

# Rubik's Cube Solver

## Introducere

Scopul proiectului este crearea unui robot ce poate manipula un cub Rubik precis și rapid, pentru a-l rezolva, amesteca sau transforma în orice stare dorită. Acesta poate fi util în mai multe circumstanțe precum:

- Amestecarea autonomă în timpul competițiilor: În timpul competițiilor este necesar ca multe cuburi să fie amestecate identic. Oamenii sunt predispuși la erori și lenți, însă robotul poate realiza această sarcină perfect de fiecare dată, și mai rapid decât un om.
- Rezolvarea cubului
- Amestecarea autonomă a cuburilor într-un mod precis pentru a realiza anumite [imagini](#).

## Descriere generală



Robotul va interacționa cu un panou de comandă, de unde va putea primi comenzi precum:

- Mutări individuale
- O anumită configurație la care trebuie să ajungă
- Rezolvare

Calcululele efective pentru identificarea mutărilor necesare vor fi efectuate de către panoul de comandă, microcontrollerul ESP32 neavând suficientă putere de calcul.

## Hardware Design

- [Modul ESP32](#)
- [Motor stepper 28BYJ-48](#) x5
- Breadboard
- Fire
- <https://www.emag.ro/breakout-conector-usb-2-0-tip-c-pololu-2-x-1-5-x-0-5-cm-00002993/pd/DSCSFY-YBM/>
- Diverse piese printate 3D pentru interacțiunea motoarelor cu cubul și scheletul robotului



# Software Design

Programul va fi scris in Arduino IDE. Acesta va comanda motoarele stepper prin biblioteca Stepper și va comunica prin serială cu un calculator pentru a primi comenzi. Rezolvarea efectivă a cubului va fi realizată de un program Python care va utiliza biblioteca [aceasta](#).

Descrierea codului aplicației (firmware):

- mediu de dezvoltare (if any) (e.g. AVR Studio, CodeVisionAVR)
- librării și surse 3rd-party (e.g. Procyon AVRlib)
- algoritmi și structuri pe care plănuți să le implementați
- (etapa 3) surse și funcții implementate

## Rezultate Obținute

Care au fost rezultatele obținute în urma realizării proiectului vostru.

## Concluzii

## Download

O arhivă (sau mai multe dacă este cazul) cu fișierele obținute în urma realizării proiectului: surse, scheme, etc. Un fișier README, un ChangeLog, un script de compilare și copiere automată pe uC crează întotdeauna o impresie bună 😊.

Fișierele se încarcă pe wiki folosind facilitatea **Add Images or other files**. Namespace-ul în care se încarcă fișierele este de tipul **:pm:prj20??:c?** sau **:pm:prj20??:c?:nume\_student** (dacă este cazul).

**Exemplu:** Dumitru Alin, 331CC → **:pm:prj2009:cc:dumitru\_alin**.

## Jurnal

Puteți avea și o secțiune de jurnal în care să poată urmări asistentul de proiect progresul proiectului.

## Bibliografie/Resurse

Listă cu documente, datasheet-uri, resurse Internet folosite, eventual grupate pe **Resurse Software** și **Resurse Hardware**.

[Export to PDF](#)

From:

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

Permanent link:

<http://ocw.cs.pub.ro/courses/pm/prj2024/rrusu/matei.hriscu>



Last update: **2024/05/19 14:37**