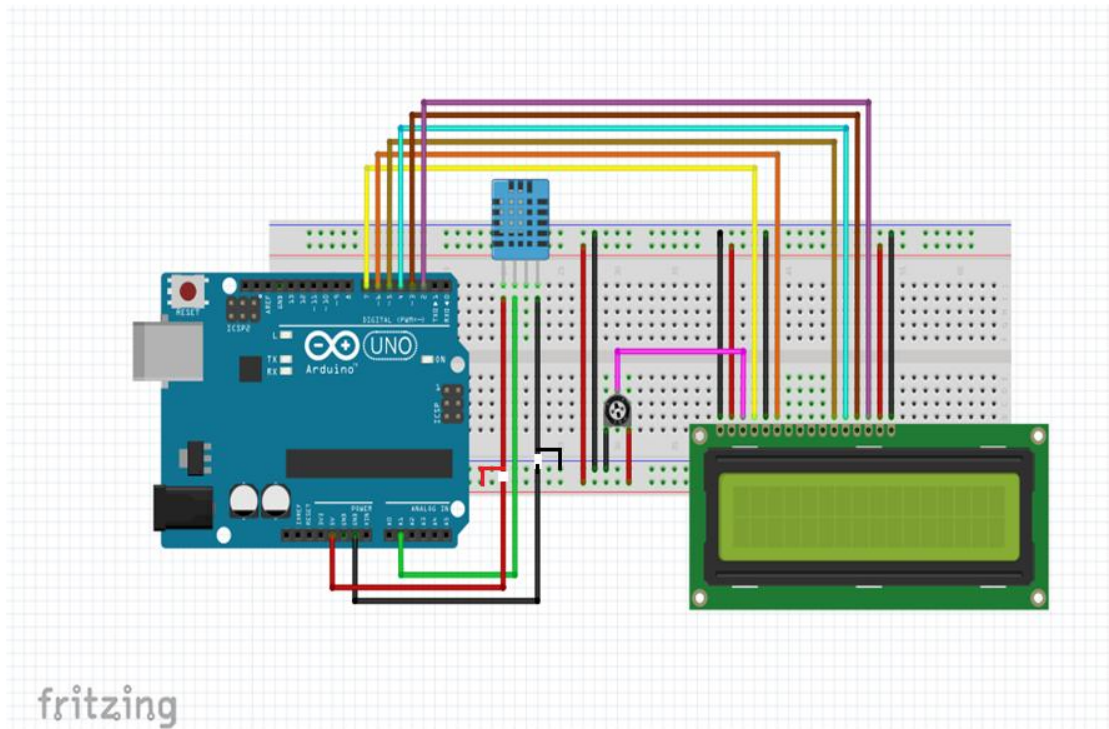


Εικόνα 14: Ποντεσιόμετρο 10k ohm

Το Ποντεσιόμετρο χρησιμοποιήθηκε για την ρύθμιση της αντίθεσης της LCD οθόνης.

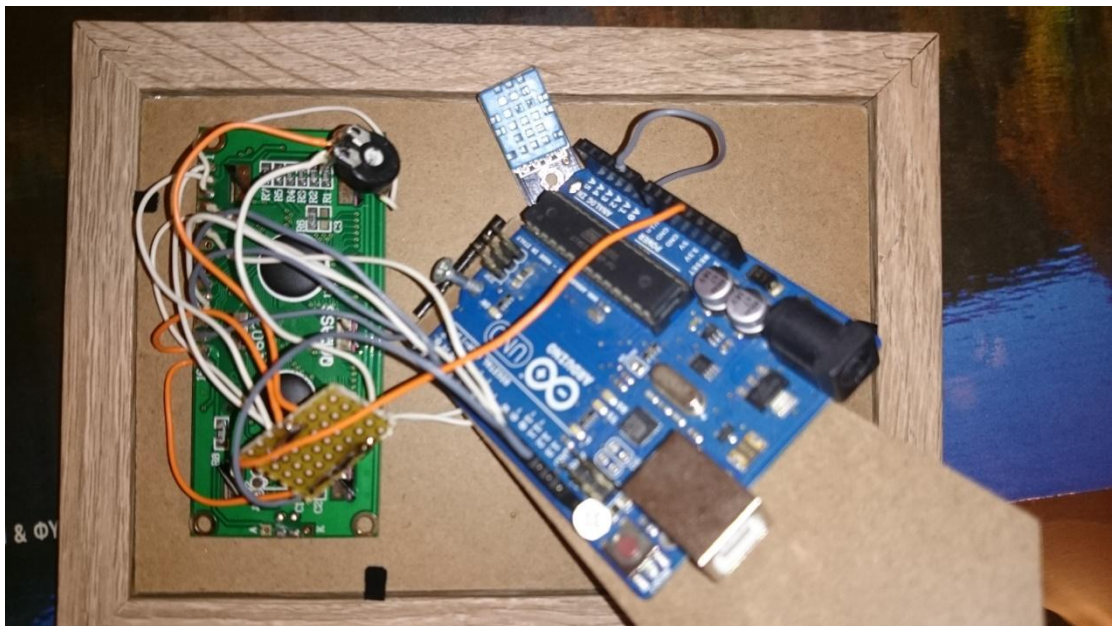
- Το αριστερό pin είναι η τάση (5V)
- Το μεσαίο pin είναι για τη σύνδεση με την LCD οθόνη
- Το δεξί pin είναι η γείωση (GND)

Όλα τα υλικά μας συνδεδεμένα μεταξύ τους μέσω του προγράμματος fritzing :



Εικόνα 15: Όλο το κύκλωμα στο Fritzing

Και στην πραγματικότητα:



Εικόνα 16: Το κύκλωμα στην πραγματικότητα

4.2 Κώδικας Μετρητή

Παρακάτω παρατίθεται ο κώδικας με τον οποίο λειτουργεί ο μετρητής θερμοκρασίας και υγρασίας.

```
#include <LiquidCrystal.h>
```

```
#include <dht.h>
```

```
LiquidCrystal lcd(7, 6, 5, 4, 3, 2); //digital ports used for lcd
```

```
int sensorPin = A0; // we use A0 as our reading port
```

```
dht sensor;
```

```
void setup()
```

```
{
```

```
  lcd.begin(16,2); //16 by 2 character display
```

```
}
```

```
void loop()
```

```
{
```

```
  delay(1000); //wait 1 sec
```

```
  sensor.read11(sensorPin); //read data
```

```
  lcd.clear(); // cleans lcd
```

```
  lcd.setCursor(0,0); //set the cursor on the point (0,0)
```

```
  lcd.print("Humidity = "); //prints on the lcd the message Humidity=
```

```
  lcd.print(sensor.humidity); //prints on the lcd the data for humidity
```

```
  lcd.setCursor(0,1); //set the cursor on the point (0,1 )
```

```
lcd.print("Temp = "); //prints on the lcd the message Temp=  
lcd.print(sensor.temperature); //prints on the lcd the data of temperature  
}
```

Και το αποτέλεσμα :



Εικόνα 17: Μέτρηση Θερμοκρασίας-Υγρασίας

ΒΙΒΛΙΟΓΡΑΦΙΑ

ΒΙΒΛΙΑ ΠΛΗΡΟΦΟΡΗΣΗΣ

- [A] Massimo Banzi (2009), Getting Started With Arduino
- [B] Enrique Ramos Melgar and Ciriaco Castro Diez ,Arduino And Kinect Projects
- [C] Hardware by Jonathan Oser and Hugh Blemings(Dec. 2009)Beginning Arduino
Practical Arduino cool Projects for Ope Source
- [D] Δρ. Βολογιαννίδης Σταύρος (2009), Ευφυής Έλεγχος, Θεωρία και Εφαρμογή

ΙΣΤΟΣΕΛΙΔΕΣ ΠΛΗΡΟΦΟΡΗΣΗΣ

- [1] <http://www.arduino.cc/>
- [2] <http://arduino.cc/en/Main/Software>
- [3] <http://arduino.cc/en/Reference/LiquidCrystal>
- [4] <http://el.wikipedia.org/wiki/Arduino>
- [5] <https://www.adafruit.com/product/386>
- [6] <http://playground.arduino.cc/Main/DHT11Lib>
- [7] <http://users.ionio.gr/~floros/thesis/Thesis%20Triantafylloy.pdf>
- [8] <http://www.teilar.gr/dbData/ProfAnn/profann-968a4b3c.pdf>
- [9] [https://en.wikipedia.org/wiki/Wiring_\(development_platform\)](https://en.wikipedia.org/wiki/Wiring_(development_platform))
- [10] <https://en.wikipedia.org/wiki/Fritzing>
- [11] <https://www.arduino.cc/en/Reference/HomePage>