

# Lego Technic RC mod

Nume: Baldovin Răzvan-Mihai-Marian

Grupa: 332CA

## Introducere

Odată cu explozia în popularitate a Formulei 1 de acum câțiva ani, am început și eu să urmăresc acest sport. Încă de la început m-a atras echipa McLaren și în special Lando Norris, care, deși nu câștiga curse atunci când am început eu să urmăresc F1, a reușit între timp să urce pe prima treaptă a podiumului. Chiar dacă McLaren nu era cea mai puternică echipă în acea perioadă, am rămas un fan al lor.

În 2022, LEGO a lansat un set Technic cu monopostul McLaren de Formula 1, la scară 1:8, semi-funcțional, în stilul clasic Technic. Am economisit bani, iar spre finalul anului am reușit să-l cumpăr. Asamblarea a durat două zile (cu tot cu pauze), urmate de săptămâni întregi în care pur și simplu am admirat cum o echipă de ingineri a reușit să creeze acest model de la zero, DIN LEGO.

Dar încă din secunda în care am terminat construcția, m-am gândit: „Are suspensii, volan funcțional, motor V6 care învârte roțile... Scheletul e acolo, ar fi mișto să o pot conduce!” Din păcate, am rămas mult timp doar cu gândul acesta, fiind descurajat de lipsa experienței în „ingineria adevărată”. Știam să scriu cod, dar nu aveam nicio idee cum să transform această idee în realitate.

Și aici intervine materia PM. Când am aflat că trebuie să realizăm un proiect cu un microcontroller și că avem libertatea să alegem orice temă, atâta timp cât îndeplinește cerințele minime, am știut imediat ce voi face. A sosit momentul să-i dau viață mașinii.

Acum că știți povestea din spatele deciziei, e timpul să vedem și implementarea.

## Descriere generală



Proiectul are ca și controller un **ESP32-S3-N16R8**, care va avea următoarele funcții:

- Va prelua comenzi de la utilizator (precum înainte, înapoi, stânga, dreapta)
- Va comunica cu servomotorul, acționând comenzile de direcție (stânga și dreapta)
- Va comunica cu driverul de motor, acționând comenzile de propulsie (înainte și înapoi)

Comunicația se va face astfel:

- ESP32 → driver de motor (GPIO + PWM)
- ESP32 → servomotor (PWM)
- driver de motor → motor (PWM)

Din moment ce comenzile utilizatorului vor genera schimbări de stare, vom folosi și întreruperi.

Astfel, conceptele din laborator utilizate sunt:

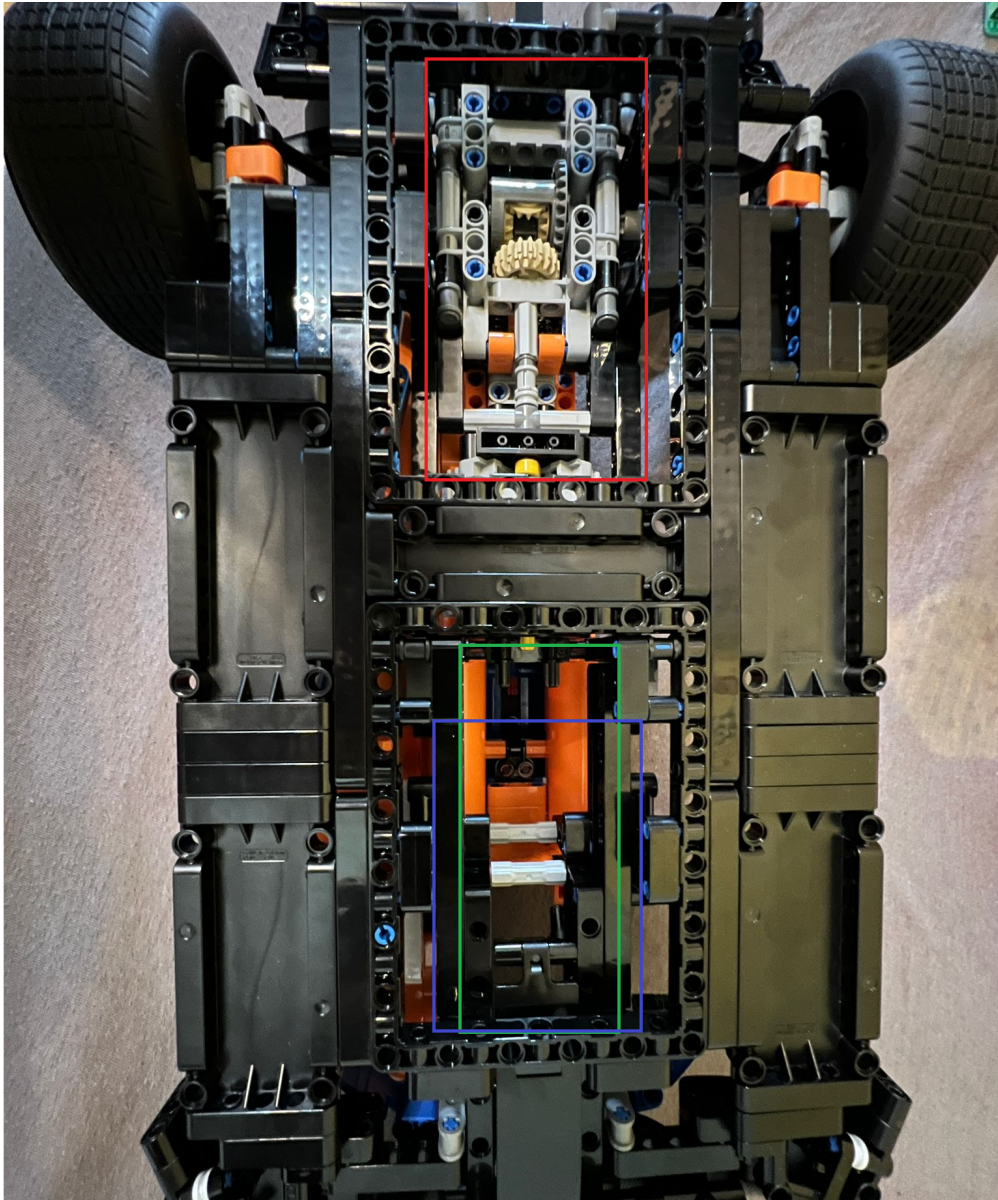
- GPIO
- Întreruperi
- PWM

## Hardware Design

Piese au fost alese cu următoarele caracteristici în gând:

- Dimensiunea spațiului util
- Caracteristicile Lego-ului (pentru a putea mișca Lego-ul)

Astfel, înainte să detaliem piesele pe care le-am achiziționat, hai să detaliem puțin cu ce lucrăm



Avem aici 2 zone in care putem lucra:

- Rosu - 10.4 x 5.5 x 3.2 cm (lungime x latime x inaltime)
- Verde - 10.4 x 4.0 x 6.0 cm
- Albastru - 8.0 x 5.5 x 6.0 cm

(In functie de nevoie, putem alege intre verde si albastru ca spatiu (ambele sunt valide in situatia de fata)



## Specificatii Lego:

- Dimensiuni: 65.0 x 27.0 x 13.0 cm
- Greutate: 1.4kg
- Diametru roti: 8.0 cm

Ca sa stabilim viteza cu care ar trebui sa se invarta motorul am facut urmatoarele calcule

1. Calculam circumferinta rotii
  - circumferinta =  $\pi * \text{diametru} = \pi * 0.08 \text{ m} = 0.2513 \text{ m}$
2. Calculam numarul de rotatii per minut al rotii
  - Presupunem o viteza de 10km/h
  - $10 \text{ km/h} = 2.78 \text{ m/s}$
  - RPS roata = viteza / circumferinta =  $2.78 \text{ m/s} / 0.2513 \text{ m} \approx 11.06 \text{ RPS}$
  - RPM roata = RPS roata \* 60 s =  $11.06 * 60 = 664 \text{ RPM}$
3. Calculam numarul de rotatii per minut al motorului
  - Observam ca o rotatie a axului de motor reuseste sa invarta aproximativ jumatate de roata
  - RPM motor = RPM roata \* 2 =  $664 \text{ RPM} * 2 = 1328 \text{ RPM}$

Totusi, viteza motorului nu este totul, ne trebuie si destul cuplu cat sa putem pleca de pe loc

1. Calculam forta de rezistenta la rulare
  - F rezistenta = coef de rezistenta la rulare \* masa \* gravitatie =  $0.01 * 1.4\text{kg} * 9.81\text{m/s}^2 = 0.137\text{N}$
  - coeficientul de rezistenta la rulare pentru cauciuc am gasit sa fie 0.01
2. Calculam Cuplul minim necesar
  - cuplu = F rezistenta \* raza rotii =  $0.137\text{N} * 4\text{cm} = 0,0055 \text{ N*m}$
  - transformam din N\*m in kg \* cm
  - $0.0055 \text{ N*m} \approx 0.056 \text{ kg * cm}$
3. Luam ca marja de eroare un factor de 4x, care sa tina cont de flexarea, rezistenta si frecarea pieselor Lego
  - cuplu final = cuplu minim \* 4 =  $0.056 \text{ kg * cm} * 4 \approx 0.2 \text{ kg * cm}$

## Bill of Materials

Piesa	Cantitate	Pret	Link
ESP32-S3-N16R8	1	5.75 (96% off)	<a href="#">AliExpress</a>
Driver de motor TB6612FNG	1	24.99	<a href="#">Optimus Digital</a>
Modul DC-DC Step-Down LM2596	1	16.49	<a href="#">Optimus Digital</a>
Motor brushless FK-180SH-3244	1	14.99	<a href="#">Optimus Digital</a>
Servomotor MG90S	1	19.33	<a href="#">Optimus Digital</a>
Suport de Baterii 2 x 18650	1	3.99	<a href="#">Optimus Digital</a>
18650 LI-ION 3450mA 8A	4	26.18 * 4	<a href="#">eMAG</a>

## Wiring



## Pinout

### ESP32

Pin ESP	Componenta legata	Pin componenta	Protocol	Funcție
5Vin	Modul DC-DC Step-Down	Vout+	Power	Sursa de 5V
GND	Modul DC-DC Step-Down	Vout-	Power	Ground
GPIO10	Servomotor	IN	PWM	Controleaza servomotorul
GND	Servomotor	GND	Power	Ground
GPIO4	Motor Driver	AIN2	GPIO	Controleaza directia motorului
GPIO5	Motor Driver	AIN1	GPIO	Controleaza directia motorului
GPIO6	Motor Driver	PWMA	PWM	Controleaza viteza motorului
GPIO7	Motor Driver	STBY	GPIO	Activeaza motor driverul
GND	Motor Driver	GND	Power	Ground

### Motor Driver

Pin driver	Componenta legata	Pin componenta	Protocol	Funcție
PWMA	ESP32	GPIO6	PWM	Controleaza viteza motorului
AIN2	ESP32	GPIO4	GPIO	Controleaza directia motorului
AIN1	ESP32	GPIO5	GPIO	Controleaza directia motorului
STBY	ESP32	GPIO7	GPIO	Activeaza motor driverul
GND	ESP32	GND	Power	Ground
VM	Modul DC-DC Step-Down	Vout+	Power	Alimentare motor
VCC	Modul DC-DC Step-Down	Vout+	Power	Alimentare logica
A1	Motor	VCC	Power	Activeaza motorul
A2	Motor	GND	Power	Ground

## Software Design

Initial am vrut sa fac o aplicatie mobila pentru a utiliza BLE (care este ceva mai responsiv), insa avand acces doar la iPhone si neavand Mac, nu am putut gasi metode "oficiale" de a dezvolta aplicatii (deoarece, din cate am gasit pe net, iti trebuie Xcode care se gaseste doar pe sisteme MacOS)

Astfel, am ales sa folosesc a doua cea mai buna optiune, care este spre norocul meu si universala, un web server hostat local care controleaza masina (design-ul a fost creat cu urmatorul gand, ca sa controlezi masina trebuie sa o vezi/urmaresti, deci ar avea mai mult sens sa fie orientata catre un telefon, in loc de laptop)

[Link GitHub](#)

Codul a fost dezvoltat in Arduino IDE.

Am folosit urmatoarele biblioteci:

- ESP32Servo.h - Pentru a controla servomotorul prin functii predefinite
- WiFi.h - Pentru a permite conexiunea la Wi-Fi
- WebServer.h - Pentru a crea un server web local

Am creat o pagina web destul de elementara, care are doua slidere:

- Cel din stanga are rolul unui volan, astfel se ocupa de virat stanga-dreapta
- Cel din dreapta are rolul de a propulsa/opri masina (inainte-inaoi)

Flow-ul programului:


- Se initializeaza conexiunea WiFi
- Se initializeaza serverul web (pe o adresa locala, deci trebuie sa ne aflam in aceeași retea)
- Asteapta input de la utilizator prin serverul web
- Comenzile, sunt transformate in semnale PWM catre componentele respective:
  - Slider stanga → servomotor (intre 0 si 180)
  - Slider dreapta → motor (intre -255 (maxim inapoi) si 255 (maxim inainte))

## 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

24.04.2025-26.04.2025 - Am inceput researchul, cautand piese potrivite pentru proiect. Am pornit cu

aproape 0 cunostinte despre proiectul pe care vreau sa il fac asa ca a trebuit sa fac ore intregi de research (cateva dintre calculele si gandurile importante sunt reprezentate in sectiunde de Hardware Design)

26.04.2025 - Am dat comanda finala de piese

07.05.2025 - Au ajuns toate piesele

08.05.2025 - Cu ajutorul unui prieten, am printat 3D niste conectori servomotor-Lego axle si motor-Lego axle

18.05.2025 - Folosind diverse datasheet-uri si videouri pe youtube, am reusit sa prototipez hardwareul care va da viata masinii. Am creat un cod simplu care ruleaza un PWM de la 0 la 255 ca sa verific, si am decis sa sudez majoritatea circuitului pentru a ma asigura ca nu voi pierde progresul facut.

19.05.2025 - Am incercat sa conectez hardware-ul la masina din Lego, sii... nu a functionat :(

19.05.2025 - Am identificat urmatoarele probleme:

- Servomotorul (SG90) nu era destul de puternic pentru steering
- Motorul se rotea in gol, deoarece conectorul nu statea fix pe acesta
- Devboard-ul este prea mare ca sa incapa in Lego

21.05.2025 - Am gasit urmatoarele solutii:

- Am achizitionat un servomotor mai puternic (MG90S)
- Am printat un conector 3D reglabil, pentru a putea strange mai mult ca motorul sa poata transmite putere la Lego
- Daca scot ESP32-ul din Devboard, si schimb din fire tata-tata in fire tata-mama, poate incapea totul

23.05.2025 - Din cauza unei singure piese din Lego, nu incapea motorul cu noul conector, am stat undeva la 2 ore pentru a desface masina (si documenta, pentru a o reface la loc :) ), urmata de a muta piesa mai in spate pentru a crea spatiu, iar apoi reasamblat.


23.05.2025 - Din nou, am observat ca motorul se invarte in gol, chit ca am strans mai mult piesa 3D.

24.05.2025 - Am incercat sa adaug diferite bucati subtire de material intre motor si conector ca sa incerc sa conectez mai bine si sa adaug frecare (poate-poate merge). Din pacate nu a mers, tot se invarte in gol.

28.05.2025 - Am desfacut lipiturile vechi pentru a putea inlocui cu fire tata-mama pentru a putea scapa de dev board.

28.05.2025 - Am incercat sa actionez MG90S, dar nu am avut succes, am schimbat temporar inapoi la SG90 pana gasesc o solutie mai buna

29.05.2025 - Am incercat sa mai printez inca un conector pentru motor. (fara succes, problema finala este puterea :( )

29.05.2025 - Am folosit un alt motor (TT motor), si ARE DESTULA PUTERE SA MISTE MASINA 

29.05.2025 - Update-uri la cod, pagina, si finalizare proiect pentru prezentare

## 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/prj2025/mdinica/razvan.baldovin>



Last update: **2025/05/30 07:37**