

## Radar

**Autor:** ILIESCU Lucian-Marius

Un radar/sonar care va afisa pe un grafic zonele de activitate:

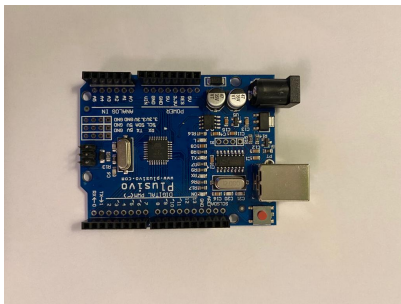
- Va roti servomotorul pe care se afla senzorul ultrasonic
- Senzorul va capta datele si le va da placutei
- Placuta va trimite datele catre softul care prelucreaza imaginile
- Softul va afisa pe ecran graficul cu verde si va arata cu rosu unde s-a gasit ceva (similar unui sonar)
- In cazul in care un obiect se afla foarte aproape de senzor, acesta va declansa buzzerul

Schema bloc

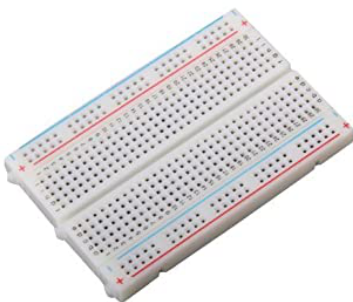


Proiectare Hardware  
Componente necesare

- Placa Arduino UNO



- Breadboard



- Fire mama-tata
- Fire tata-tata
- Senzor ultrasonic HC-SR04 - pentru a detecta obiectele



- Servomotor - pentru a roti senzorul



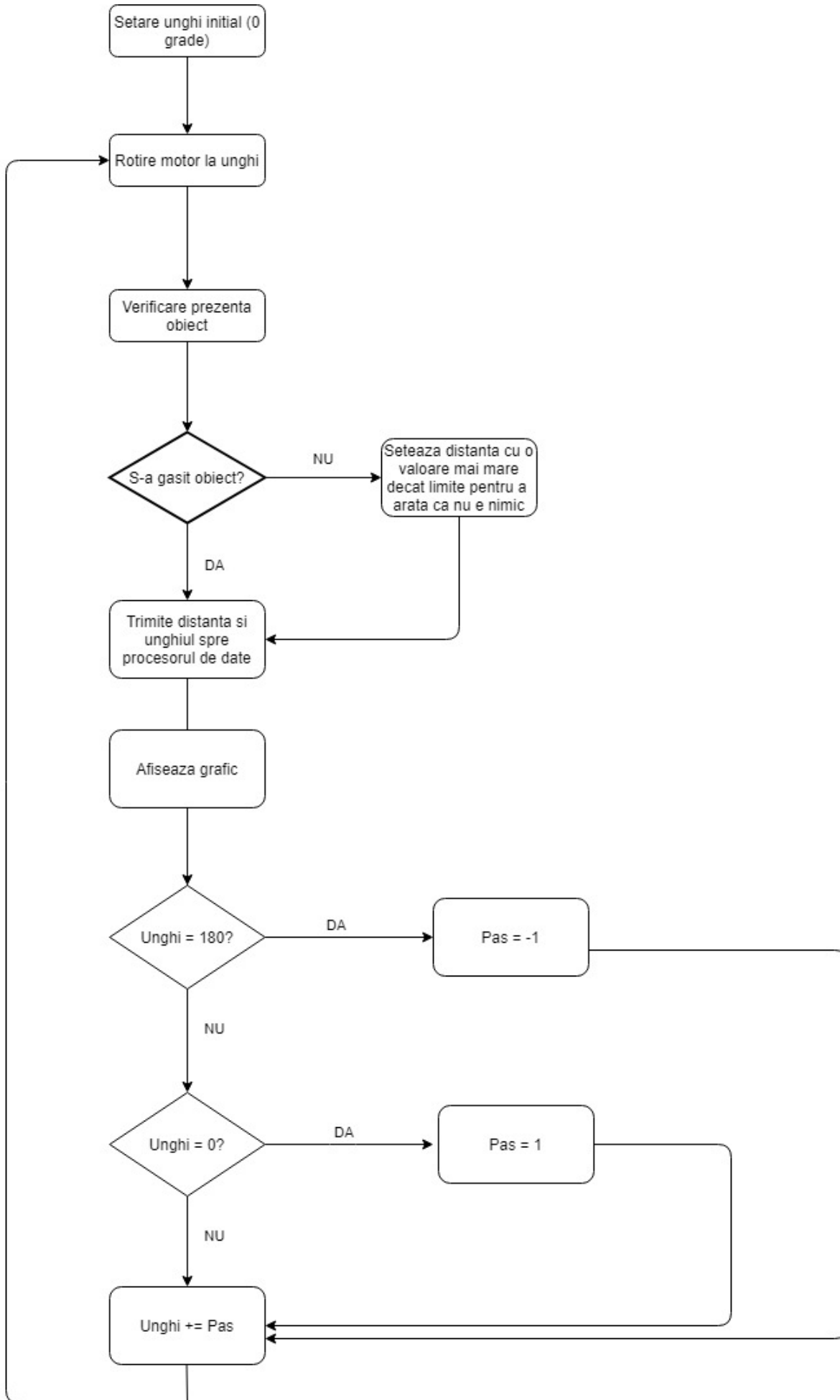
- Buzzer - pentru a anunta prezenta obiectelor apropiate



Schema Electrica



Software Design



Partea de soft este implementata din doua parti:

1. Preluarea datelor de la senzori
2. Procesarea/Afisarea datelor

1) Programul Arduino are urmatoarele roluri:

- Roteste servomotorul
- Preia distanta de la senzorul ultrasonic
- Suna alarma cand se afla un obiect foarte aproape
- Trimite datele (unghiul/distanta) spre procesare

Codul este scris folosind C/C++.

Programul va efectua doua bucle care vor reprezenta rotatiile radarului de la 0-180° si invers. La fiecare iteratie programul va primi distanta obtinuta de senzor si o va afisa pe ecran impreuna cu unghiul motorului. Datele afisate pe interfata seriala sunt trimise spre procesare.

2) Programul Processing are urmatoarele roluri:

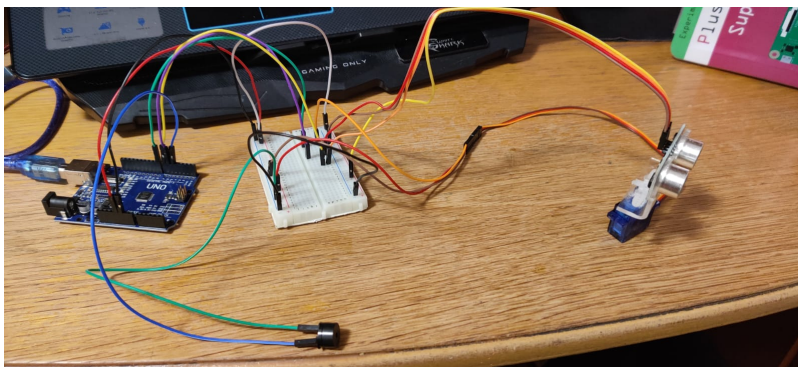
- Primeste datele de la Arduino
- Le prelucreaza sub forma unui grafic

Codul este scris folosind Java si biblioteca Processing.

Se porneste comunicatia cu portul serial unde e conectata placuta arduino. Programul va desena intai scheletul graficului, trasand arcele de cerc si liniile radarului. Primeste datele de pe seriala si daca sunt valide, acesta va adauga obiectul in lista cu obiecte gasite. Se afiseaza toate obiectele gasite folosind functia drawObjects. De asemenea, se va afisa unghiul si distanta ultimului obiect gasit. Fiecare obiect are un camp de timp, care ii determina durata lui in lista de obiecte si la fiecare iteratie scade durata acestuia, cand ajunge la 0 se va scoate obiectul din lista de obiecte.

Rezultate

Proiectul functioneaza conform asteptarilor, dar sunt cateva aspecte care nu sunt perfecte.



Un demo video se poate gasi aici:

[https://www.youtube.com/watch?v=McZXUy0mHV0&ab\\_channel=lucianiliescu](https://www.youtube.com/watch?v=McZXUy0mHV0&ab_channel=lucianiliescu)

Observatii

Senzorul nu poate oferi o precizie exacta, putand sa apara mai multe erori de masurat, cum ar fi reflexia undelor pe suprafete diferite (unele suprafete gestionand reflexia intr-un mod diferit).

## Concluzii

Proiectul mi s-a parut interesant si a fost distractiv sa lucrez la el. M-a invatat cum se lucreaza cu pinii si cum se conecteaza componentele/logica lor de conectare.

## Bibliografie/Resurse

1. Idee de proiect: <https://www.udemy.com/course/arduino-build-arduino-practical-projects/>
2. Resurse ajutatoare: <https://howtomechatronics.com/projects/arduino-radar-project/>
3. Idee pentru procesare: <https://processing.org/>
4. Cod sursa: <https://github.com/lucian606/Radar>
5. [radar.pdf](#)

Link catre pagina: <https://ocw.cs.pub.ro/courses/pm/prj2021/abirlica/radar>

From:

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

Permanent link:

<http://ocw.cs.pub.ro/courses/pm/prj2021/abirlica/radar>

Last update: **2021/06/01 23:31**

