

Plant monitor and watering system

Introducere

Proiectul constă într-un sistem de monitorizare a temperaturii, luminii și umidității ce va acționa o pompă de apă, cu scopul de a iriga automat și eficient plantele dintr-o locuință. Acesta poate alerta utilizatorul atunci când e nevoie de apă în rezervor.

Descriere generală

Schema bloc



Descriere componente

1. Placa de dezvoltare UNO R3 ATmega 328p

Placa de dezvoltare Arduino servește ca unitate centrală de procesare, gestionând toate funcțiile și comunicațiile în proiect. Arduino primește date de la senzorii de temperatură și presiune atmosferică și le afișează pe ecranul LCD. De asemenea, primește date de la senzorul de umiditate a solului și de la senzorul ultrasonic pentru a controla pompa de apă și pentru a emite semnale sonore prin buzzer.

2. LCD

Afișajul LCD cu interfață I2C oferă o modalitate convenabilă de a afișa informații text pe un ecran cu 16 caractere pe 2 linii. Arduino comunică, prin protocolul I2C, cu display-ul (LCD) pentru a afișa datele de la senzorii de temperatură și presiune atmosferică.

3. Senzor ATH10 pentru umiditate atmosferică și temperatură:

Acest senzor măsoară umiditatea și temperatura ambientală. Arduino citește datele de la senzorul

ATH10 și le afișează pe ecranul LCD.

4. Senzor ultrasonic

Acest senzor utilizează ultrasunete pentru a măsura distanța între el și un obiect. Arduino citește datele de la senzorul ultrasonic pentru a determina nivelul apei și pentru a controla pornirea și oprirea pompei de apă în funcție de acest nivel. De asemenea, poate alerta utilizatorul atunci când nivelul apei este scăzut.

5. Senzor analogic de umiditate a solului

Acest senzor măsoară umiditatea solului. Arduino citește datele de la senzorul de umiditate a solului și utilizează aceste informații pentru a decide când să activeze sau să dezactiveze pompa de apă.

6. Buzzer

Buzzerul este un dispozitiv audio utilizat pentru a emite semnale sonore. Arduino controlează buzzerul pentru a emite semnale sonore în cazul unui nivel scăzut al apei.

7. Pompa de apă submersibilă

Pompă electrică utilizată pentru a pompa apă dintr-un rezervor într-un alt recipient sau sistem. Arduino controlează pornirea și oprirea pompei de apă în funcție de datele primite de la senzori (cu precădere umiditatea solului și nivelul apei din rezervor).

8. Modul tranzistor IRF520 MOSFET

Acest modul conține un tranzistor MOSFET utilizat pentru a controla puterea furnizată către dispozitive de mare putere, cum ar fi motoarele. Arduino utilizează modulul IRF520 pentru a controla alimentarea pompei de apă, permițându-i să pornească sau să oprească fluxul de apă în funcție de necesități.

Hardware Design

Diagrama circuit

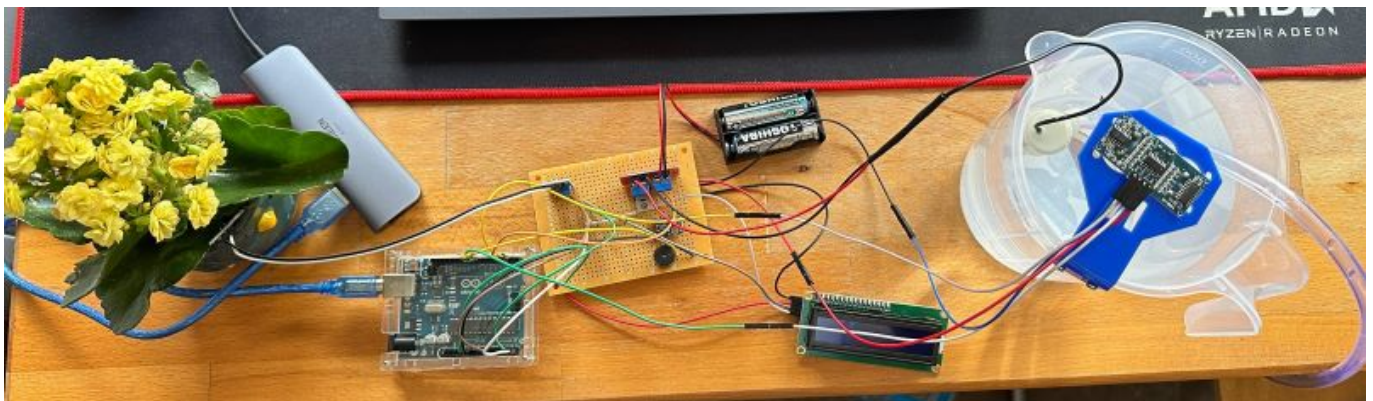


Lista piese:

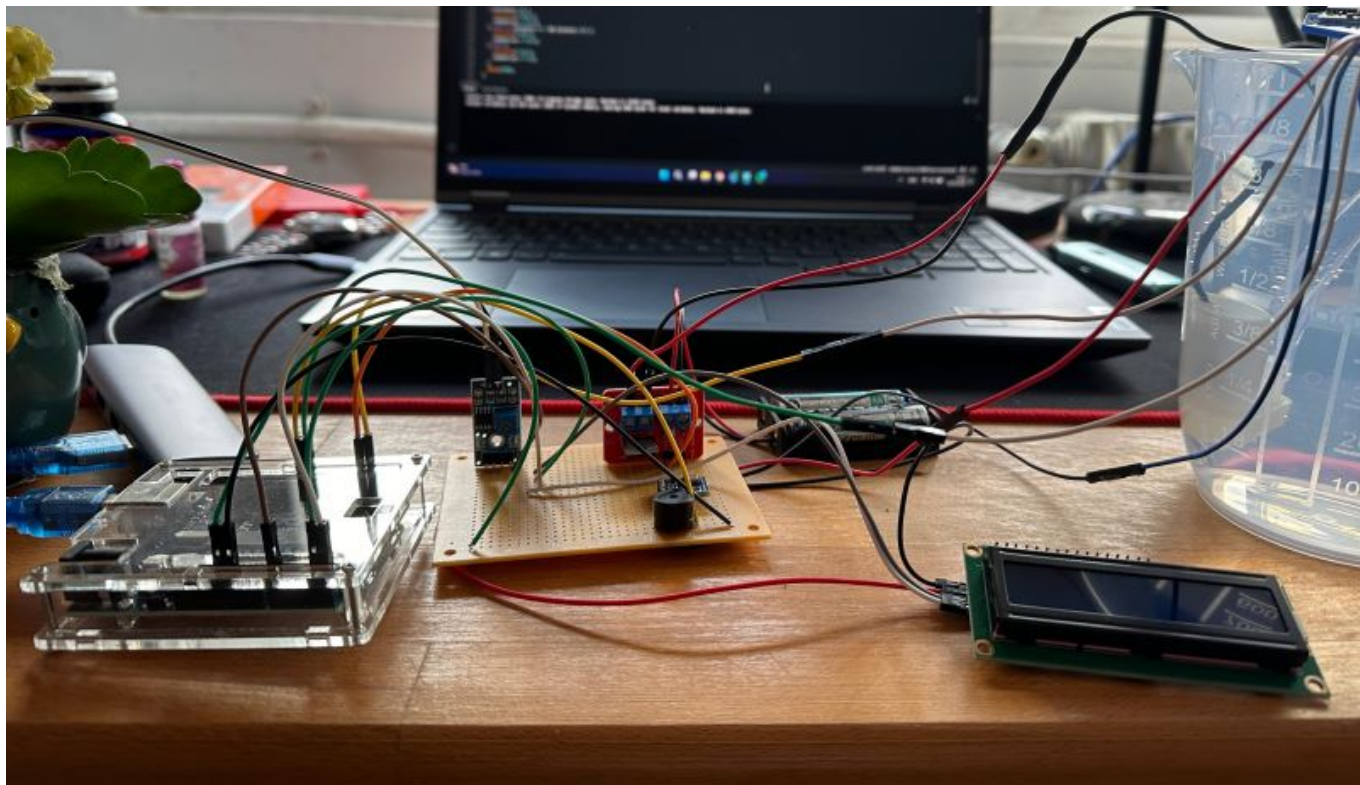
- Placa de Dezvoltare UNO R3 ATmega328p
- 1602 LCD with I2C Interface and Blue Backlight
- Senzor ATH10 pentru umiditate atmosferică și temperatură
- HC-SR04 Ultrasonic Sensor Module
- Senzor analogic umiditate sol
- Buzzer pasiv
- Pompa apa submersibila, 3-6VDC
- IRF520 MOSFET power transistor module
- Suport baterii 2xR6

Montaj & setup hardware

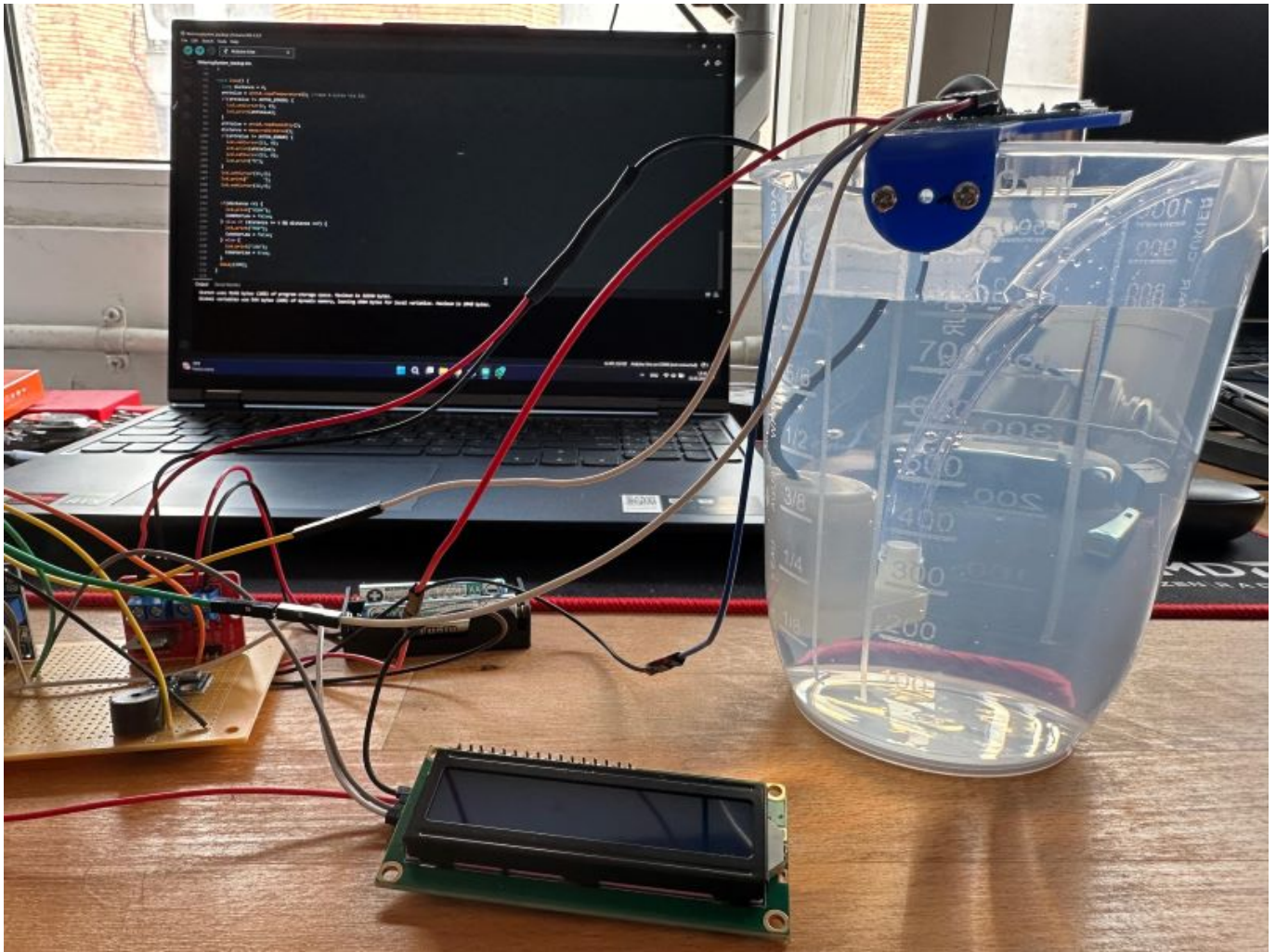
General



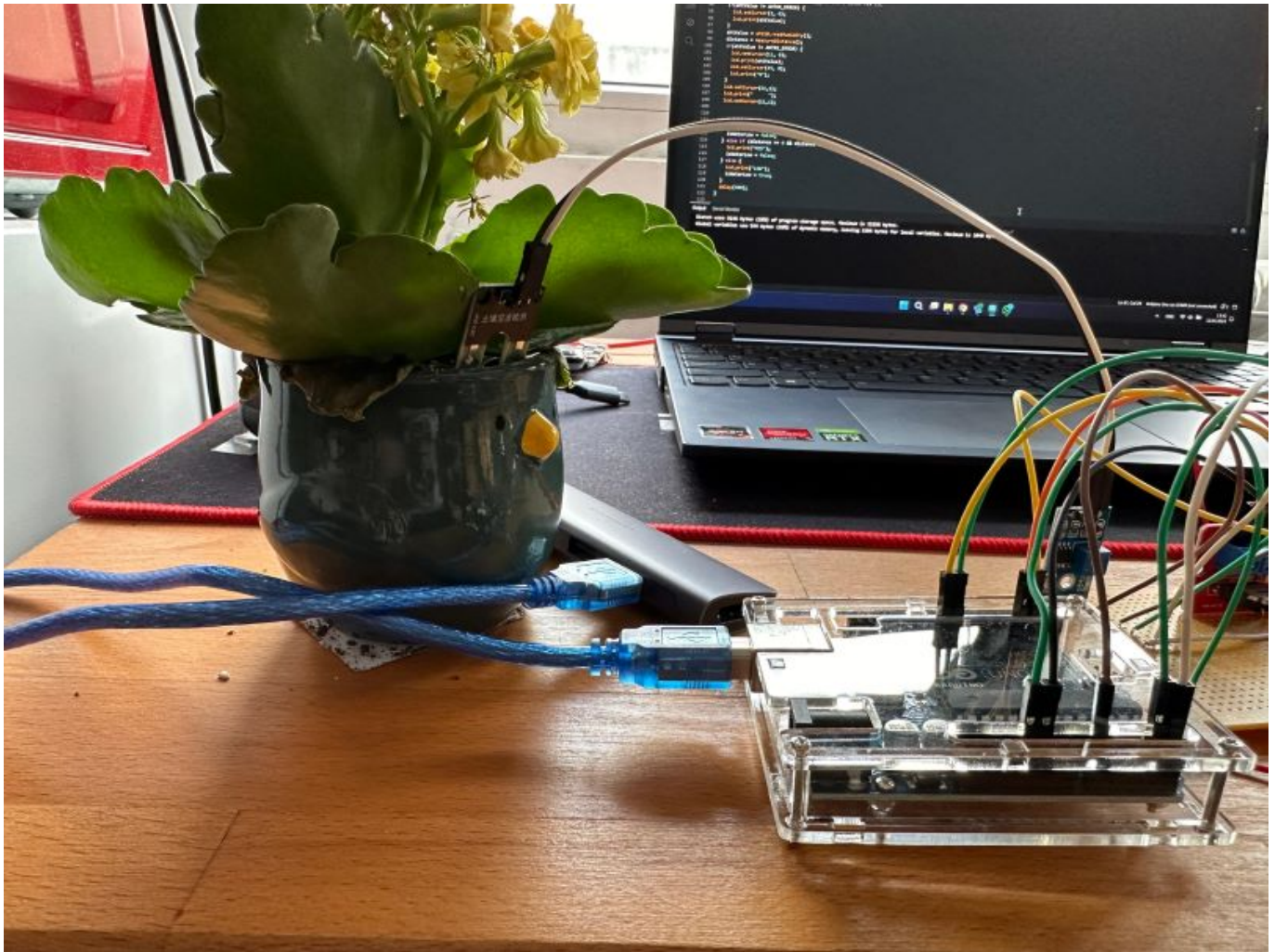
Cablaj



Sistemul de gestiune a apei



Monitorizarea plantei



Software Design

Initializarea și scrierea pe display (prin I2C) se realizează utilizând biblioteca LiquidCrystal_I2C.h.



Citirea și afisarea valorilor captate de senzorul pentru temperatura și umiditate (AHT10 prin I2C) se realizeaza utilizand biblioteca AHTxx.h.



Determinarea nivelului de apa din rezervor se realizeaza prin funcția următoare, ce măsoară durata între trimiterea unui impuls sonor și captarea acestuia, utilizând un senzor ultrasonic de distanță. Apoi se realizează conversia în centimetri, luând în calcul viteza sunetului.



Utilizez 2 tipuri de întreruperi:

* hardware interrupt pentru oprirea funcționării atunci când apa din rezervor nu este suficientă.

* pin change interrupt pentru determinarea schimbărilor de umiditate în sol, captate de senzorul analogic plasat în pământ.



Buzzer-ul pasiv și pompa sunt actionate, utilizând funcțiile `tone()` și `analogWrite()` (prin semnale de tip PWM)

Mediu de dezvoltare: Arduino IDE

Librării folosite: `Wire.h`, `LiquidCrystal_I2C.h`, `AHTxx.h`

Rezultate Obținute

[Demo funcționare \(Google Drive Link\)](#)

Concluzii

Acest proiect m-a ajutat sa consolidez materia din cadrul laboratoarelor. Mi-a placut sa creez ceva fizic, care functioneaza si are o utilitate practica. Cred ca acest proiect poate fi rafinat si utilizat in propria casa :)

Download

Cod Github: [aici](#)

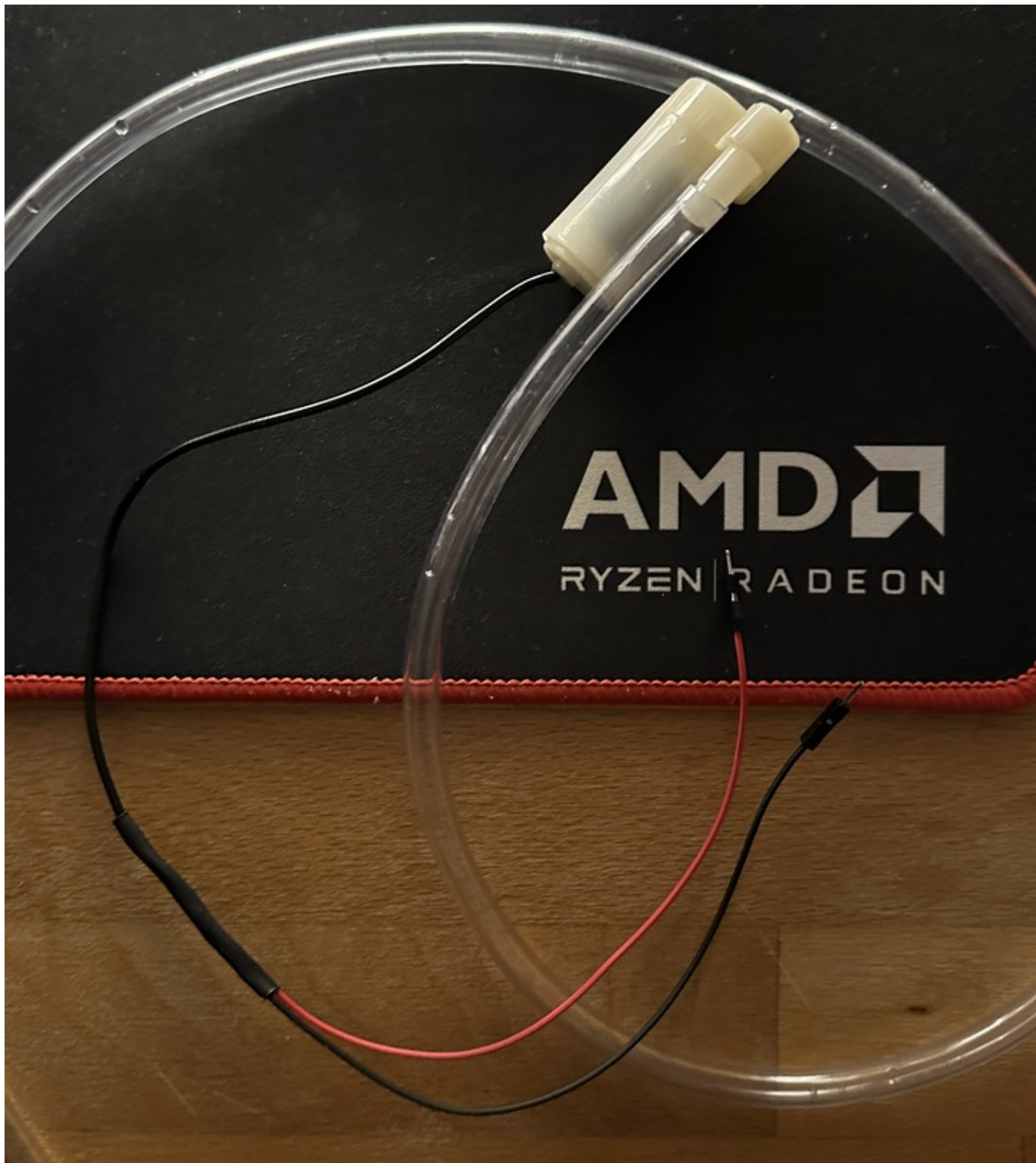
Arhivă proiect: [plantmonitoringwateringsystem.zip](#)

Jurnal

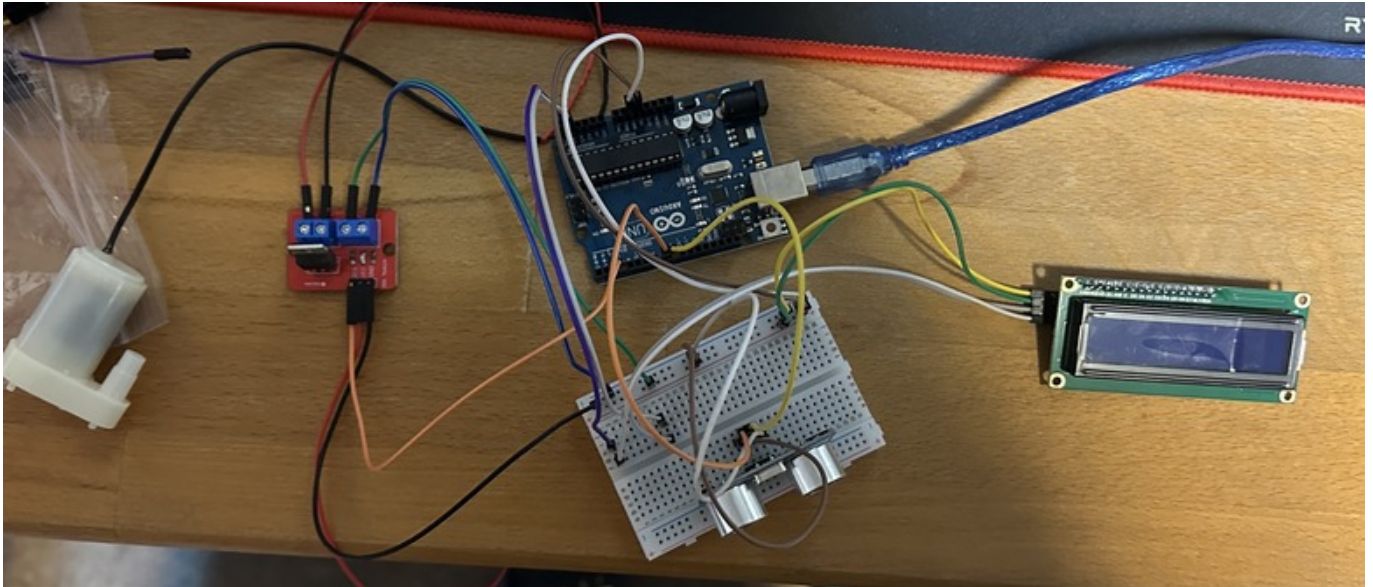
* 1 mai - am conceput si formulat tema de proiect

* 5 mai - am finalizat lista de piese si am comandat cele necesare

* 6 mai - am cablat pompa de apa



* 7 mai - am realizat cablarea componentelor, folosind doar breadboard-ul; am testat fiecare componenta in parte, utilizand cod simplu

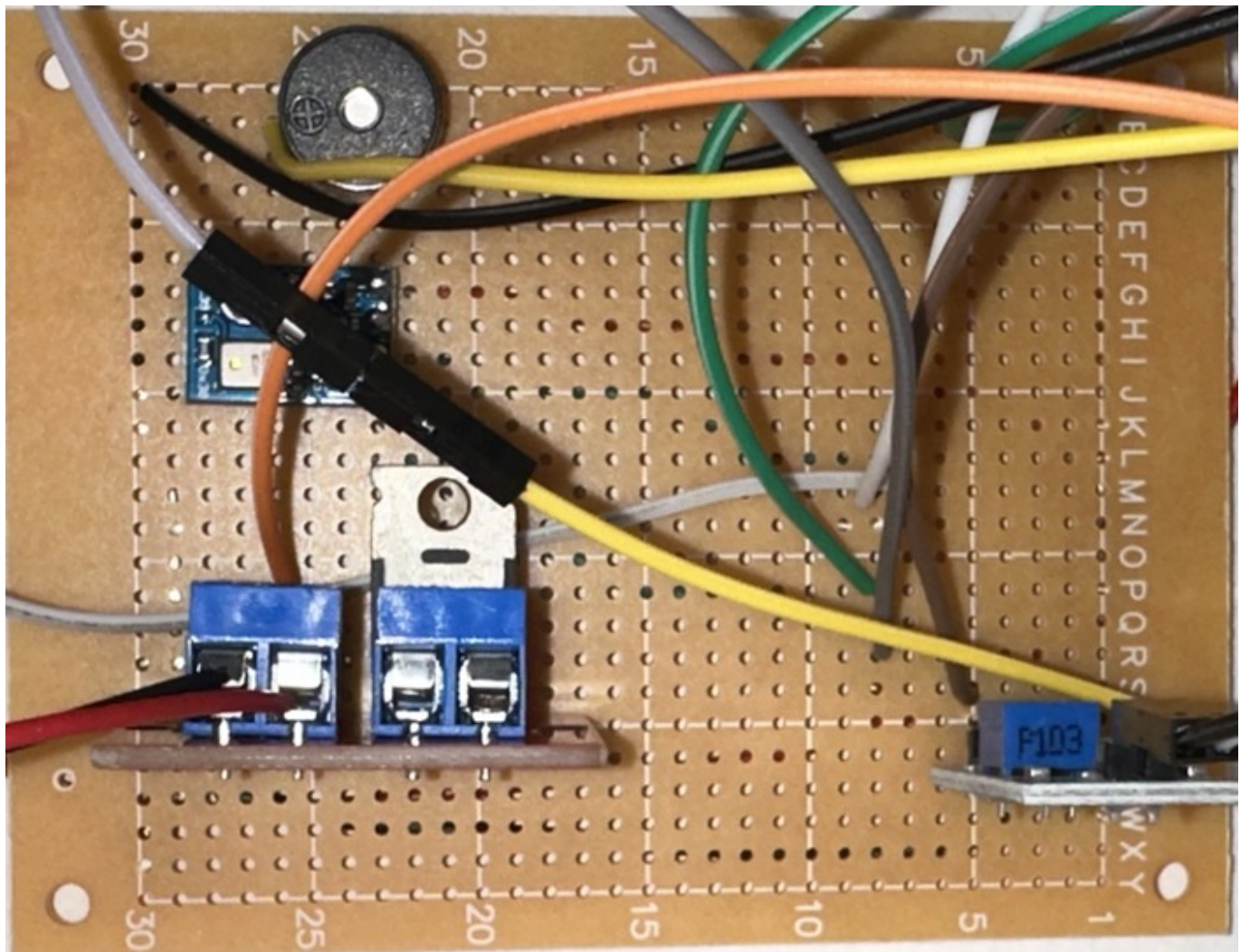


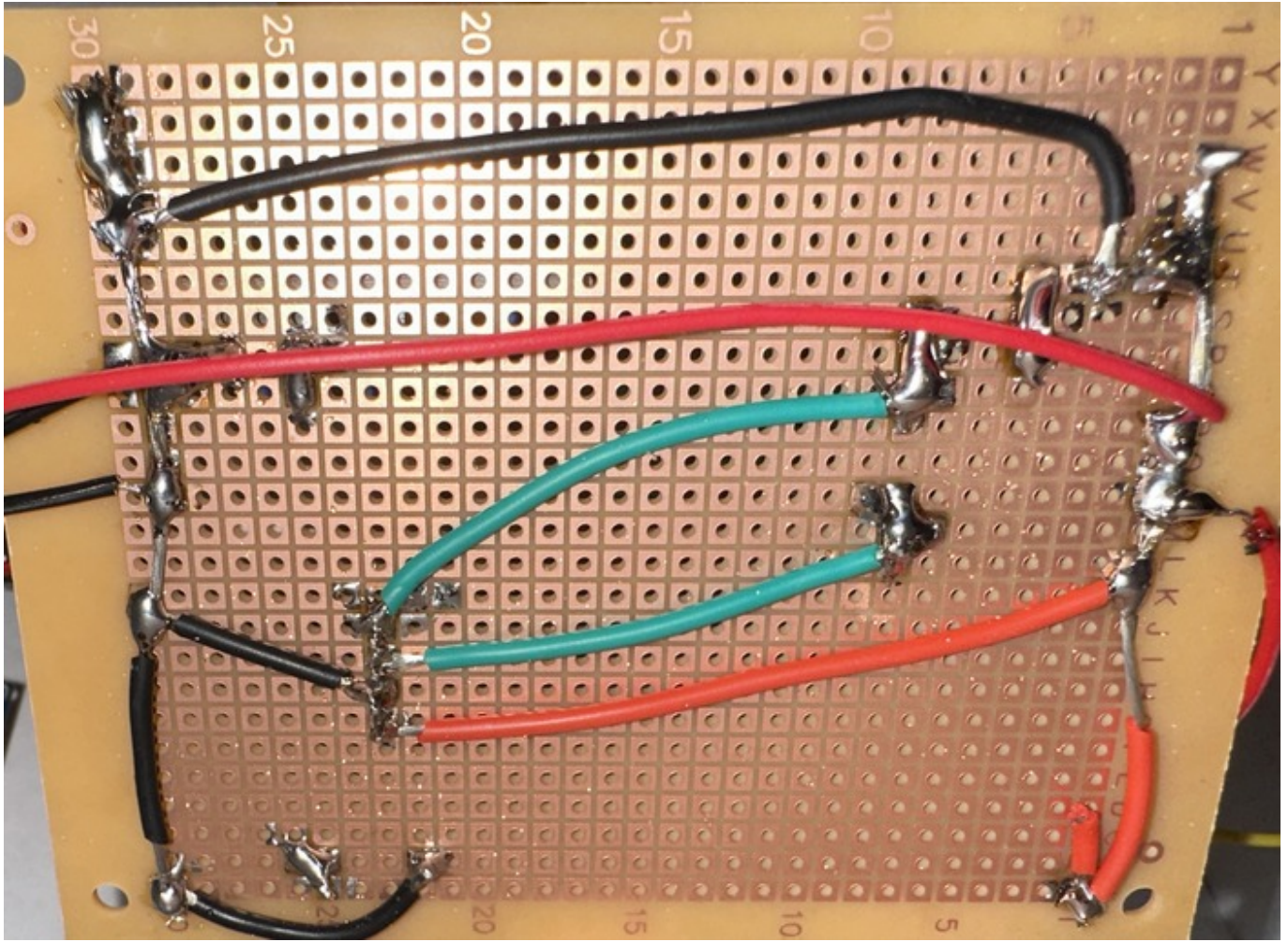
* 8 mai - am lipit pinii senzorului de temperatura si umiditate AHT10



* 10 mai - sistemul actioneaza pompa cat timp exista apa in rezervor, altfel se aude buzzer-ul pasiv

* 15 mai - am finalizat lipirea tuturor pieselor pe o placa de prototipare (milestone hardware)





* 22 mai - sistemul actioneaza pompa cat timp este apa in rezervor, insa doar atunci cand solul este uscat (milestone software)

Bibliografie/Resurse

Resurse hardware

Datasheets

- * ATMEGA328P - [atmega48a.pdf](#)
- * Senzor temperatura & umiditate AHT10 - [aosong_aht10_en_draft_0c.pdf](#)
- * Senzor ultrasonic de distanta HC-SR04 - [hc-sr04.pdf](#)
- * IRF520 tranzistor de putere - [irf520.pdf](#)

Resurse software

- * Arduino hardware interrupts tutorial - [YouTube Link](#)
- * Pin change interrupts guide - [Link to blog](#)
- * Tone library in Arduino - [Arduino official website](#)
- * Liquid crystal I2C Arduino library - [Arduino official website](#)

[Export to PDF](#)

From:

<http://ocw.cs.pub.ro/courses/> - **CS Open CourseWare**

Permanent link:

<http://ocw.cs.pub.ro/courses/pm/prj2024/vstoica/tudor.popescu0210>



Last update: **2024/05/27 00:09**