

Μέτρηση θερμοκρασίας με μικροελεγκτή και αισθητήριο θερμίστορ

Ομάδα 23

Καλβουρτζής Αθανάσιος (7364)

Σπαφαρίδης Ξενοφών (7336)

23 Ιουνίου 2013

# Περιεχόμενα

Εισαγωγή	Σελίδα 3
Θεωρητική ανάλυση	Σελίδα 6
Σχεδίαση	Σελίδα 8
Παρουσίαση και ανάλυση κώδικα	Σελίδα 10
Προβλήματα υλοποίησης	Σελίδα 12
Μετρήσεις	Σελίδα 13
Συμπεράσματα	Σελίδα 15
Βιβλιογραφία	Σελίδα 15

# Εισαγωγή

Στην εργασία αυτή θα κατασκευαστεί ψηφιακό θερμόμετρο με την χρήση αισθητηρίου θερμίστορ, μικροελεγκτή Arduino και μίας οθόνης για την απεικόνιση της θερμοκρασίας.

Τα θερμίστορ είναι αντιστάσεις μεταβαλλόμενες με τη θερμοκρασία, η αντίσταση των NTC θερμίστορ μειώνετε εκθετικά με την αύξηση της θερμοκρασίας.

## Τα υλικά που θα χρειαστούν

- NTC Θερμίστορ 10KΩ
- 10KΩ Αντίσταση 1%
- Arduino Leonardo
- Οθόνη LCD 2x16
- Μπαταρία 9V
- Clip τροφοδοσίας 9V μπαταρίας
- 330Ω Αντίσταση
- Breadboard

## NTC Θερμίστορ 10KΩ



Τα θερμίστορ χρησιμοποιούνται για εφαρμογές μέτρησης θερμοκρασίας, αισθητήρια και έλεγχο, αντιστάθμισης θερμοκρασίας σε ηλεκτρονικές διατάξεις.

## Τεχνικά Χαρακτηριστικά

Τιμή αντίσταση στους 25°C	10KΩ
B25°C	3977
Tolerance on B25	±0.75
Θερμοκρασία λειτουργίας	- 40°C με + 125°C
Χρόνος απόκρισης	1.2sec
Βάρος	≈0.3g

## Οθόνη LCD 2x16



Η οθόνη είναι συμβατή με τον driver Hitachi HD44780, και μπορεί να ελεγχθεί με τον μικροελεγκτή Arduino με την χρήση της βιβλιοθήκης LiquidCrystal.

## Arduino Leonardo



Το Arduino είναι μια υπολογιστική πλατφόρμα βασισμένη σε μια απλή μητρική πλακέτα με ενσωματωμένο μικροελεγκτή και εισόδους/εξόδους, και η οποία μπορεί να προγραμματιστεί με τη γλώσσα Wiring.

## Τεχνικά Χαρακτηριστικά

Microcontroller	ATmega32u4
Operating Voltage	5V
Input Voltage (recommended)	7-12V
Input Voltage (limits)	6-20V
Digital I/O Pins	20
PWM Channels	7
Analog Input Channels	12
DC Current per I/O Pin	40 mA
DC Current for 3.3V Pin	50 mA
Flash Memory	32 KB (ATmega32u4) of which 4 KB used by bootloader
SRAM	2.5 KB (ATmega32u4)
EEPROM	1 KB (ATmega32u4)
Clock Speed	16 MHz

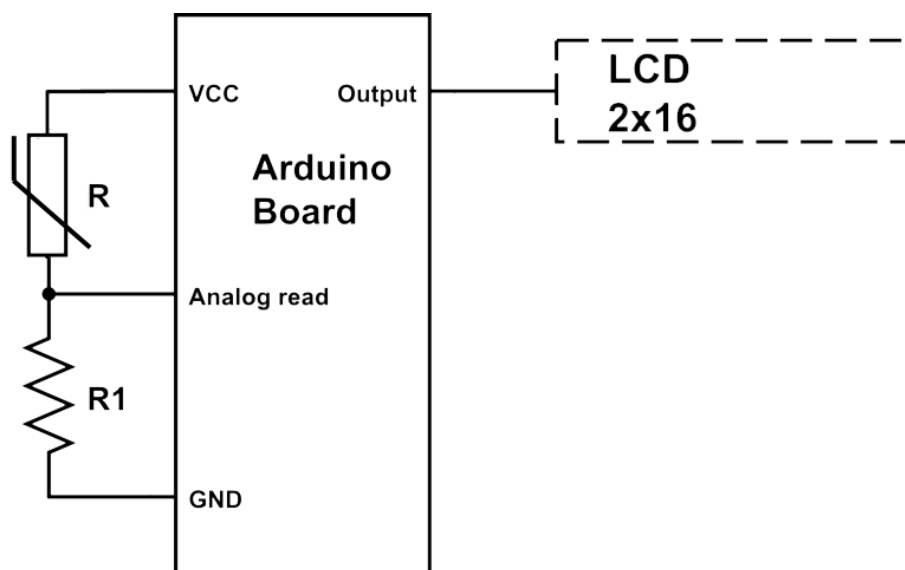
Για την παροχή τροφοδοσίας μπορεί να χρησιμοποιηθεί η θύρα USB ή πηγή τάσης με τιμή μέσα στα όρια των προδιαγραφών.

# Θεωρητική ανάλυση

## Μέτρηση αντίστασης

Ο μικροελεγκτής δεν μπορεί να μετρήσει αντίσταση για αυτόν τον λόγο θα γίνει μέτρηση τάσης με την χρήση των αναλογικών εισόδων του Arduino.

Για να είναι δυνατή η μέτρηση θα συνδεσμοποιηθεί το θερμίστορ μαζί με μια αντίσταση R1 10KΩ σε σειρά συνδεσμοποιία διαιρέτη τάσης, στον ένα ακροδέκτη του θερμίστορ θα συνδεθεί η τροφοδοσία των 5V, στο ένα ακροδέκτη της αντίστασης η γείωση και στο κοινό ακροδέκτη των δυο θα συνδεθεί η αναλογική είσοδος. Όπως περιγράφει το σχήμα 1.



Σχήμα 1 : Μέτρηση αντίστασης

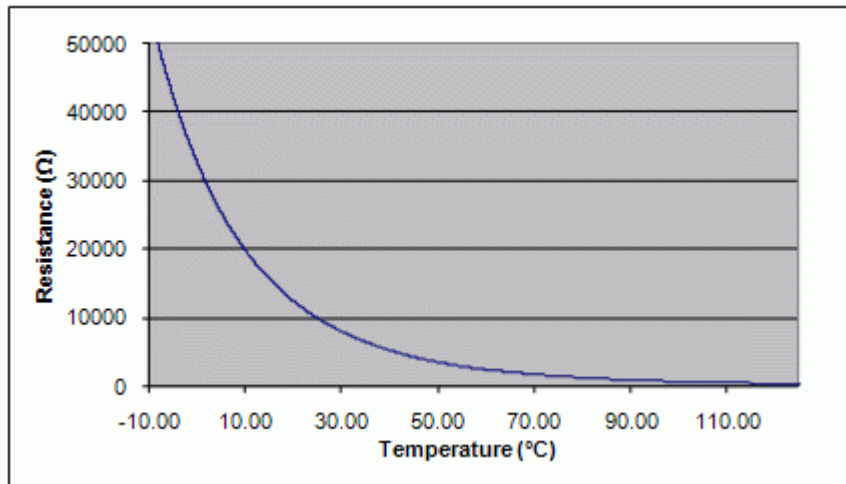
Τότε η τάση  $V_0$  στο κοινό άκρο θα είναι  $V_0 = V_{cc} \frac{R}{R+R_1}$

Η τάση που διαβάζει η αναλογική είσοδος του μικροελεγκτή είναι  $ADC = V_0 \frac{1023}{V_{cc}}$

οπότε  $ADC = V_{cc} \frac{R}{R+R_1} \frac{1023}{V_{cc}} = 1023 \frac{R}{R+R_1}$

λύνοντας ως προς την αντίσταση  $R = \frac{R_1}{\frac{1023}{ADC} - 1}$

## Υπολογισμός Θερμοκρασίας



Σχήμα 2 : Χαρακτηριστική θερμίστορ

Για τον υπολογισμό της θερμοκρασίας θα γίνει χρήση της συνάρτησης

$$T = A + B \ln \frac{R}{R_{ref}} + C \ln^2 \frac{R}{R_{ref}} + D \ln^3 \frac{R}{R_{ref}}$$

T η θερμοκρασία σε βαθμούς Kelvin

R η τιμή της αντίστασης του θερμίστορ

$R_{ref}$  είναι η τιμή της αντίστασης στην θερμοκρασία αναφοράς (εδώ 25°C)

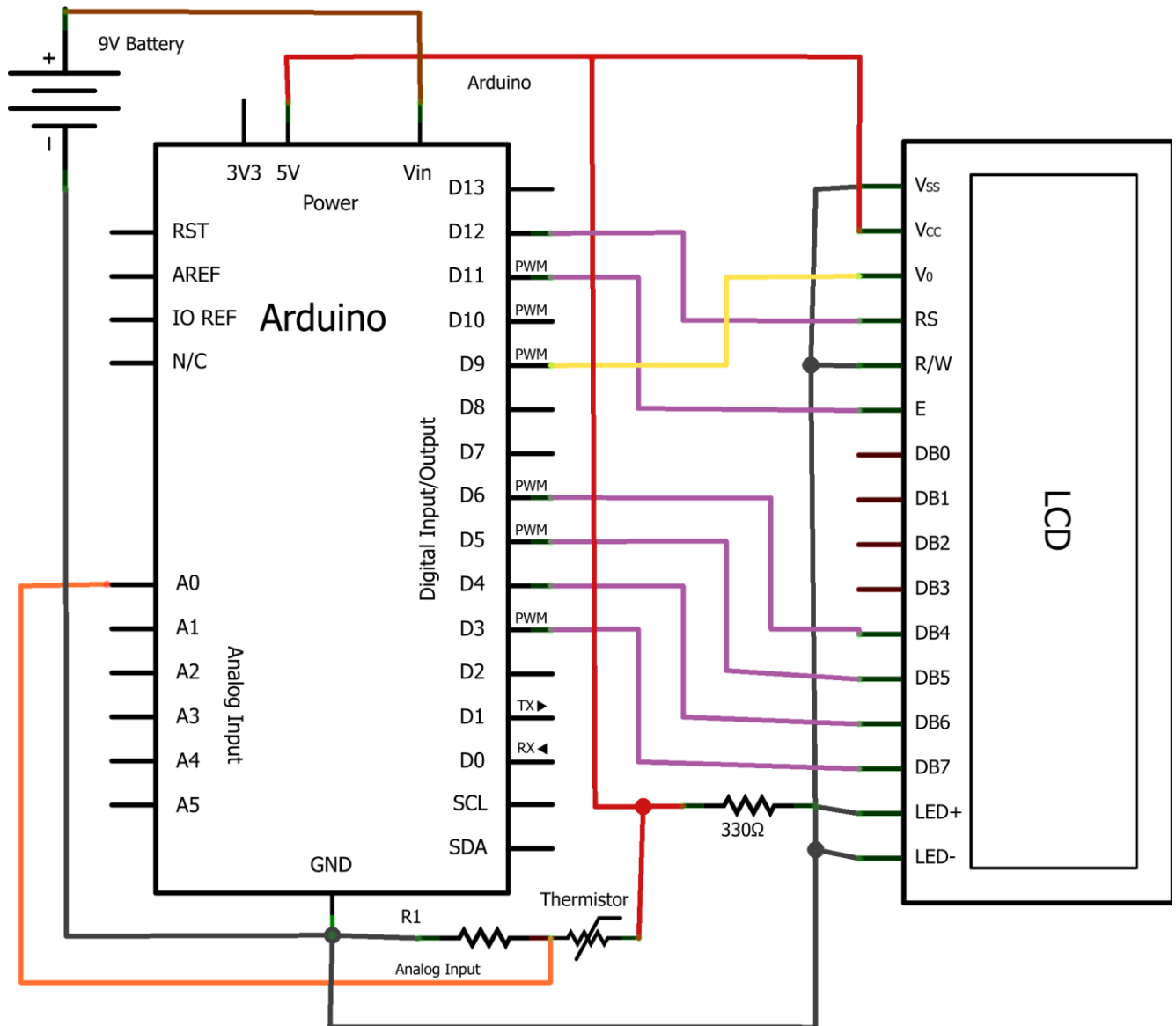
A, B, C, D σταθερές εξαρτώμενες από το συγκεκριμένο θερμίστορ.

Στην περιοχή τιμών από 25°C μέχρι 125°C το εισαγόμενο σφάλμα έχει μέγιστο 0.005°C και στην περιοχή από -40°C μέχρι +25°C μέγιστο 0.015°C όπως αναφέρει το φυλλάδιο του κατασκευαστή.

Στα σφάλματα μέτρησης συμπεριλαμβάνεται το σφάλμα του A/D converter του μικροελεγκτή κατά την μέτρηση της τάσης πάνω στη γνωστή αντίσταση R1.

# Σχεδίαση

Ακολουθεί το Σχήμα 2 με την τελική συνδεσμολογία του οργάνου.



Σχήμα 3 : Συνδεσμολογία οργάνου

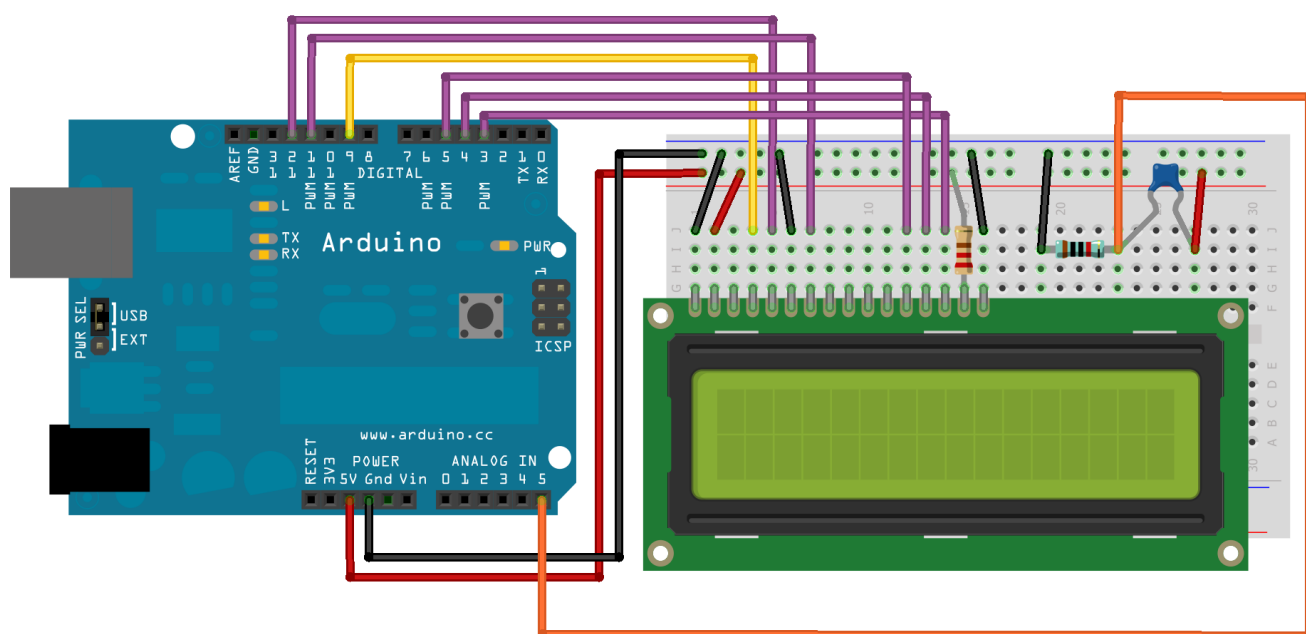
Το άκρο της αντίστασης 10K συνδέετε απευθείας στο GND ( μαύρο ) το κοινό άκρο της αντίστασης 10K και του θερμίστορ συνδέετε στο Analog In 5 ( πορτοκαλί ) το άλλο άκρο του θερμίστορ στην τροφοδοσία 5V ( κόκκινο ) .



Η συνδεσμολογία της οθόνης LCD για να ελεγχθεί με τον μικροελεγκτή Arduino.

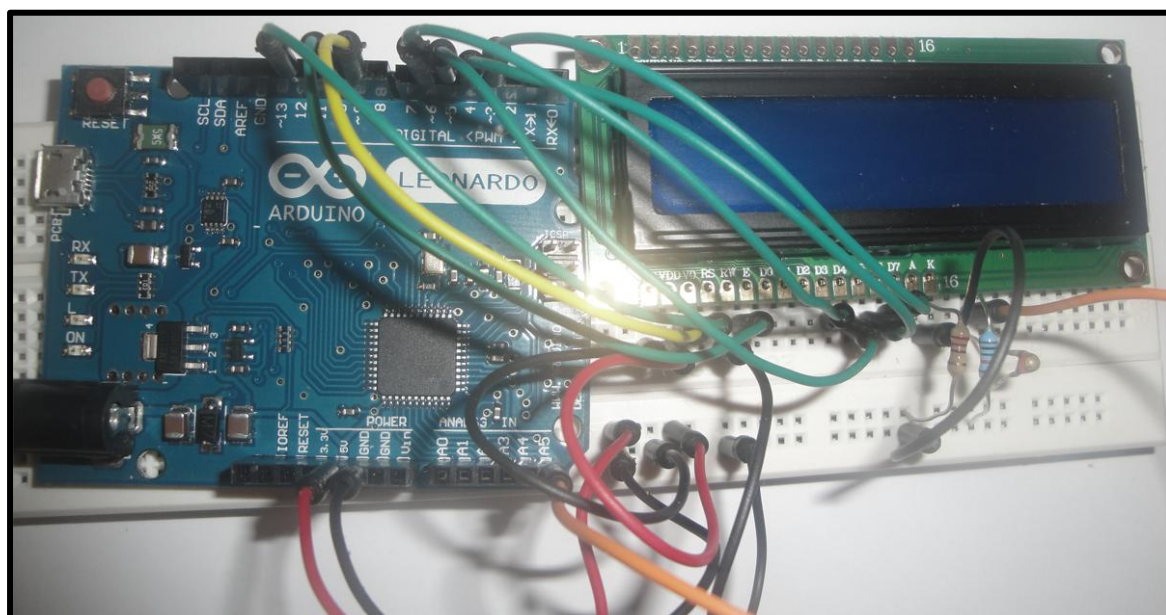
Pin	Arduino	Pin	Arduino
Vss	GND	DB4	D5
Vdd	5V	DB5	D4
Vo	D9	DB6	D3
RS	D12	DB7	D2
R/W	GND	LED+	330Ω-Vcc
E	D10	LED-	GND

Το σχέδιο με την συνδεσμολογία επάνω στο breadboard :



Σχήμα 4 : Συνδεσμολογία οργάνου

Φωτογραφία από την κατασκευή πάνω στο breadboard :



# Παρουσίαση και ανάλυση κώδικα

```
//Required Libraries
#include <math.h>
#include <LiquidCrystal.h>

//Define constants
#define RESISTANCE_R1 10000
#define RESISTANCE_REF 8600
#define CONST_A 0.003354016
#define CONST_B 0.000256985
#define CONST_C 0.000002620131
#define CONST_D 0.00000006383091
#define INPUT_PIN 0
#define OUTPUT_PIN_V0 9

//Calculate temperature for given input voltage
double Temperature( int InputVoltage ){

    //Calculate Resistance
    float R = ( ( ( 1023.0*RESISTANCE_R1 ) / InputVoltage ) - RESISTANCE_REF );

    double logarithm = log( R / RESISTANCE_REF );
    //Return temperature results in Celsius
    return ( 1 / ( CONST_A + CONST_B * logarithm +
        CONST_C * logarithm * logarithm +
        CONST_D * logarithm * logarithm * logarithm ) ) - 273.15;

}

//Define LCD object using selected pins
LiquidCrystal lcd( 12, 11, 5, 4, 3, 2 );

//Arduino setup function
void setup(){

    //Setup digital pins
    pinMode( OUTPUT_PIN_V0 , OUTPUT); // LCD Vo Voltage
    analogWrite( OUTPUT_PIN_V0, 50 ); // 50/1023 of Vcc

    //Initialize LCD
    lcd.begin(16, 2);
    lcd.print("Temperature");

}

//Arduino Loop function
void loop() {

    //Read input voltage
    int InputVoltage = analogRead( INPUT_PIN );

    //Calculate temperature
    double T = Temperature( InputVoltage );

    //Display result
    lcd.setCursor(0, 1);

    //Print temperature
    lcd.print( T );
    lcd.print(" *C");

    delay(500);

}
```

Το βασικό μέρος ενός προγράμματος για τον μικροελεγκτή αποτελείται από τις συναρτήσεις `setup()` και `loop()`, η `setup()` εκτελείται μία φορά κατά την εκκίνηση του προγράμματος με σκοπό την αρχικοποίηση παραμέτρων και η `loop()` εκτελείται επανειλημμένως κατά την λειτουργία του μικροελεγκτή.

Αρχικά στον κώδικά μας εισάγου τις απαιτούμενες βιβλιοθήκες `math.h` και `LiquidCrystal.h` για τον υπολογισμό του λογαρίθμου και τον έλεγχο της οθόνης LCD αντίστοιχα.

Στη συνέχεια ορίζουμε τις σταθερές του προγράμματος όπου `RESISTANCE_R1` η τιμή της γνωστής αντίστασης R1, `CONST_A`, `CONST_B`, `CONST_C`, `CONST_D` οι σταθερές από το συγκεκριμένο θερμίστορ για τον υπολογισμό της αντίστασης, `RESISTANCE_REF` είναι η τιμή της αντίστασης στην θερμοκρασία αναφοράς (εδώ 25°C), `INPUT_PIN` το pin της αναλογικής εισόδου `OUTPUT_PIN_V0` το pin εξόδου για την παροχή τάση στην υποδοχή V<sub>0</sub> της οθόνης LCD.

Η συνάρτηση `Temperature` υπολογίζει την θερμοκρασία σε βαθμούς Κέλσιου. Δέχεται ως όρισμα την τιμή που διαβάζει ο A/D `InputVoltage`, βάση της οποίας υπολογίζει την τιμή της αντίστασης του θερμίστορ. Στη συνέχεια υλοποιώντας την κατάλληλη συνάρτηση (όπως παρουσιάστηκε στον υπολογισμό της θερμοκρασίας ) υπολογίζει την θερμοκρασία και επιστρέφει την τιμή της.

Ορίζεται το αντικείμενο `lcd` του τύπου `LiquidCrystal` με τα αντίστοιχα pins της συνδεσμολογίας της οθόνης.

Στην συνάρτηση `setup()` ορίζεται το pin εξόδου και δίνεται η απαραίτητη τάση στον ακροδέκτη της οθόνης lcd. Στη συνέχεια ενεργοποιείται η οθόνη και τυπώνει στην πρώτη γραμμή την λέξη "Temperature".

Η επαναληπτική `loop()` κάθε 500ms διαβάζει την τάση της αναλογικής εισόδου και καλεί την `Temperature` για τον υπολογισμό της θερμοκρασίας . Στην συνέχεια εκτυπώνει το αποτέλεσμα.

# Προβλήματα υλοποίησης

Το πιο σοβαρό πρόβλημα αρχικά ήταν η εύρεση των σωστών παραμέτρων για την ακριβή υλοποίηση της συνάρτησης του συγκεκριμένου θερμίστορ, επίσης το μαγαζί που τα προμηθευτήκαμε δεν βοήθησε ιδιαίτερα. Τελικά μετά από ψάξιμο στο διαδίκτυο βρέθηκε το αντίστοιχο datasheet.

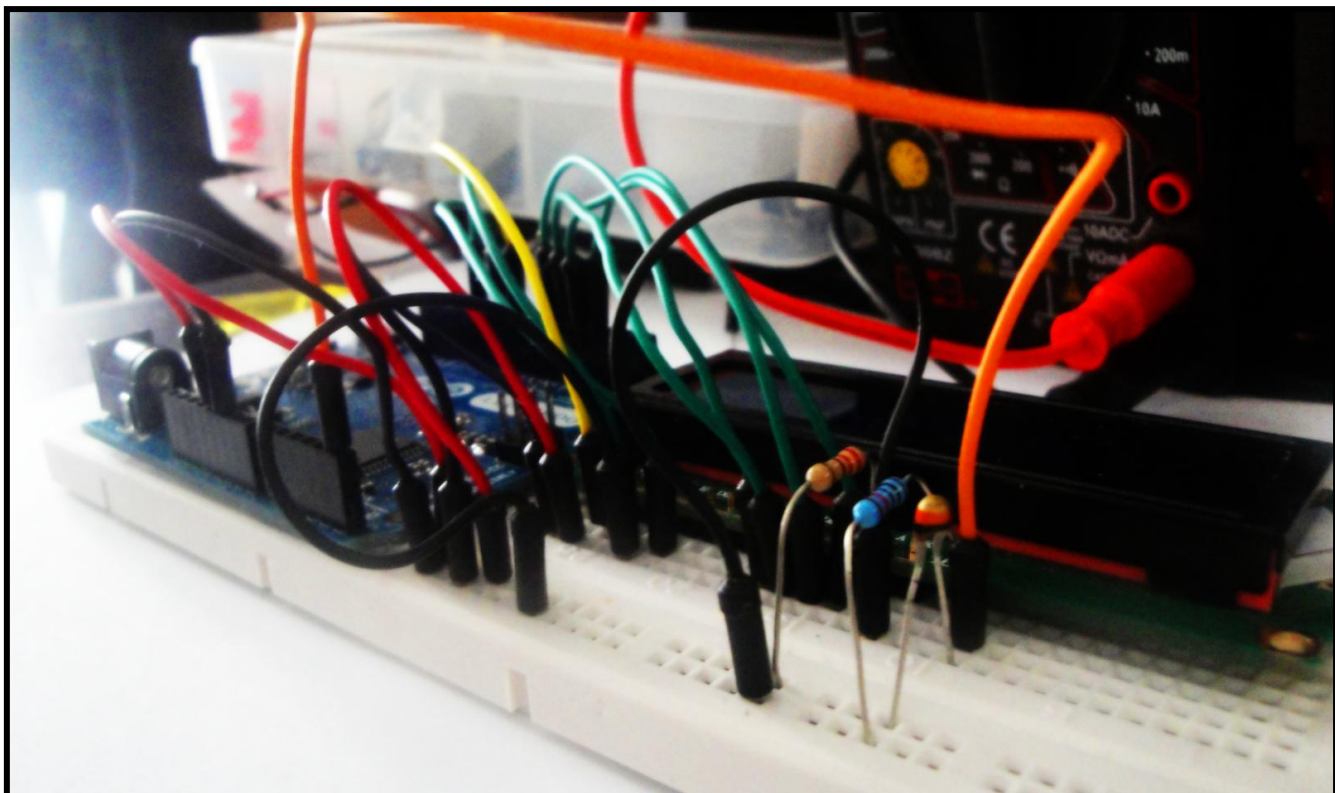
Μικροπροβλήματα είχαμε αρχικά με την συνδεσμολογία, κυρίως της οθόνης η οποία για να λειτουργήσει αξιόπιστα απαιτεί μεγάλη σταθερότητα ( χρειάζεται συγκόλληση ). Η οθόνη επίσης αναπαράγει ένα σφάλμα εμφανίζοντας έναν εξτρά χαρακτήρα κάποιες φορές ,όχι πάντα .Την αιτία δεν την έχουμε εντοπίσει.

Ένα ακόμη πρόβλημα που ίσως διορθωθεί είναι η έλλειψη των κατάλληλων συνθηκών για την ακριβή και αξιόπιστη μέτρηση της  $R_{ref}$  στους  $25^{\circ}\text{C}$  ,αναγκαία για την υλοποίηση της συνάρτησης.

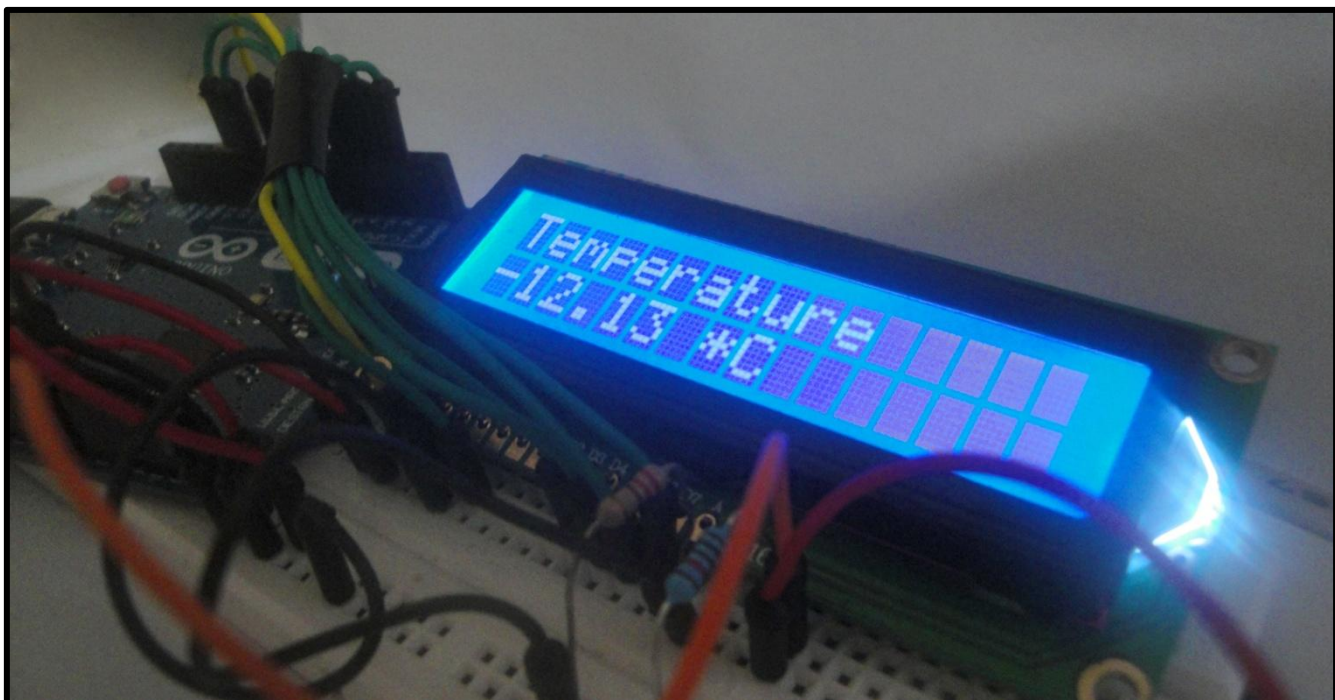




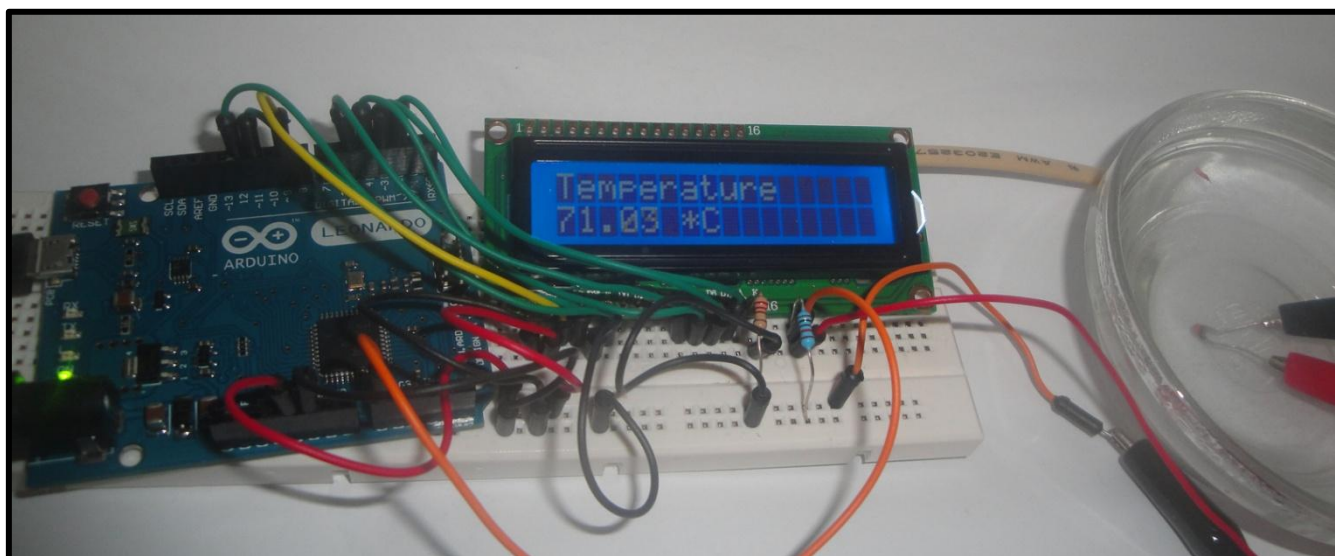
# Μετρήσεις



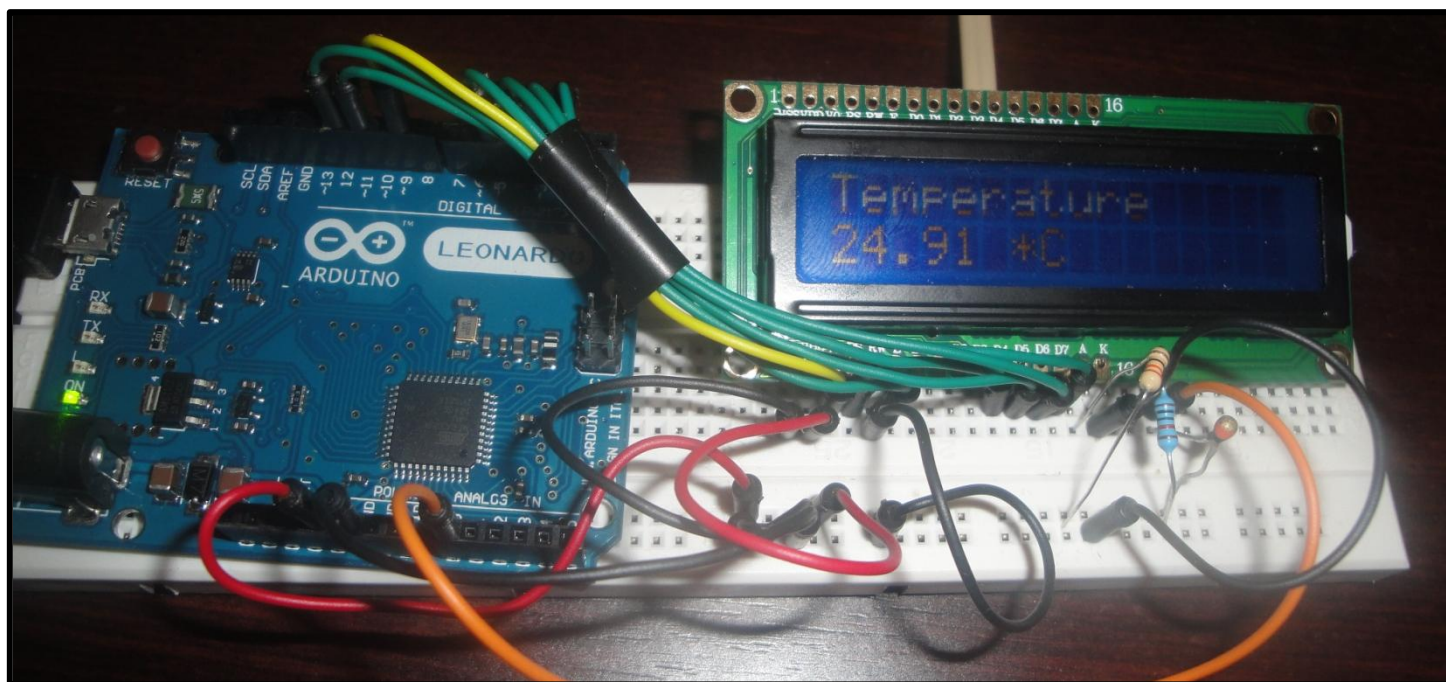
Έχουν γίνει μετρήσεις σε θερμοκρασία δωματίου, σε κατάψυξη και σε καυτό νερό.



Μέτρηση στην κατάψυξη



Μέτρηση σε καυτό νερό



Μέτρηση σε θερμοκρασία δωματίου

# Συμπεράσματα

Η υλοποίηση ενός οργάνου μέτρησης θερμοκρασίας με θερμίστορ αποτελεί μια πολύ οικονομική λύση, επίσης σε συνδυασμό με τον μικροελεγκτή Arduino ξεπερνιέται εύκολα το πρόβλημα του υπολογισμού της θερμοκρασίας λόγω μεγάλης μη-γραμμικότητας στη συμπεριφορά του θερμίστορ. Στα αρνητικά συμπεριλαμβάνεται το ότι αναπόφευκτα εισάγονται κάποια σφάλματα στις μέτρησης λόγω ευαισθησίας του οργάνου.

Εναλλακτικά η συναρμολόγηση των εξαρτημάτων του κυκλώματος θα μπορούσε να γίνει μόνιμα πάνω στη διάτρητη πλακέτα, η οποία θα προσαρτηθεί πάνω στο Arduino.

Μια ιδέα για βελτίωση της κατασκευής, χάριν χρηστικότητας και ευστάθειας της διάταξης, είναι η τοποθέτηση όλων των συστατικών του κυκλώματος σε ένα κατάλληλο κουτί από άποψη μεγέθους. Για αυτήν υλοποίηση πιο ταιριαστή θα ήταν μια μικρότερη έκδοση Arduino όπως το Micro ή το Nano.

## Βιβλιογραφία

- [1] <http://en.wikipedia.org/wiki/Thermistor>
- [2] <http://www.arduino.cc/>
- [3] <http://arduino.cc/en/Main/arduinoBoardLeonardo>
- [4] <http://arduino.cc/en/Reference/LiquidCrystal>
- [5] <http://en.wikipedia.org/wiki/Arduino>
- [6] Συστήματα μετρήσεων, Β. Πετρίδη
- [7] Αισθητήρες μέτρησης και έλεγχου, P. Elgar
- [8] Vishay BCcomponents NTCLE100E3 NTC Thermistors, Radial Leaded, Standard Precision, Datasheet