|  |
| --- |
| Universitatea Imagini pentru electronica ettiImagini pentru university politehnica of bucharest POLITEHNICA Bucuresti  Facultatea de electronica, telecomunicatii si  Tehnologia informatiei |
| Proiect Optional Microcontrolere |
| Placa Intel Galileo |
|  |
| **Student: Ciurescu Alexandra Mihaela** |
|  |

|  |
| --- |
| Bucuresti, 2016 |

**Cuprins**

1. Introducere 3
   1. Descrierea proiectului 3
2. Resurse software si hardware 4
   1. Resurse hardware 4
      1. Placa de dezvoltare Intel Galileo Gen 1 4
      2. Senzor umiditate 6
      3. Senzor temperatura 7
      4. Celula fotoconductoare 8
      5. Afisaj LCD...................................................................................... 8
      6. Pompa de apa...................................................................................9
   2. Resurse software 9
3. Etapele de realizare ale proiectului....................................................................................10

3.1 Implementare hardware 10

3.2 Implementare software 12

1. Concluzii 15

5. Bibliografie........................................................................................................................ 16

1**. Introducere**

Acest proiect isi propune familiarizarea studentilor cu o placa de dezvoltare realizata in jurul unui microprocesor de tip Intel x86. Se urmareste realizarea unui proiect practic, prin care studentii sa-si insuseasca informatii atat despre placa Intel Galileo, cat si despre modul de functionare al diverselor componente conectate la placa.

**1.1. Descrierea proiectului**

Proiectul „Gradinarul Electronic” are ca scop monitorizarea cantitatii de lumina la care are acces o planta, umiditatea solului din ghiveci si temperatura ambientala. Pe baza acestor parametri, controlerul Galileo va lua decizia cand floarea va trebui udata si va actiona o valva care va deschide circuitul de udare al plantei.

Componente necesare:

* Breadboard
* Display
* Senzor de lumina
* Senzor de temperatura
* Senzor de umiditate

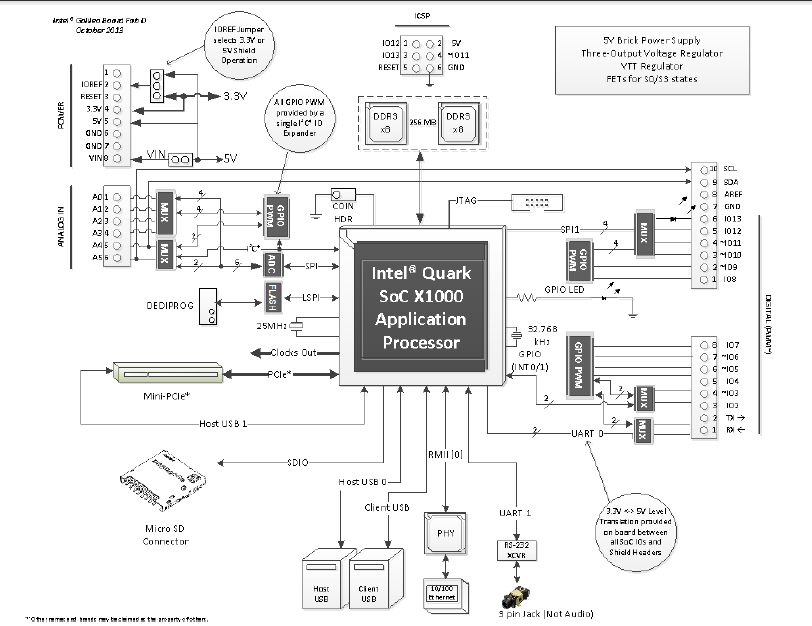
Tipuri de conexiuni:

* 5V
* GND
* I/O digitali si analogici

**2.Resurse software si hardware**

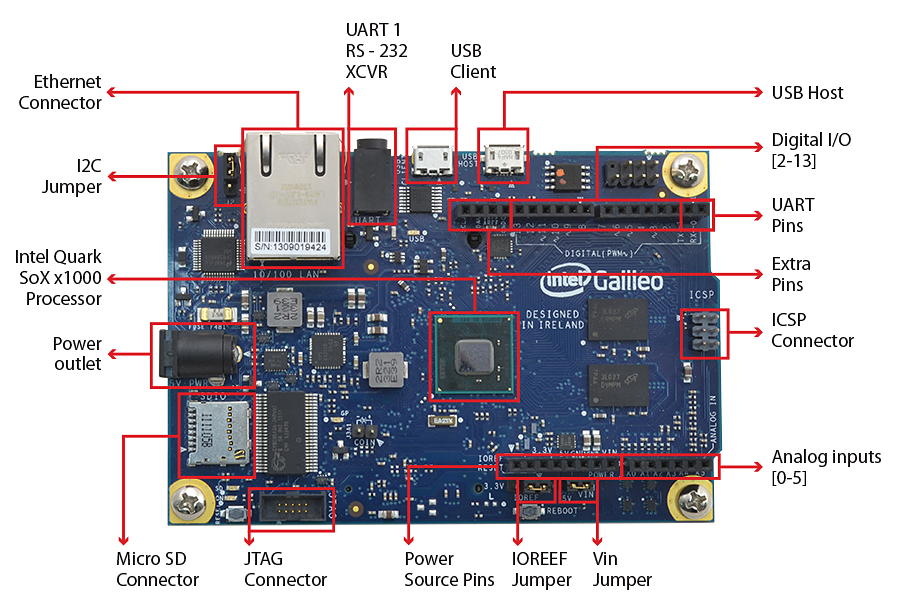
2.1. Resurse hardware

2.1.1.Placa de dezvoltare Intel Galileo Gen 1



Intel® Galileo Gen 1 se bazeaza pe un procesor de tip Intel Quark SoC X1000, un sistem de clasa Intel Pentium® pe 32 de bits. Este prima placa bazata pe arhitectura Intel proiectata pentru a fi compatibila din punct de vedere hardware si software cu pinii de pe shield-urile pentru Arduino Uno R3(pinii digitali de la 0 la 13, intrarile analog de la 0 la 5, pini porturilor pentru UART se afla in aceeasi pozitie).

In acest proiect, am folosit alimentarea de 5V disponibila pe placa si o serie de pini digitali si analogi.

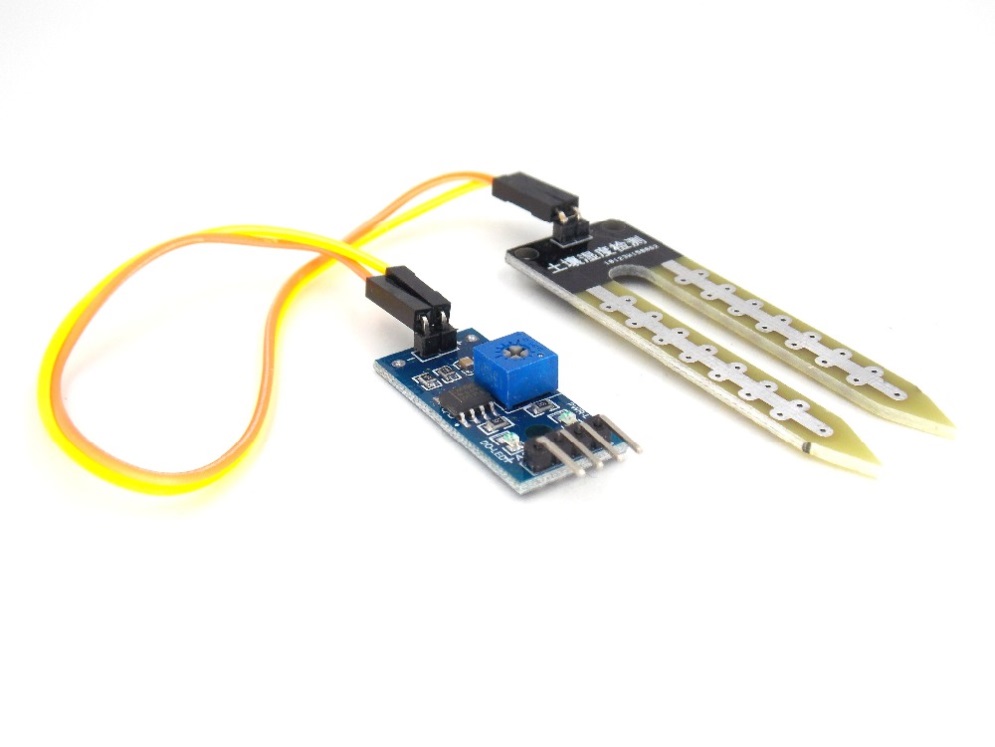
Astfel, iesirea senzorului de temperatura este conectata pe **pinul analogic A0**, urmand ca informatia preluata de pe acesta sa treaca printr-un multiplexor si apoi printr-un convertor analog-digital. Dupa cum se poate observa si pe schema, rolul multiplexorului analogic este acela de a selecta, la intrarea convertorului analog-digital, unul dintre cele sase canale analogice. Ulterior, informatia din ADC este transmisa procesorului prin interfata periferica seriala(SPI-Serial Pheriferical Interface).Acelasi lucru se realizeaza si pentru pinii **A2** si **A3** care sunt folositi pentru fotocelula si respectiv pentru senzorul de umiditate.

In ceea ce priveste pinii digitali, **pinul IO12** este legat la pinul RS al Lcd-ului. Cand RS este 0, informatia este tratata ca o instructiune (exemplu: pozitionare cursor, clear screen,etc.), iar cand RS este 1informatia este vazuta ca un text care trebuie afisat pe ecran (exemplu: pentru a afisa litera „T”, RS trebuie sa fie 1). **Pinul IO11**se conecteaza la pinul Enable de pe LCD si are rolul de a transmite informatii pe magistrala de date. Pinul RW al display-ului are rolul de a determina daca informatia de pe magistrala de date se va scrie pe LCD(RW este 0), sau va fi citita(RW este 1). Ambii pini sunt pini de uz general de intrare/iesire (GPIO).

Restul de pini analogici folositi**(IO2,IO3,IO4,IO5)** au rolul de a transmite informatia pe magistrala de date, acestia fiind conectati la pinii D6,D5,D4 si D3 ai LCD-ului.

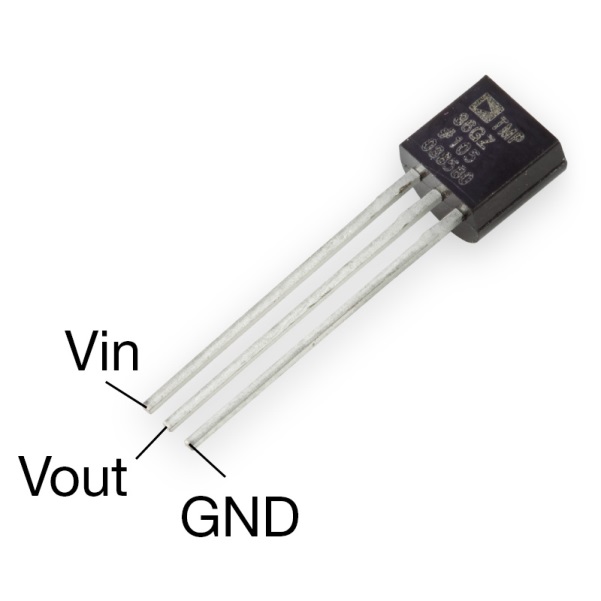
2.1.2.Senzorul de umiditate

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Componenta | Supply Voltage | Precizie | Interval temperatura | Output | Pret |
| VH400 Soil Moisture Sensor | 3.5V to 20 V | 2% | -40ºC to 85ºC | 0 to 3V | $39.95 |
| Soil Humidity Hygrometer Moisture Detection Sensor YL-69 | 3.3V~5V | N/A | -40 to +125°C | Dual output mode:  -analog output more accurate  -digital for simple high vs low | 5$ |
| Capacitive Soil Moisture Sensor SKU:SEN0193 | 3.3 ~ 5.5 V | N/A | -40 to +125°C | 0 ~ 3.0V | 7$ |

Am ales senzorul de umiditate a solului Soil Hygrometer Humidity, compatibil cu Intel Galileo Gen2, motivul principal fiind pretul scazut fata de celelalte. De asemenea,el functioneaza la o tensiunea de lucru de la 3,3 V la 5 V.  
Acesta detectează cantitatea apei din sol, electrozii senzorului acționează ca un rezistor variabil care se va schimba de la 100kohmi (atunci când solul este umed) la 2Mohmi în stare uscată.

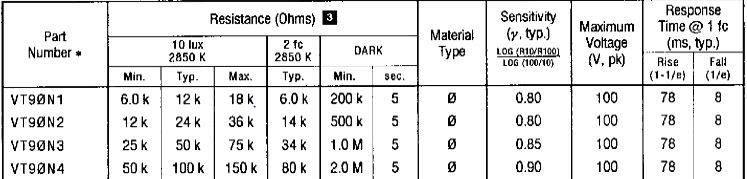
2.1.3.Senzorul de temperatura

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Producator | Numar Componenta | Tip Iesire | Utilizare | Interval temperatura | Precizie  (Valoare tipica) | Precizie  (Valoare maxima) | Input Voltage Range | Output Voltage Range |
| Microchip | EMC1072 | Digital | SMBus/I2C Multi Temperature Sensor | -40 to +125°C | 0.25°C | +/-1°C | 3.0 to 3.6V | - |
| Analog Devices | TMP36 | Analog | Voltage Output Temperature Sensor | -40 to +125°C | +/-1-2°C | +/-2-4°C | 2.7 to 5.5V | 750mV la 25 °C |
| Texas Instruments | LM34 | Analog | Analog Output Temperature Sensor | -50 to +300°F | +/-1°F | +/-4°F | 5 to 30V | - |
| Analog Devices | TMP35 | Analog | Voltage Output Temperature Sensor | +10 to +135°C | +/-1-2°C | +/-2-4°C | 2.7 to 5.5V | 250mV la 75°C |

Din senzorii prezentati in tabel am ales senzorul TMP36 deoarece functioneaza intr-un interval de temperatura corespunzator cerintelor proiectului (TMP35 functioneaza doar pentru temperaturi pozitive) si functioneaza in limitele de tensiune impuse de placa de dezvoltare (2.7V-5.5V). N-am ales EMC1072 intrucat avea o iesire digitala noi avand nevoie de iesire analog. In cazul LM34 se observa ca precizia este mult mai scazuta.

Senzorul de temperatura TMP36 furnizeaza la iesire o tensiune ce depinde de temperatura din mediu. De asemenea este convenabil deoarece tensiunea de la iesire se schimba direct proportional cu temperatura in grade Celsius. In datasheet se precizeaza ca o variatie cu 10mV este echivalenta cu cea a unui grad Celsius.

2.1.4.Celula fotoconductoare



Din aceasta serie de celule fotoconductoare noi am ales componenta VT90N1 intrucat rezistenţa sa electrică este mare, iar variatia ei la iluminare este mai uşor de măsurat; Acest lucru ne-a permis sa avem o afisare mai precisa a luminozitatii. De asemenea am dispus in prealabil de aceasta componenta ceea ce a dus la scaderea costurilor proiectului.

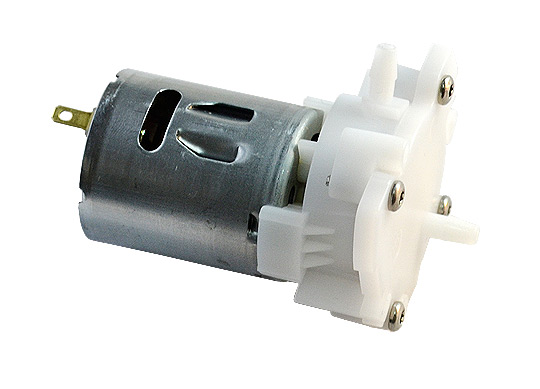
2.1.5. Afisaj LCD

Am ales un LCD de 20x4 caractere deoarece aveam nevoie de mai multe linii pentru afisarea tuturor parametrilor: temperatura, umiditate, luminozitate. Puteam alege din urmatoarele variante de LCD-uri: 16x2, 16x4, 8x2, insa acestea nu indeplineau cerintele pentru proiect.



2.1.6. Pompa de apa

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Producator | Componenta | Tensiune de lucru | Diametru pompa | Diametru tub | Material | Pret |
| <https://www.aliexpress.com/item/5-x-DC-3v-6v-Mini-Micro-Submersible-Water-Pump-Low-Noise-Motor-pump-120L-H/32689942954.html?spm=2114.40010208.4.36.QA20Ct> | Mini Submersible Water Pump Low Noise Motor | 2.5-6V | 24mm | 5mm | Plastic | 8.53$ |
| Olimex | Micro-Water-Pump | 3-12V | 29mm | 3mm | Plastic | 3€ |

Am ales aceasta pompa de apa deoarece functioneaza la tensiuni de 3-12V, se pot folosi tuburi pentru apa de 3mm, iar pretul este foarte scazut.

.

**2.2.Resurse software**

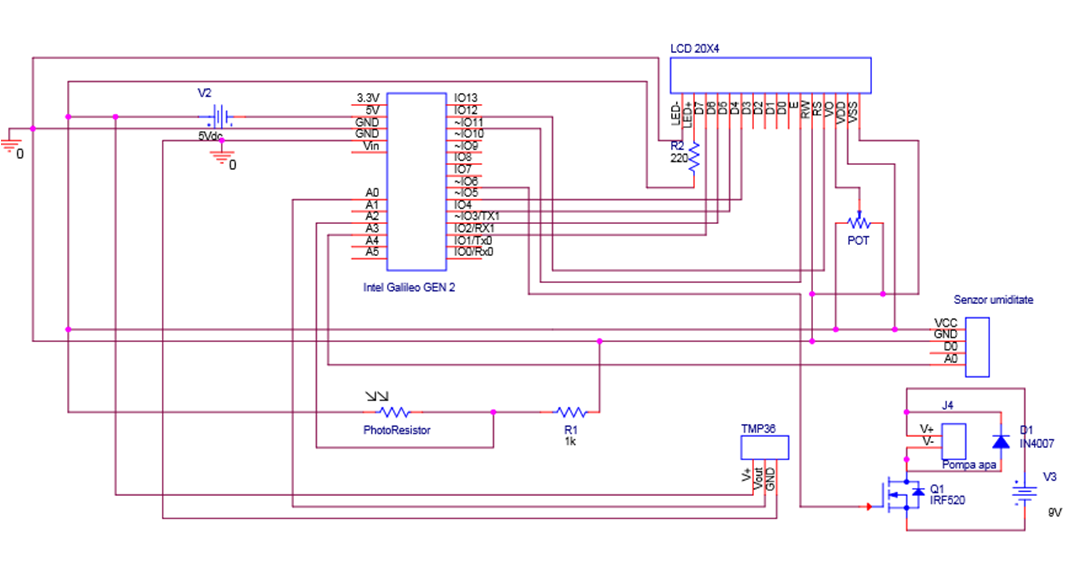
Mediul de dezvoltare integrat Arduino (IDE ) este o aplicație cross-platform scrisă în Java. Acesta este conceput pentru a introduce in arta  programarii nou-veniții nefamiliarizați cu dezvoltarea de software. Mediul de dezvoltare Arduino include un editor de cod cu o serie de caracteristici, cum ar fi evidențierea sintaxei, indentarea automata, și este de asemenea, capabil de compilarea și încărcarea programelor in placa de dezvoltare cu un singur clic. Un program sau cod scris pentru Arduino este numit ” schiță ” (sketch).

Programele Arduino sunt scrise în C sau C++. Pentru executia ciclica a unui program, utilizatorul nu trebuie decat sa defineasca doua functii:  
**- setup ( )** : o funcție ce ruleaza o singură dată, la începutul unui program, care are rolul de a initializa setările programului;  
**- loop ( )** : o funcție ce ruleaza în mod repetat până la intreruperea alimentarii placii de dezvoltare.

**3.Etapele de realizare ale proiectului**

3.1. Implementare hardware

Schema electrica in OrCad:

**Observatii:**

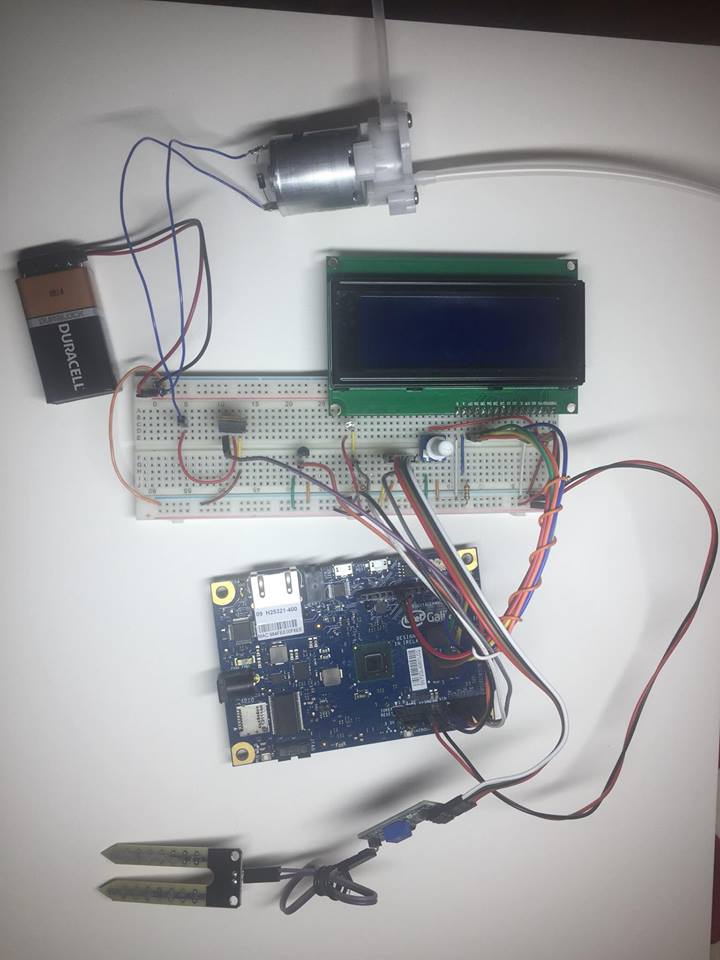
1. Pentru functionarea **pompei de apa** a fost necesara o dioda redresoare 1N4007 conectata in paralel cu pompa. Aceasta se foloseste pentru protectia la inversarea polaritatii si pentru protectia impotriva curentului invers. Astfel dioda permita trecerea curentului intr-o singura directie protejand restul circuitului. Pentru pornirea pompei a fost necesar un tranzistor IRF520 al carei porti va inchide circuitul dintre drena si sursa odatata alimentata. De asemenea, pompa de apa este alimentata de la o baterie de 9V.

2. Pentru reglarea contrastului **LCD**-ului s-a folosit o rezistenta variabila de 10kΩ, iar pentru a porni lumina de fundal s-a folosit o rezistenta de 220Ω pentru limitarea curentului.

3. Pentru functionarea **fotocelulei** este necesara o rezistenta de 1k Ω, care impreuna cu celula fotorezistiva formeaza un divizor de tensiune.

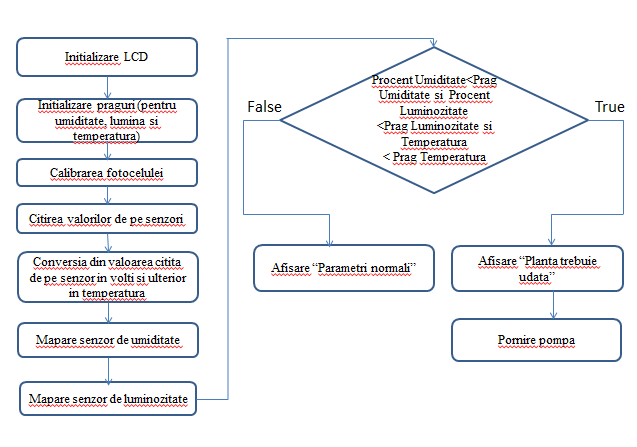
4. In locul **senzorului de umiditate** am fi putut folosi o rezistenta care varia cu cantitatea de apa din sol(cu cat solul era mai umed, cu atat valoarea rezistentei era mai mica). Nu s-a preferat aceasta implementare deoarece am fi pierdut mult la precizie si rezultatele nu ar mai fi fost cele asteptate.

**Implementarea schemei pe breadboard**



**3.2.Implementare software**

Organigrama pentru cod:



Codul in limbajul compatibil Intel Galileo:

//includem libraria specifica LCD

#include <LiquidCrystal.h>

// #define aref\_voltage 3.3

//se initializeaza libraria cu numarul pinilor digitali pe care i-am folosit

LiquidCrystal lcd(12, 11, 5, 4, 3, 2);

//definim constantele corespunzatoare pinilor analogici folositi pentru senzori

//definim constanta corespunzatoare pinului digital folosit pentru pompa de apa

const int TsensorPin=A0; //senzor temperatura

const int UsensorPin=A3; //senzor umiditate

const int LsensorPin=A2; //senzor luminozitate

const int pompaPin=6; //pompa de apa

//initializam corespondentii in procente a valorilor primite de la senzori

int procentUm;

int procentLum;

//definim pragurile in functie de care planta va fi udata sau nu

int pragU=45;

int pragL=25;

int pragT=25;

//initializam variabilele de calibrare a valorii minime si maxime

int sensorLow = 1024;

int sensorHigh = 0;

void setup() {

// se initializeaza pinul pompei ca output

pinMode(pompaPin,OUTPUT);

// setam numarul de coloane si linii pe care le are LCD-ul folosit

lcd.begin(20, 4);

//se realizeaza calibrarea fotocelulei

while (millis() < 5000) {

// inregistreaza valoarea de maxim

LsensorVal = analogRead(LsensorPin);

if (LsensorVal > sensorHigh) {

sensorHigh = LsensorVal;

}

// inregistreaza valoarea de minim

if (LsensorVal < sensorLow) {

sensorLow = LsensorVal;

}

}

// era folosit pentru definirea lui aref si configureaza tensiunea de referinta pentru intrarea analog

// analogReference(0);

}

void loop() {

//se citesc valoarile de pe A0,A3,A2

//si se stocheaza in mai multe variabile

int TsensorVal = analogRead(TsensorPin);

int UsensorVal = analogRead(UsensorPin);

int LsensorVal = analogRead(LsensorPin);

//Temperatura

//se realizeaza conversia dintre valoarea citita de pe ADC in Volti

float voltage = (TsensorVal / 1024.0) \* 5.0;

//Valabil in cazul in care folosim aref\_voltage la 3.3V

//float voltage = (TsensorVal/1024.0)\*aref\_voltage;

//face conversia din volti in grade Celsius

//sensorul se schimba cu 10mv per grad

//in datasheet se precizeaza un offset de 500mV

// ((voltage - 500mV) times 100)

float temp = (voltage - .5) \* 100;

lcd.print("Temp=");

lcd.print(temp);

lcd.print(" C");

//Umiditate

//maparea senzorului de umiditate

procentUm=map(UsensorVal,509,211,0,100);

lcd.setCursor(0,1); //cursorul LCD-ului se pozitioneaza pe coloana 0, linia 1

lcd.print("Umiditate:");

if (procentUm < 100) lcd.print(" ");

if (procentUm < 10) lcd.print(" ");

lcd.print(procentUm);

//lcd.print(UsensorVal);

lcd.print("%");

//Luminozitate

//maparea senzorului de luminozitate

procentLum=map(LsensorVal,0,950,0,100);

lcd.setCursor(0,2); //cursorul LCD-ului se pozitioneaza pe coloana 0, linia 2

lcd.print("Luminozitate:");

if (procentLum < 100) lcd.print(" ");

if (procentLum < 10) lcd.print(" ");

lcd.print(procentLum);

//lcd.print(LsensorVal);

lcd.print("%");

//Pornire/Oprire pompa in functie de pragul de udare stabilit

if(procentUm<pragU && procentLum < pragL && temp<pragT )

{

lcd.setCursor(0,3); //cursorul LCD-ului se pozitioneaza pe coloana 0, linia 3

lcd.print("Planta trebuie udata");

digitalWrite(pompaPin,HIGH);

}

else{

lcd.setCursor(0,3); //cursorul LCD-ului se pozitioneaza pe coloana 0, linia 3

lcd.write("Parametrii normali");

digitalWrite(pompaPin,LOW);

}

delay(1000);

lcd.clear();

}

**Observatii:**

1. Pentru realizarea proiectului, in cod am introdus 3 praguri (de umiditate, de temperatura si de lumina) in functie de care se stabileste udarea plantei. Astfel, am decis ca planta sa fie udata doar pe timpul noptii, atunci cand temperatura este sub 25 de grade. Acest lucru este in primul rand benefic plantei deoarece vara, cand temperaturile ambientale sunt foarte ridicate nu este recomandat sa udam plantele in mijlocul zilei deoarece solul este inca cald si, astfel, apa se va evapora in proportii mari in timpul udarii.

2. Calibrarea fotocelulei se realizeaza pentru a avea o mai buna precizie a rezultatelor. Astfel, pe parcursul primelor 5 secunde se inregistreaza valorile minime si maxime furnizate de senzor si inlocuiesc cu acestea valorile date in foaia de catalog: sensorLow = 1024; sensorHigh = 0.

4. Deoarece am avut o problema cu placa de dezvoltare Galileo Gen 2 (nu ni se mai conecta la usb), aceasta ne-a fost schimbata cu un Galileo Gen 1. In urma trecerii de la o generatie la alta, senzorul de umiditate nu mai arata valori normale, aceasta problema putand fi din cauza aparitiei unui offset. De aceea, am schimbat valorile pentru mapare ale senzorului, cu dezavantajul ca am pierdut din precizie.

3. Pentru optimizarea valorilor furnizate de senzorul de temperatura, am incercat conectarea acestuia la pinul aref\_voltage la 3.3V. Ulterior, uitandu-ne in documentatia placii de dezvoltare am descoperit ca acest pin nu este mapat si din aceasta cauza rezultatele noastre nu erau corecte.

**4.Concluzii**

Acest proiect foloseste atat resurse hardware, cat si software pentru a implementa un sistem de udare pentru o planta, tinandu-se cont de parametrii din mediul ambiant.

Din punctul meu de vedere, acest proiect a reprezentat o oportunitate de a lucra cu o placuta de dezvoltare si desi nu a fost prima oara cand am luat contact cu o astfel de componenta electronica (am facut diverse aplicatii pe Arduino Uno) , mi s-a parut primul lucru practic pe care l-am realizat in aceasta facultate.

De asemenea, proiectul a necesitat o parte de acumulare de noi cunostinte pentru a descoperi cum functioneaza placuta si senzorii de care am avut nevoie. Pe langa aceste aptitudini tehnice dobandite, am invatat sa lucram in echipa, sa impartim sarcinile si sa ne ajutam la depanarea erorilor aparute.

**6. Bibliografie**

1. Arduino Projects Book

2. <http://www.analog.com/media/en/technical-documentation/data-sheets/TMP35_36_37.pdf>

3. <http://enable.excelitas.com/downloads/DTS_vt900seriesdatasheet.pdf>

4. <http://roboromania.ro/produs/senzor-umiditate-sol-soil-hygrometer-humidity-compatibil-arduino/>

5. <http://roboromania.ro/datasheet/4x20-roboromania.pdf>

6. <https://www.olimex.com/Products/Components/Misc/MICRO-WATER-PUMP/>

7. <http://www.onsemi.com/pub_link/Collateral/1N4001-D.PDF>

8. <https://www.arduino.cc/en/ArduinoCertified/IntelGalileo>

9. <https://www.arduino.cc/en/Tutorial/LiquidCrystalDisplay>

10. <https://www.arduino.cc/en/Tutorial/HomePage>