

Touring Car

Introducere

- Proiectul constă în construirea unei mașini de tip rover 4x4, care poate fi controlată de la distanță printr-o aplicație pe telefon și permite manevrarea în toate cele 4 direcții. Mașina prezintă un mecanism inteligent de detecție a obstacolelor frontale: în cazul în care mașina s-ar lovi în partea din față de un obiect, aceasta blochează orice încercare a utilizatorului de a se mișca drept înainte, forțându-l să schimbe direcția de mers, pentru a evita impactul.
- Scopul proiectului este de a simula un prototip de mașină cu control la distanță, care să se folosească și de un senzor, asemenea mașinilor reale moderne.
- Ideea de la care am pornit a fost să creez un proiect practic, pe care l-aș putea folosi și în afara cursului de PM, având ca inspirație mașinile cu telecomandă pentru copii.
- Aș spune despre proiect că este util pentru mine și ceilalți pentru că îmbină design-ul și folosirea mai multor electronice cu ceva distractiv, putând fi folosit drept exemplu pentru cineva care nu are experiență cu microcontrollere, ca motivație pentru a realiza ceva practic.

Descriere generală



Mașina așteaptă comenzi prin Bluetooth de pe smartphone. Odată primită o comandă, aceasta ajunge prin HC-05 la microcontroller, care o traduce în semnale către driver-ul L298N (PWM pentru viteză și GPIO pentru direcție), punând cele 4 motoare în mișcare. Pentru viraj, motoarele de pe o parte se rotesc mai repede decât cele de pe cealaltă.

Senzorul ultrasonic rulează în permanentă în fundal. Dacă detectează un obstacol prea aproape în față, microcontrollerul ignoră automat orice comandă de mers înainte, obligând utilizatorul să schimbe direcția. Când drumul e liber, mașina răspunde din nou normal la toate comenzile.

Hardware Design

Listă de componente:

- 1 x Microcontroller ATmega328P Xmini Xplained

- 1 × Kit șasiu 4WD (plăci bază + 4 motoare DC cu reductoare + 4 roți)
- 1 × Driver motoare L298N
- 1 × Modul Bluetooth HC-05
- 1 × Senzor ultrasonic HC-SR04
- 4 × baterii AA / 2 × Acumulatori Li-Ion 18650 (în funcție de nevoia de putere, încă nu sunt sigur până nu asamblez proiectul)
- 1 × Mini-breadboard
- n × Jumpere

Aici puneți tot ce ține de hardware design:

- listă de piese
- scheme electrice (se pot lua și de pe Internet și din datasheet-uri, e.g. <http://www.captain.at/electronic-atmega16-mmc-schematic.png>)
- diagrame de semnal
- rezultatele simulării

Software Design

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/prj2026/ciprian.popescu0411/235141> 

Last update: **2026/05/10 15:03**