

Intelligent irrigation system

Nume: Pîrvănescu Sandra-Ioana

Grupa: 331CA

Introducere

Proiectul constă într-un sistem de irigare ce poate fi comandat prin Bluetooth. Acesta udă plantele dacă anumite condiții din mediul înconjurător se respectă.

Ideea a pornit de la faptul că, în momentul în care eu împreună cu familia mea plecăm în vacanță, trebuie să rugăm un prieten să vină să ne ude plantele. Pentru acest lucru, am venit cu ideea de a crea un sistem de irigare inteligent, simplu, care poate fi extins. Astfel, proiectul creat poate fi folositor atât pentru mine și familia mea, cât și pentru alte familii ce se află în aceeași situație.

Descriere generală

O schemă bloc cu toate modulele proiectului este următoarea:



Proiectul folosește 2 Arduino Uno R3 ce vor comunica prin intermediul SPI.

Modulul Bluetooth va prelua comenzi de la utilizator și le va transmite către Arduino.

Senzorul de temperatura va măsura temperatura din aer, iar dacă temperatura va depăși o anumită valoare, nu se vor uda plantele pentru a nu exista riscul de ofilire a acestora.

Senzorul de lumină va măsura prezența luminii în mediul înconjurător, plantele fiind udate doar dacă există lumină pentru a facilita fotosinteza. Arduino-ul din dreapta primește comenzi de la cel din stânga dacă se primește cerere de la utilizator prin Bluetooth și dacă cei 2 senzori indică valori ce îndeplinesc condițiile puse.

Se măsoară umiditatea solului pentru a decide dacă mai trebuie udate sau nu plantele. Senzorul de prezență verifică dacă este cineva în preajma plantelor, iar dacă este, pompa de apă își va întrerupe funcționarea. Ledul roșu va fi aprins în momentul în care pompa nu funcționează, iar ledul verde va fi aprins în momentul în care pompa funcționează și plantele sunt udate de către aceasta.

Hardware Design

- 2 Arduino Uno R3
- Modul Bluetooth HC-05
- Senzor temperatură și umiditate DHT11
- Senzor lumină
- Senzor prezență PIR
- Pompă de apă 3-6V
- Senzor umiditate sol
- Led-uri
- Rezistențe
- Tranzistor
- Cablu USB

Arduino-ul cu rol de Master are conectat modulul de Bluetooth HC-05, senzorul de temperatură și umiditate DHT11 și senzorul de lumină.

Arduino-ul cu rol de Slave are conectat senzorul de prezență PIR, pompa de apă, senzorul de măsurare a umidității solului și led-uri, rezistențe și un tranzistor.

Pentru că pompa are nevoie de 200mA pentru a funcționa, am realizat un circuit cu un tranzistor, iar pompa este alimentată printr-un cablu USB conectat la priză.

Led-ul roșu este aprins când pompa nu este alimentată, iar led-ul verde este aprins când pompa este alimentată, adică udă planta. Pompa va fi alimentată pentru 5 secunde, folosind timere și întreruperi. Pompa se va opri dacă senzorul PIR detectează mișcare și NU își va relua procesul de udare a plantelor.

Software Design

- Master

Pe pinii 2, respectiv 3 se află conectați pinii RX, respectiv TX, ai modulului Bluetooth. Pe pinul 7 se află conectat semnalul digital al senzorului de lumină. Pe pinul 8 se află conectat semnalul digital al senzorului DHT11.

În `setup()`, am inițializat modulele, folosind bibliotecile "SPI.h" (pentru comunicarea prin SPI), "SoftwareSerial.h" (pentru modulul Bluetooth) și "DHT.h" (pentru modulul DHT11). De asemenea, am setat pinii 7 și 8 ca INPUT.

În `loop()`, preluat informațiile trimise de senzori. Pentru a trimite către Slave informația primită prin Bluetooth, în mediul înconjurător trebuie să fie lumină, caracterul primit prin Bluetooth să fie "1" și temperatura să fie sub 50 de grade Celsius. Dacă aceste condiții sunt îndeplinite, atunci semnalul este trimis către Slave prin SPI.

- Slave

Pe pinul 2 am conectat senzorul PIR pentru a folosi întreruperi. Pe pinii 7 și 8 se află 2 leduri, verde și roșu. Pe pinul 4 se află conectată pompa.

Pentru comunicarea prin SPI, am realizat o funcție de tratare a întreruperii pentru momentul în care se primește informație. Pentru a contoriza timpul pe care pinul pompei îl petrece pe HIGH, mai exact 5 secunde, am folosit Timer1 și o funcție de tratare a întreruperii. Timer-ul este setat la o frecvență de 2Hz, adică se intră în întrerupere o dată la 0.5 secunde. Astfel, după a 10-a intrare în întrerupere, alimentarea pompei este oprită.

Pentru senzorul PIR, am realizat o funcție de tratare a întreruperii în care se intră numai dacă semnalul emis de PIR trece din LOW în HIGH (adică detectează mișcare). Prin această întrerupere, se oprește alimentarea pompei, indiferent dacă timpul de 5 secunde a expirat sau nu.

În setup(), am setat pinii pentru comunicarea SPI, am configurat timer-ul și pinii senzorilor și led-urilor. De asemenea, am folosit funcția "attachInterrupt()" pentru a activa întreruperea pe pinul PIR-ului. La final, am activat întreruperile globale.

În loop(), am verificat dacă am primit semnal prin SPI. Slave-ul va porni irigarea solului dacă umiditatea solului este sub 60% și dacă a primit prin SPI caracterul "1". Irigarea se oprește dacă timpul a expirat sau s-a detectat mișcare de către PIR. Irigarea se reia doar dacă se mai primește un semnal prin SPI.

Arhiva cu codul poate fi descărcată de la secțiunea "Download".

Rezultate Obținute

Sistemul de irigare inteligentă poate fi folosit în casă și poate fi extins. Acesta poate să aibă mai multe Arduino de tip Slave pentru a se iriga mai multe plante.

Condițiile de irigare se pot schimba cu ușurință, fiind posibilă adaptarea acestuia nevoilor utilizatorilor.

O imagine cu rezultatul final al proiectului este următoarea:



Demo

Download

În arhivă se află codul de Arduino pentru funcționarea proiectului, atât pentru Arduino ce are rol de Master, cât și pentru cel ce are rol de Slave.

[intelligentirrigationsystem.zip](#)

Bibliografie/Resurse

Laburi PM: Laboratoare 2, 3, 4, 5.

<https://docs.arduino.cc/tutorials/generic/introduction-to-the-serial-peripheral-interface>

<https://www.circuitbasics.com/how-to-set-up-spi-communication-for-arduino/>

<https://makersportal.com/blog/2019/5/27/arduino-interrupts-with-pir-motion-detector>

<https://projecthub.arduino.cc/arcaegecengiz/using-dht11-12f621>

[Export to PDF](#)

From:

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

Permanent link:

<http://ocw.cs.pub.ro/courses/pm/prj2023/gpatru/sandra.pirvanescu>



Last update: **2023/05/31 09:12**