



**UNIVERSITATEA POLITEHNICA DIN  
BUCUREŞTI**  
**FACULTATEA DE INGINERIE MECANICĂ ŞI  
MECATRONICĂ**



**Programul de studii: Mecatronica**

# **PROIECT DE DIPLOMĂ**

Coordonator științific,

Conf. Dr. Ing. Bogdan GRĂMESCU

Absolvent,

Alexandru-Cristian TROACĂ

**Bucureşti**

**2022**



UNIVERSITATEA POLITEHNICA DIN  
BUCUREŞTI  
FACULTATEA DE INGINERIE MECANICĂ ŞI  
MECATRONICĂ



Programul de studii: Mecatronică

# SISTEM MECATRONIC MOBIL PENTRU INSPECȚII AUTO

Coordonator științific,

Conf. Dr. Ing. Bogdan GRĂMESCU

Absolvent,

Alexandru-Cristian TROACĂ

București

2022

# Cuprins

<b>CAPITOLUL 1. INTRODUCERE .....</b>	<b>6</b>
1.1    MOTIVAȚIA ALEGERII TEMEI “SISTEM MECATRONIC MOBIL PENTRU INSPECȚII AUTO”	6
1.2    PREZENTAREA TEMEI.....	6
1.3    SCURT ISTORIC .....	7
<b>CAPITOLUL 2. STADIUL ACTUAL .....</b>	<b>8</b>
2.1    ROBOTI MOBILI / SISTEME MOBILE GENERALE .....	8
2.2    TIPURI DE ROBOȚI MOBILI .....	9
2.2.1    Roboți staționari (manipulatori) .....	9
2.2.2    Roboți terestre .....	10
2.2.3    Roboți aerieni .....	16
2.2.4    Roboți acvatici .....	16
2.3    SOLUȚII EXISTENTE.....	17
2.3.1    Robot pentru inspecții cu șenile GPK-32 .....	17
2.3.2    Robotnik Guardian .....	18
2.3.3    CoroWare Explorer Ex-D.....	19
2.3.4    Dr. Robot Sentinel3 .....	20
2.3.5    InspectorBots Mega Bot .....	21
<b>CAPITOLUL 3. PREZENTARE SOLUȚIA PROPRIE.....</b>	<b>22</b>
<b>CAPITOLUL 4. MEMORIU TEHNIC.....</b>	<b>25</b>
4.1    PROIECTAREA STRUCTURII MECANICE .....	25
4.1.1    Proiectarea carcasei .....	25
4.1.2    Proiectarea suporturilor pentru motoare .....	30
4.1.3    Proiectarea suport senzori ultrasonici.....	33
4.1.4    Proiectare suport camera .....	34
4.1.5    Proiectare suport acumulator .....	38
4.1.6    Realizare roată omnidirecțională .....	38

4.2	ALEGEREA COMPONENTELOR ELECTRICE .....	40
4.3	ALEGEREA ELEMENTELOR DE ACȚIONARE .....	41
4.3.1	Servomotor – Dynamixel XM430-W210-T .....	41
4.3.2	Driver motor .....	44
4.3.3	Testarea motoarelor .....	46
4.4	MICROCONTROLER .....	50
4.5	ELEMENTE SENZORIALE.....	53
4.5.1	Senzori ultrasonici HC-RS04 .....	53
4.5.2	Testare senzor ultrasonic .....	56
4.5.3	Camera Raspberry Pi High Quality Camera.....	59
4.5.4	Testare camera .....	60
4.6	ALIMENTARE .....	63
4.6.1	Baterie.....	63
4.6.2	Surse coborâtoare de tensiune .....	64
<b>CAPITOLUL 5.</b>	<b>TESTARE ȘI EXPERIMENTARE A SISTEMULUI ROBOTIC ....</b>	<b>68</b>
5.1	ASAMBLARE .....	68
5.2	MOD DE FUNCȚIONARE AUTONOM .....	71
5.3	MOD DE FUNCȚIONARE MANUAL .....	72
5.4	STUDIU SOLICITARE ÎNCOVOIÈRE PLACĂ INFEROARĂ.....	76
5.4.1	Cazul I - Încastrare simplă plan frontal, $m = 3 \text{ kg}$ .....	77
5.4.2	Cazul II – Încastrare dublă plan frontal, $m = 3 \text{ kg}$ .....	82
<b>CAPITOLUL 6.</b>	<b>EFICIENTĂ ECONOMICĂ .....</b>	<b>86</b>
6.1	CALCULAREA CHELTUIELILOR MATERIALE.....	86
6.2	CALCULAREA CHELTUIELILOR CU RESURSELE UMANE.....	88
6.3	CALCULUL AMORTIZĂRII ECHIPAMENTELOR.....	89
6.4	CALCULUL PROFITULUI .....	89
<b>CAPITOLUL 7.</b>	<b>CONCLUZII .....</b>	<b>90</b>

DICȚIONAR EXPLICATIV DE TERMENI ȘI ABREVIERI.....	92
BIBLIOGRAFIE .....	93
ANEXE .....	ERROR! BOOKMARK NOT DEFINED.

# **Capitolul 1. Introducere**

## **1.1 Motivația alegerii temei “Sistem mecatronic mobil pentru inspecții auto”**

Roboții umanoizi, roverele fără pilot, roboti tip animale de companie pentru divertisment, dronele, sunt exemple excelente de roboți mobili. Aceștia pot fi distinși de alți roboți prin capacitatea lor de a se deplasa în mod autonom, cu suficientă inteligență pentru a reacționa și a lua decizii pe baza informațiilor primite din mediul înconjurător. Roboții mobili trebuie să dispună de senzori pentru a primi date de intrare, de o modalitate de decodare a acestor date și de o modalitate de a efectua acțiuni (inclusiv acțiuni proprii - mișcarea proprie).

În zilele noastre, există roboți mobili care pot merge, alerga, sări sau care se târasc asemenea organismelor biologice. Au apărut mai multe domenii ale roboticii, cum ar fi roboții mobili pe roți, roboții cu picioare, roboții zburători, viziune robotică, inteligență artificială, acestea îmbină diferite domenii tehnologice, cum ar fi mecanica, electronica și informatica.

Aceste noi tendințe sunt conduse de inteligență artificială, conducere autonomă, comunicarea în rețea, munca cooperativă, nanorobotica, roboți umanoizi prietenoși, interacțiune sigură om-robot, precum și exprimarea și perceperea emoțiilor.

În plus, aceste noi tendințe sunt aplicate în diferite domenii, cum ar fi medicină, asistență medicală, sport, industrie și distribuția de bunuri. Aceste tendințe își vor continua evoluția în anii următori.

## **1.2 Prezentarea temei**

În prezent pentru a putea observa ceea ce se ascunde sub o mașină sau pentru a face o simplă inspecție este necesară folosirea unui elevator sau a unei gropi de inspecții. Aceste două soluții implica automat deplasarea cu mașina la un service dotat cu aceste echipamente.

Pentru a verifica partea inferioară a mașinii se poate folosi un robot mobil de dimensiuni mici, controlat de la distanță, echipat cu o camera și capabil de a transmite imaginea în timp real pe ecranul unui laptop. Astfel nu mai este necesara deplasarea la un service și folosirea unui elevator, mașina putând fi verificată în orice moment, indiferente de locul în care se află.

În continuare, în aceasta lucrare se vor prezenta etapele necesare proiectării și construirii acestui robot.

### 1.3 Scurt istoric

**1966–72** Institutul de Cercetare Stanford a dezvoltat robotul Shakey, care avea o cameră foto, un sonar, senzori de detectare a coliziunii și o legătură wireless. Acesta a fost primul robot mobil de uz general capabil să detecteze și să interpreteze mediul înconjurător și apoi să navigheze singur printre obstacole.[1]

**1970** Uniunea Sovietică a reușit să aselenizeze cu succes pe suprafața Lunii primul rover lunar Lunokhod 1, care a fost controlat de la distanță de pe Pământ și care avea la bord mai multe camere și alți senzori. În 301 zile de funcționare, roverul a parcurs aproximativ 10 km, a returnat peste 25.000 de imagini și a efectuat mai multe analize ale solului.[2]

**1973** Uniunea Sovietică a amplasat un al doilea rover lunar, Lunokhod 2, pe suprafața Lunii. În timpul misiunii de 4 luni, roverul a parcurs o distanță de 39 km, un record pentru distanța de deplasare în afara Pământului, care a rezistat până în 2014.[3]

**1976** Navele spațiale fără echipaj uman ale NASA Viking 1 și Viking 2 (fiecare dintre ele formată dintr-un orbitor și un modulator) au intrat pe orbita lui Marte, iar modulatoarele au aterizat pe suprafața planetei câteva zile mai târziu.[4]

**1979** The Stanford Cart (modelul inițial introdus în 1962) a devenit capabil să navigheze pe bază de viziune printr-un traseu cu obstacole.[5]

**1986** Echipa lui Ernst Dieter Dickmanns a dezvoltat mașina-robot VaMoRs, care este capabilă să se depleteze singură pe străzi fără trafic, atingând viteze de până la 90 km/h.[6]

**1996-97** La bordul navei Mars Pathfinder, NASA a trimis pe Marte un rover, Sojourner, care a fost comandat de pe Pământ, dar care a reușit să se depleteze în mod autonom de-a lungul traectoriei predefinite și să evite situațiile periculoase de pe parcurs.[7]

**2000** FDA a aprobat utilizarea robotului chirurgical da Vinci în Statele Unite.[8]

**2002** A fost lansat un prim model al robotului aspirator de podea Roomba pentru uz casnic.[9]

**2004** Roverii gemeni Spirit și Opportunity au aterizat pe Marte. Roverul Spirit a rămas blocat în 2009, dar roverul Opportunity și-a continuat misiunea pana în anul 2019, iar în 2014 a depășit recordul pentru cea mai lungă distanță parcursă de un rover în afara Pământului, stabilit de Lunokhod 2.[10]

**2009** Versiunea inițială a sistemului de operare pentru roboți – Robot Operating System (ROS 0.4) a fost lansată.[11]

**2009** Google a început să-și testeze tehnologia de autopilotare cu o Toyota Prius modificată pe autostrăzile din California.[12]

**2010** În cadrul VisLab Intercontinental Autonomous Challenge, patru vehicule autonome au parcurs o cursă de aproape 16.000 km, de la Parma, Italia, până la Shanghai, China, aproape fără intervenție umană.[13]

**2012** NASA a reușit să aterizeze cu succes roverul Curiosity pe Marte; acesta este încă activ.[14]

**2014** Google și-a prezentat noul prototip de vehicul autonom care nu are volan și nici pedale.[15]

Mulți roboți noi, vehicule autonome, drone, senzori și dispozitive asociate au apărut și au devenit obișnuite.

## Capitolul 2. Stadiul actual

### 2.1 Roboti mobili / sisteme mobile generale

Domeniul roboților mobili autonomi este un domeniu de cercetare și dezvoltare extrem de popular. În cadrul acestui domeniu, majoritatea roboților mobili utilizează roți pentru deplasare. Roboții mobili pe roți lasă urme acolo unde niciun om nu a fost vreodată (de exemplu, explorări extraterestre), omul nu ar trebui să meargă (de ex, zone periculoase sau contaminate), sau acționează alături de om (de exemplu, sprijin și asistență pentru oameni).

Roboții mobili pe roți au început deja să pătrundă în în casele noastre, sub forma roboților de curățat podelele și a mașinilor de tuns iarba, vehiculele autonome pot fi găsite în interiorul multor fabrici industriale, prototipuri de mașini care se conduc singure circulă deja în traficul normal, iar în viitorul apropiat multe aplicații noi ale roboților mobili autonomi pe roți urmează să apară, chiar și în unele moduri neprevăzute.

Principalele părți mecanice și electronice ale unui robot mobil autonom sunt următoarele:

- **Piese mecanice:** piese rigide și mobile (corp, roți, șenile, picioare etc.)
- **Actuatori:** motoare electrice (DC, pas cu pas, servomotor etc.)
- **Senzori:** encodere de rotație, senzori de proximitate și de distanță, unitate de navigație inerțială (accelerometru și giroscop), sistem global de navigație prin satelit etc.
- **Calculatoare:** microcontrolere, calculatoare personale portabile, sisteme integrate etc.
- **Unitate de alimentare:** baterii, panouri solare etc.

- **Electronica:** drivere pentru actuatori, distribuție de energie, distribuție de semnale, telecomunicații etc.

Un robot mobil poate fi definit ca un sistem mechatronic capabil să se deplaseze în mediul său înconjurător în mod autonom. Pentru a putea fi capabil de acest lucru trebuie să fie echipat cu:

- **Senzori** care ajuta la determinarea locației și cunoașterea împrejurimilor (de care robotul este mai mult sau mai puțin conștient)
- **Actuatori** necesari pentru a se putea deplasa
- **Inteligenta** (algoritm, regulator) care îi va permite să calculeze, pe baza datelor colectate de senzori, comenziile pe care trebuie să le trimită actuatorilor pentru a îndeplini o anumită sarcină.

În cele din urmă, la acestea trebuie să adăugăm împrejurimile robotului care corespund lumii în care își desfășoară activitatea și misiunea sa, care este sarcina pe care o are de îndeplinit și pe care trebuie să o îndeplinească.

## 2.2 Tipuri de roboți mobili

În funcție de tipul sistemului de locomoție, roboții pot fi clasificați în următoarele mari categorii:

- Roboți staționari (manipulatori)
- Roboți terestre
  - a) Roboți mobili cu roți (WMR – Wheeled Mobile Robot)
  - b) Roboți păsitori (sau cu picioare)
  - c) Roboți mobili cu șenile
  - d) Roboți hibrid
- Roboți aerieni
- Roboți acvatici

### 2.2.1 Roboți staționari (manipulatori)

Manipulatoarele și roboții industriali sunt exemple de acest tip. Baza robotului este fixă, iar aceștia sunt compuși dintr-un lanț cinematic deschis, în principal cu un efectuator final cu unele speciale care nu numai că manipulează obiecte, dar pot efectua și sarcini precum sudarea, vopsirea, asamblarea, prelucrarea și aşa mai departe. Printre roboții de acest tip se numără Abb, Kuka, Fanuc, Staubli, Kawasaki, Comau, Wittman



Figură 2.2.1 Roboți manipulatori KUKA [16]

Alte sisteme robotice staționare importante sunt dispozitivele de prindere. Prehensiunea este o parte importantă a manipulării și, încă de la început, dispozitivele de prehensiune au fost concepute în principal pentru a ajuta oamenii în sarcinile de manipulare, oferind soluții care pot fi clasificate în două categorii: unelte și proteze.

## 2.2.2 Roboți tereștri

### a) Roboți mobili cu roți (WMR)

Acest tip de robot este cel mai des întâlnit deoarece folosirea roților este mai simplă decât folosirea șenilelor sau a picioarelor. Robotul este mai ușor de proiectat, construit și programat atunci când se deplasează pe un teren plat, neaglomerat. De asemenea, costurile pentru acest tip de robot sunt mult mai scăzute decât cele pentru un robot cu picioare.

Controlul roților este mai puțin complex și provoacă o uzură mai mică a suprafeței pe care se deplasează, în comparație cu alte soluții. Un alt avantaj este că nu prezintă mari dificultăți în ceea ce privește problemele de echilibru, deoarece robotul este de obicei în contact cu o suprafață. Principalul dezavantaj al roților este că acestea nu se descurcă foarte bine la trecerea peste obstacole, cum ar fi terenul stâncos, suprafețele ascuțite sau zonele cu frecare redusă.

Pentru roboții mobili cu roți se folosesc în principal următoarele patru tipuri de roți:

- **Roată standard fixă** - Acestea sunt roți convenționale cu un singur grad de libertate (DOF), se rotesc în jurul punctului de prindere. (Figura 2.2.2)



Figură 2.2.2 Roată fixă [17]



Figură 2.2.3 Roată pivotantă [19]



Figură 2.2.4 Roată omnidirecțională [18]



Figură 2.2.5 Roată sferică [20]

- **Roată pivotantă** - Are 2 grade de libertate (2 DOF) și permite rotirea în jurul articulației sale. (Figura 2.2.3)
- **Roată omnidirecțională** - Sunt roți cu discuri mici (numite role) în jurul circumferinței, care sunt perpendiculare pe direcția de rotație. Efectul este că roata poate fi acționată cu

toată forța, dar va aluneca și lateral cu mare ușurință. Au trei grade de libertate (3 DOF). (Figura 2.2.4)

- **Roată sferică sau cu bilă** - acest tip de roată este dificil de implementat (Figura 2.2.5)

### b) Roboți pășitori (sau cu picioare)

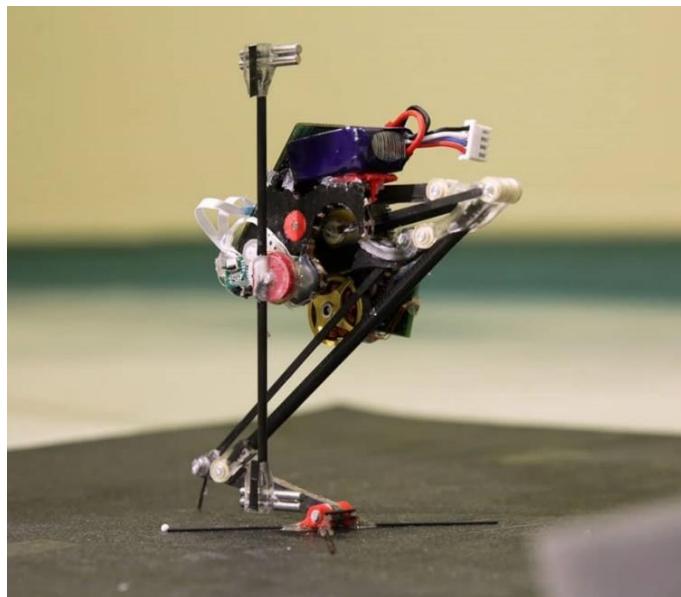
Deși sunt de obicei mai scumpe decât roțile, robotii cu picioarele prezintă mai multe avantaje decât cei cu roți. Cel mai mare avantaj este transversalitatea și eficiența lor și faptul că se pot deplasa și pe terenuri moi și denivelate, o mai bună mobilitate, o mai bună eficiență energetică, o mai bună stabilitate și un impact mai mic asupra solului. Roboții pășitori au, de asemenea, avantajul de a face față cu ușurință obstacolelor sau fisurilor întâlnite în mediul înconjurător; pe scurt, adaptabilitate și manevrabilitate pe teren accidentat.

În cazul roboților care pășitori, stabilitatea este principala problemă, deoarece echilibrul robotului în timpul deplasării este de o importanță extremă, în special în cazul unui robot cu două picioare sau al unui humanoid.

Există două tipuri de stabilitate: statică și dinamică. Stabilitatea statică este reprezentată de capacitatea robotului de a-și menține poziția atunci când este supus doar la forțe de reacțiune. Stabilitatea dinamică este legată de capacitatea de a rămâne în poziție verticală atunci când este supus atât la forțe de reacțiune, cât și la forțe inerțiale.

### • Roboți cu un singur picior

"Hoppers" sunt roboții cu un singur picior. Cea mai mare provocare a acestor roboți este faptul că nu pot sta nemîșcați; trebuie să țopăie continuu pentru a-și menține echilibrul. Uniroo și Salto sunt exemple de astfel de roboți.



Figură 2.2.6 Robot săritor SALTO [21]

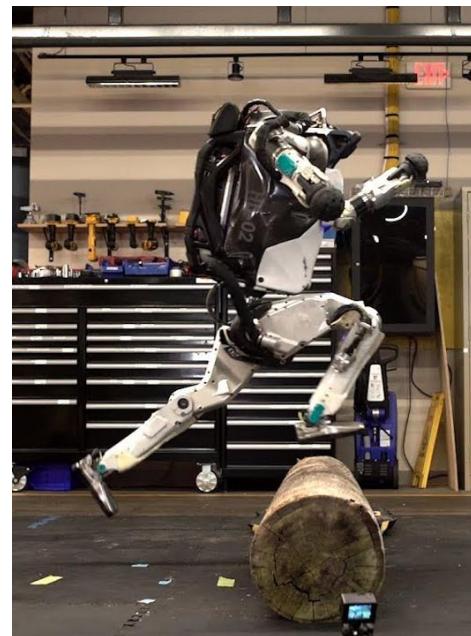
- **Roboți cu două picioare (umanoizi)**

Unul dintre cele mai importante tipuri de roboți pășitori este reprezentat de roboții umanoizi. Pentru a reproduce capacitatele umane, aceștia trebuie să utilizeze foarte bine senzorii. Aceștia pot să meargă, să vorbească, să reproducă emoțiile etc. Una dintre cele mai importante probleme din punct de vedere al locomoției este pierderea echilibrului. Mișcarea roboților bipezi depinde de stabilitatea dinamică.

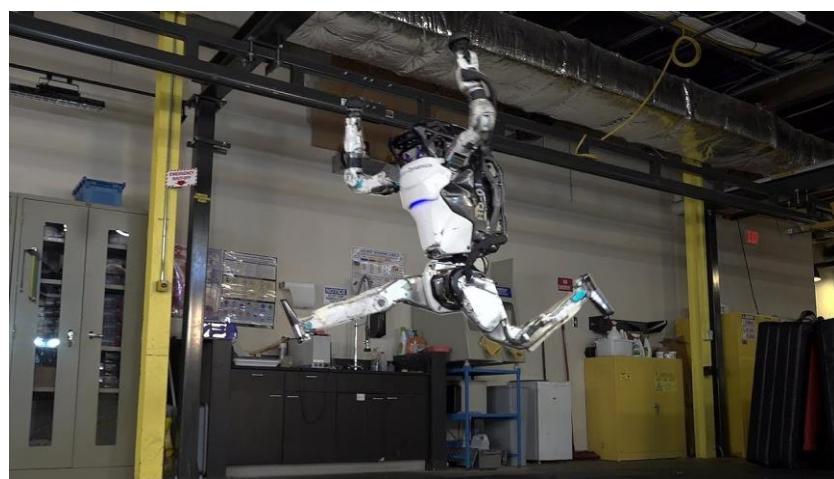
Roboții cu două picioare pot să meargă, să alerge, să urce și să coboare scări, să sară și chiar să facă salturi. Mișcările sunt sofisticate, deoarece picioarele sunt foarte mici și echilibrul trebuie să fie dinamic în permanență; chiar și statul pe loc necesită un control sofisticat.



Figură 2.2.7 NASA – Valkyrie robot [23]



Figură 2.2.8 Boston Dynamics – Atlas robot



Figură 2.2.9 Boston Dynamics – Atlas robot [22]

- **Roboți cu patru picioare (patrupezi)**

Atunci când este nevoie de o siguranță sporită sau de o capacitate de a susține și transporta sarcini mari, se folosesc roboții patrupezi sau roboții cu un număr mai mare de picioare. Acești roboți au avantajul de a fi stabili static atunci când nu sunt în mișcare, dar necesită un control dinamic al mersului, deoarece centrul de greutate al robotului trebuie să fie schimbat cu ușurință în timpul mersului. Controlul și coordonarea picioarelor acestor roboți este, totuși, mai complicată.



Figură 2.2.10 Big Dog



Figură 2.2.11 Spot dog

- c) **Roboți mobili cu șenile**

Aceștia sunt un tip de roboți care utilizează benzi de rulare sau șenile în loc de roți. În configurațiile cu roți, presupunem că roțile nu pot aluneca pe suprafață. Deoarece șenilele nu pot fi inclinate în raport cu corpul robotului (în orice construcție operațională), direcția trebuie să fie realizată prin accelerarea unei șenile, încetinirea celeilalte (sau inversarea acesteia) sau o combinație a celor două. [24]

Roboții cu șenile au suprafețe de contact cu solul mult mai mari, iar acest fapt joacă un rol important în îmbunătățirea manevrabilității lor pe terenuri accidentate în comparație cu roboții convenționali pe roți.



Figură 2.2.12 NASA – GROVER [25]



Figură 2.2.13 Dr. Robot Jaguar

#### d) Roboți hibrid

Aceștia sunt roboți a căror structură constă într-o combinație a oricărui dintre tipurile descrise mai sus. De exemplu, un dispozitiv articulat segmentat și cu roți. Soluțiile hibride combină caracterul adecvat al roților cu adaptabilitatea picioarelor. Rover-ul Curiosity creat de NASA este un rover hibrid.



Figură 2.2.14 NASA - Curiosity Rover

### 2.2.3 Roboți aerieni

Acest grup este format din sisteme mobile care zboară într-un anumit spațiu aerian (avioane, elicoptere, drone, rachete, sisteme de zbor care imită animale; atunci când sunt utilizate fără pilot, acestea sunt denumite vehicule aeriene fără pilot) sau care orbitează în jurul Pământului sau al unui alt corp ceresc (sateliți).

Un robot aerian fără pilot, cunoscut sub numele de "dronă", este o mașină care îndeplinește o sarcină preprogramată cu sau fără interacțiune umană și este inspirat din funcționarea unui avion. Cele mai avansate drone pot acum să decoleze și să aterizeze complet independent. Inițial, acestea au fost utilizate mai ales în aplicații militare, dar s-au extins rapid către alte aplicații, cum ar fi cele științifice, agricole, comerciale, recreative, de supraveghere, de livrare, distribuție și logistică a produselor, de fotografie aeriană etc.



Figură 2.2.15 Octocopter

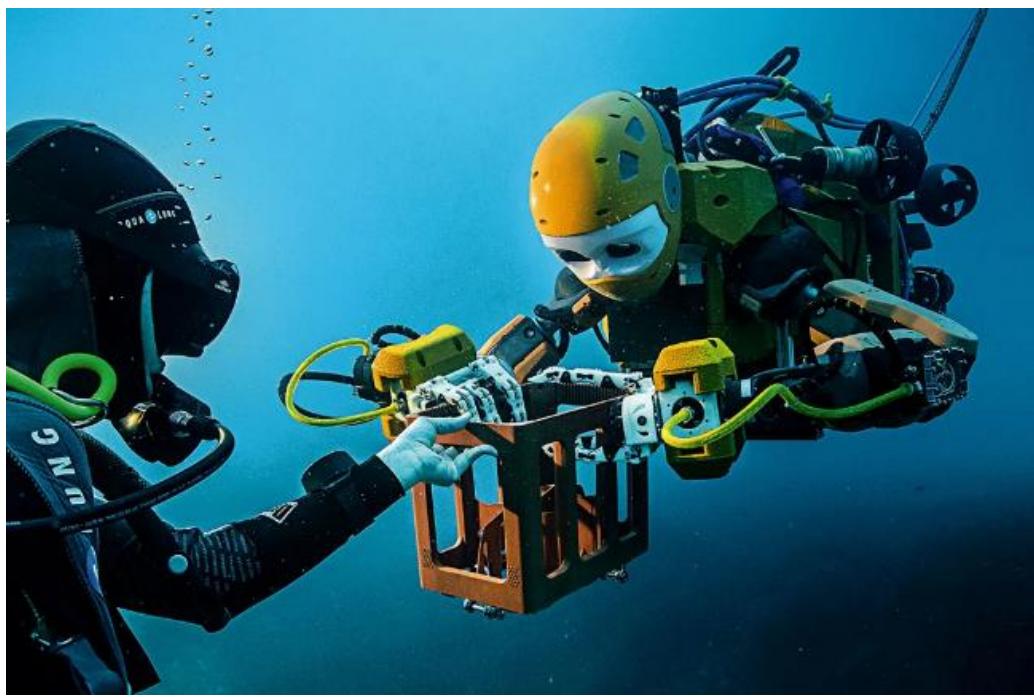


Figură 2.2.16 Dronă folosită pentru a livra pizza [26]

### 2.2.4 Roboți acvatici

Unul dintre cele mai vechi obiective ale omului a fost acela de a explora oceanele și zonele subacvatice inaccesibile pentru el. În acest scop au fost construite multe dispozitive, inclusiv sisteme robotizate.

OceanOne este un exemplu de robot submarin. Aceasta este un robot humanoid care explorează fundul mării. Acesta profită de cele mai bune avantaje ale vehiculelor operate de la distanță și de avantajele roboților umanoizi, cum ar fi faptul că dispune de o mâna robotică cu care poate prelua obiecte ca și cum ar fi o ființă umană.



Figură 2.2.17 OceanOne

## 2.3 Soluții existente

### 2.3.1 Robot pentru inspecții cu șenile GPK-32



Figură 2.3.1 GPK-32 Robot [38]

Acesta este un pachet complet de robot pentru inspecție cu șenile, foarte ușor de folosit, conceput pentru utilizatorul casnic.

Robotul utilizează o cameră 1080P cu zoom 10x montată pe un suport de înclinare actionată (-20° până la +45°) și un set de lumini LED puternice pentru a asigura vizibilitatea în locuri întunecate. Camera cu zoom 10x de înaltă rezoluție oferă posibilitatea de a inspecta de aproape și în mare detaliu. Un emițător separat trimite imaginea video înapoi către controler fără latență. Aceste caracteristici permit ca GPK-32-Z să abordeze cu ușurință zonele greu accesibile.

Acest robot cu șenile este construit pe un șasiu din aluminiu și utilizează componente durabile (roți UHMW, rulmenți din bronz etc.). Fiecare șenilă este acționată de un motor cu angrenaj planetar care oferă o putere fiabilă și lină. Roțile sunt montate direct pe arborele de ieșire al motorului, iar roțile inerțiale sunt înșurubate direct pe șasiu.

Combinarea dintre motor și șenile permite acestui robot să meargă pe majoritatea terenurilor, dar dimensiunile sale îl fac foarte manevrabil și în interior. Robotul utilizează o direcție diferențială (mișcarea se bazează pe două roți acționate separat, plasate de o parte și de alta a corpului robotului. Astfel, acesta își poate schimba direcția prin variația vitezei relative de rotație a șenilelor sale).[38]

### 2.3.2 Robotnik Guardian

Platforma Guardian de la Robotnik este o platformă robotică extrem de versatilă, de mare mobilitate, utilizabilă în aproape orice mediu. Este echipată cu un sistem hibrid de rulare pe roți și șenile și poate urca pante de până la 45 de grade. Carcasa sa este clasificată la IP54, dar se poate implementa o protecție optională împotriva intemperiilor de până la IP66.



Figură 2.3.2 Robotnik Guardian [39]

Poate fi utilizat în nenumărate aplicații, de la navigație sau aplicații științifice mai avansate de tip swarming, la eliminarea dispozitivelor explozive și chiar aplicații militare.

De asemenea, poate fi operat de la distanță și pot fi instalate componente opționale, cum ar fi brațele robotizate.

Suportă orice tip de senzor, cum ar fi camere PTZ, 3D, IP și IR, microfoane, module GPS și multe altele. În versiunea sa de bază, platforma este doar controlată prin radio, dar versiunea avansată este prevăzută cu un PC încorporat care rulează sistemul de operare Linux Real Time OS cu interfețe WiFi, USB și RS232. Are chiar spațiu pentru a găzdui un computer secundar pentru o putere de procesare sporită, necesară pentru aplicații avansate, cum ar fi DGPS sau laser ranging.

Platforma are o greutate de 85 kg și suportă o sarcină utilă de până la 50 kg. Are două motoare de tracțiune de 1000 de wați și este dotată cu două baterii de 12 V, 50 Ah. Este construită la comandă, în conformitate cu specificațiile clientului. Are un preț de aproximativ 25.000 de dolari americani, dar prețul poate varia în funcție de configurația sa.

### 2.3.3 CoroWare Explorer Ex-D

Platforma CoroWare Explorer este, în esență, un computer pe roți. Platforma de 13 kg este extrem de robustă, fiind destinată funcționării în aer liber pe teren accidentat. Roțile de pe fiecare parte sunt montate pe boghiuri care pivotează în jurul unei axe centrale, iar întoarcerea se realizează cu ajutorul unui sistem de direcție cu patine.



Figură 2.3.3 CoroWare Explorer Ex-D [39]

Ca o caracteristică foarte interesantă, în această versiune, computerul vine cu Windows XP și Linux preinstalate într-o configurație dual-boot și un pachet software de teleoperare este, de asemenea, preinstalat, în timp ce serviciile MRDS sunt furnizate pentru toate componentele hardware. Calculatorul are un procesor de 2 GHz, un hard-disk de 80 GB și este inclusă și o cameră color de 2 Megapixeli. Este furnizat un încărcător de baterii și este posibilă, de asemenea, operarea prin cablu cu alimentare de curent alternativ. Prețul este de aproximativ 13.000 de dolari americanii.

#### 2.3.4 Dr. Robot Sentinel3

Dr. Robot Sentinel3 este o platformă robotică mobilă autonomă complet echipată. Acest robot poate naviga continuu, se poate reîncărca singur, poate fi teleoperat și poate monitoriza de la distanță anumite zone. Acesta cântărește 6 kg, are un șasiu robust din aluminiu și poate transporta până la 15 kg. Platforma este acționată de două motoare duble de 12 VDC cu codere optice integrate. Robotul se poate conecta la o rețea WiFi 802.11 b/g standard, iar antenele cu rază de acțiune extinsă, precum și suportul 802.11a/n sunt disponibile opțional. De asemenea, este disponibilă o interfață RS232.



Figură 2.3.4 Dr Robot Sentinel3 [39]

Acesta are o baterie dublă NiMH de 12 V și 3,8 Ah, un modul I/O, un modul de comandă a motorului cu feedback de poziție și curent și un mic afișaj LCD color. De asemenea, sunt

furnizate un încărcător de perete, un joystick pentru teleoperare, cabluri și chiar un router wireless. Este furnizat și un software de teleoperare și monitorizare. Toate acestea se obțin la un preț de aproape 12.000 de dolari americanii.

### 2.3.5 InspectorBots Mega Bot



Figură 2.3.5 Mega Bot [39]

Aceasta este o platformă pentru sarcini grele, echipată cu motoare electrice cu cuplu mare, care asigură o capacitate de tractare de 900 kg și poate transporta chiar și un om. Poate fi utilizată atât în scopuri de divertisment, cât și în aplicații de salvare sau de securitate. Platforma are o greutate de aproximativ 80 kg și poate urca pante de până la 30 de grade. Șasiul este fabricat din oțel și este rezistent la apă.

Este o platformă modulară, astfel încât orice hardware poate fi instalat pe ea, în funcție de cerințe și aplicații. Are un preț de aproximativ 7.000 de dolari americanii.

## Capitolul 3. Prezentare soluția proprie

Soluția proprie este reprezentată de un robot compact cu dimensiuni de gabarit relativ mici, 220mm lungime, 200mm lățime și înălțimea maxima de aproximativ 92mm. Datorita dimensiunilor reduse acesta este potrivit pentru a inspecta partea inferioara a autoturismelor fără a mai fi necesară folosirea unui elevator.

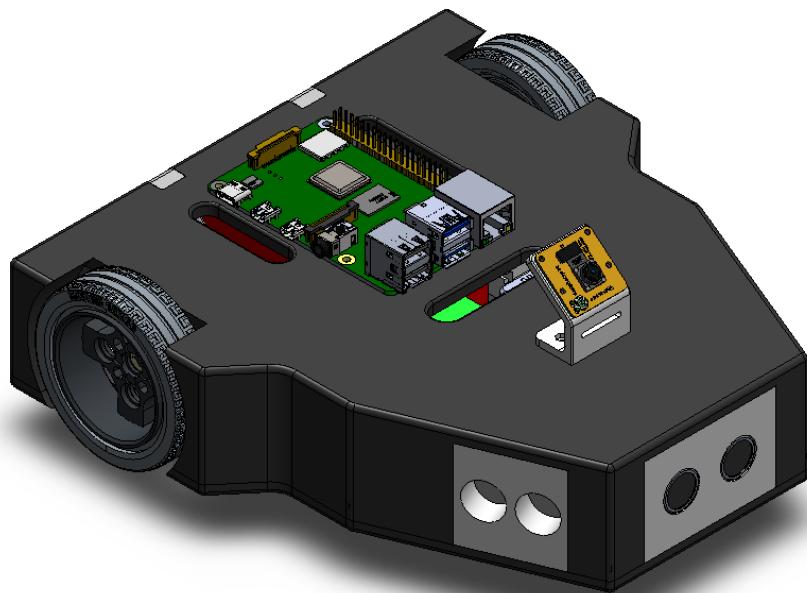


Figura 2.3.1 Ansamblu model 3D - vedere izometrică

Robotul este echipat cu o cameră potrivită pentru a transmite imaginea în timp real la o rezoluție maxima de 1920 x 1080. Imaginea este transmisă prin intermediul unui web browser către orice dispozitiv conectat la aceeași rețea wireless la care este conectat și robotul. Camera este înclinată la un unghi de 40° față de orizontală, caracteristică ce oferă o perspectivă mărită asupra componentelor autoturismului.

De asemenea, senzorii ultrasonici, poziționați atât în fața robotului cât și pe lateral (Figura 2.3.2), împiedică deplasarea robotului în direcția în care ar putea exista un obstacol sau un perete. Această serie de senzori îl vor ține la o distanță sigură atât pentru acesta cât și pentru obiectele din jurul său.

Pentru a crește gradul de siguranță, robotul va fi limitat în a executa numai anumite mișcări predefinite, mișcări care vor fi specificate utilizatorului și vor fi ușor de intuit de către acesta. Mișcările principale ale robotului sunt mers înainte, mers înapoi, viraj stânga și viraj dreapta.

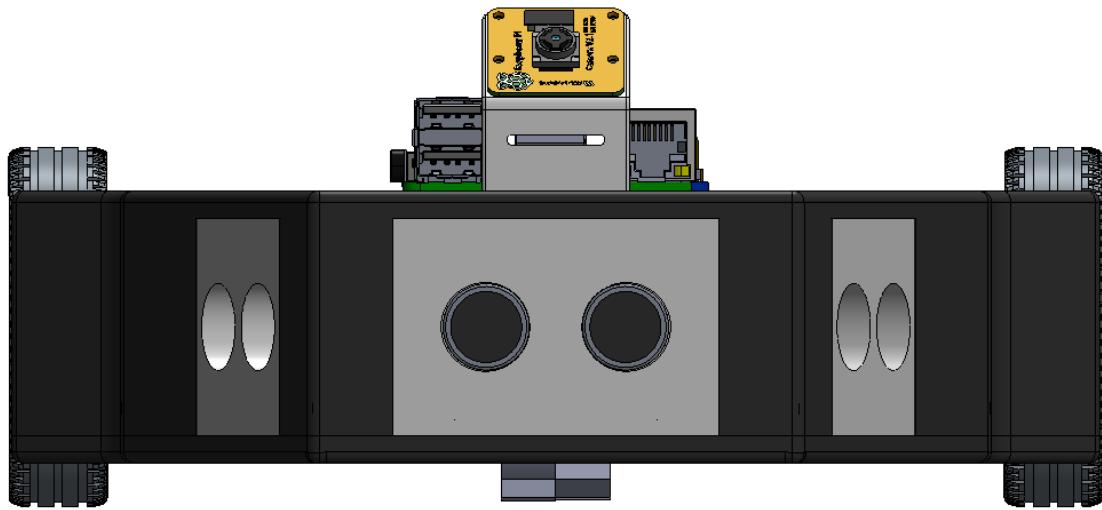


Figura 2.3.2 Ansamblu model 3D - vedere din față

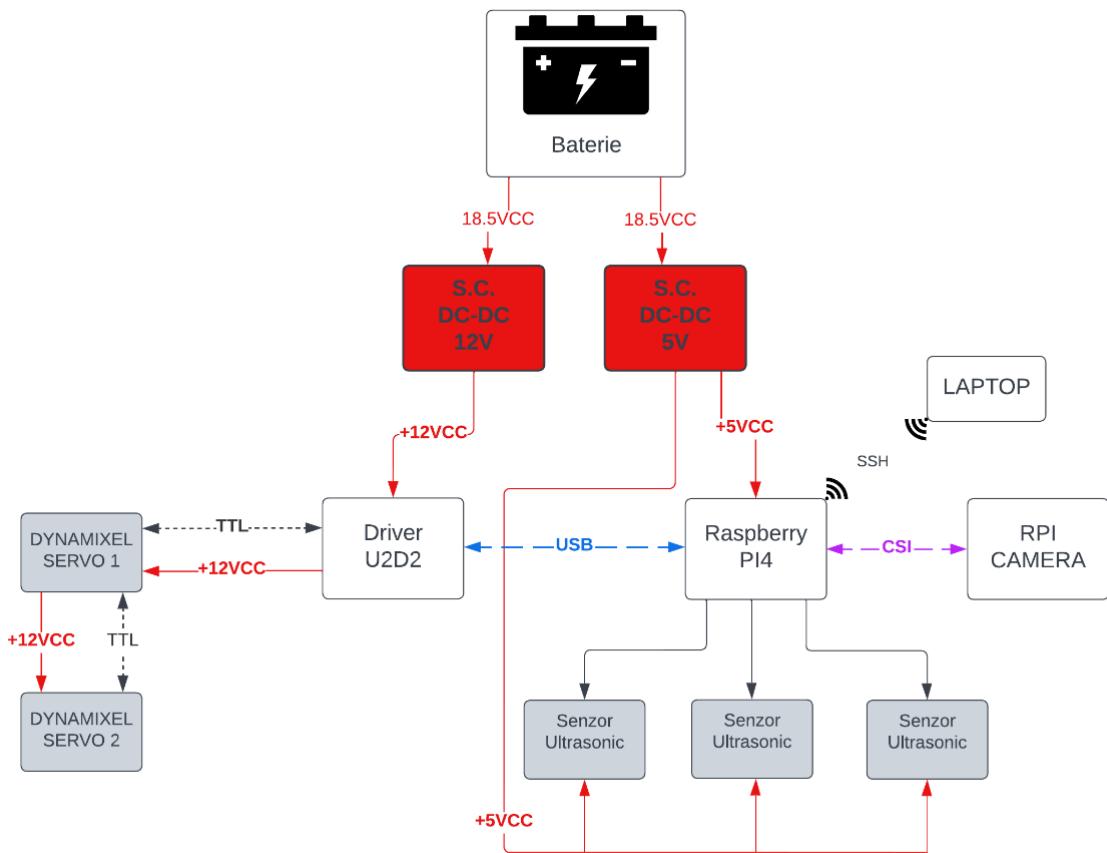


Figura 2.3.3 Schema bloc a sistemului

Sistemul este alcătuit din piese realizate folosind tehnologia de printare 3D: sașiu inferior, sașiu superior, suport motoare, roată omnidirecțională, suport senzori ultrasonici, suport cameră, suport baterie.

Conform figurii 3.3, din punct de vedere electronic, sistemul este alcătuit din două servomotoare acționate de un singur driver, trei senzori ultrasonici, o camera Raspberry și un microcontroler Raspberry Pi 4. Toate componentele sunt alimentate de o baterie de 18.5V prin intermediul a două surse coborâtoare de tensiune, una folosită pentru a furniza tensiunea de 12V pentru motoare, iar cealaltă alimentează restul componentelor cu o tensiune de 5V.

Alegerea PLA-ului și ABS-ului ca materiale pentru realizarea structurii mecanice oferă robotului o greutate redusă, de aproximativ 1.5 kg. De asemenea, alegerea culorilor alb și negru pentru piesele realizate prin imprimare 3D conferă robotului un aspect elegant, plăcut.

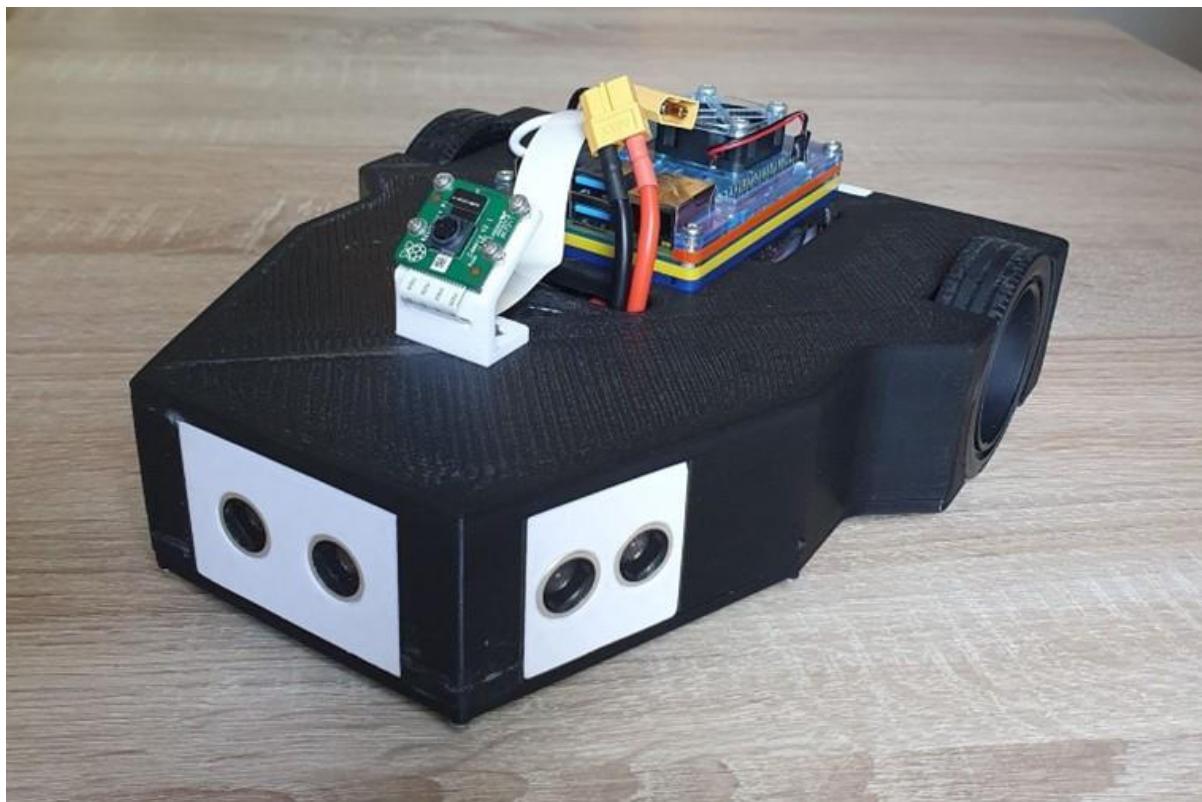


Figura 2.3.4 Ansamblu real

Sistemul este alcătuit din piese printate la imprimanta 3D: placă șasiu inferior, placă șasiu superior, suport motoare, suport senzori ultrasonici, suport camera, suport baterie.

Electric: Motoare (care se deplasează cu viteza variabilă), driver pentru motoare, senzori ultrasonici, cameră, regulatoare de tensiune, baterie și microcontroler.

Microcontroler-ul Raspberry Pi 4 face ca toate componente sa comunice intre ele si sa fie comandate

## Capitolul 4. Memoriu Tehnic

### 4.1 Proiectarea structurii mecanice

Ținând cont de funcția pe care trebuie sa o îndeplinească acest robot, de a intra sub mașină, este esențial ca gabaritul sa îi permită acest lucru. Drept urmare înălțimea este un factor limitator in proiectarea acestui robot. Pentru se putea deplasa pe dedesubtul majoritatea modelelor de autoturisme, am ales ca înălțimea maxima să nu depășească 100mm.

Dimensiunile finale ale robotului sunt 220mm lungime, 200.7 mm lățime și înălțimea maxima de doar 91.57mm.

In timpul proiectării au fost luate in calcul limitările tehnologiei de execuție si anume tehnologia de printare 3D si de materialele disponibile PLA și ABS.

#### 4.1.1 Proiectarea carcasei

Pentru proiectarea carcasei sistemului, s-a ținut cont de mediul de exploatare a robotului, și anume străzi si șosele unde posibilitatea de a întâlni praf este ridicata. Drept urmare, am decis ca zonele prin care praful poate pătrunde sa fie inexistente sau cât mai puține.

Pentru a satisface această condiție am ales sa folosesc două plăci care se îmbină etans, prin șuruburi, rămânând astfel neprotejate doar alezajele pentru rotile motoare si cel pentru roata omnidirecțională.

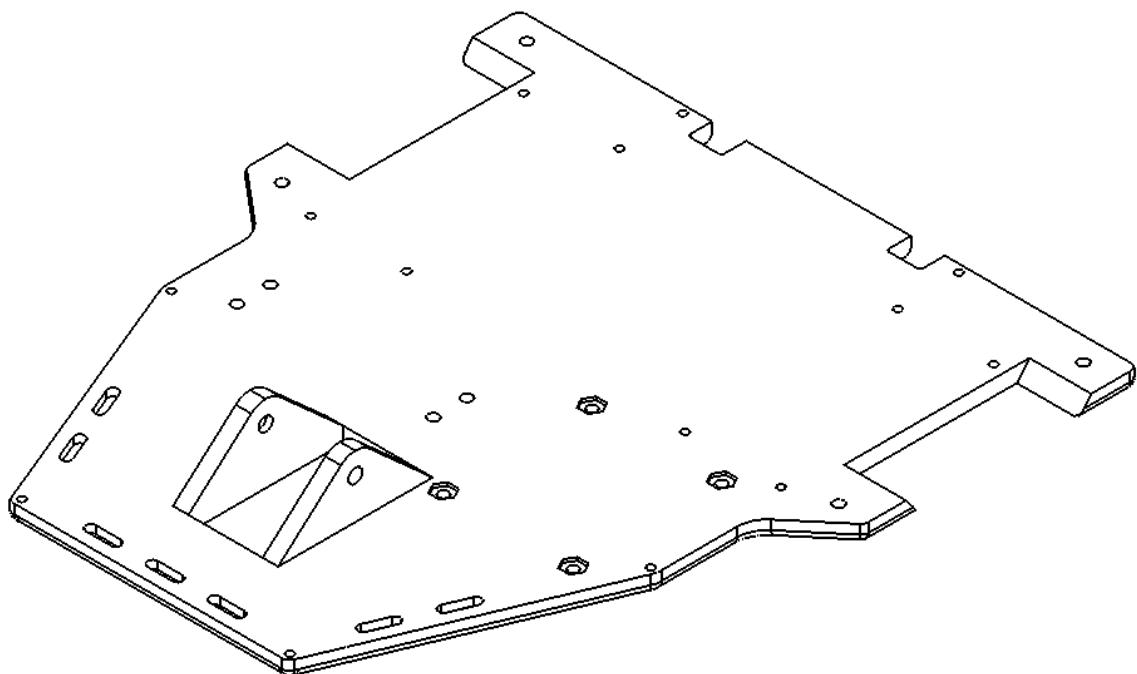


Figura 4.1.1 Placa șasiu inferior

In figura 4.1.1 este reprezentata placa inferioară a șasiului. Aceasta este placa pe care se montează marea majoritate a componentelor. In partea laterală a plăcii au fost create alezajele pentru rotile motoarelor și in spatele plăcii alezajul pentru prinderea suportului de baterie.

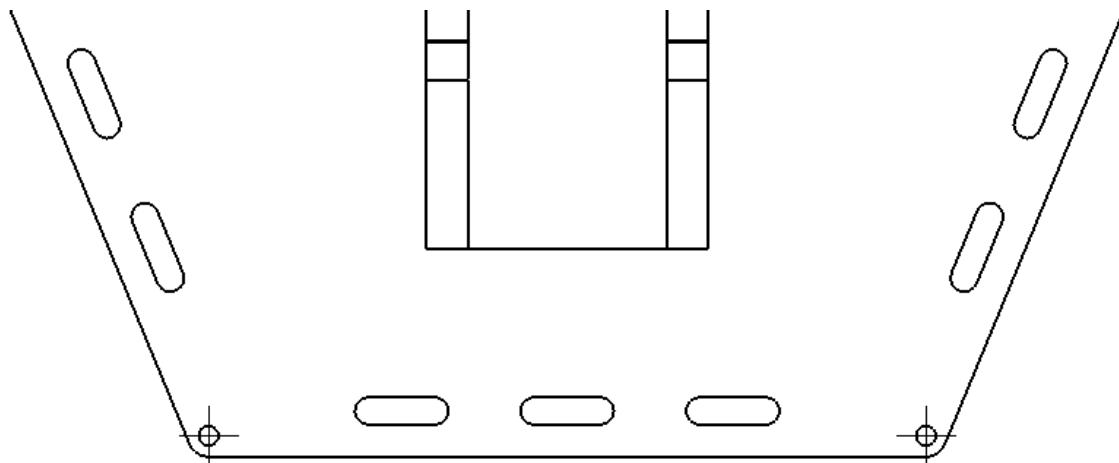


Figura 4.1.2 Evidențiere zona alezaj suport senzori ultrasonici

In partea din față a plăcii a fost creat alezajul pentru roata omnidirecțională. Pentru lărgărirea roții am ținut cont de diametrul roții și de garda de la sol a robotului.

De asemenea, in partea din față si in lateral am creat alezajele pentru prinderile suporturilor pentru senzorii ultrasonici. (Figura 4.1.2)

Totodată au fost create găurile străpunse pentru fixarea suporturilor pentru motoare, pentru fixarea driver-ului, fixarea surselor coborâtoare de tensiune și găurile pentru etanșarea robotului aflate în zona exterioară a plăcii.

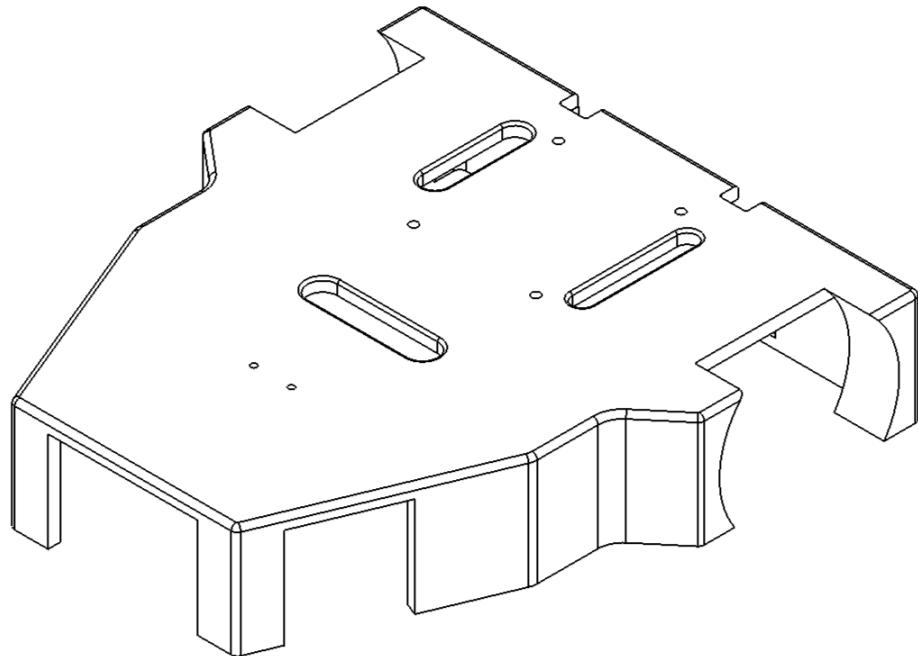


Figura 4.1.3 Placa șasiu superior vedere izometrică

In figura 4.1.3 este prezentata placa șasiului superior. Pe partea superioara a plăcii există găuri pentru fixarea suportului de camera și găuri pentru fixarea microcontrolerului Raspberry PI.

În partea din spate a plăcii, în jurul găurilor de fixare pentru microcontroler există trei alezaje ce permit accesul cablurilor la porturile HDMI, USB, GPIO și portul la alimentare al microcontrolerului Raspberry Pi.

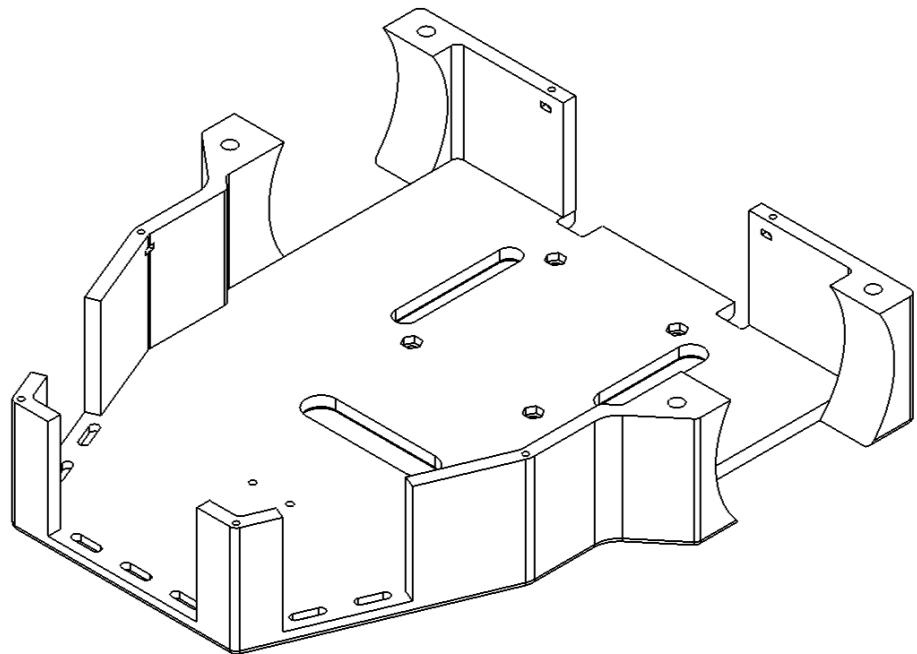


Figura 4.1.4 Placa șasiu superior vedere in interior

Placa superioară este aproape identica cu placa inferioara, având alezajele pentru suporturile senzorilor ultrasonici în aceeași poziție cu alezajele de pe placa inferioară. De asemenea găurile pentru etanșare se află pe perimetru plăcii. Acest lucru se observă în figura 4.1.4.

În plus față de placa inferioară, aceasta placă a fost proiectată cu pereți exteriori cu grosimea de 5mm și înălțimea de 40mm. Pentru a determina înălțimea pereților exteriori s-a ținut cont de dimensiunile de gabarit ale componentelor ce urmau a fi montate și de dimensiunile finale ale suportului pentru motoare. Suporturile pentru motoare fiind de asemenea un punct de sprijin pentru placa superioară.

Pereții lateralii ai șasiului superior prezintă alezaje pentru roți și pentru suporturile senzorilor ultrasonici. (Figura 4.1.5)

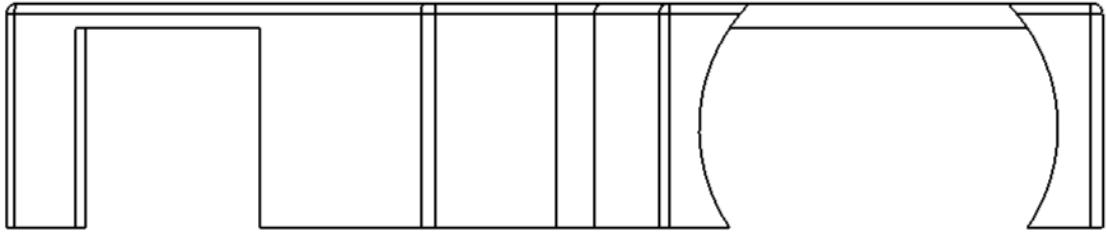
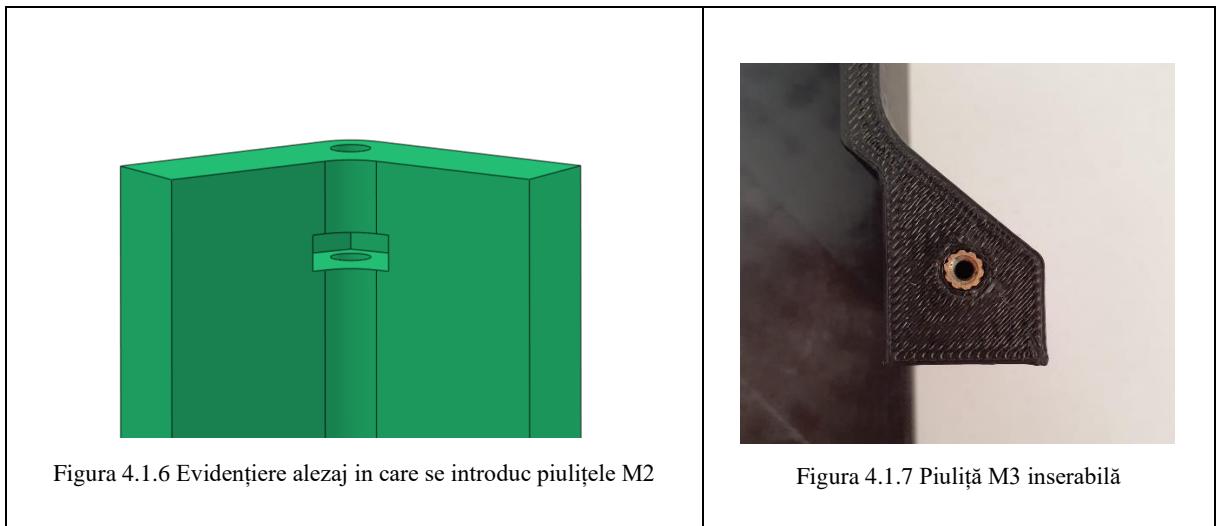


Figura 4.1.5 Placa superioara vedere din lateral

Pentru a putea fixa cele 2 placi se folosesc șuruburi de tip M2 pentru pereții exteriori și șuruburi M3 pentru pereții mai groși din zona roților. Șuruburile de tip M2 se fixează folosind piulițe standard M2 care se introduc în alezajele create interiorul pereților exteriori, zona evidențiată în figura 4.1.6.



Șuruburile M3 se fixează folosind piulițe M3 inserabile care se montează în alezajul de la suprafața pereților groși din zona roților. (Figura 4.1.7)

#### 4.1.2 Proiectarea suporturilor pentru motoare

Şasiul va avea două roți motoare acționate individual și o roata omnidirecțională pentru sprijin. În acest scop a fost necesară proiectarea unor suporturi pentru susținerea motoarelor roților motoare.

Un pas esențial înaintea proiectării suporturilor menționate anterior este acel de alegere a roților și implicit a diametrului acestora. În acest scop a fost ales un tip de roți standard pentru servomotoare de curent continuu ce se găsesc în comerț la un preț de achiziție bun (Figura 4.1.8). Aceste roți au un diametru de 66 mm ceea ce le face potrivite pentru acest sistem, astfel obținându-se și o viteză de deplasare mai mare a întregului sistem. De asemenea, acestea sunt echipate cu anvelope din cauciuc, fapt ce mărește aderența cu suprafețele ce se pot întâlni în mediul de lucru al sistemului.



Roata aleasa are lățimea de 18mm, diametrul de 66mm și se prinde de motor folosind 4 șuruburi M2.5. Știind dimensiunile roților, au putut fi proiectate suporturile pentru motoare.

Figura 4.1.8 Roata Aleasa

Suporturile motoarelor (Figura 4.1.9) au fost proiectate pentru a oferi întregului sistem o gardă la sol mică, de doar 8 mm, în acest fel centrul de greutate al acestuia a fost adus mai aproape de sol, măringind astfel stabilitatea robotului. De asemenea, această gardă la sol, împreună cu inexistența cablurilor de conexiune sau a componentelor electronice mai jos de linia șasiului, protejează robotul la contactul cu zone umede.

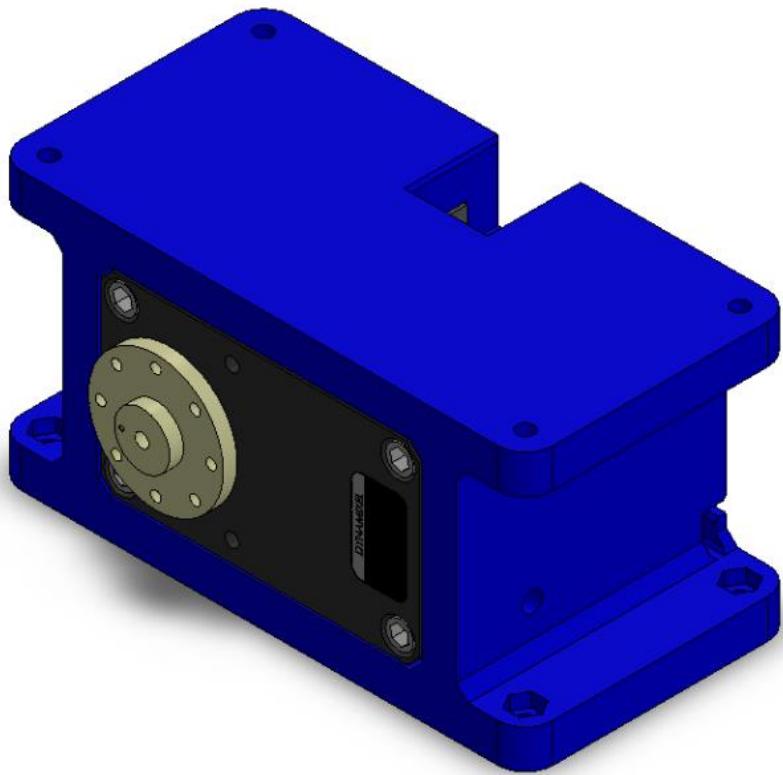


Figura 4.1.9 Ansamblu suport motor + motor

Pentru a putea avea acces la mufele motorului, cat si pentru a putea scoate motorul din suport, au fost create trei alezaje in spatele suportului. Alezajele sunt evidențiate in figura 4.1.10.

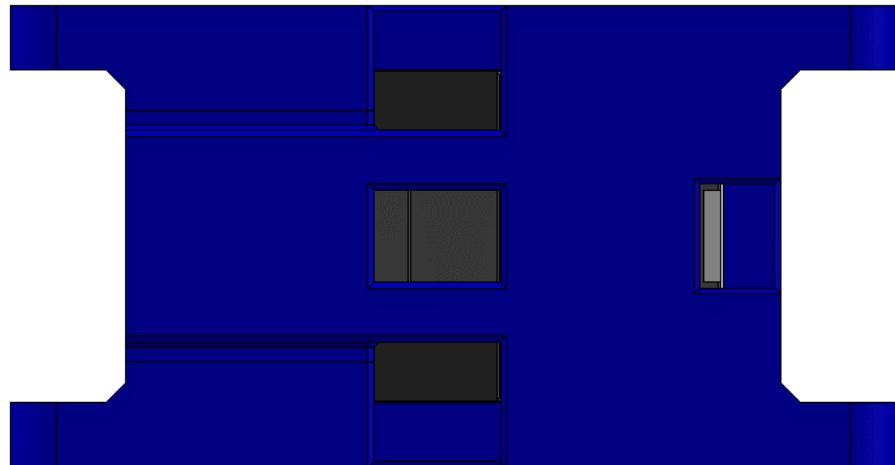


Figura 4.1.10 Suport motor - vedere din spate

De asemenea, motoarele alese au un led pentru feedback, alezajul din partea laterală a suportului face posibila monitorizarea acestui led. Pentru a facilita wire management-ul au fost

create două șanțuri prin care se pot introduce cablurile de comanda și alimentare ale motoarelor. Motoarele se fixează pe suport folosind șuruburi M2.5 care se strâng pe suprafața laterală a suportului, în găurile create. Toate aceste lucruri se observă în figura 4.1.11.

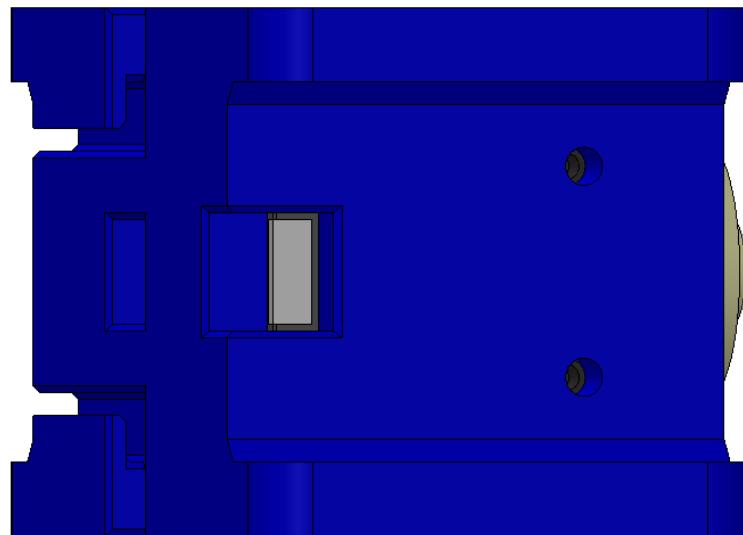


Figura 4.1.11 Evidențiere alezaj led, șanțuri, găuri fixare

Suportul are un design simetric, putând fi montat atât pe șasiul superior cât și pe cel inferior, atât pe partea dreaptă a robotului cât și pe partea stângă. Montajul pe șasiu se face folosind șuruburi și piulițe M2 care se introduc în alezajele de pe suprafețele laterale evidențiate în figura 4.1.12



Figura 4.1.12 Evidențiere alezaje piulițe M2

#### 4.1.3 Proiectarea suport senzori ultrasonici

Suportul pentru senzorul ultrasonic este relativ simplu de proiectat, prezinta două găuri cu diametrul de 16mm pentru introducerea difuzoarelor senzorului. Înălțimea suportului este egală cu înălțimea pereților robotului, 40mm, iar lățimea a fost aleasă astfel încât planul frontal al difuzoarelor să fie coplanar cu planul frontal al suportului.

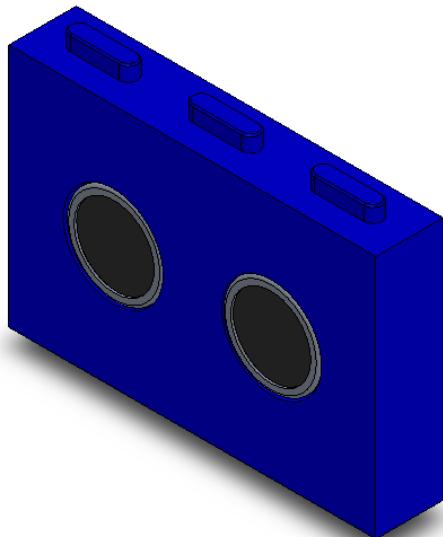


Figura 4.1.13 Subansamblu suport + senzor

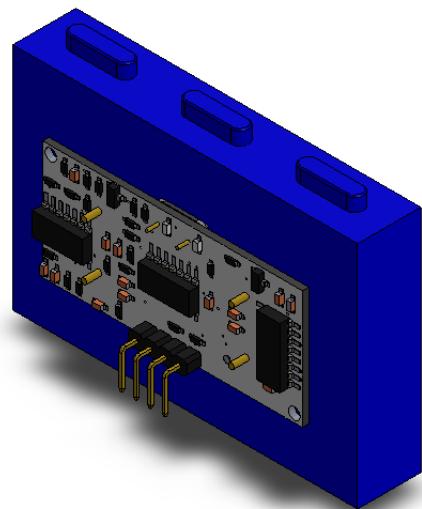


Figura 4.1.14 Subansamblu vedere din spate

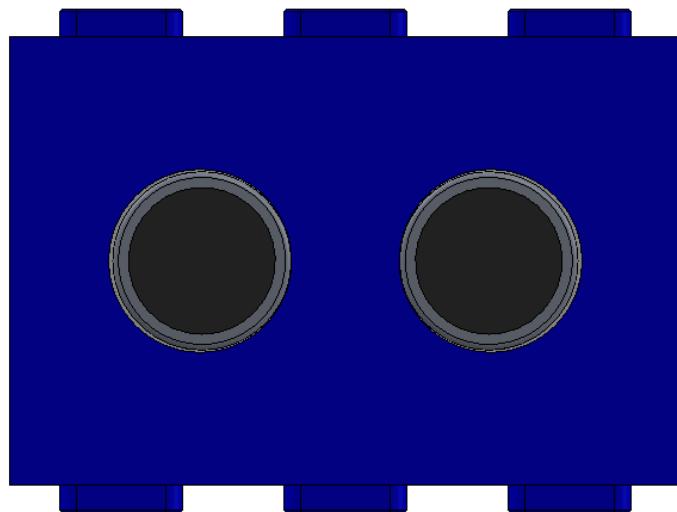


Figura 4.1.15 subansamblu vedere din față

Suportul se fixează pe ambele șasiuri folosind cele trei prinderi dispuse simetric pe latura superioara și pe cea inferioară a suportului. Prinderile sunt evidențiate în figura 4.1.15.

Deoarece senzorii de pe laturile laterale ale robotului sunt diferenți din punct de vedere al dimensiunilor de gabarit față de senzorul din față robotului, au fost proiectate două suporturi asemănătoare cu cel prezentat anterior. (Figura 4.1.16 și Figura 4.1.17)

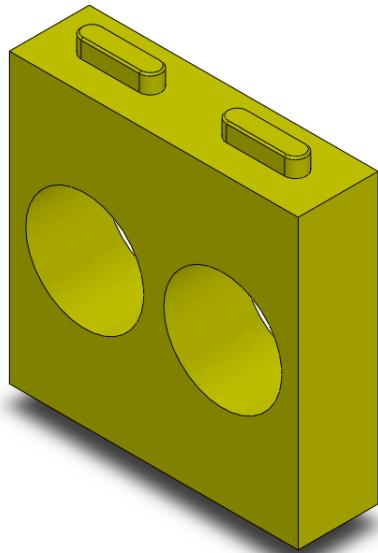


Figura 4.1.16 suport senzor lateral

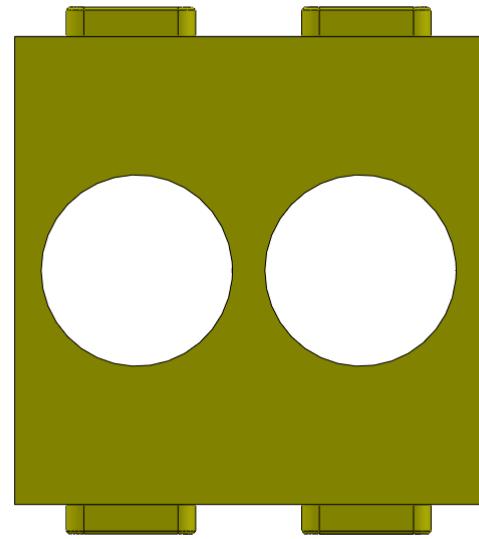


Figura 4.1.17 suport senzor lateral vedere din fata

#### 4.1.4 Proiectare suport camera

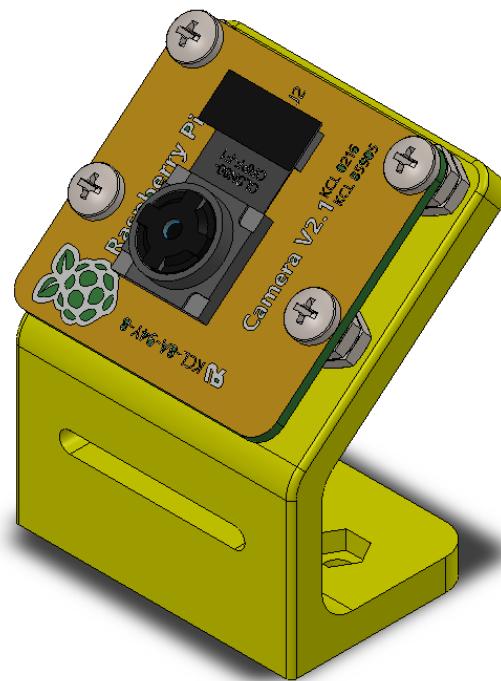


Figura 4.1.18 Subansamblu Camera – vedere izometrică

Deși în modelul 3D nu este prezent, în realitate camera are pe spate în zona inferioară portul pentru cablul CSI. Acest port are o înălțime de aproximativ 3mm și împiedică fixarea camerei direct pe suport. Portul CSI este evidențiat în Figura 4.1.19.



Figura 4.1.19 Camera raspberry - spate

Pentru a rezolva aceasta problema au fost adăugate câte 2 piulițe M2 pe fiecare șurub de prindere. (Figura 4.1.20)

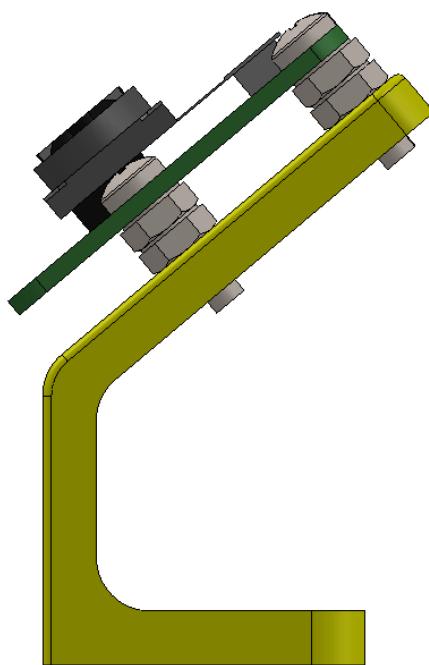


Figura 4.1.20 Subansamblu Camera - vedere din lateral

Camera este înclinată la un unghi de  $40^{\circ}$  față de orizontală, pentru a oferi o perspectivă mărită asupra componentelor autoturismului.

De asemenea, în zona inferioară a suportului este prezent un alezaj prin care se introduce cablul pentru conectarea camerei. (Figura 4.1.21)

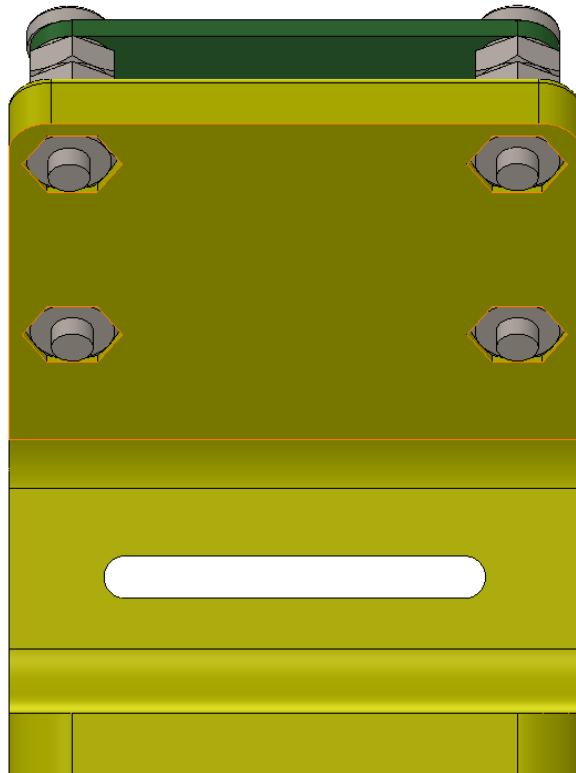


Figura 4.1.21 Subansamblu Camera - Vedere din spate

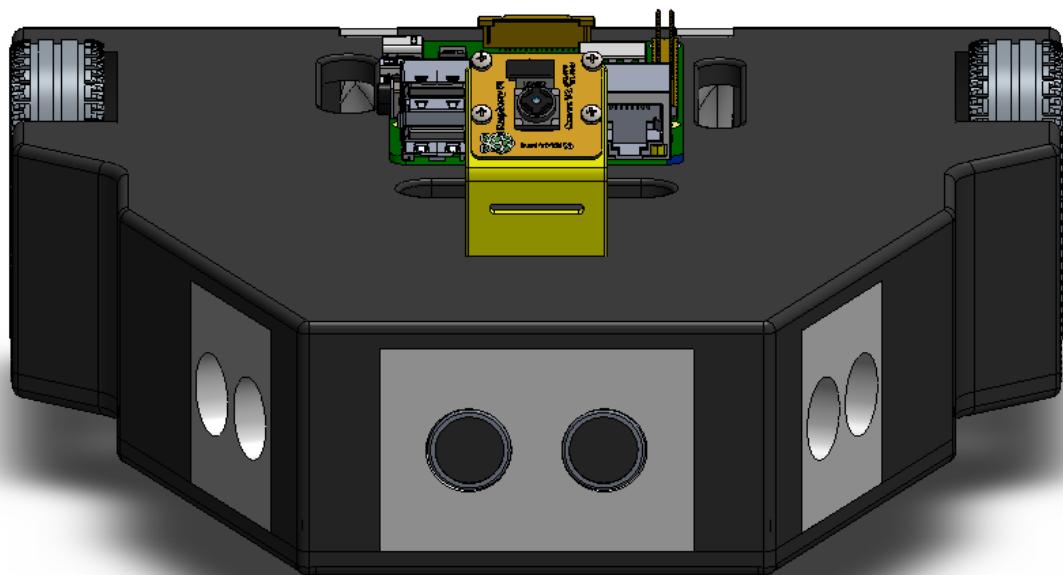


Figura 4.1.22

In figurile următoare se vor reprezenta etapele asamblării camerei pe suport:



Figura 4.1.23 Componentele necesare

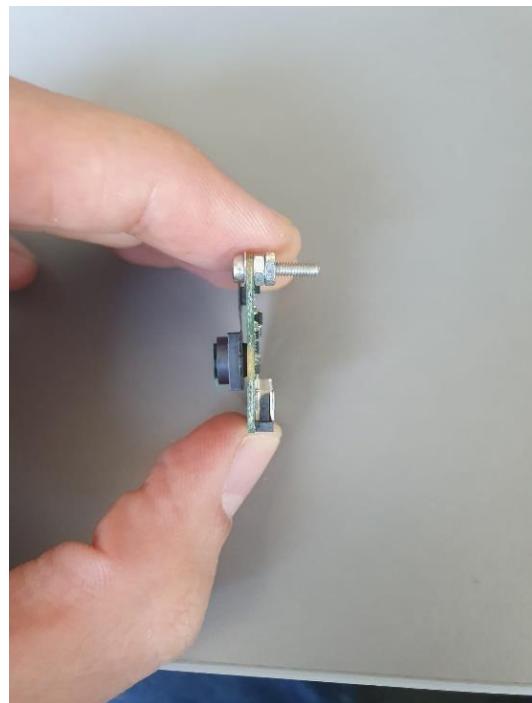


Figura 4.1.24 Adăugarea piulițelor pentru distanțare



Figura 4.1.25 Subansamblu final - vedere din lateral

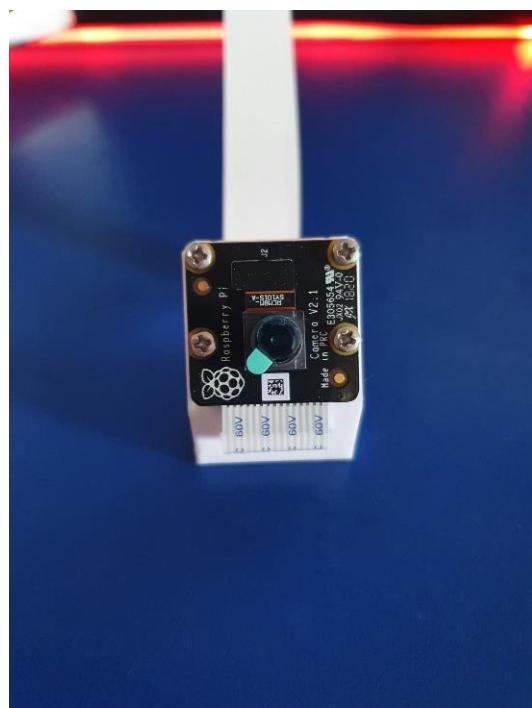


Figura 4.1.26 Subansamblu final - vedere din fata

#### 4.1.5 Proiectare suport acumulator

Suportul acumulatorului se fixează direct pe sașiu, fără șuruburi, caracteristică ce facilitează înlocuirea sau încărcarea bateriei. Cele patru găuri de diametru 10mm au fost proiectate pentru reduce consumul de filament și pentru a reduce timpul de printare.

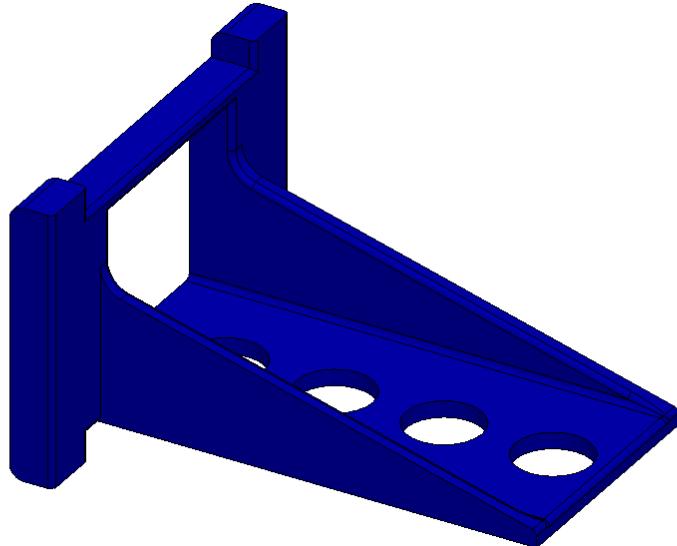


Figura 4.1.27 Suport acumulator

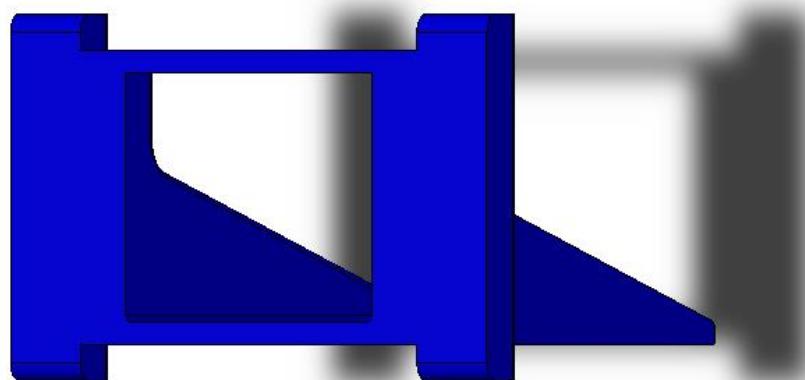


Figura 4.1.28 Suport acumulator

#### 4.1.6 Realizare roată omnidirecțională

Roata omnidirecțională este compusă dintr-un cilindru cu diametrul de 50mm care prezintă alezaje pentru role. Sunt 18 role de diametru 8mm și înălțime 8mm dispuse simetric pe ambele părți ale roții. Rolele sunt prinse cu tija metalică de diametru 1mm. Atât cilindrul cat și rolele au fost realizate prin procedeul de printare 3D.



Figura 4.1.29 roată omnidirecțională

In următoarele figuri se vor reprezenta etapele asamblării roții:



Figura 4.1.30 Dimensionarea tijelor metalice

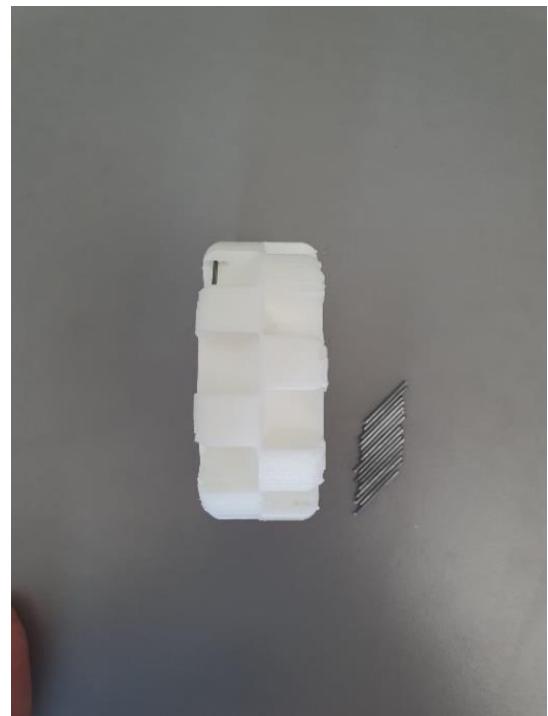


Figura 4.1.31 Verificare tijelor metalice

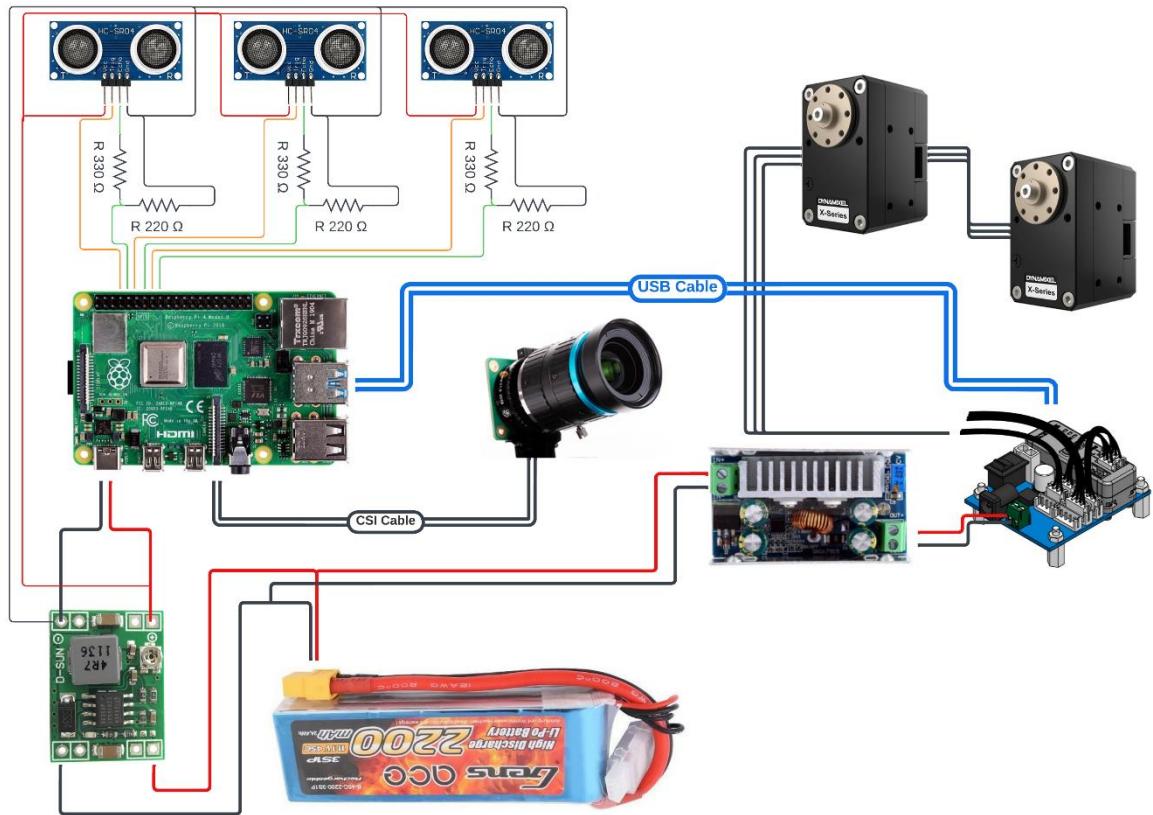


Figura 4.1.32 Montarea rolelor



Figura 4.1.33 Ansamblu final

## 4.2 Alegerea componentelor electrice



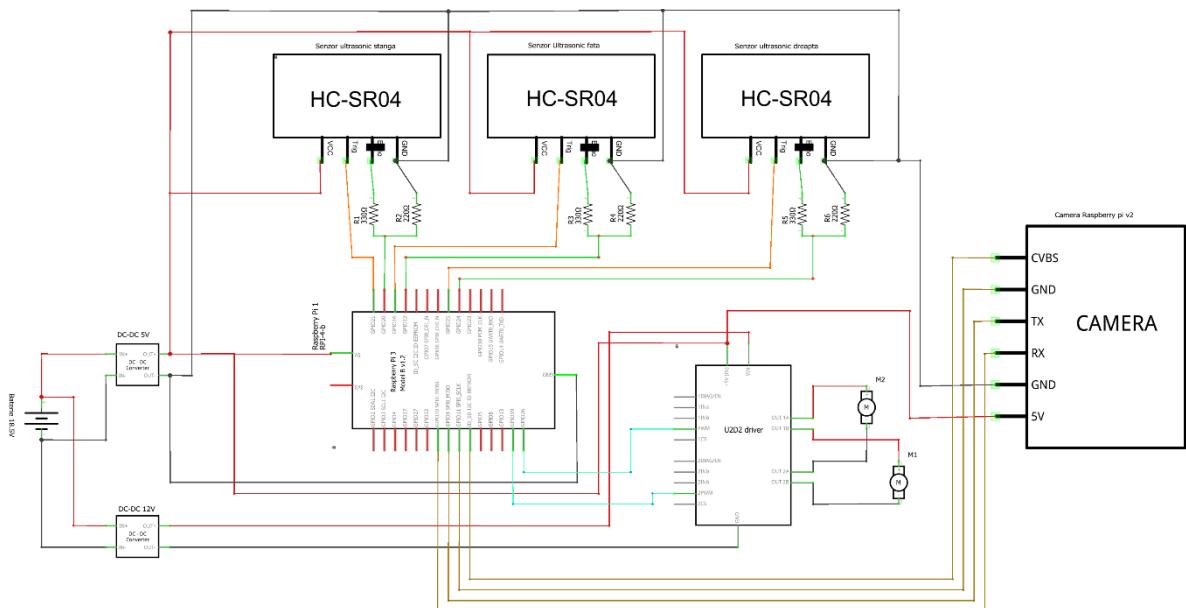
Figură 4.2.1 Schema componentelor electrice

Conform figurii 4.2.1, din punct de vedere electronic, sistemul este alcătuit din două servomotoare Dynamixel XM430-W210-T acționate de driver-ul U2D2, trei senzori ultrasonici HC-SR04, o camera Raspberry High Quality și un microcontroler Raspberry Pi 4 Model B.

Toate componentele sunt alimentate de bateria Gens Ace de 18.5V 2200mAh prin intermediul a două surse coborâtoare de tensiune, una folosită pentru a furniza tensiunea de 12V pentru motoare, iar cealaltă, MP1584EN, alimentează restul componentelor cu o tensiune de 5V.

Alimentarea motoarelor se face printr-o componentă dedicată, U2D2 Power HUB, care transferă energia furnizată de baterie către ambele motoare în același timp.

De asemenea, există 3 circuite de tip divizor de tensiune, câte unul pentru fiecare senzor ultrasonic, care au rolul de a proteja porturile GPIO ale microcontrolerului Raspberry Pi de o tensiune mai mare de 3.3V. Modul în care funcționează acest circuit va fi prezentat în următoarele capituloare.



Figură 4.2.2 Schema electrică

## 4.3 Alegerea elementelor de acționare

### 4.3.1 Servomotor – Dynamixel XM430-W210-T

Pentru a cunoaște poziția robotului în timp real este necesară folosirea unui motor dotat cu encoder. Un motor cu encoder este un tip de motor în care un encoder rotativ este montat pe

partea sa din spate, care oferă feedback sistemului prin urmărirea vitezei sau a poziției arborelui motorului.

Motorul ales este Dynamixel XM430-W210-T deoarece permite citirea poziției și a vitezei în timp real și are caracteristicile tehnice potrivite pentru sistemul propus.

Seria Dynamixel X-Series este actuala serie de servomotoare de rețea de la ROBOTIS. Servomotorul XM430-W210-T este echipat cu un encoder magnetic fără contact care permite rotiri de  $360^{\circ}$  cu până la 95RPM. Este capabil de un cuplu de până la 3,7N.m cu o tensiune de 14,8V. Această caracteristică remarcabilă, combinată cu o greutate redusă de 82 g, rezultă într-un servomotor foarte eficient pentru un domeniu universal de aplicații.



Figură 4.3.1 Dynamixel XM430-W210 [27]

Caracteristici importante ale motorului Dynamixel XM430-W210 (ANEXA Tabel 1):

- Tensiune de alimentare: 10.0 – 14.8 V (Recomandat 12V)
- Cuplu maxim: 3.0 Nm la 12V 2.3A
- Viteza mers în gol: 77 rev/min la 12V
- Microcontroler integrat (MCU): ARM CORTEX-M3
- Senzor poziție: encoder fără contact 12bit  $360^{\circ}$
- Rezoluție encoder: 4096 [pulsuri/rotație]
- Moduri de operare: control viteză / control poziție
- Sarcină maximă radială: 40N la 10mm față planul frontal al motorului
- Sarcină maximă axială: 20N
- Curent consumat în standby: 40mA
- Temperatura de operare:  $-5^{\circ} - +80^{\circ}\text{C}$
- Dimensiuni: 28.5 x 46.5 x 34 [mm]

- Greutate: 82 [g]
- Feedback senzori: Poziție, Viteză, Curent, Temperatură, Tensiune de alimentare

Pentru alegerea motoarelor se va tine cont de cuplul minim necesar pentru a putea pune în mișcare robotul. Înținând cont de greutatea finală a robotului, aproximativ 1.5Kg și distanța de la axul motorului la suprafață de contact a roții (39.75mm), momentul minim necesar rezultă din ecuația:

$$M_m = d * m * g \quad (1)$$

$$M_m = 39.75 * 10^{-3} * 1.5 * 9.81 \quad (2)$$

$$M_m = \sim 0.6 \text{ Nm} \quad (3)$$

Motorul ales produce un cuplu maxim de 3 Nm la tensiunea de 12V, cuplul necesar pentru punerea în mișcare a robotului este dat de ecuația (2) și este aproximativ 0.6 Nm, rezultă că motorul ales este potrivit pentru aplicația prezentată.

Tabel 1 Moduri de funcționare pentru motorul Dynamixel

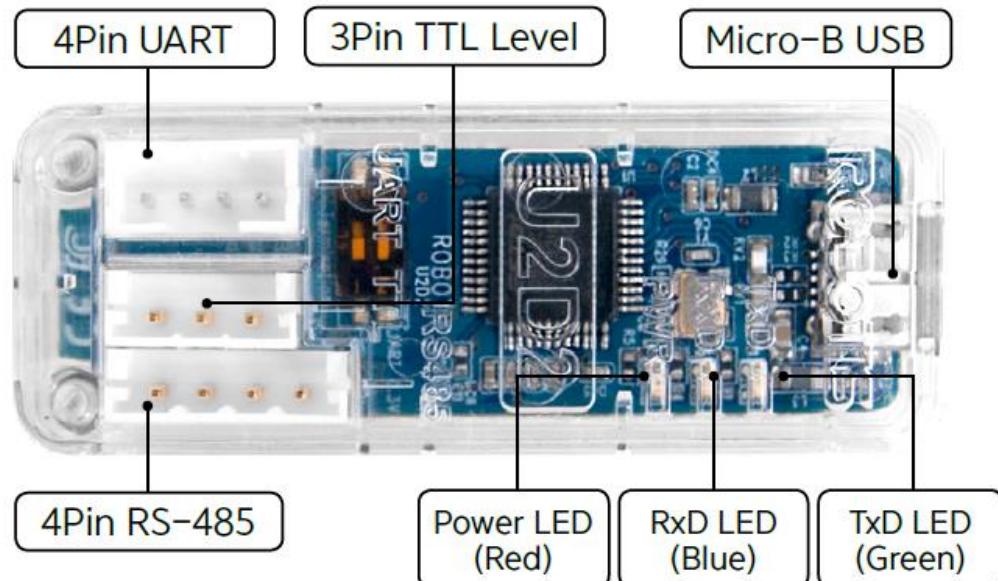
Mod de operare	Descriere
Modul de control al curentului	DYNAMIXEL controlează numai curentul (cuplul) indiferent de viteză și poziție. Acest mod este ideal pentru un dispozitiv de prindere sau pentru un sistem care utilizează numai controlul curentului(cuplului) sau pentru un sistem care dispune de controlere suplimentare de viteză/poziție.
Modul de control al vitezei	Acest mod controlează viteza. Acest mod este identic cu Wheel Mode(endless) din motoarele DYNAMIXEL existente. Este ideal pentru roboții cu roți.
Modul de control al poziției	Acest mod controlează poziția. Intervalul de funcționare a poziției este limitat de Limita de poziție maximă și limita de poziție minimă . Este folosit pentru roboții articulați unde fiecare articulație se rotește mai puțin de 360 de grade.

Modul de control al poziției extinse	Acest mod controlează poziția. Sunt acceptate 512 ture (-256 ~ 256). Acest mod este ideal pentru sistemele de transport sau pentru un sistem care necesită un reductor suplimentar. Limita de poziție maximă și limita de poziție minimă nu sunt utilizate în modul de control extins al poziției.
Modul de control al poziției bazat pe curent	Acest mod controlează atât poziția, cât și curentul (cuplul). Sunt acceptate până la 512 ture (-256 ~ 256). Acest mod este ideal pentru un sistem care necesită atât controlul poziției, cât și al curentului, cum ar fi roboții articulați sau dispozitivele de prindere.
Mod de control PWM (mod de control al tensiunii)	Acest mod controlează direct ieșirea PWM. (Modul de control al tensiunii)

#### 4.3.2 Driver motor

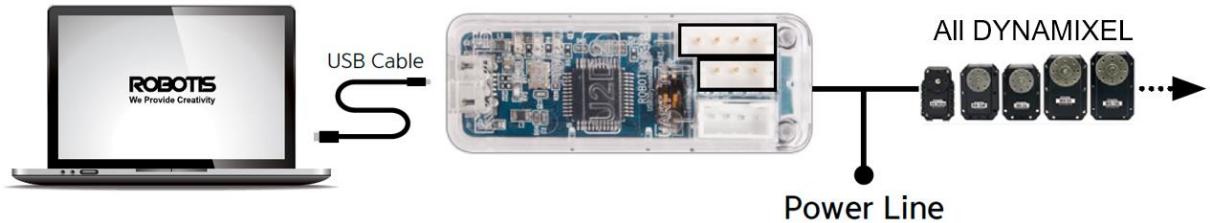
Ținând cont că motoarele alese se pot controla doar prin semnal tip TTL (interfață serial) este necesară alegerea unui driver. Driver ul a fost ales ținând-se cont de tipul de semnal pe care motoarele îl pot folosi și de tensiunea maxima și curentul maxim acceptate de motoare.

Driverul ales pentru controlul acestor motoare este U2D2. U2D2 este un convertor de comunicație USB de dimensiuni mici care permite controlul și operarea motoarelor DYNAMIXEL cu un PC.



Figură 4.3.2 Driver U2D2 [30]

U2D2 poate fi conectat la portul USB al PC-ului cu ajutorul cablului USB. Suportă atât conectorul TTL cu 3 pini, cât și conectorul RS-485 cu 4 pini pentru a se conecta cu diverse motoare DYNAMIXEL-uri. U2D2 nu alimentează motoarele, prin urmare, o sursă de alimentare externă trebuie să furnizeze energie pentru motoare.[30]

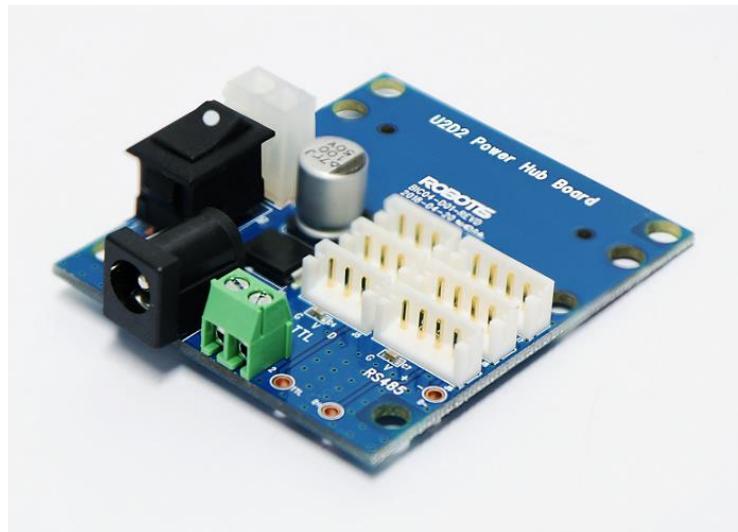


Figură 4.3.3 Schema conexiuni driver [30]

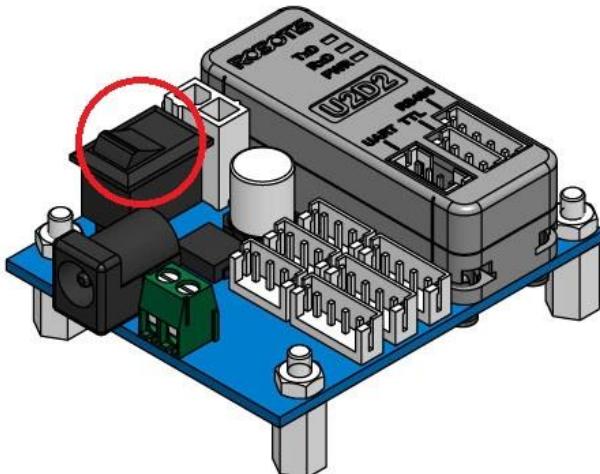
Specificații:

- Greutate: 9g
- Dimensiuni: 48 x 18 x 14.6 [mm]
- Baudrate: Maximum 6Mbps
- Porturi: - 3Pin TTL Level(TTL Communication supported DYNAMIXEL)
  - 4Pin RS-485(RS-485 Communication supported DYNAMIXEL)
  - 4Pin UART(Controllers that support 4Pin UART)

Alimentarea motoarelor se face separat folosind hub-ul de alimentare U2D2 POWER HUB. Combinat cu driverul U2D2, acesta poate folosi diverse tipuri de surse de alimentare externe pentru a furniza energie motoarelor DYNAMIXEL.



Figură 4.3.4 U2D2 POWER HUB [31]

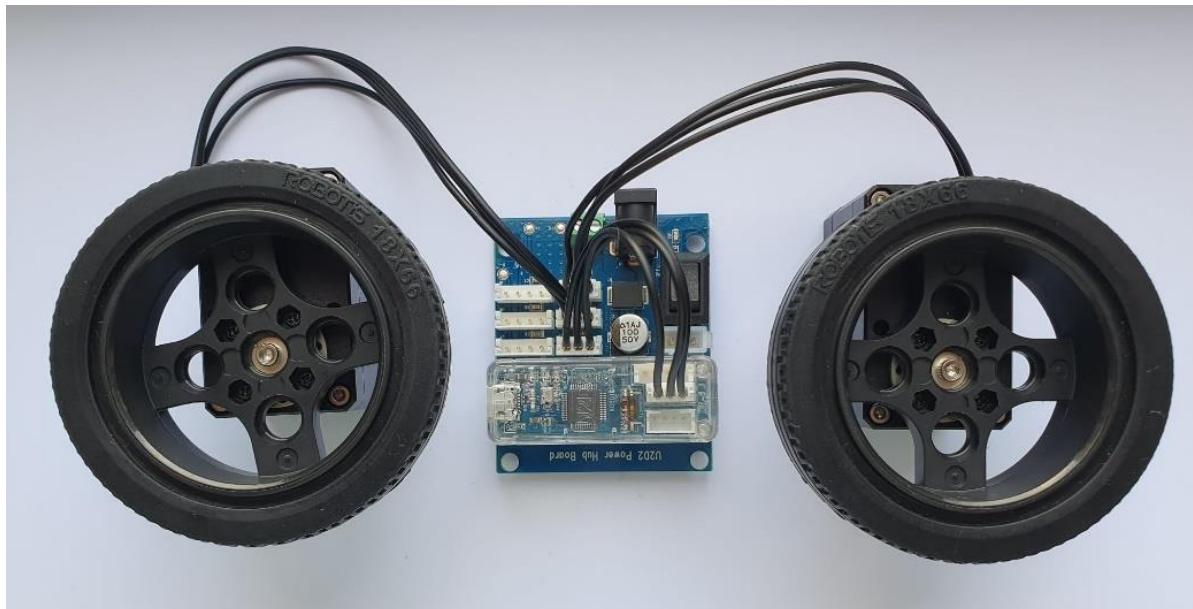


Figură 4.3.5 Ansamblu driver U2D2 + power hub [31]

Specificații tehnice:

- Tensiune de operare: 3.5V – 24V
- Curent maxim: 10A
- Buton Power ON/OFF
- 3 Tipuri de conectori pentru alimentare

### 4.3.3 Testarea motoarelor



Figură 4.3.6 Conectarea motoarelor la driver

Pentru a citi poziția motorului folosind U2D2 se foloseste următorul script python:

```
#!/usr/bin/env python
# -*- coding: utf-8 -*-

import os

if os.name == 'nt':
    import msvcrt
    def getch():
        return msvcrt.getch().decode()
```

```

else:
    import sys, tty, termios
    fd = sys.stdin.fileno()
    old_settings = termios.tcgetattr(fd)
    def getch():
        try:
            tty.setraw(sys.stdin.fileno())
            ch = sys.stdin.read(1)
        finally:
            termios.tcsetattr(fd, termios.TCSADRAIN, old_settings)
        return ch

from dynamixel_sdk import * # Uses Dynamixel SDK library

***** DYNAMIXEL Model definition *****
***** (Use only one definition at a time) *****
MY_DXL = 'X_SERIES'      # X330 (5.0 V recommended), X430, X540, 2X430

# Control table address
if MY_DXL == 'X_SERIES' or MY_DXL == 'MX_SERIES':
    ADDR_TORQUE_ENABLE      = 64
    ADDR_GOAL_POSITION       = 116
    ADDR_PRESENT_POSITION    = 132
    DXL_MINIMUM_POSITION_VALUE = 0          # Refer to the Minimum Position Limit of product
eManual
    DXL_MAXIMUM_POSITION_VALUE = 4095        # Refer to the Maximum Position Limit of product
eManual
    BAUDRATE                = 1000000
# DYNAMIXEL Protocol Version (1.0 / 2.0)
# https://emanual.robotis.com/docs/en/dxl/protocol2/
PROTOCOL_VERSION         = 2.0

# Factory default ID of all DYNAMIXEL is 1
DXL_ID                   = 1

# Use the actual port assigned to the U2D2.
# ex) Windows: "COM*", Linux: "/dev/ttyUSB*", Mac: "/dev/tty.usbserial-*"
DEVICENAME                = '/dev/ttyUSB0'

TORQUE_ENABLE              = 1      # Value for enabling the torque
TORQUE_DISABLE              = 0      # Value for disabling the torque
DXL_MOVING_STATUS_THRESHOLD = 20     # Dynamixel moving status threshold

index = 0
dxl_goal_position = [DXL_MINIMUM_POSITION_VALUE, DXL_MAXIMUM_POSITION_VALUE]      # Goal
position

# Initialize PortHandler instance
# Set the port path

```

```

# Get methods and members of PortHandlerLinux or PortHandlerWindows
portHandler = PortHandler(DEVICENAME)

# Initialize PacketHandler instance
packetHandler = PacketHandler(PROTOCOL_VERSION)

# Open port
if portHandler.openPort():
    print("Succeeded to open the port")
else:
    print("Failed to open the port")
    print("Press any key to terminate...")
    getch()
    quit()

# Set port baudrate
if portHandler.setBaudRate(BAUDRATE):
    print("Succeeded to change the baudrate")
else:
    print("Failed to change the baudrate")
    print("Press any key to terminate...")
    getch()
    quit()

# Enable Dynamixel Torque
dxl_comm_result,     dxl_error      =     packetHandler.write1ByteTxRx(portHandler,      DXL_ID,
ADDR_TORQUE_ENABLE, TORQUE_ENABLE)
if dxl_comm_result != COMM_SUCCESS:
    print("%s" % packetHandler.getTxRxResult(dxl_comm_result))
elif dxl_error != 0:
    print("%s" % packetHandler.getRxPacketError(dxl_error))
else:
    print("Dynamixel has been successfully connected")

while 1:
    print("Press any key to continue! (or press ESC to quit!)")
    if getch() == chr(0x1b):
        break

    # Write goal position
    if (MY_DXL == 'XL320'): # XL320 uses 2 byte Position Data, Check the size of data in your
DYNAMIXEL's control table
        dxl_comm_result, dxl_error = packetHandler.write2ByteTxRx(portHandler,   DXL_ID,
ADDR_GOAL_POSITION, dxl_goal_position[index])
    else:
        dxl_comm_result, dxl_error = packetHandler.write4ByteTxRx(portHandler,   DXL_ID,
ADDR_GOAL_POSITION, dxl_goal_position[index])
    if dxl_comm_result != COMM_SUCCESS:
        print("%s" % packetHandler.getTxRxResult(dxl_comm_result))

```

```

        elif dxl_error != 0:
            print("%s" % packetHandler.getRxPacketError(dxl_error))

    while 1:
        # Read present position
        if (MY_DXL == 'XL320'): # XL320 uses 2 byte Position Data, Check the size of data in
        your DYNAMIXEL's control table
            dxl_present_position,           dxl_comm_result,           dxl_error      =
        packetHandler.read2ByteTxRx(portHandler, DXL_ID, ADDR_PRESENT_POSITION)
        else:
            dxl_present_position,           dxl_comm_result,           dxl_error      =
        packetHandler.read4ByteTxRx(portHandler, DXL_ID, ADDR_PRESENT_POSITION)
        if dxl_comm_result != COMM_SUCCESS:
            print("%s" % packetHandler.getTxRxResult(dxl_comm_result))
        elif dxl_error != 0:
            print("%s" % packetHandler.getRxPacketError(dxl_error))

        print("[ID:%03d] GoalPos:%03d PresPos:%03d" % (DXL_ID, dxl_goal_position[index],
        dxl_present_position))

        if      not      abs(dxl_goal_position[index]      -      dxl_present_position)      >
DXL_MOVING_STATUS_THRESHOLD:
            break

        # Change goal position
        if index == 0:
            index = 1
        else:
            index = 0

        # Disable Dynamixel Torque
        dxl_comm_result,   dxl_error      =   packetHandler.write1ByteTxRx(portHandler,       DXL_ID,
ADDR_TORQUE_ENABLE, TORQUE_DISABLE)
        if dxl_comm_result != COMM_SUCCESS:
            print("%s" % packetHandler.getTxRxResult(dxl_comm_result) )
        elif dxl_error != 0:
            print("%s" % packetHandler.getRxPacketError(dxl_error))

        # Close port
        portHandler.closePort()

```

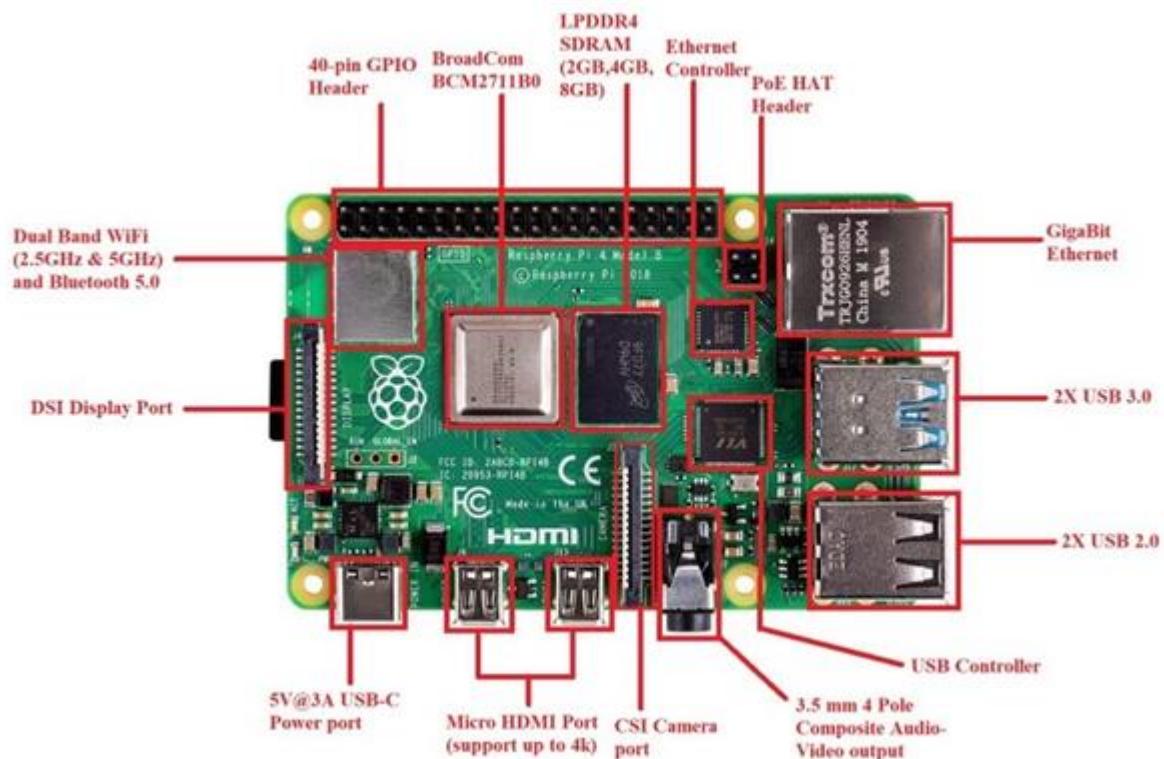
Codul prezentat mai sus comunica cu unul dintre motoarele dynamixel si comandă o rotire de 180° la apăsarea oricărei taste. În timpul rotirii programul citește și afișează poziția curentă a motorului pana când acesta ajunge în poziția dorită. După ce motorul a efectuat o rotire de 180° și se apasă din nou orice tastă, motorul se rotește cu 180° în direcția opusă și transmite în timp real poziția curentă.

## 4.4 Microcontroler

Raspberry Pi 4 Model B este cea mai recentă placă lansată de Fundația Raspberry Pi în iunie 2019. Acest model dispune de cel mai recent procesor quad-Core de înaltă performanță Broadcom 2711 pe 64 de biți, Cortex A72, cu o viteză de 1,5 GHz. Acest procesor utilizează cu 20% mai puțină energie și oferă o performanță cu 90% mai mare decât modelul anterior.

Celelalte caracteristici noi ale plăcii sunt suportul pentru două ecrane cu rezoluții de până la 4k prin intermediul unei perechi de porturi micro-HDMI, decodare video hardware până la 4Kp60, LAN wireless dual-channel 2.4/5.0GHz, Gigabit Ethernet, două porturi USB 3.0, Bluetooth 5.0 și capacitate PoE (prin intermediul unei plăci PoE HAT separate).[42]

Microcontroler-ul Raspberry PI 4 este ideal pentru aceasta aplicație deoarece interfața GPIO ne permite conectarea la senzorii ultrasonici, interfața USB este folosită pentru driverul motorului și de asemenea portul MIPI CSI permite conectarea camerei RPI.

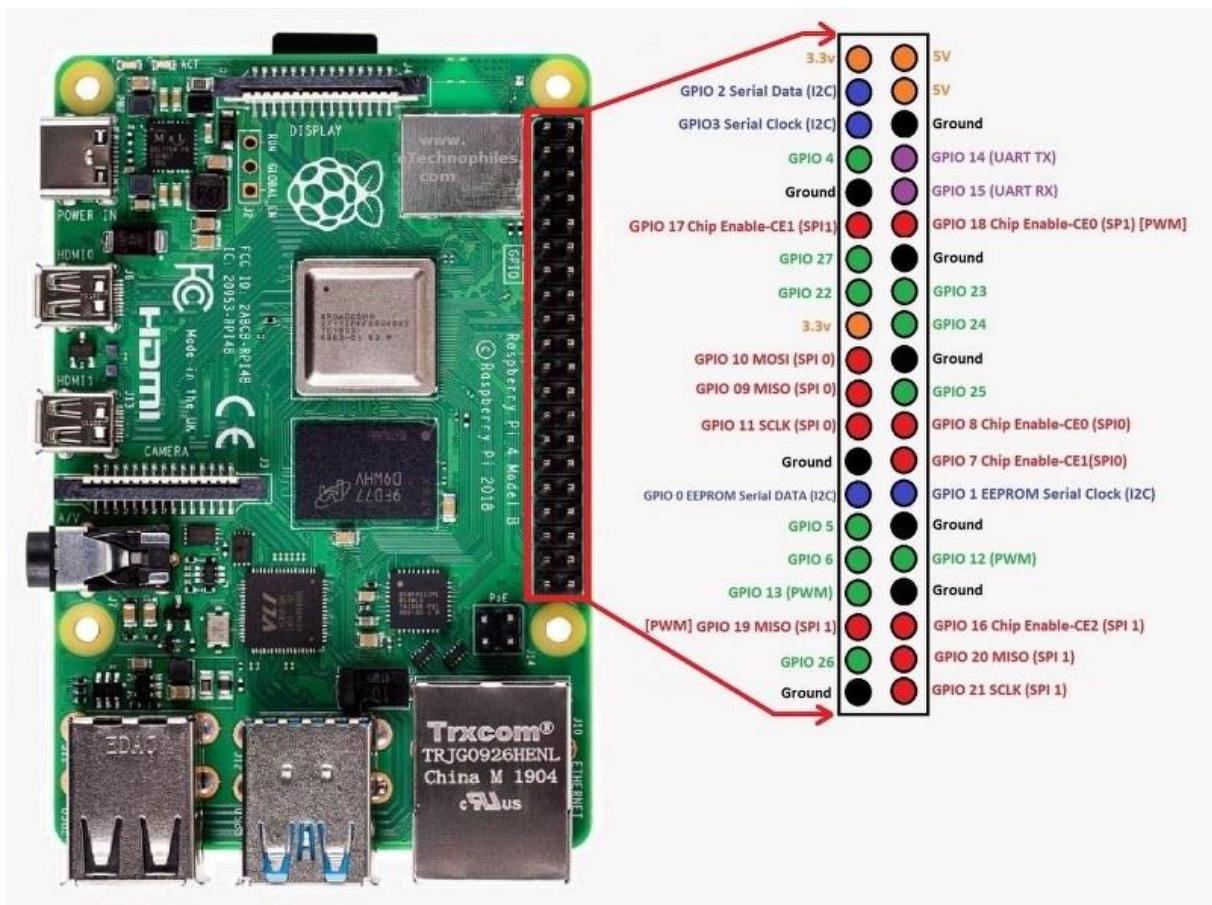


Figură 4.4.1 Componente Raspberry PI 4 Model B[42]

Principalele caracteristici ale computerului Raspberry Pi 4 sunt:

- Procesor Broadcom BCM 2711, Quad core Cortex-A72 pe 64 de biți;

- 2 porturi USB 3.0 și 2 porturi USB 2.0;
- Port MIPI CSI pentru conectarea camerei web
- Memorie RAM de 2GB, 4GB sau 8GB;
- Suporta Gigabit Ethernet;
- Tensiunea de alimentare este de 5V, curent continuu;
- Temperatură de operare cuprinsă între 0 °C - 50 °C;
- 2 porturi Micro-HDMI;
- 40 de pini GPIO (General-Purpose Input/Output);
- Conexiune Wireless de 2.4 GHz sau de 5.0 GHz



Figură 4.4.2 Raspberry PI GPIO Pinout

Raspberry Pi GPIO reprezintă pinii de intrare și ieșire cu scop general. Acești pini sunt utilizați pentru a conecta placă Raspberry pi la dispozitive periferice externe de intrare/ieșire. Acest model B conține un bloc GPIO cu 40 de pini. Din acești 40 de pini, 26 de pini sunt pini GPIO.[41][42]

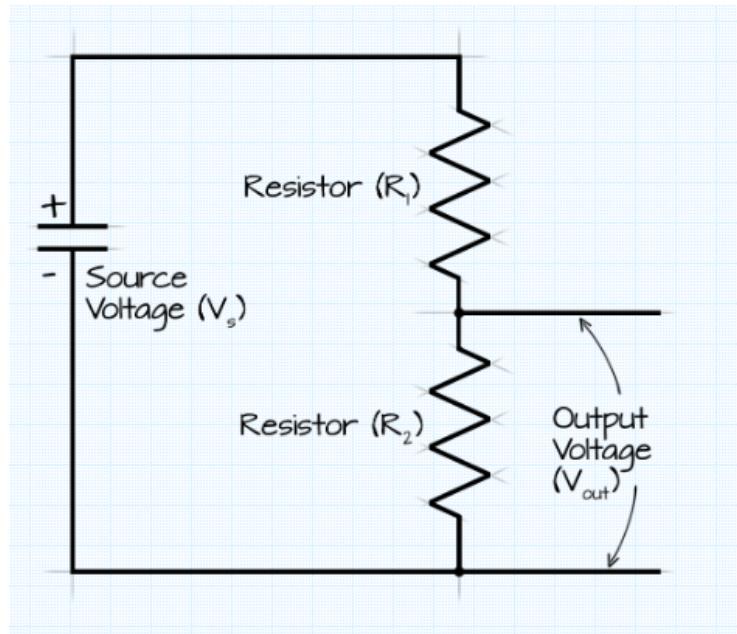
Pentru sistemul prezentat este necesară folosirea pinilor de alimentare de 5V, pin-ul de Ground și o parte din pinii GPIO pentru conectarea senzorilor ultrasonici.

Un pin care poate fi setat ca intrare sau ieșire și care este controlat în timpul funcționării se numește pin GPIO. Un pin GPIO setat ca intrare permite ca semnalul transmis de orice dispozitiv extern (conectat la acest pin) să fie recepționat de Raspberry Pi. Tensiunea de intrare între 1,8V și 3,3V este citită ca fiind HIGH de către Raspberry pi. Iar atunci când tensiunea de intrare este mai mică de 1,8V, aceasta este citită ca fiind LOW. Un pin GPIO setat ca ieșire furnizează HIGH/3,3V sau LOW/0V.

Spre deosebire de microcontroler-ul Arduino, unde nivelul logic 1/HIGH este reprezentat de tensiunea de 5V, nivelul logic 1/HIGH al microcontrolerului Raspberry Pi 4 este de doar 3.3V. De aceea nu trebuie conecta niciun dispozitiv extern cu o tensiune de ieșire mai mare de 3,3 V la niciunul dintre pinii GPIO. Dacă se întâmplă acest lucru placa Raspberry Pi se arde.

Pentru a rezolva aceasta problema, se poate folosi un simplu circuit pentru a reduce tensiunea de ieșirea a componentelor de la 5V la 3.3V. Acest circuit se numește divizor de tensiune.

Un circuit divizor de tensiune este un circuit foarte comun care ia o tensiune mai mare și o transformă într-o tensiune mai mică prin utilizarea unei perechi de rezistori. [40]



Figură 4.4.3 Divizor de tensiune - Schema de principiu [40]

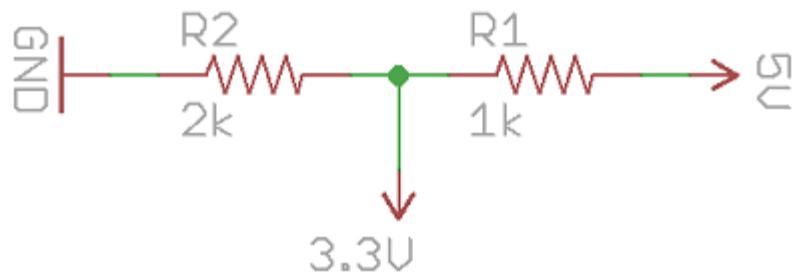
Formula de calcul a tensiunii de ieșire se bazează pe legea lui Ohm și este prezentată mai jos.

$$V_{out} = V_s \cdot \frac{R_2}{R_1 + R_2} \quad (1)$$

Unde:

- $V_s$  este tensiunea sursei, măsurată în volți (V).
- $R_1$  este rezistența primului rezistor, măsurată în ohmi ( $\Omega$ ).
- $R_2$  este rezistența celui de-al doilea rezistor, măsurată în ohmi ( $\Omega$ ).
- $V_{out}$  este tensiunea de ieșire, măsurată în volți (V).

Rezolvând ecuația (1) folosind valorile  $V_s = 5V$ ,  $V_{out} = 3.3V$  și  $R_1 = 1k\Omega$  rezultă  $R_2 \cong 2k\Omega$ .



Figură 4.4.4 Circuit electric divizor tensiune de la 5V la 3.3V

## 4.5 Elemente senzoriale

### 4.5.1 Senzori ultrasonici HC-RS04

Senzorul ultrasonic ales este HC-SR04 deoarece are dimensiuni reduse, este ușor de folosit și prețul de achiziție este scăzut.

Senzorul de distanță cu ultrasunete HC-SR04 este un senzor utilizat pentru detectarea distanței până la un obiect cu ajutorul sonarului.



Figură 4.5.1 Senzor ultrasonic HC-SR04 [28]

HC-SR04 utilizează un sonar cu ultrasunete fără contact pentru a măsura distanța până la un obiect și este format din două transmițătoare cu ultrasunete (practic difuzeoare), un receptor și un circuit de control. Transmițătoarele emit un sunet ultrasonic de înaltă frecvență, care ricoșează în orice obiect solid din apropiere, iar receptorul ascultă orice ecou de întoarcere. Ecoul este apoi procesat de circuitul de control pentru a calcula diferența de timp dintre semnalul transmis și cel recepționat. Acest timp poate fi utilizat ulterior, împreună cu câteva calcule matematice, pentru a calcula distanța dintre senzor și obiectul reflectorizat.

Caracteristici ale senzorului HC-SR04[28]:

- Tensiune de alimentare: 5V;
- Curent maxim: 20mA;
- Output digital HIGH: 5V;
- Output digital LOW: 0V;
- Temperatura de operare: -15° - 70°C;
- Interval de măsurare: 2 – 400[cm];
- Frecvența ultrasunetelor: 40kHz
- Unghi de detectare: 30°
- Unghi de acțiune: 15°
- Dimensiuni: 43 x 20 x 14 [mm]

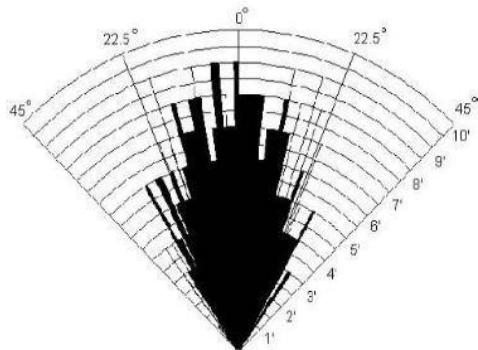
Pinout:

- Pin1: VCC
- Pin2: TRIG (Input pin)
- Pin3: ECHO (Output pin)
- Pin4: GND

Senzorul HC-SR04 are patru pini, VCC, GND, TRIG și ECHO; toți acești pini au funcții diferite. Pinii VCC și GND sunt cei mai simpli, ei alimentează senzorul. Acești pini trebuie să fie conectați la o sursă de +5 volți și, respectiv, la ground.

Există un singur pin de control: pinul TRIG. Pinul TRIG este responsabil pentru trimiterea impulsului ultrasonic. Acest pin trebuie să fie setat la HIGH timp de 10  $\mu$ s, moment în care HC-SR04 va trimite o rafală de 8 cicluri sonice la 40 kHz. După ce a fost trimisă o

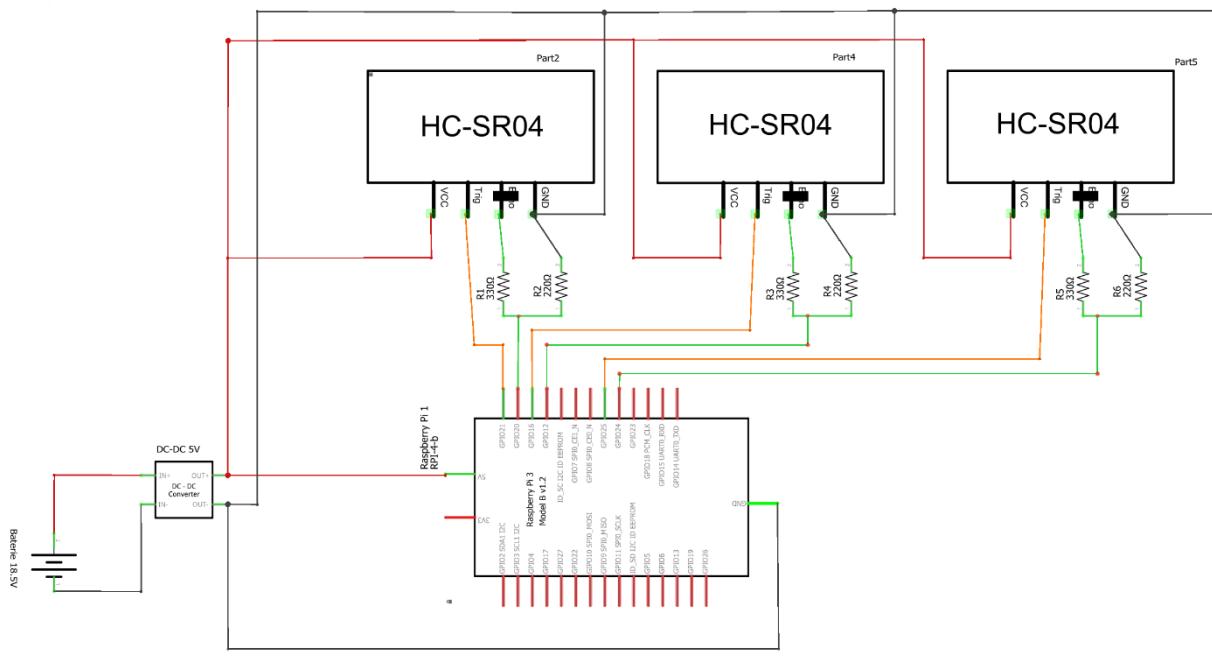
rafală sonică, pinul ECHO va deveni HIGH. Pinul ECHO este pinul de date și este utilizat pentru măsurarea distanței. După ce o rafală ultrasonică este trimisă, pinul va deveni HIGH, va rămâne HIGH până când o rafală ultrasonică este detectată înapoi, moment în care va deveni LOW.[29]



Figură 4.5.2 Intervalele unghiulare de eficiență ale senzorului ultrasonic HC-SR04

Conform figurii 4.5.2 acest tip de senzor este eficient în măsurarea distanței față de obiect atunci când acesta se află într-un interval descris de un con cu unghiul de  $45^{\circ}$  în fața senzorului. De aceea este necesară utilizarea unui număr de 3 astfel de senzori, toți fiind instalati pe placă de jos a șasiului.

Schema de conectare a acestor senzori este prezentată în figura 4.5.3.



Figură 4.5.3 Schema electrică senzori ultrasonici conectați la Raspberry pi

#### 4.5.2 Testare senzor ultrasonic

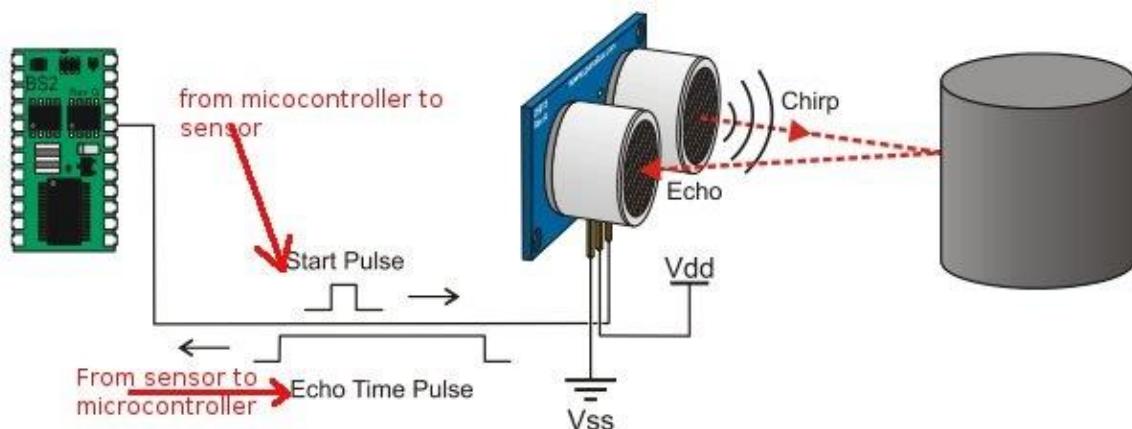
Măsurarea distanței se face trimițând pe pinul GPIO 23 al plăcuței Raspberry Pi un semnal cu o durată de 10 micro secunde. Pe durata semnalului, senzorul generează unde sonore la o frecvență de 40kHz. Undele acustice se deplasează până când întâmpină un obstacol, moment în care sunt reflectate înapoi la senzor. În momentul în care senzorul detectează undele acustice reflectate, trimit un semnal electric plăcuței Raspberry Pi. Intervalul de timp dintre trimitera semnalului de 10 microsecunde către senzor, până la primirea semnalului corespunzător undei reflectate, este cronometrat, iar pe baza lui se calculează distanța de la senzor la obiect.

Viteza sunetului este aproximativ 340 de metri pe secundă în aer la temperatura de 20° Celsius și presiune de 1 atm. Formula de calcul a distanței este (1), iar cea folosită în codul python este (2).

$$d = v \cdot t \quad (1)$$

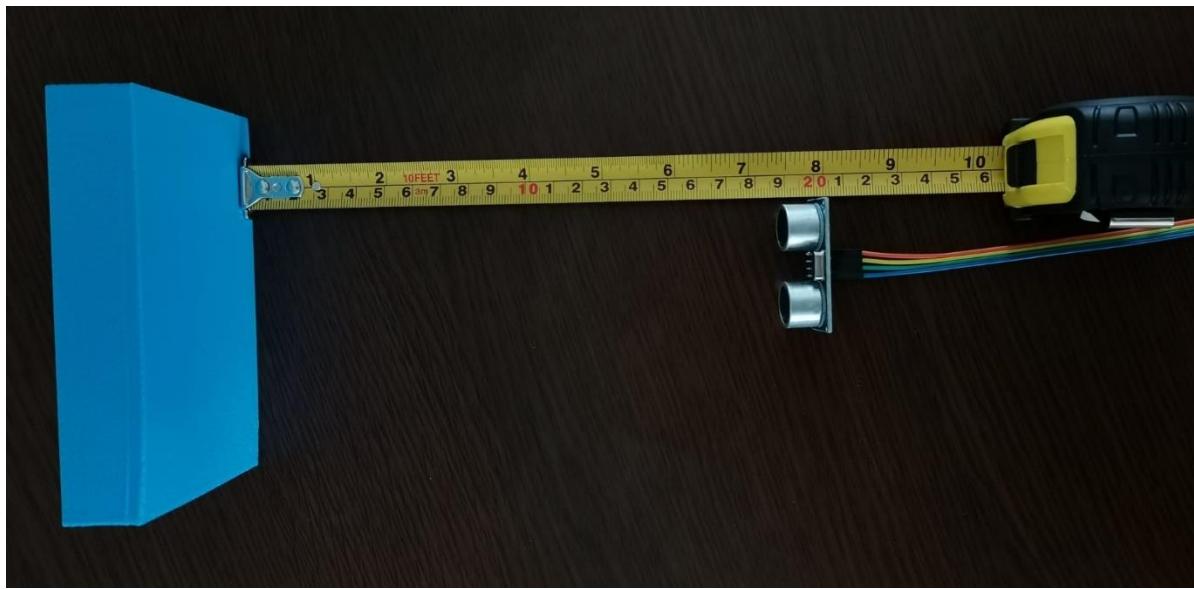
$$distance = pulse\_duration * 0.034 / 2 \quad (2)$$

Intervalul de timp este înjumătățit, deoarece unda sonoră parurge o distanță de 2 ori mai mare decât cea dintre senzor și obiect (dus-întors), așa cum se vede în figura următoare.

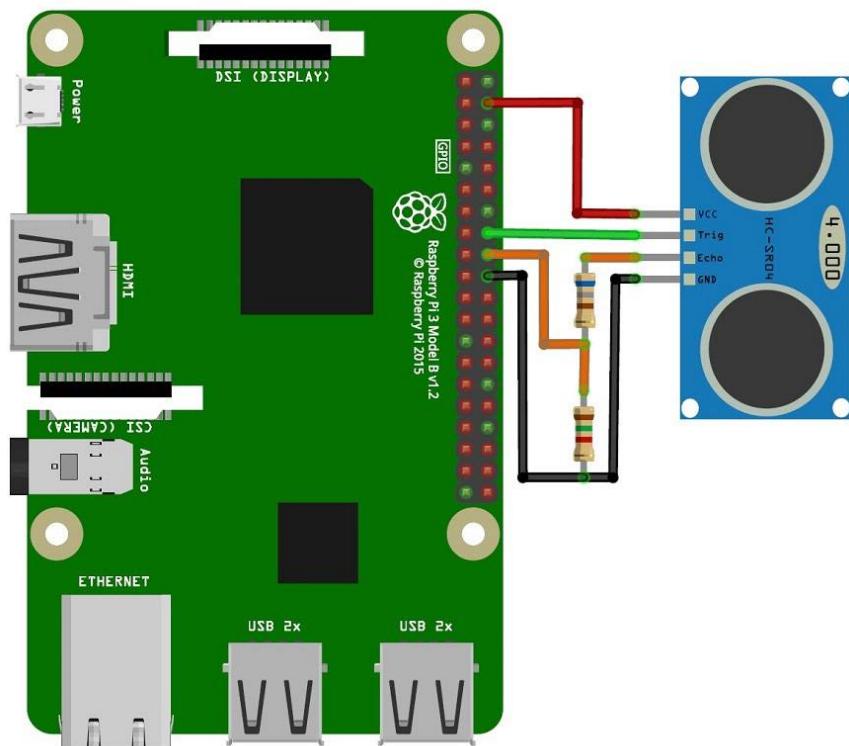


Figură 4.5.4 Principiu de funcționare al senzorului

Totodată, intervalul de timp este măsurat în milisecunde, așa că pentru a obține distanță în centimetri este necesar ca viteza de 340m/s să fie împărțită la 10000, de aici coeficientul de 0.034 din formulă.



Figură 4.5.5 Scenariul de testare



Figură 4.5.6 Schema conectare senzor ultrasonic la Raspberry pi

Pentru a proteja placă Raspberry, a fost folosit un circuit divizor de tensiune pentru a limita tensiunea maxima ce poate ajunge la Raspberry. Motivul fiind descris în capitolul anterior. Pentru a mari gradul de siguranță, am ales  $R_1 = 220\Omega$  și  $R_2 = 330\Omega$  rezultând astfel tensiune de ieșire  $V_{out} = 3V$  pentru o tensiune de intrare  $V_s = 5V$

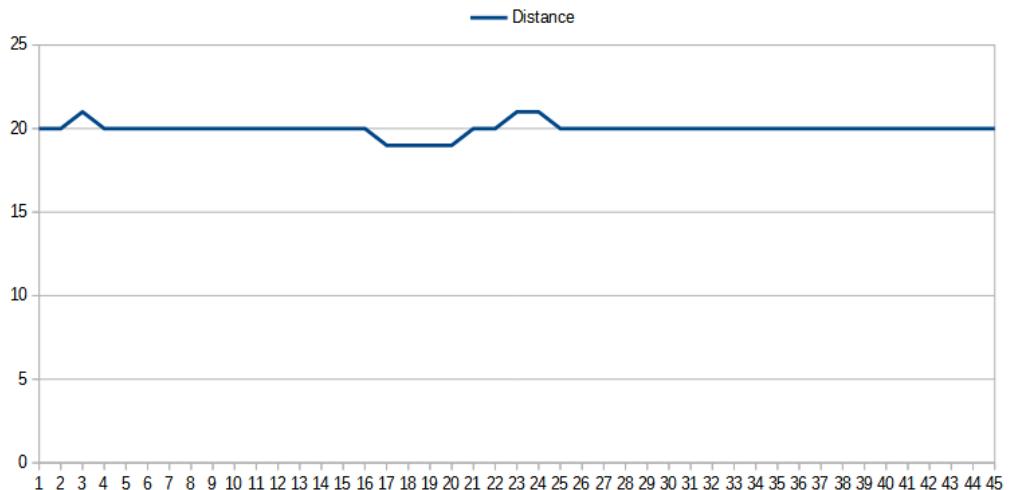
```

Distance: 20 cm
Distance: 19 cm
Distance: 19 cm
Distance: 19 cm
Distance: 19 cm
Distance: 20 cm
Distance: 20 cm
Distance: 21 cm

```



Figură 4.5.7 Valorile obținute de senzor



Figură 4.5.8 Reprezentarea grafica a valorilor citite de senzor

Scriptul python folosit pentru testare senzorilor ultrasonici:

```

import RPi.GPIO as GPIO
import time

GPIO.setmode(GPIO.BOARD)

TRIG = 16
ECHO = 18
i=0

GPIO.setup(TRIG,GPIO.OUT)
GPIO.setup(ECHO,GPIO.IN)

GPIO.output(TRIG, False)
print "Calibrating....."
time.sleep(2)

print "Place the object....."

```

```

try:
    while True:
        GPIO.output(TRIG, True)
        time.sleep(0.00001)
        GPIO.output(TRIG, False)

        while GPIO.input(ECHO)==0:
            pulse_start = time.time()

        while GPIO.input(ECHO)==1:
            pulse_end = time.time()

        pulse_duration = pulse_end - pulse_start

        distance = pulse_duration * 0.034 / 2

        distance = round(distance+1.15, 2)

        if distance<=20 and distance>=5:
            print „distance:”,distance,”cm”
            i=1

        if distance>20 and i==1:
            print „place the object....”
            i=0
            time.sleep(2)

except KeyboardInterrupt:
    GPIO.cleanup()

```

#### 4.5.3 Camera Raspberry Pi High Quality Camera

Camera de înaltă calitate Raspberry Pi este cea mai recentă camera de la Raspberry Pi. Aceasta oferă o rezoluție mai mare (12 megapixeli, față de 8 megapixeli) și o sensibilitate mai mare (cu aproximativ 50% mai mare pe pixel pentru o performanță îmbunătățită în condiții de iluminare slabă) decât modulul de cameră v2 existent și este concepută pentru a funcționa cu obiective interschimbabile atât cu montură C, cât și cu montură CS. Alte obiective pot fi atașate folosind adaptoare care se pot achiziționa separat.

Modulul este alcătuit dintr-un PCB pe care este amplasat un senzor Sony IMX477, un cablu FPC pentru conectarea la un computer Raspberry Pi, o montură de obiectiv din aluminiu frezat cu montură integrată pentru trepied și inel de reglare a focalizării, precum și un adaptor pentru montură C- la montură CS.[36]



Figură 4.5.9 High Quality RPI Camera [37]

Specificații tehnice[36]:

- Senzor: Sony IMX477R 12.3 megapixeli
- Dimensiune pixeli:  $1.55 \times 1.55 [\mu\text{m}]$
- Focus ajustabil 12.5 – 22.4 [mm]
- Filtru IR integrat
- Standard lentile: CS-Mount / C-Mount (adaptor C-CS inclus)
- Lungime cablu date: 200 [mm]
- Format output video: RAW12/10/8, COMP8
- Dimensiuni:  $38 \times 38 \times 20$  [mm] (fără obiectiv)

#### 4.5.4 Testare camera

Pentru a testa camera am folosit un script python prezentat in tabelul de mai jos.

După ce se realizează conectarea la microcontroler, se rulează programul camera.py, programul care deschide un port ce poate fi accesat prin intermediul unui browser.

Comanda folosita pentru a rula programul camera.py este: 'python3 camera.py'.

In browser-ul web se introduce adresa ip a microcontrolerului urmata de numărul portului deschis. Exemplu '192.168.0.100:8000'.

După ce a fost introdusa adresa de mai sus, in browser apare imaginea transmisa in timp real de camera montata pe robot.

Tabel 2 Script Python testare camera

```

import io
import picamera
import logging
import socketserver
from threading import Condition
from http import server

PAGE="""\
<html>

<head>
<title>Picamera MJPEG streaming demo</title>
</head>

```

```

<body>
    
    <script>
        var img = document.querySelector('img');
        var source = new EventSource('/stream.mjpeg');
        source.addEventListener('image', function (e) {
            img.src = 'data:image/jpeg;base64,' + e.data;
        });
    </script>
</body>

</html>
"""
class StreamingOutput(object):
    def __init__(self):
        self.frame = None
        self.buffer = io.BytesIO()
        self.condition = Condition()

    def write(self, buf):
        if buf.startswith(b'\xff\xd8'):
            # New frame, copy the existing buffer's content and notify all
            # clients it's available
            self.buffer.truncate()
            with self.condition:
                self.frame = self.buffer.getvalue()
                self.condition.notify_all()
            self.buffer.seek(0)
        return self.buffer.write(buf)

class StreamingHandler(server.BaseHTTPRequestHandler):
    def do_GET(self):
        if self.path == '/':
            self.send_response(301)
            self.send_header('Location', '/index.html')
            self.end_headers()
        elif self.path == '/index.html':
            content = PAGE.encode('utf-8')
            self.send_response(200)
            self.send_header('Content-Type', 'text/html')
            self.send_header('Content-Length', len(content))
            self.end_headers()
            self.wfile.write(content)
        elif self.path == '/stream.mjpeg':
            self.send_response(200)
            self.send_header('Age', 0)
            self.send_header('Cache-Control', 'no-cache, private')
            self.send_header('Pragma', 'no-cache')

```

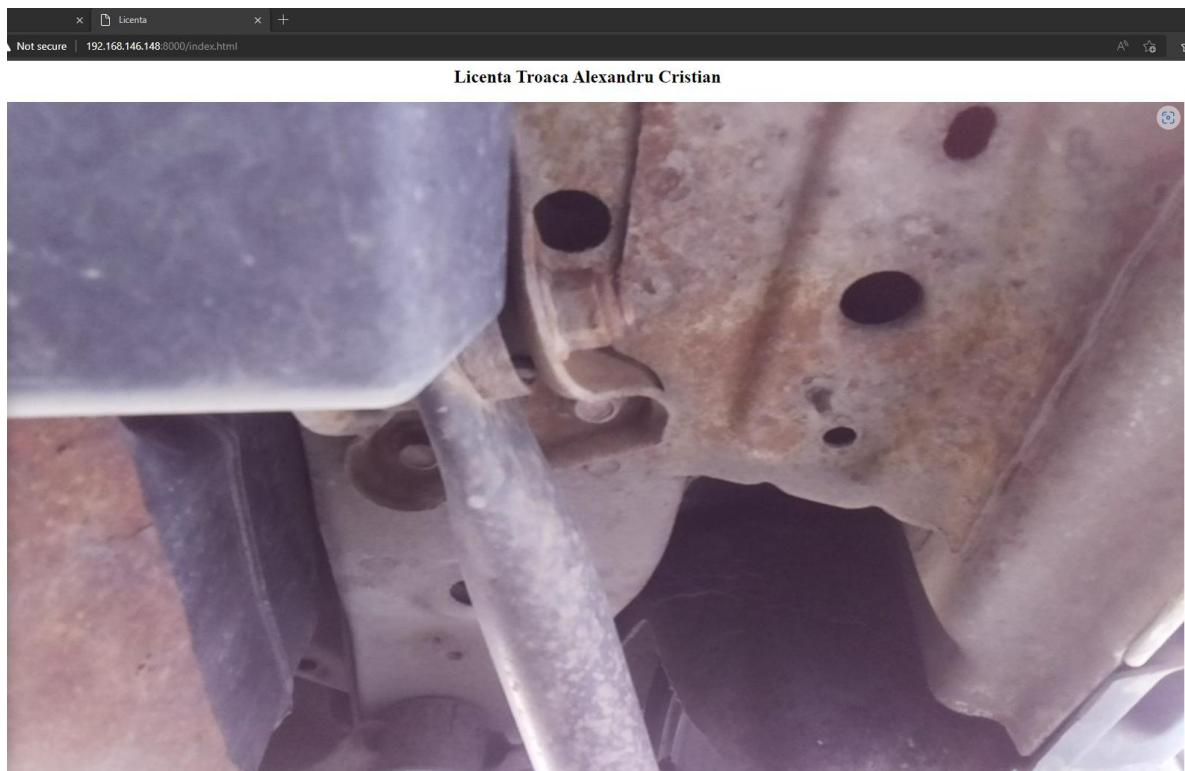
```

        self.send_header('Content-Type', 'multipart/x-mixed-replace; boundary=FRAME')
        self.end_headers()
        try:
            while True:
                with output.condition:
                    output.condition.wait()
                    frame = output.frame
                self.wfile.write(b'-FRAME\r\n')
                self.send_header('Content-Type', 'image/jpeg')
                self.send_header('Content-Length', len(frame))
                self.end_headers()
                self.wfile.write(frame)
                self.wfile.write(b'\r\n')
        except Exception as e:
            logging.warning(
                'Removed streaming client %s: %s',
                self.client_address, str(e))
        else:
            self.send_error(404)
            self.end_headers()

class StreamingServer(socketserver.ThreadingMixIn, server.HTTPServer):
    allow_reuse_address = True
    daemon_threads = True

with picamera.PiCamera(resolution='1280x720', framerate=24) as camera:
    output = StreamingOutput()
    camera.start_recording(output, format='mjpeg')
    try:
        address = ('', 8080)
        server = StreamingServer(address, StreamingHandler)
        server.serve_forever()
    finally:
        camera.stop_recording()

```



Figură 4.5.10 Testare camera – imagine browser

## 4.6 Alimentare

### 4.6.1 Baterie

Bateria Lipo Gens Ace 2200mah 5S 45C este o baterie Lipo de înaltă calitate, cu greutate redusă și descărcare ridicată și oferă suficientă putere pentru aplicația prezentată. Descărcarea maximă continuă a bateriei lipo Gens ace 2200mah 18.5V poate fi de 45C, iar descărcarea maximă în vârf este de 90C.[32]

În descrierea bateriilor, curentul de descărcare este adesea exprimat ca rată C pentru a fi standardizat în funcție de capacitatea bateriei, care este adesea foarte diferită de la o baterie la alta. Rata C este o măsură a ratei la care o baterie se descarcă în raport cu capacitatea sa maximă. O rată de 1C înseamnă că curentul de descărcare va descărca întreaga baterie în 1 oră. Pentru o baterie cu o capacitate de 2.2 Ah, acest lucru echivalează cu un curent de descărcare de 2.2 amperi. O rată de 45C pentru această baterie ar fi de 99 de amperi, iar o rată C/2 ar fi de 1.1 de amperi.[33]



Specificații [32]:

- Tip: LiPo
- Număr de celule: 5
- Capacitate: 2200mAh
- Tensiune: 18.5V
- Rata de descărcare continuă: 45C
- Rata de descărcare în vârf: 90C
- Încărcare echilibrată
- Dimensiuni: 99 x 33 x 39 [mm]
- Greutate: 272g

Figură 4.6.1 Baterie Gens Ace [32]

#### 4.6.2 Surse coborâtoare de tensiune

Deoarece tensiunea oferita de baterie este prea mare pentru a putea conecta direct componentele robotului este obligatorie folosirea unei surse coborâtoare de tensiune.

Tensiunea nominală obținută la bornele acumulatorului Lipo Gens Ace 5S, aşa cum este menționat anterior, este de 18.5V, o tensiune care poate afecta unele componente electronice ale sistemului, întrucât aceasta depășește tensiunea maximă de alimentare a acestora. În acest sens este necesară și utilizarea a două surse coborâtoare de tensiune, care să ne asigure la ieșirea acesteia tensiune necesară de 5V pentru microcontroler și senzori ultrasonici, respectiv 12V pentru alimentarea motoarelor. Pentru a realiza acest lucru a fost utilizată o sursă coborâtoare de tensiune din comerț ce folosește circuitul integrat MP1584EN (Figura 4.6.2) și un modul convertor de tensiune de 12V (Figura 4.6.3).

##### Convertor de tensiune DC-DC 5V:

Acest convertor de tensiune DC/DC se bazează pe MP1584, transformă tensiunea de intrare cuprinsă între 4,5V și 28V într-o tensiune mai mică între 0,8V și 18V, capabil să conducă o sarcină de 3A cu o reglare excelentă a tensiunii de linie și a sarcinii.[35]

Această sursă folosește un potențiometru pentru a regla tensiunea de la intrare în funcție de tensiunea dorită la ieșire. Printre avantajele utilizării acestei soluții se enumeră: dimensiuni foarte mici de gabarit ( 22.2mm x 17.4mm x 4mm), ușor de utilizat și preț de achiziție redus.



#### Caracteristici tehnice[34]:

- Tensiunea de intrare: 4.5 – 28V
- Tensiunea de ieșire: 0.8V – 20V
- Curent de ieșire: maximum 3A
- Dimensiuni: 22 x 17 x 4 [mm]
- Temperatura de funcționare: -50 °C - +8 °C

Figură 4.6.2 Convertor DC-DC MP1584 [35]

MP1584 este un regulator de comutare în trepte de înaltă frecvență cu un MOSFET de putere de înaltă tensiune integrată internă. Oferă ieșire 3A cu comanda curentă a modului de reacție pentru o buclă rapidă și o compensare ușoară.

Intervalul larg de intrare de la 4.5V la 28V permite o varietate de aplicații pas cu pas, inclusiv cele dintr-un mediu de intrare pentru automobile. Un cip operațional de 100 $\mu$ A permite utilizarea în aplicații cu baterie.

Prin comutarea la 1,5MHz, MP1584 este capabil să prevină probleme de zgomot de interferență electromagnetică, cum ar fi cele găsite în aplicațiile AM radio și ADSL. MP1584 este disponibil într-un pachet SOIC8E îmbunătățit termic.

Tensiunea de ieșire este setată folosind un divizor de tensiune rezistiv de la tensiunea de ieșire la pinul FB. Divizorul de tensiune împarte tensiunea de ieșire până la tensiunea de reacție cu raportul:  $V_{FB} = V_{OUT} \cdot \frac{R_2}{R_1+R_2}$

Un exemplu de utilizare al regulatorului de tensiune MP1584EN este obținerea unei tensiuni la ieșire de 5V, pentru o tensiune de intrare cuprinsă între 8V-28V. Conform fișei de catalog, prin varierea rezistenței R<sub>3</sub> se poate obține tensiunea dorită la ieșire, celelalte componente dimensionându-se în funcție de aceasta conform pașilor:

V <sub>OUT</sub> (V)	L (μH)	C <sub>2</sub> (μF)	R <sub>3</sub> (kΩ)	C <sub>3</sub> (pF)	C <sub>6</sub>
1.8	4.7	47	105	100	None
2.5	4.7 - 6.8	22	54.9	220	None
3.3	6.8 - 10	22	68.1	220	None
5	15 - 22	22	100	150	None
12	22 - 33	22	147	150	None

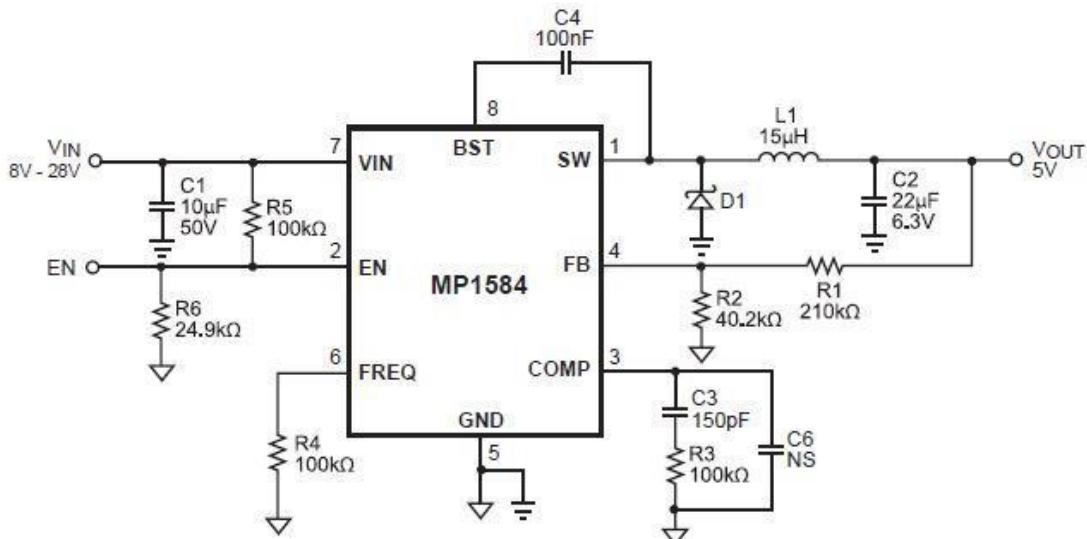
Figura 4.6.1 Tabel de valori pentru diferite tensiuni la ieșire dorite

$$\text{Se calculează valoare rezistenței } R_3: R_3 = \frac{2\pi \cdot C_2 \cdot f_c}{R_1 + R_2} \frac{R_2}{R_1 + R_2}$$

$$\text{Se calculează valoare condensatorului } C_3: C_3 > \frac{4}{2\pi \cdot R_3 \cdot f_c}$$

Se verifică relația:  $\frac{1}{2\pi \cdot C_3 \cdot R_{ESR}} < \frac{f_c}{2}$ . Dacă aceasta este adevărată se alege un al doilea condensator

$$C_6 = \frac{C_2 \cdot R_{ESR}}{R_3}$$



Figură 4.6.3 Mod de utilizarea pentru MP1584EN

## **Converter de tensiune DC-DC 12V:**

Pentru alimentarea motoarelor am ales o sursa capabila de a furniza constant tensiunea de 12V și un curent minim de ieșire de minimum 2.3A, conform cerințelor din fișa tehnica a motorului prezentat in capitolele anterioare.

Modul de reglare al tensiunii de ieșire este asemănător cu cel prezentat la convertorul de tensiune de mai sus.



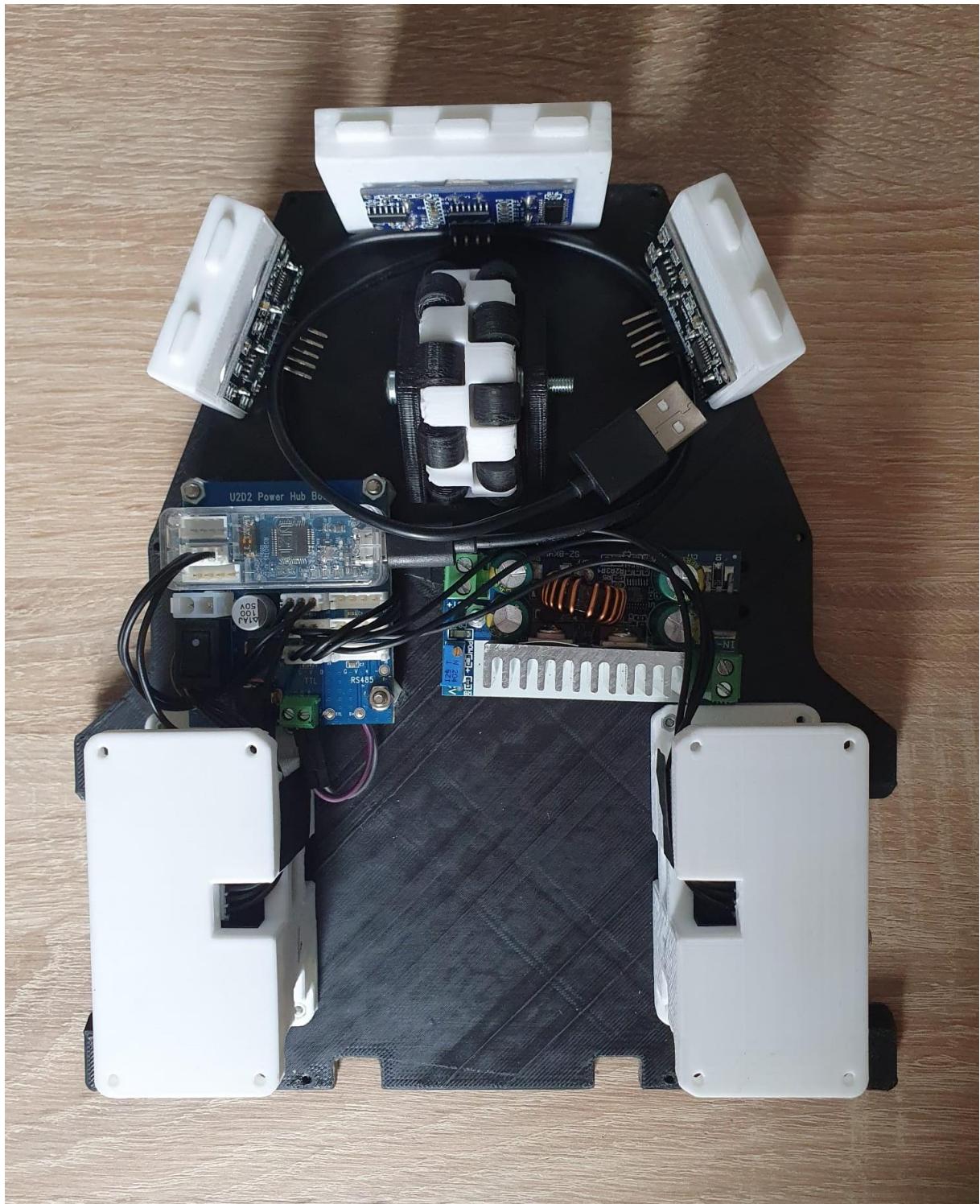
Figură 4.6.4 Convertor tensiune 12V

### Caracteristici tehnice:

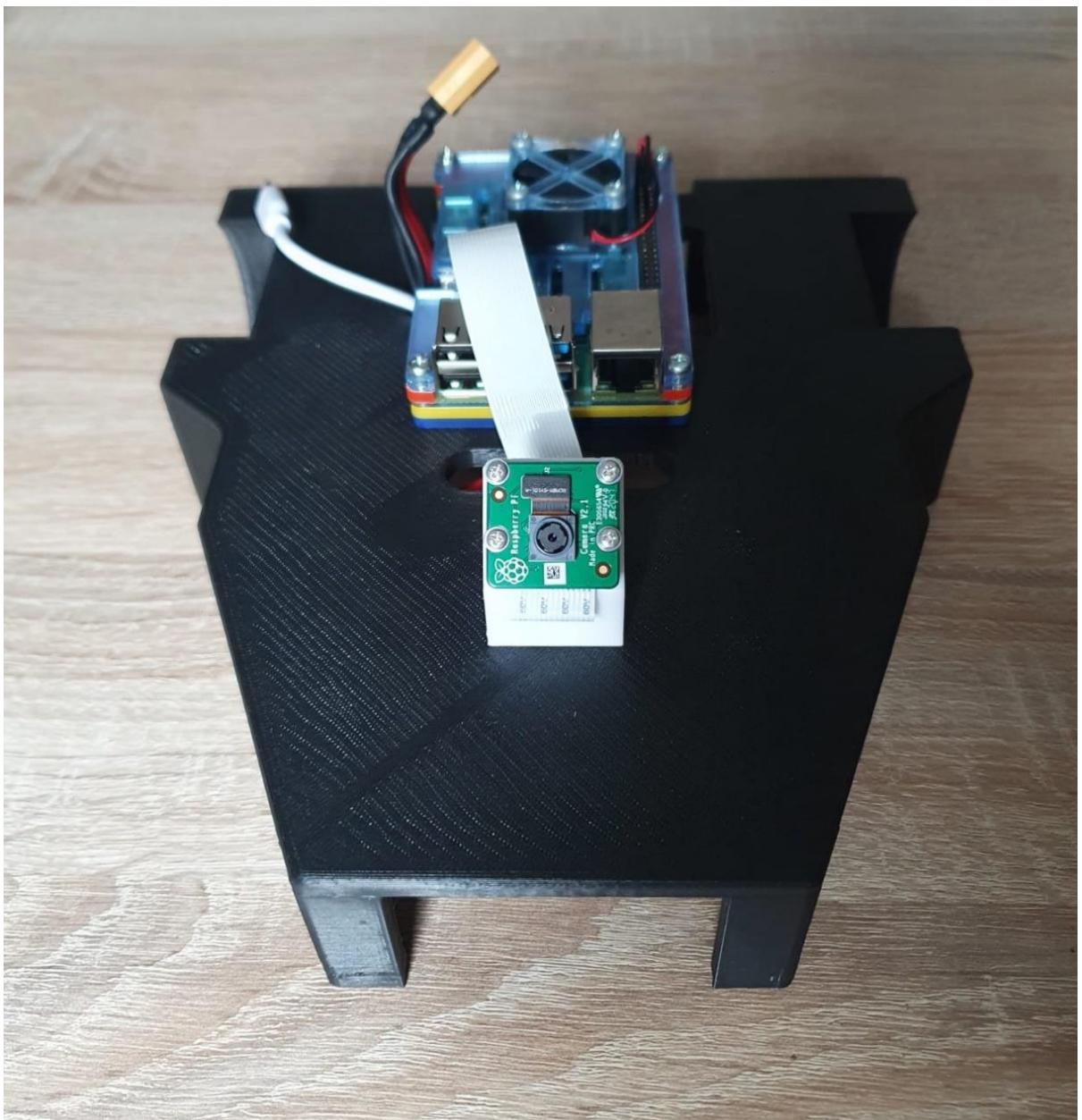
- Tensiune de intrare recomandată: 8V – 55V (max 60V)
- Tensiune de ieșire reglabilă: 1V – 36V
- Curent de ieșire maxim: 15A
- Putere disipată în mod normal: 100W
- Putere disipată maximă: 200W
- Frecvență de funcționare: 180 kHz
- Până la 94% eficiență
- Protecție la scurtcircuit și alimentare inversă
- Dimensiuni: 70 x 38 x 31 [mm]
- Masă: 80g

# Capitolul 5. Testare și experimentare a sistemului robotic

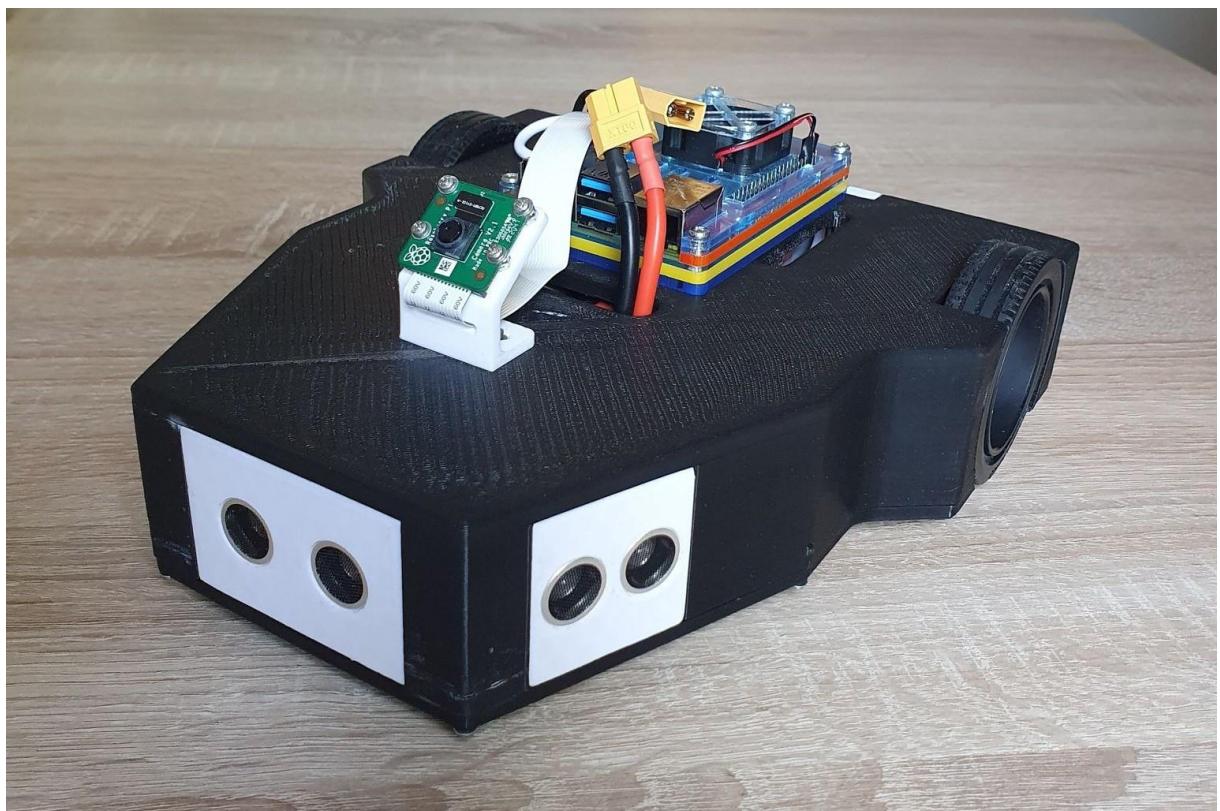
## 5.1 Asamblare



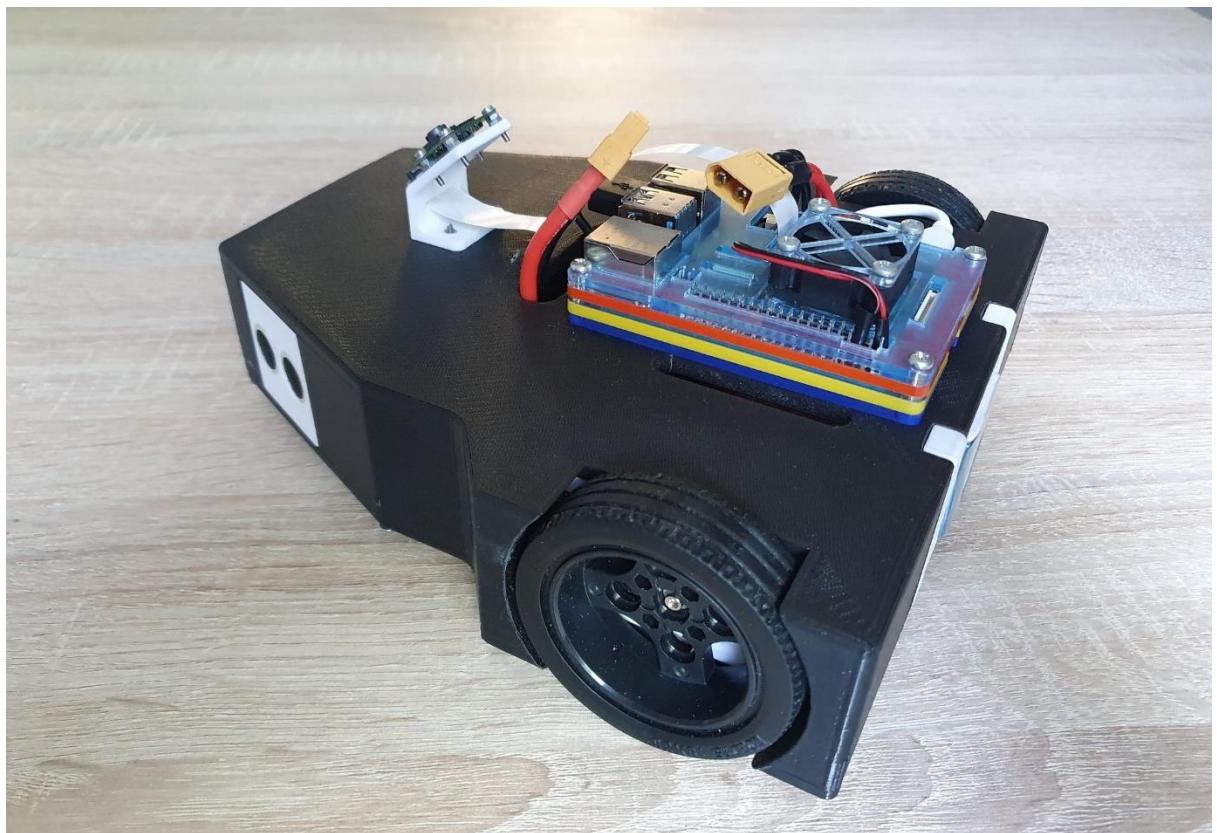
Figură 5.1.1 Asamblarea componentelor pe sășiu inferior



Figură 5.1.2 Asamblarea componentelor pe sășiul superior



Figură 5.1.3 Ansamblu final



Figură 5.1.4 Ansamblu final

## 5.2 Mod de funcționare autonom

Robotul a fost conceput pentru a fi manevrat manual sau pentru a naviga autonom într-o parcare sau depozit în care există autoturisme parcate.

Modul în care robotul parcurge spațiul de lucru are reflexii importante în eficiența întregului sistem, prin acoperirea unui procentaj cât mai mare din zona de lucru dar și prin timpul de realizare a sarcinilor.

În acest sens au fost studiate mai multe strategii de deplasare în zona de lucru, diferențiate prin forma traseului ce trebuie parcurs, și anume: strategia spirală (figura 5.1.1 a), strategia dreptunghiurilor (figura 5.1.1 b) și strategia arcelor de cerc (figura 5.1.1 c). Aceste strategii au fost studiate pentru funcționarea într-un perimetru având forma unu dreptunghi.

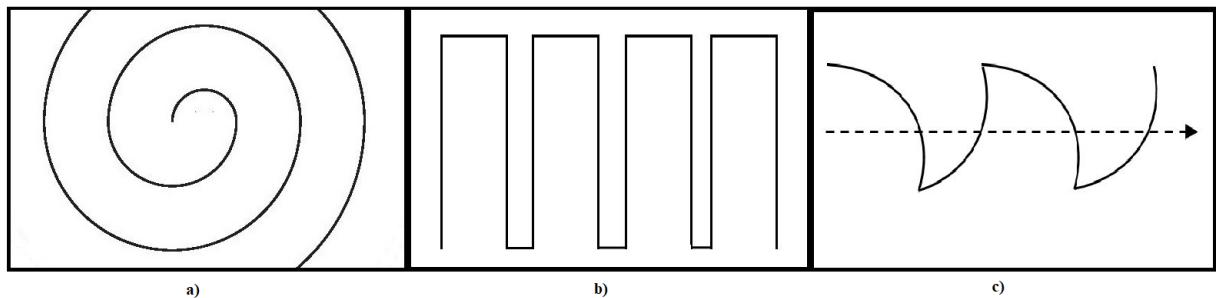


Figura 5.2.1 Strategii de deplasare

Din strategiile prezentate mai sus, cea potrivită pentru sistemul prezentat este strategia dreptunghiurilor. Deoarece autoturismele sunt parcate în ordine, paralel una față de cealaltă.

Strategia dreptunghiurilor presupune o parcurgere liniară a spațiului de lucru, fiind puțin sensibilă la obstacole staționare și putând realiza un procentaj mai mare de acoperire al spațiului de lucru.

Un avantaj îl constituie faptul că, prin această strategie, sunt percepute numai obiectele care se află în fața robotului, mai exact în dreptul senzorului ultrasonic frontal. Acel obiect poate fi evitat verificând dacă în fața senzorilor laterali există vreun obiect și prin virajul în direcția în care nu există obiect. După ce obiectul a fost evitat, robotul continua să se deplaceze pe traseul deja stabilit. În figura 5.1.2 este prezentat modul în care funcționează programul pentru evitarea obstacolelor.

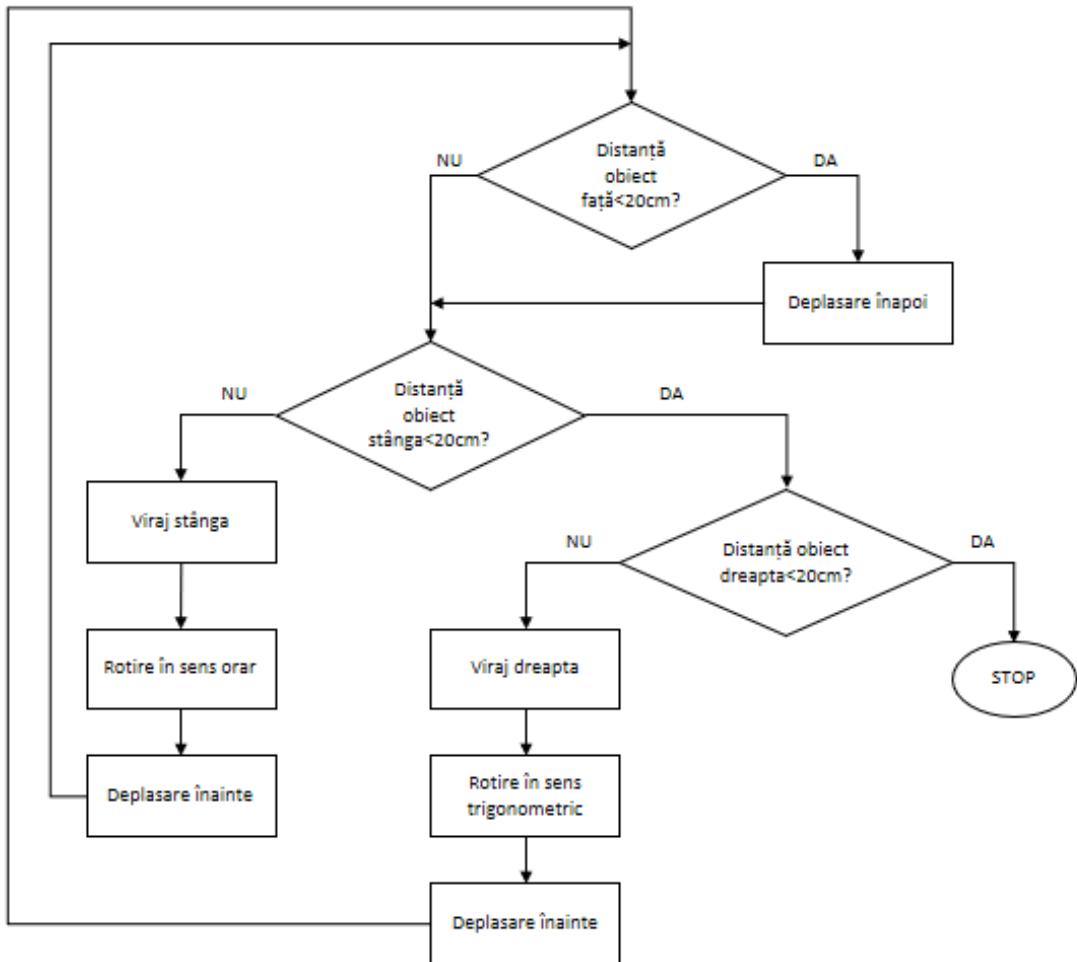


Figura 5.2.2 Descriere program de evitare al obstacolelor

### 5.3 Mod de funcționare manual

Înaintea realizării programului propriu-zis a fost creată o diagramă logică a acestuia, având rolul de a face mult mai bine înțeles programul dar și urmărirea înlănțuirii logice de execuție a comenziilor. Această diagramă logică este prezentată în figura 5.1.3.

Pentru a folosi sistemul este necesar un laptop conectat la aceeași rețea Wi-Fi cu microcontrolerul Raspberry Pi. Conectarea la microcontroler se face folosind comanda ssh prin intermediul comand prompt-ului.

Comanda folosită este: 'ssh pi@adresa\_ip', unde adresa\_ip se înlocuiește cu adresa ip a microcontrolerului. Exemplu 'ssh pi@192.168.0.100'

După ce s-a realizat conectarea la microcontroler, se rulează programul camera.py, programul care deschide un port ce poate fi accesat prin intermediul unui browser.

Comanda folosită pentru a rula programul camera.py este: 'python3 camera.py'.

In browser-ul web se introduce adresa ip a microcontrolerului urmata de numărul portului deschis. Exemplu '192.168.0.100:8000'.

După ce a fost introdusa adresa de mai sus, in browser apare imaginea transmisa in timp real de camera montata pe robot.

Pentru a comanda robotul este necesara deschiderea celui de-al doilea terminal si conectarea prin ssh la Raspberry Pi. După ce au fost realizati acești pași se rulează programul de control al motoarelor.

Comanda folosita pentru a rula programul de control al motoarelor este: 'DISPLAY=:0.0 python3 control\_wasd.py'. După rularea acestui program robotul se poate controla folosind tastatura, modul de functionare al programului fiind prezentat in figura 5.2.2.

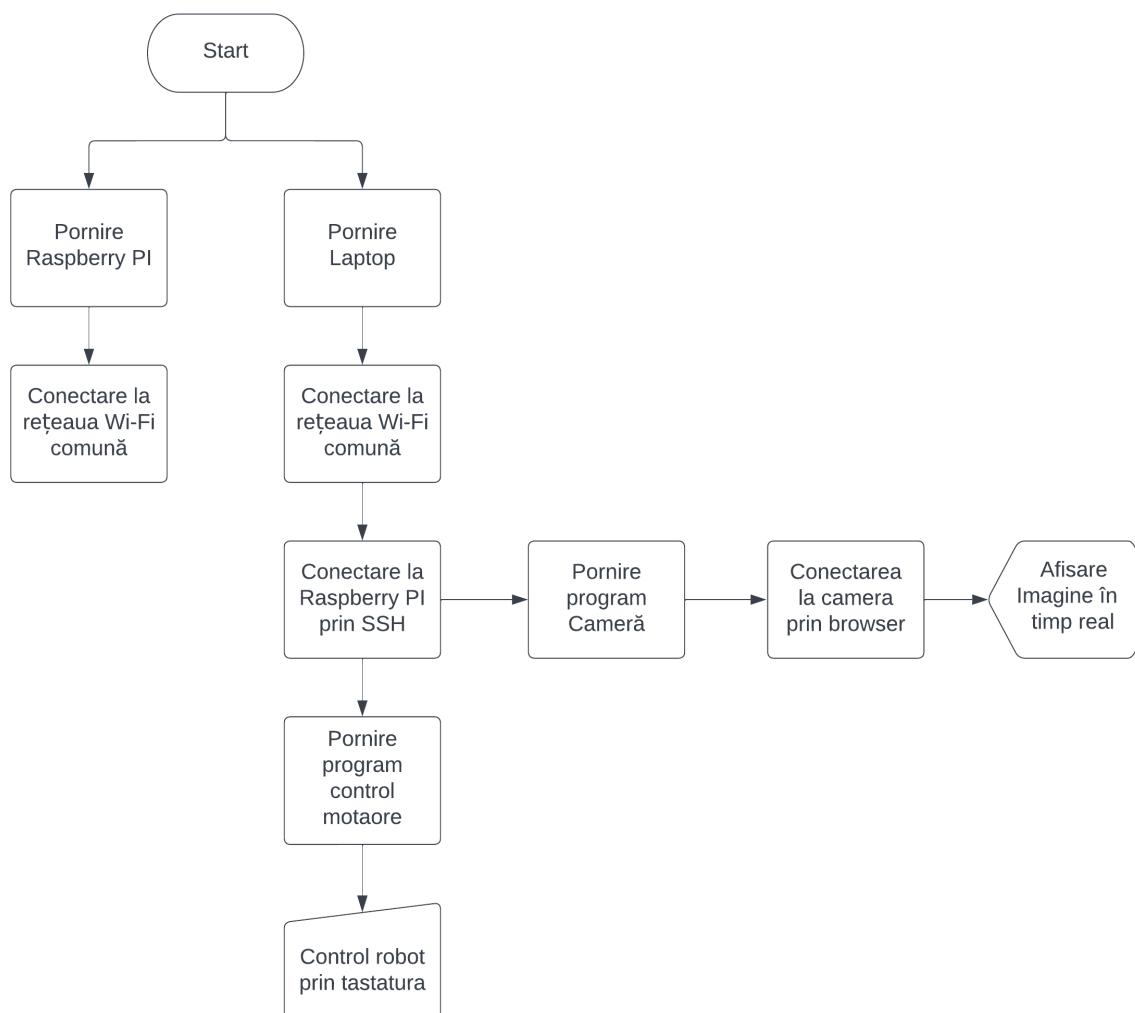


Figura 5.3.1 Mod de functionare general

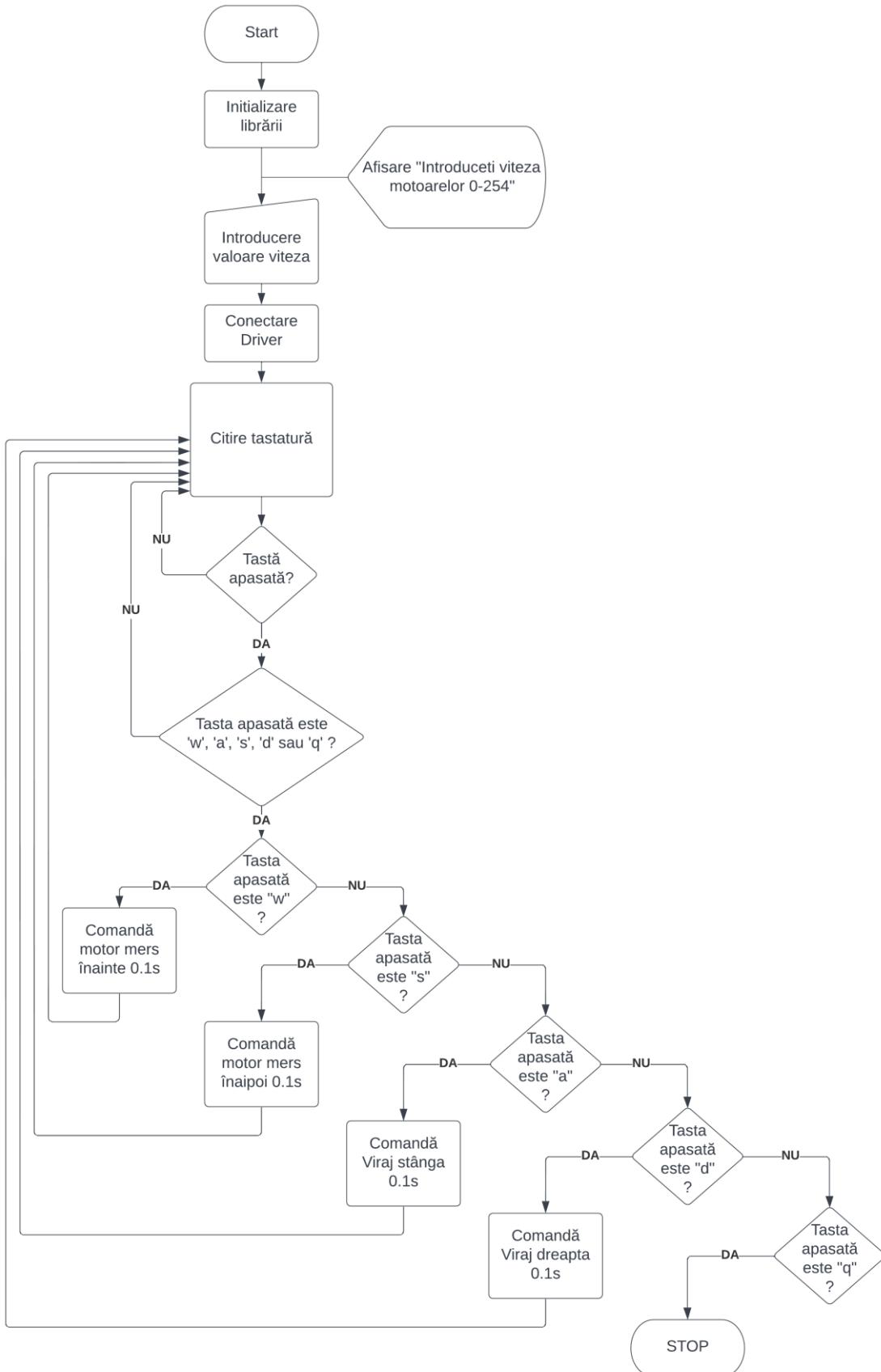
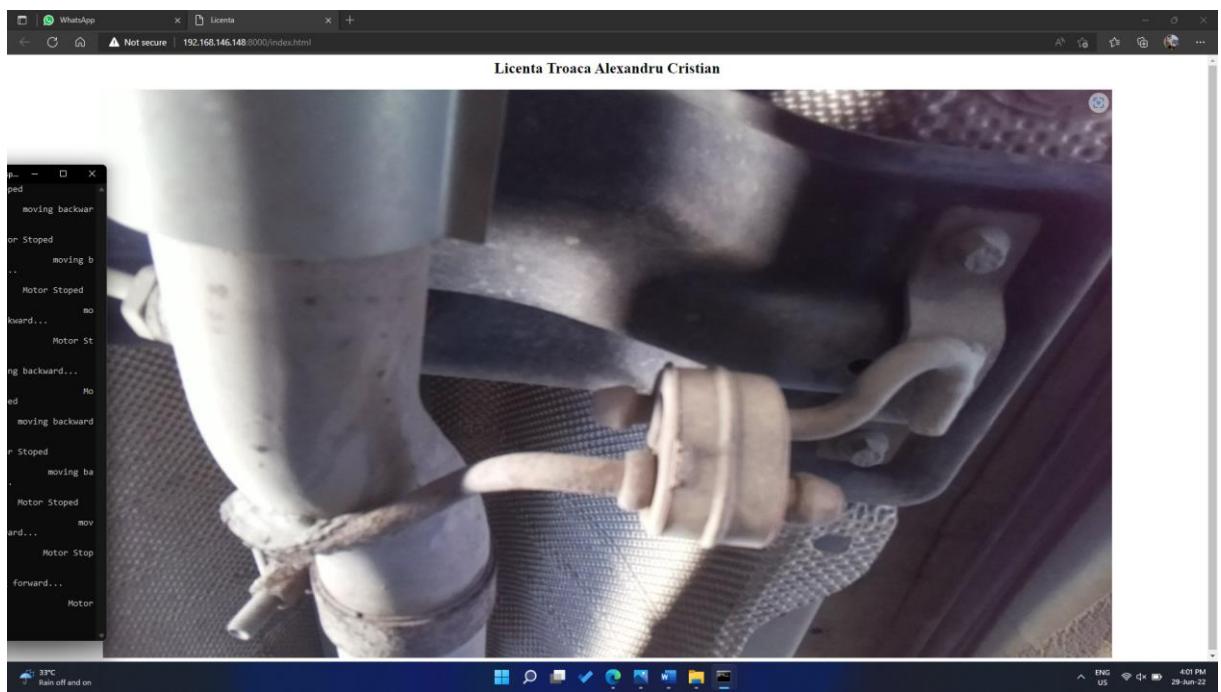
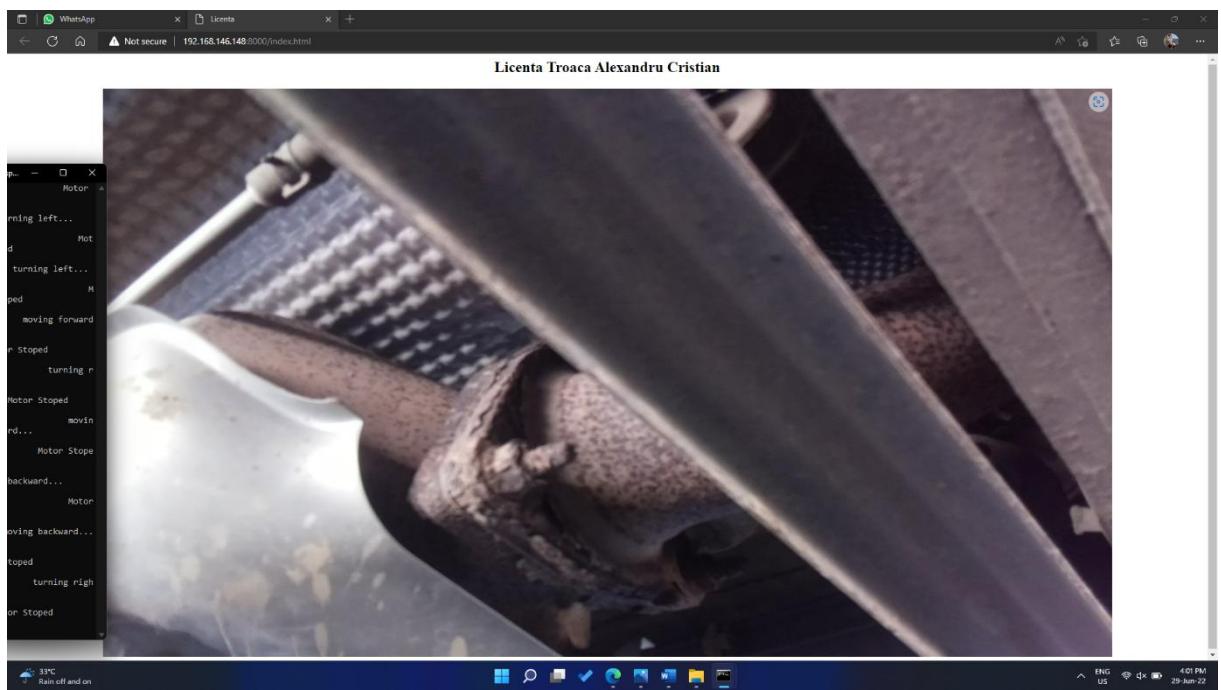


Figura 5.3.2 Mod de functionare al programului de comanda



Figură 5.3.1 Imagine din timpul utilizării sistemului



Figură 5.3.2 Imagine din timpul utilizării sistemului

## 5.4 Studiu solicitare încovoiere placă inferioară

Componenta va fi testată pentru a studia comportamentul geometriei în urma solicitării de încovoiere. Astfel, placa va fi testată celor două cazuri evidențiate în Tabel A.1: a) Placă simplu încastrată pe planul frontal, Figura A.1, b) Placă dublu încastrată pe planul frontal, Figura A.2. După ce placa a fost încastrată, se va aplica o masă uniformă distribuită pe suprafața piesei, Figura A.5,  $m = 3\text{ kg}$ .

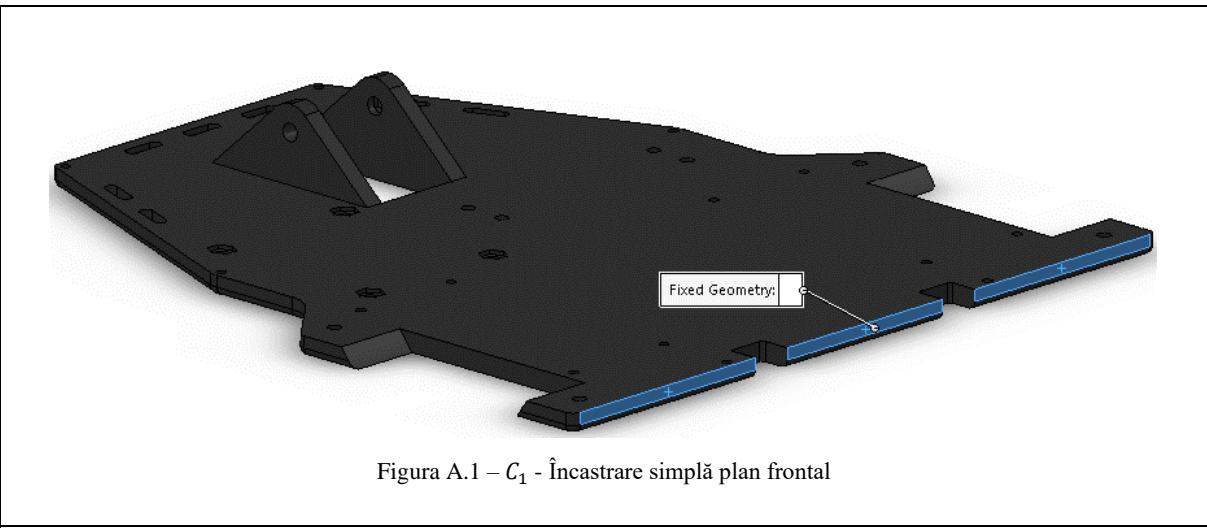


Figura A.1 –  $C_1$  - Încastrare simplă plan frontal

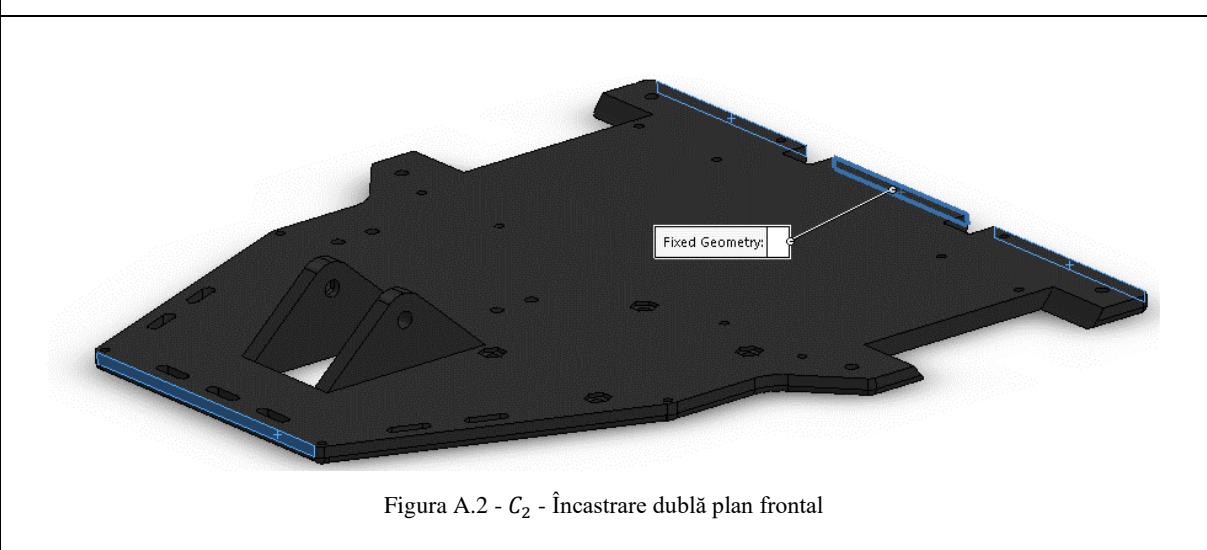
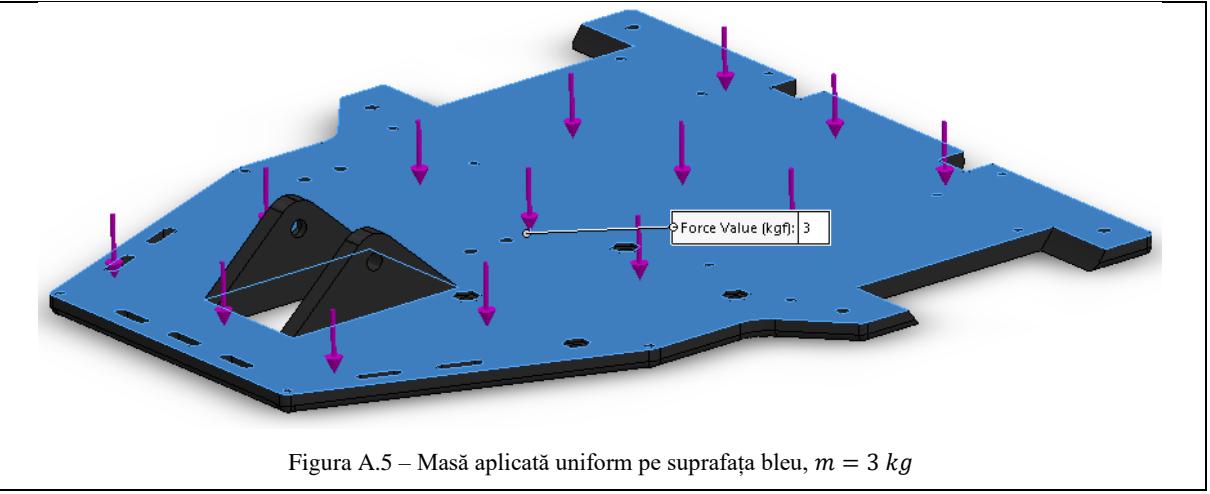


Figura A.2 -  $C_2$  - Încastrare dublă plan frontal



Tabel A.1 Condiții simulare

În consecință, cele 2 cazuri de simulare sunt evidențiate în Tabelul A.2:

	Tip încastrare	Masă distribuită uniform
$C_1$	Încastrare simplă plan frontal	$m = 3 \text{ kg}$
$C_2$	Încastrare dublă plan frontal	

Tabel A.2 – Cazuri simulare

În continuare, se vor simula cele două cazuri și se vor documenta valorile următoarelor componente:

- $\sigma_{ech}$  [MPa] - energia potențială de deviație (criteriul von Mises);
- $v$  [mm] – săgeata;
- $\varepsilon$  – alungirea relativă

#### 5.4.1 Cazul I - Încastrare simplă plan frontal, $m = 3 \text{ kg}$

**Energia potențială de deviație (criteriul von Mises)  $\sigma_{ech}$ :** În tabelul următor, Tabel 1.1, sunt evidențiate rezultatele obținute în urma simulării. Astfel, înaintea rulării simulării, Figura 1.1, ansamblul este reprezentat în formă incipientă. După rularea simulării, rezultatul obținut este reprezentat în Figura 1.2 și evidențiază distribuția energiei potențiale de deviație în placă, plaja de gradient turcoaz - verde - galben.

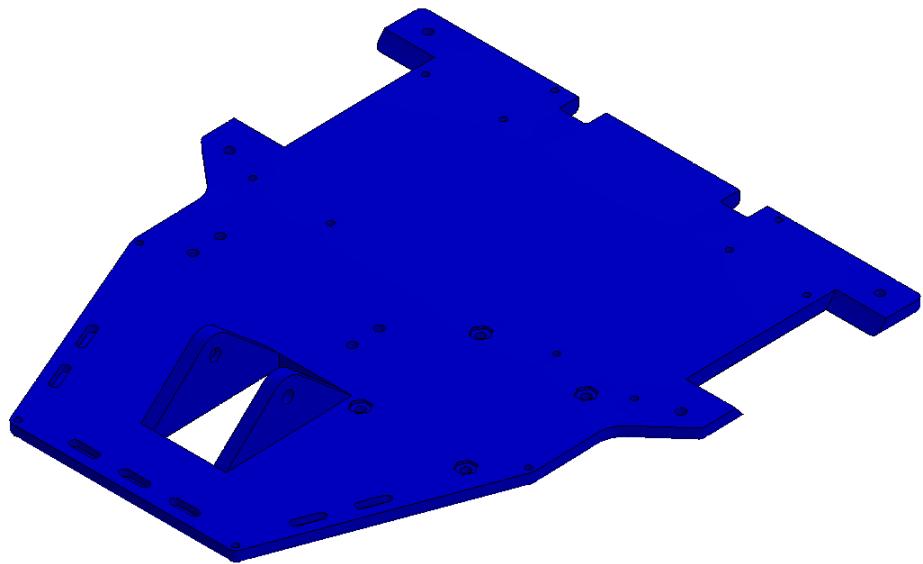


Figura 1.1 – Stare pre-simulare – vedere izometrică

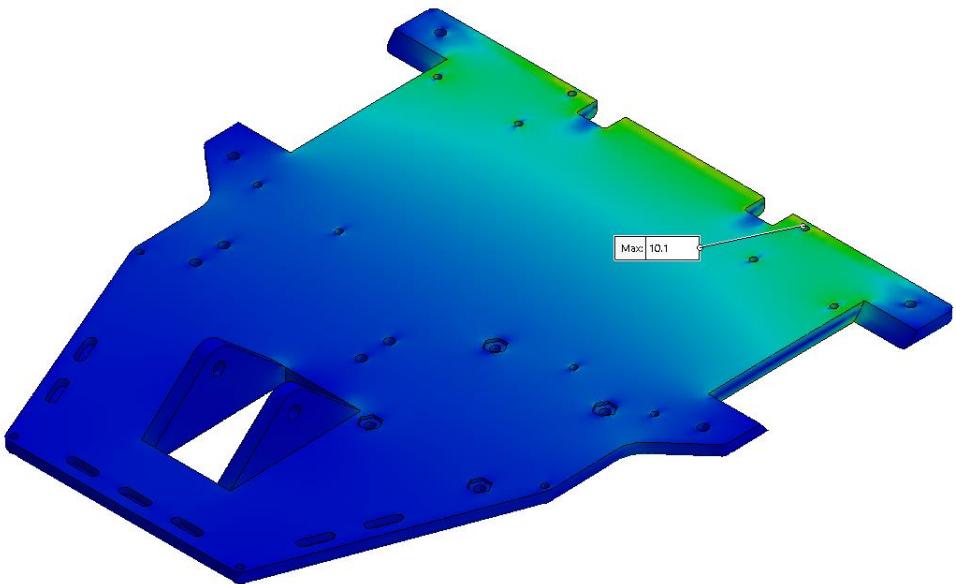


Figura 1.2 – Stare post-simulare – vedere izometrică

Tabel 1.1 – Rezultate simulare

În tabelul următor, Tabel 1.2, este evidențiat punctul de solicitare maximă, Figura 1.4, având un cuantum  $\sigma_{ech} = \sigma_{max} = 10.1 \text{ MPa}$  conform gradientului din Figura 1.5.

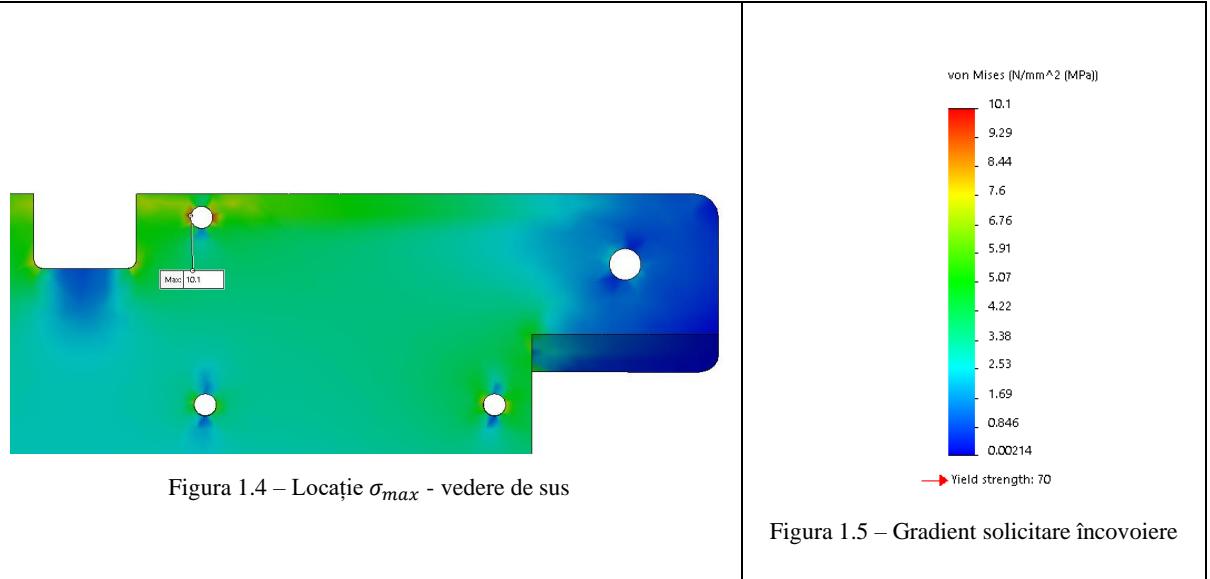


Figura 1.4 – Locație  $\sigma_{max}$  - vedere de sus

Figura 1.5 – Gradient solicitare încovoiere

Tabel 1.2 – Rezultate  $\sigma_{max}$

Conform Figura 1.5, valoarea maximă a tensiunii echivalente în placa inferioară este  $\sigma_{ech} = \sigma_{max} \cong 10.1 \text{ MPa}$ . Știind că tensiunea admisibilă la încovoiere a materialului, PLA, din care este realizată placa este  $\sigma_a \cong 70 \text{ MPa}$ , placa nu va suferi deformări plastice.

$$\sigma_{max} < \sigma_a \quad (1)$$

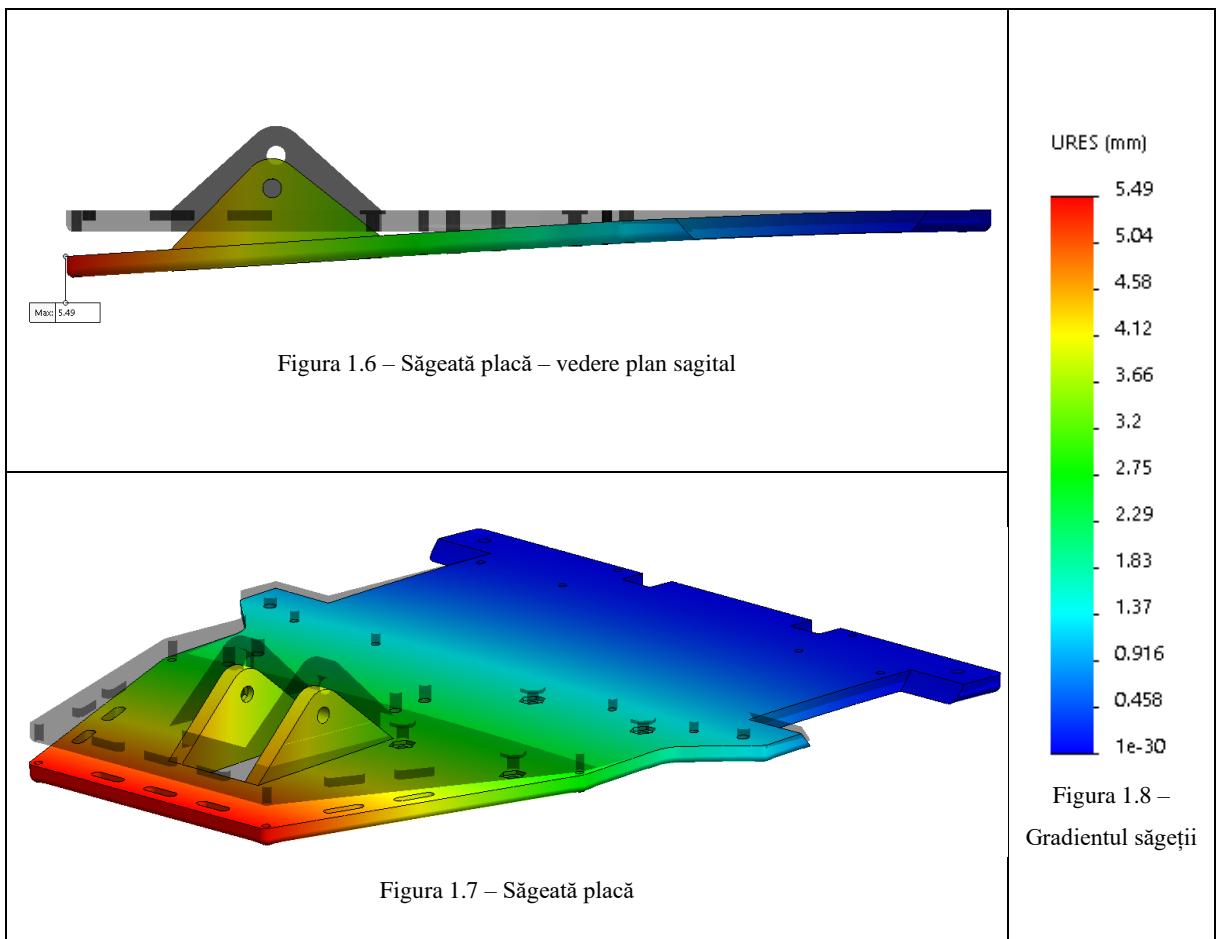
Astfel, poate fi calculat factorul de siguranță,  $c$ , al plăcii pentru **Cazul 1 - Încastrare simplă plan frontal,  $m = 3 \text{ kg}$ :**

$$c = \frac{\sigma_{ac}}{\sigma_{ech}} \quad (2)$$

Din (2) rezultă factorul de siguranță:

$$c = \frac{70 \text{ MPa}}{10.1 \text{ MPa}} = 7.01$$

**Sägeata  $v$ :** În Tabel 1.3 este reprezentată săgeata cauzată de efortul de încovoiere. În Figura 1.6 și 1.7 este reprezentată forma inițială a plăcii peste care este suprapusă forma rezultată în urma simulării și în Figura 1.8 este evidențiat gradientul săgeții  $v$ . Reprezentarea formei obținute a fost amplificată pentru a evidenția tendința de deformare în momentul în care este supusă solicitării de încovoiere și săgeata maximă este  $v_{max} = 5.49 \text{ mm}$ .



Tabel 1.3 – Rezultate săgeată

**Alungirea relativă  $\epsilon$ :** În tabelul următor, sunt evidențiate rezultatele obținute în urma simulării. Astfel, în Figura 1.9, placa este reprezentată în starea pre-simulare. În Figura 1.10 este reprezentată placa în stare post-simulare. După cum se poate observa, zonele supuse alungirii relative sunt evidențiată prin plaja de gradient verde-portocaliu

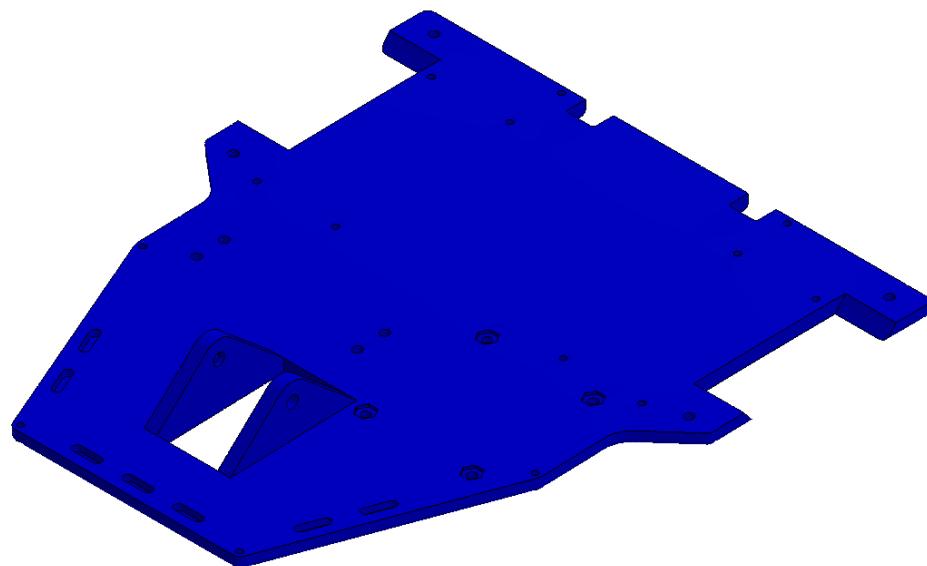


Figura 1.9 – Stare pre-simulare

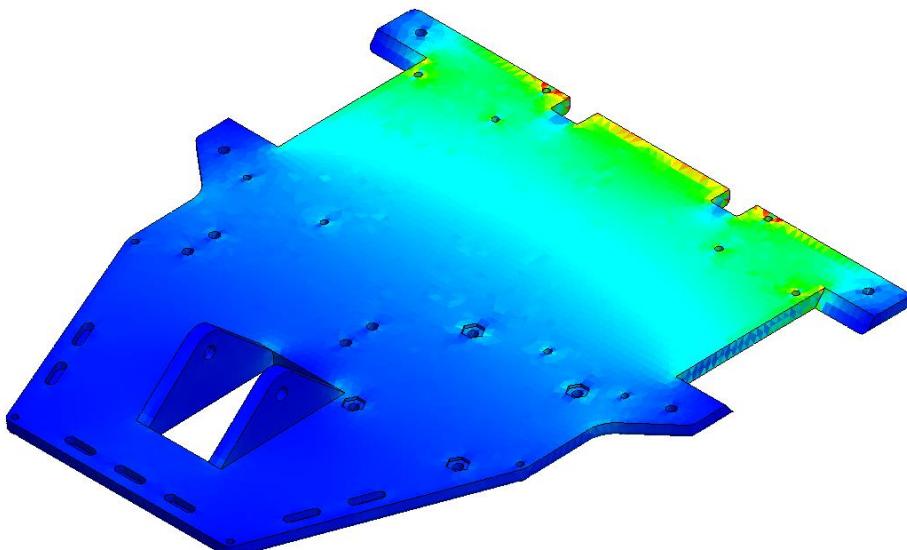


Figura 1.10 – Stare post-simulare

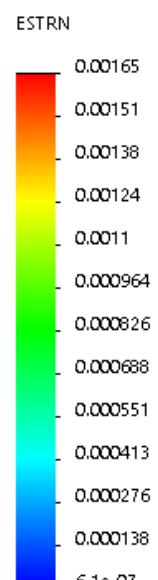


Figura 1.11 –  
Gradient alungire  
relativă

Tabel 1.4 – Rezultate alungire relativă

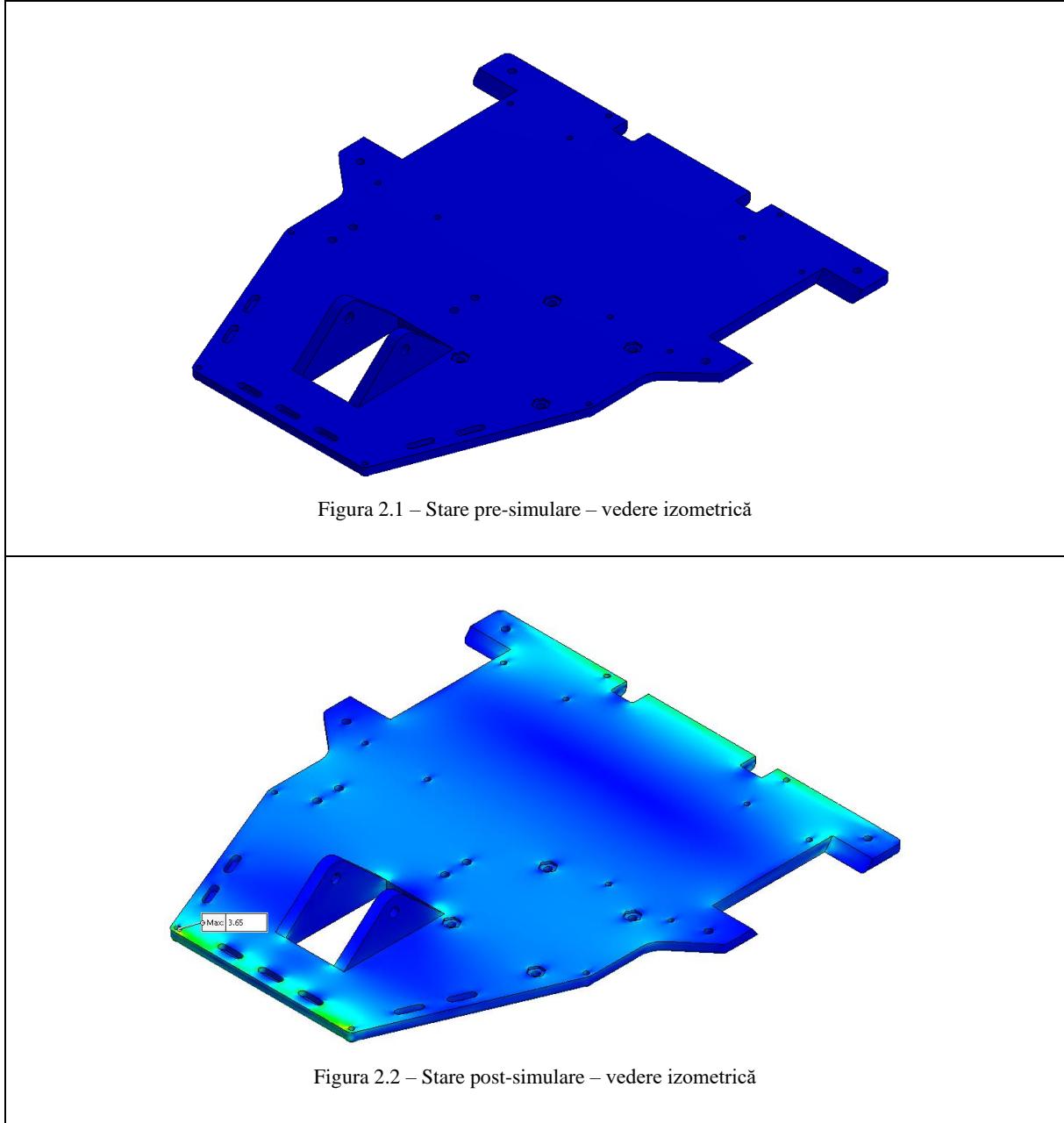
Conform Figura 1.11, alungirea relativă generală maximă  $\varepsilon_{max}$  este:

$$\varepsilon_{max} = 1,65 \cdot 10^{-3} \quad (3)$$

Din (3) rezultă faptul că alungirea relativă este neglijabilă în **Cazul 1 - Încastrare simplă plan frontal,  $m = 3\text{ kg}$ .**

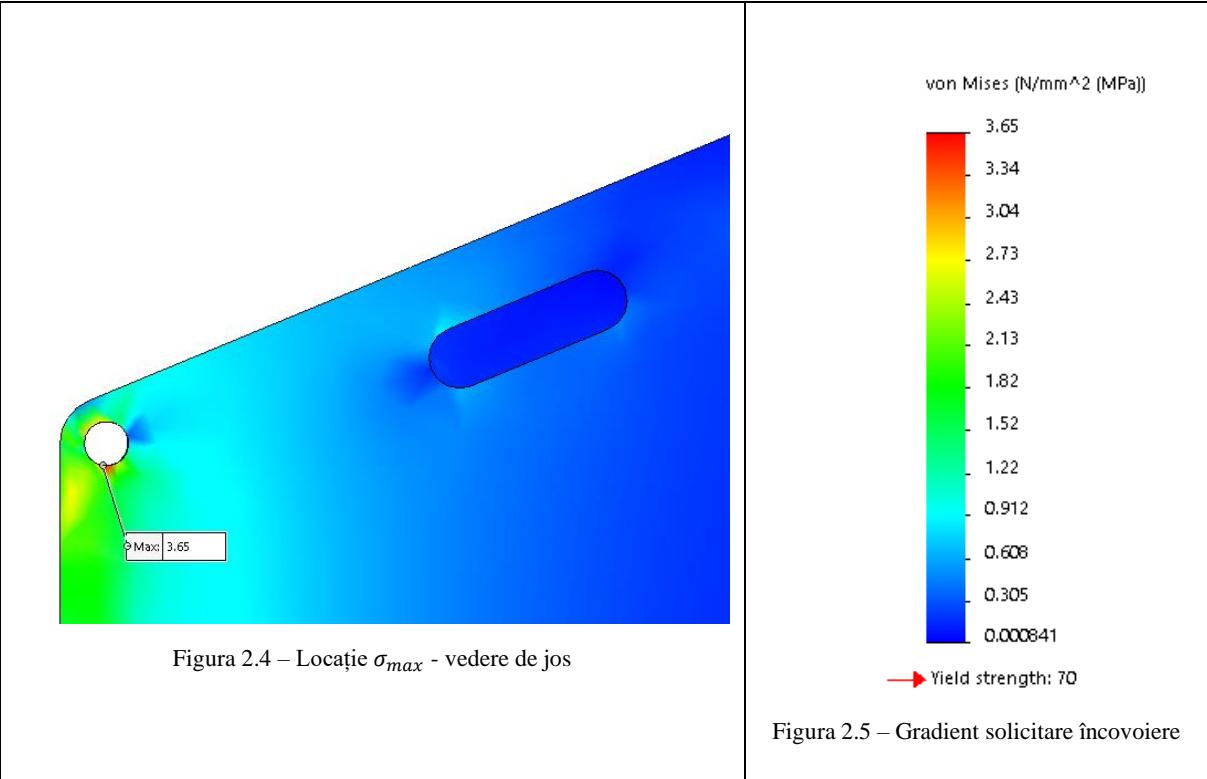
### 5.4.2 Cazul II – Încastrare dublă plan frontal, $m = 3 \text{ kg}$

**Energia potențială de deviație (criteriul von Mises)  $\sigma_{ech}$ :** În tabelul următor, Tabel 2.1, sunt evidențiate rezultatele obținute în urma simulării. Astfel, înaintea rulării simulării, Figura 2.1, ansamblul este reprezentat în formă incipientă. După rularea simulării, rezultatul obținut este reprezentat în Figura 2.2 și evidențiază distribuția energiei potențiale de deviație în placă, plaja de gradient turcoaz - verde - galben.



Tabel 2.1 – Rezultate simulare

În tabelul următor, Tabel 2.2, este evidențiat punctul de solicitare maximă, Figura 2.4, având un cuantum  $\sigma_{ech} = \sigma_{max} = 3.65 \text{ MPa}$  conform gradientului din Figura 2.5.



Tabel 2.2 – Rezultate  $\sigma_{max}$

Conform Figura 2.5, valoarea maximă a tensiunii echivalente în placa inferioară este  $\sigma_{ech} = \sigma_{max} \cong 3.65 \text{ MPa}$ . Știind că tensiunea admisibilă la încovoiere a materialului, PLA, din care este realizată placa este  $\sigma_a \cong 70 \text{ MPa}$ , placa nu va suferi deformări plastice.

$$\sigma_{max} < \sigma_a \quad (1)$$

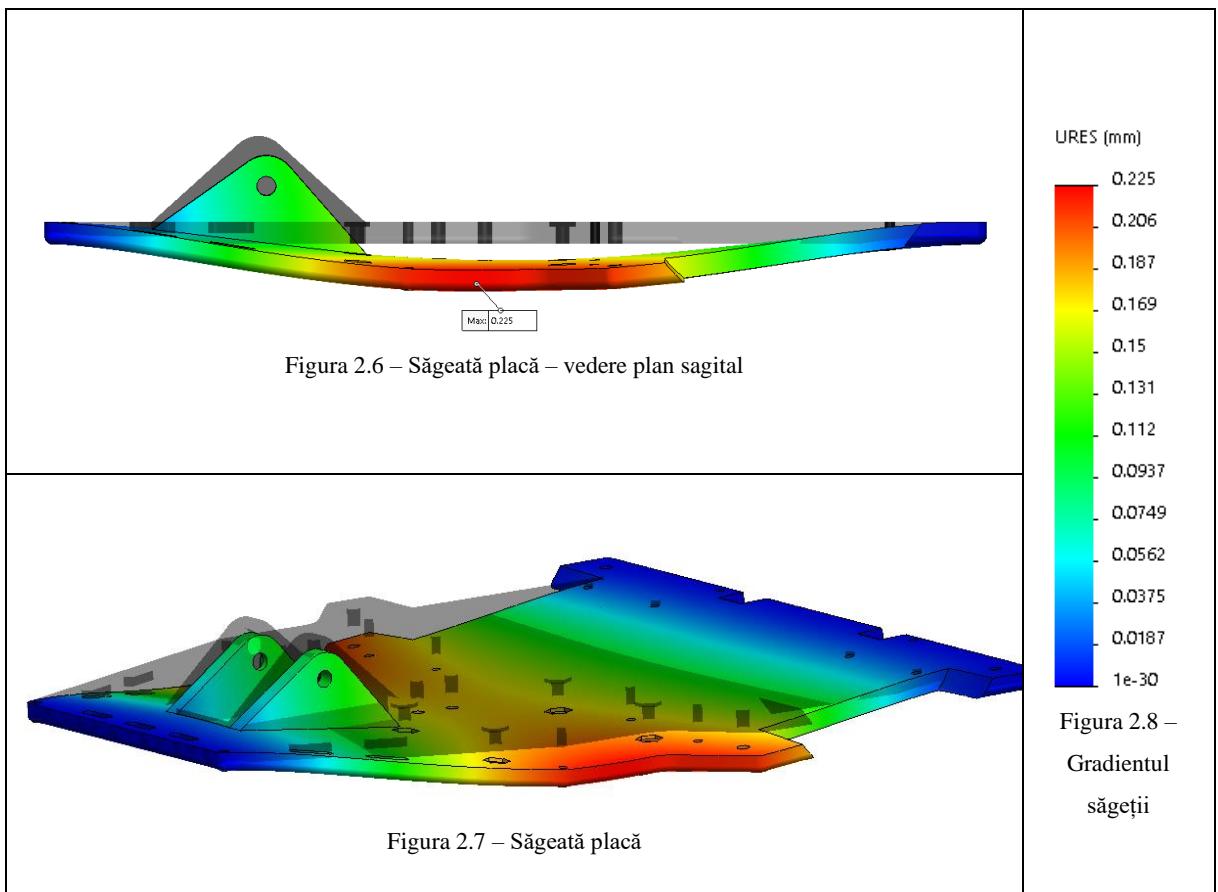
Astfel, poate fi calculat factorul de siguranță,  $c$ , al plăcii pentru **Cazul 2 - Încastrare dublă plan frontal,  $m = 3 \text{ kg}$ :**

$$c = \frac{\sigma_{ac}}{\sigma_{ech}} \quad (2)$$

Din (2) rezultă factorul de siguranță:

$$c = \frac{70 \text{ MPa}}{3.65 \text{ MPa}} = 19.17$$

**Săgeata  $v$ :** În Tabel 2.3 este reprezentată săgeata cauzată de efortul de încovoiere. În Figura 2.6 și 2.7 este reprezentată forma inițială a plăcii peste care este suprapusă forma rezultată în urma simulației și în Figura 2.8 este evidențiat gradientul săgelei  $v$ . Reprezentarea formei obținute a fost amplificată pentru a evidenția tendința de deformare în momentul în care este supusă solicitării de încovoiere și săgeata maximă este  $v_{max} = 0.225 \text{ mm}$ . Scara 50:1



**Alungirea relativă  $\epsilon$ :** În tabelul următor, sunt evidențiate rezultatele obținute în urma simulării. Astfel, în Figura 2.9, placa este reprezentată în starea pre-simulare. În Figura 2.10 este reprezentată placa în stare post-simulare. După cum se poate observa, zonele supuse alungirii relative sunt evidențiată prin plaja de gradient verde-portocaliu.

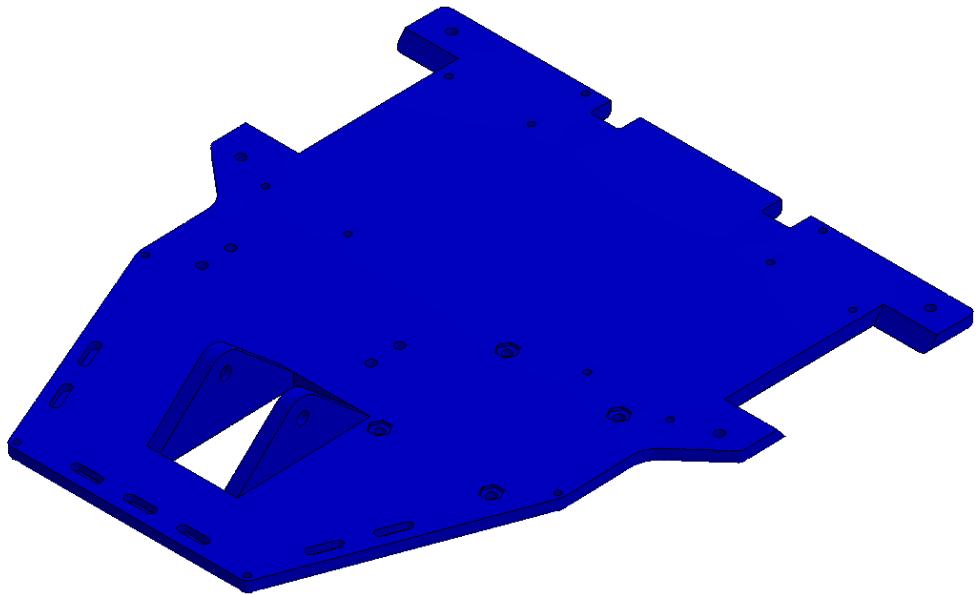


Figura 2.9 – Stare pre-simulare

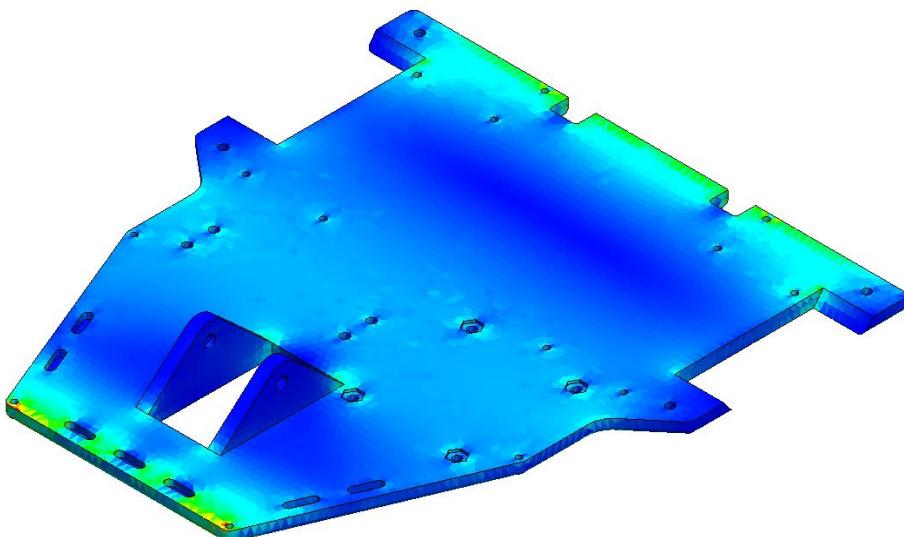


Figura 2.10 – Stare post-simulare

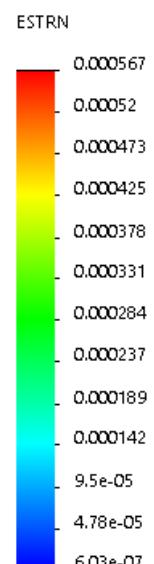


Figura 2.11 –  
Gradient alungire  
relativă

Tabel 2.4 – Rezultate alungire relativă

Conform Figura 2.11, alungirea relativă generală maximă  $\varepsilon_{max}$  este:

$$\varepsilon_{max} = 5.67 \cdot 10^{-4} \quad (3)$$

Din (3) rezultă faptul că alungirea relativă este neglijabilă în **Cazul 2 - Încastrare dublă plan frontal,  $m = 3 \text{ kg}$ .**

# Capitolul 6. Eficiență economică

Eficiența economică reprezintă raportul dintre efect și efort, comparând astfel efectul economic reprezentat de preț și efortul economic reprezentat de cost. Prețul include profitul, implicit acesta trebuie să fie mai mare decât costul.

Calculul de eficiență al proiectului de investiții analizat reprezintă stabilirea cheltuielilor de investiții, precum și a efectelor acestora, rezultate în urma materializării unui proiect.

Dependența costurilor este constituită de următorii factori:

- prețurile resurselor;
- tipul de produs fabricat;
- tehnologia de lucru;
- calitatea managementului aplicat.

Dependența prețului cuprinde următorii factori:

- performanțele produsului;
- capacitatea de absorbție a pieței.

Studiul economic are ca scop evidențierea caracteristicilor economice pentru resursele utilizate în vederea realizării sistemului holografic cu destinație comercială.

Pentru determinarea profitului se vor utiliza următoarele formule de calcul:

$$\Pi = P_g \cdot Q - C_T \cdot Q, \text{ unde:} \quad (6.1)$$

$P_g$  → prețul de vânzare

$Q$  → cantitatea de produse vândute

$C_T$  → costuri totale de producție

Costurile totale de producție se calculează după următoarea formulă:

$$C_T = C_{mat} + C_{ru} + C_{am}, \text{ unde:} \quad (6.2)$$

$C_{mat}$  → cheltuieli materiale

$C_{ru}$  → cheltuieli resurse umane

$C_{am}$  → amortizarea echipamentelor

## 6.1 Calcularea cheltuielilor materiale

Cheltuielile materiale se calculează după următoarea formulă:

$$C_{mat} = C_{pf} + C_{mp} + C_u, \text{ unde:} \quad (6.3)$$

$C_{pf}$  → cheltuieli produse finite

$C_{mp}$  → cheltuieli materie primă

$C_u$  → cheltuieli cu unități

Tabelul 6.1 - Prețurile produselor finite necesare robotului

Nr. crt.	Produs	Cantitate	Unitate măsură	Preț unitar	Preț total
1.	Raspberry PI	1	buc	389	389
2.	Camera Raspberry Pi	1	buc	150	150
3.	Senzor Ultrasonic HC-SR04	3	buc	10	30
4.	Cablu USB C	1	buc	20	20
5.	Baterie 2200mAh	1	buc	240	240
6.	Motor DYNAMIXEL W210	2	buc	1270	2540
7.	Driver Motor U2D2	1	buc	140	140
8.	Power HUB Driver U2D2	1	buc	190	190
9.	Convertor tensiune 5V	1	buc	7	7
10.	Convertor tensiune 12V	1	buc	70	70
11.	Set roți	1	buc	45	45
12.	Set șuruburi M2	1	buc	15	15
13.	Set piulițe M2	1	buc	15	15
14.	Set șuruburi M3	1	buc	20	20
15.	Set piulițe M3	1	buc	20	20

Cheltuielile pentru produsele finite se obțin însumând prețul total al fiecărui produs din tabelul 6.1.

$$C_{pf} = \sum T_{pfi} = 3891 \text{ lei} \quad (6.4)$$

Tabel 6.2 - Prețurile materiilor prime necesare fabricării componentelor specifice

Nr. crt.	Produs	Cantitate	Unitate măsură	Preț unitar	Preț total
1.	Filament PLA Negru	1	buc	140	140
2.	Filament ABS Alb	1	buc	100	100
3.	Fludor	1	buc	50	50

Cheltuielile pentru materiile prime se obțin însumând prețul total al fiecărui produs din tabelul 6.2.

$$C_{mp} = \sum T_{mpi} = 290 \text{ lei} \quad (6.5)$$

Cheltuielile cu unitățile reprezintă consumul de curent electric utilizat pentru realizarea sistemului. Valoarea estimată este de aproximativ 150 lei.

$$C_u = \frac{150}{Q} \left[ \frac{\text{lei}}{\text{buc}} \right] \quad (6.6)$$

## 6.2 Calcularea cheltuielilor cu resursele umane

Cheltuielile cu resursele umane ( $C_{ru}$ ) sunt egale cu suma dintre cheltuielile cu salariile ( $C_s$ ) și cu sumele datorate bugetului de stat ( $C_{bs}$ ):

$$C_{ru} = \frac{C_s + C_{bs}}{Q} \text{ unde:} \quad (6.7)$$

Operațiile care necesită resurse umane pentru a fi realizate sunt prezentate în tabelul 6.3.

Tabelul. 6. 3. Durata de timp a operațiilor de prelucrare necesare

Nr.	Denumire	Durată [ore]
1.	Printare placă sășiu inferior + superior	16
2.	Printare suport motoare	8
3.	Printare suport senzori	6
4.	Printare roata, suport camera, suport baterie	12
5.	Lipire componente electrice	1
6.	Finisare piese printate	2
7.	Asamblare	2

Pentru operațiile descrise mai sus, este nevoie de un tehnician mecanic cu un salariu brut de 4000 RON.

$$C_s = 4000 \text{ lei} \quad (6.8)$$

$$C_{bs} = \frac{38.5}{100} \cdot C_s = 1540 \text{ lei} \quad (6.9)$$

$$C_{ru} = \frac{C_s + C_{bs}}{Q} = \frac{5540}{Q} \left[ \frac{\text{lei}}{\text{buc}} \right] \quad (6.10)$$

### 6.3 Calculul amortizării echipamentelor

Deoarece echipamentele au un cost ridicat, amortizarea acestora se realizează în timp. În tabelul de mai jos sunt prezentate echipamentele necesare operațiilor de prelucrare.

Tabelul 6.4. Echipamente necesare

Nr.	Denumire	Preț achiziție [lei]	Amortizare[ani]	Amortizare lunară [lei]
1.	Imprimanta 3D	9000	4	187.5
2.	Unelte electrice de mâna	520	2	20
3.	Unelte de prelucrat plastic	60	1	5

Însumând amortizarea lunară a fiecărui echipament din tabelul de mai sus, s-au obținut cheltuielile cu amortizarea :

$$C_{am} = \sum \frac{T_{am}}{Q} = \frac{212.5}{Q} \left[ \frac{\text{lei}}{\text{buc}} \right] \quad (6.11)$$

### 6.4 Calculul profitului

Un tehnician mecanic va realiza în medie 5 de sisteme roboti pe lună. Astfel cheltuielile cu unitățile vor fi egale cu:

$$C_u = \frac{C_e}{Q} = \frac{150}{5} = 30 \left[ \frac{\text{lei}}{\text{buc}} \right] \quad (6.12)$$

Se calculează cheltuielile materiale:

$$C_{mat} = C_{pf} + C_u = 3891 + 290 + 30 = 4211 \text{ lei} \quad (6.13)$$

Se calculează cheltuielile cu resursele umane:

$$C_{ru} = \frac{C_s + C_{bs}}{Q} = \frac{5540}{5} = 1180 \left[ \frac{\text{lei}}{\text{buc}} \right] \quad (6.14)$$

Se calculează cheltuielile cu amortizarea:

$$C_{am} = \sum \frac{T_{am}}{Q} = \frac{212.5}{10} = 21.25 \left[ \frac{\text{lei}}{\text{buc}} \right] \quad (6.15)$$

Se calculează cheltuielile totale:

$$C_T = C_{mat} + C_{ru} + C_{am} = 4196 + 1180 + 21.25 = 5397.25 \text{ lei} \quad (6.16)$$

Pentru calcularea prețului de vânzare al unui robot se stabilește rata profitului ca fiind egală cu 60%.

$$P_g = C_T \cdot 1.6 = 5397.25 \cdot 1.60 = 8635.6 \text{ lei} \quad (6.17)$$

Se calculează profitul lunar impozabil ca fiind :

$$\Pi = P_g \cdot Q - C_T \cdot Q = 8635.6 \cdot 5 - 5397.25 \cdot 5 = 16191.75 \text{ lei} \quad (6.18)$$

Se calculează impozitul datorat bugetului de stat:

$$I_p = \frac{16}{100} \cdot \Pi = 2590 \text{ lei} \quad (6.19)$$

Astfel, profitul net lunar este:

$$\Pi_{net} = \Pi - I_p = 16191.75 - 2590 = 13601 \text{ lei} \quad (6.20)$$

Prin urmare, pentru a produce în serie sistem mechatronic pentru inspecții auto cu destinație comercială, cu o investiție inițială de aproximativ 20000 de lei, se poate obține un profit de aproximativ 13601 RON/lună din comercializarea acestui produs.

## Capitolul 7. Concluzii

În cadrul acestui capitol sunt prezentate parametrii tehnico-funcționali ai sistemului mechatronic mobil pentru inspecții auto, precum și condițiile de exploatare.

Conform celor prezentate anterior a fost proiectat și realizat un sistem mechatronic care are rolul de inspecta partea inferioară a automobilelor. În urma proiectării mecanice a fost realizat un sistem compact, având dimensiuni de gabarit mici, făcându-l fiabil pentru mediul de lucru al acestuia; întregul sistem se încadrează într-un volum de 220x200x91.6 mm<sup>3</sup>.

De asemenea, masa întregului sistem este mică, acesta fiind ușor de transportat; această masă este de 1.5 kg.

Acest sistem este prietenos cu mediul în care acționează dar și cu utilizatorul. Robotul se controlează simplu, folosind tastatura, mișările sunt predefinite, vor fi specificate utilizatorului și vor fi ușor de intuit de către acesta. Mișările principale ale robotului sunt mers înainte, mers înapoi, viraj stânga și viraj dreapta.

Imaginea este transmisă în timp real, are o calitate bună și se pot face inspecții pentru aproape toate modelele de autoturisme folosind acest robot.

**O condiție esențială de exploatare** este utilizarea unui laptop conectat la aceeași rețea WiFi cu microcontrolerul pentru a controla și a avea acces la imaginea transmisa de robot.

### Direcții de viitor:

- Cupolă transparentă pentru a proteja camera de praf
- LED-uri pentru a îmbunătății calitatea imaginii
- Sistem pentru reglarea înclinației camerei
- Îmbunătățire program de comandă

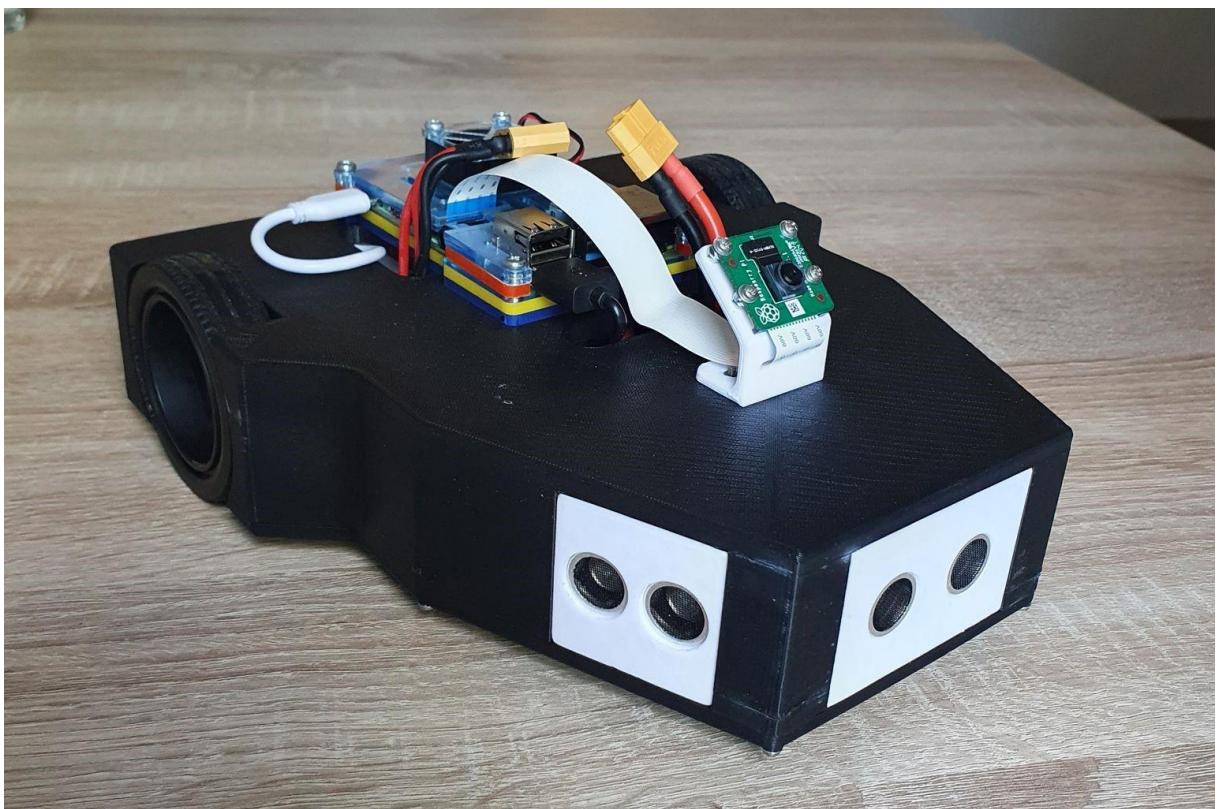


Figura 6.4.1 Ansamblu final

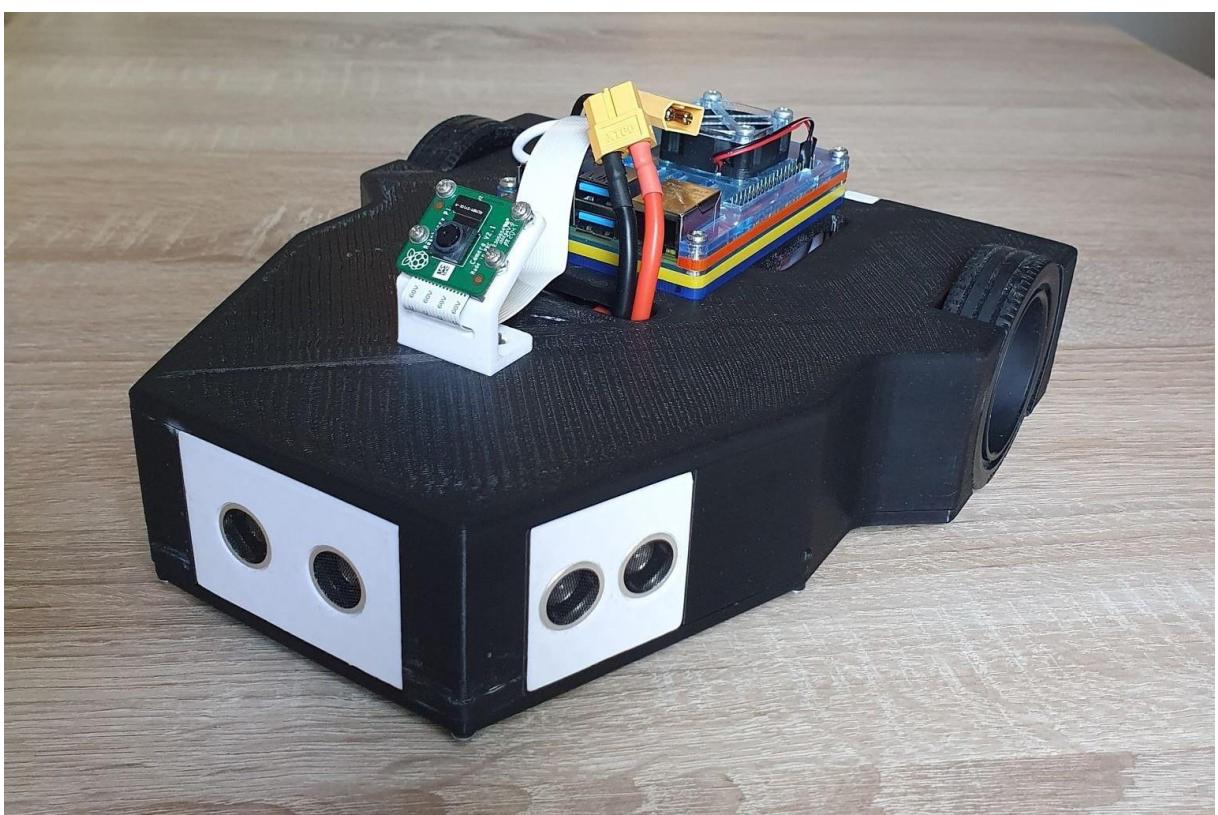


Figura 6.4.2 Ansamblu final

# Dicționar explicativ de termeni și abrevieri

ABS	- Acrylonitrile butadiene styrene
CSI	- Camera Serial Interface
DC	- Direct Current
DOF	- Degree Of Freedom
GPIO	- General-purpose input/output
HDMI	- High Definition Multimedia Interface
IP	- Internet Protocol
IR	- Infrared
LCD	- Liquid Crystal Display
LED	- Light-emitting Diode
MCU	- Microcontroller unit
MRDS	- Managed Reference Data Services (Accenture)
NASA	- National Aeronautics and Space Administration
OS	- Operating System
PLA	- Polylactic acid
PTZ	- Pan-tilt-zoom
PWM	- Pulse Width Modulation
RAM	- Random Access Memory
RPM	- Revolutions per minute
SSH	- Secure Shell
TTL	- Time-to-live
UART	- Universal asynchronous receiver-transmitter
UHMW	- Ultra High Molecular Weight Polyethylene
USB	- Universal Serial Bus
Wi-Fi	- Wireless Fidelity
WMR	- Wheeled Mobile Robot

# Bibliografie

## Articole științifice și publicații:

- 1) Gheorghe Buzdugan – „Rezistența materialelor”, Ediția 11, Editura Tehnică, București, 1979;
- 2) Pavel Tripa, Mihai Hlușcu - “Rezistența materialelor”, Editura Mirton, Timișoara, 2006;
- 3) Pavel Tripa – „Rezistența materialelor - Solicitați compuse, deformații, stabilitate, soc, oboseală, vase de rotație, tuburi cu pereți groși, solicitări peste limita de elasticitate”, Editura Mirton, Timișoara, 2001;
- 4) Curs electronic „ Proiectarea Sistemelor Mecatronice”, conf.dr.ing. Bogdan Grămescu, prof.dr.ing. Constantin Nițu
- 5) Tianyun Yao, Zichen Deng, Kai Zhang, Shiman Li – “A method to predict the ultimate tensile strength of 3D printing polylactic acid (PLA) materials with different printing orientations”, Volume 163, 2019, Pages 393-402, <<https://doi.org/10.1016/j.compositesb.2019.01.025>>;
- 6) Zbigniew Oksiuta, Marek Jalbrzykowski, Joanna Mystkowska, Eliza Romanczuk, Tomasz Osiecki – “Mechanical and Thermal Properties of Polylactide (PLA) Composites Modified with Mg, Fe, and Polyethylene (PE) Additives”, *Polymers*, 2020, <<https://doi.org/10.3390/polym12122939>>;

## Site-uri web și fișe tehnice:

- [1] [N.J. Nilsson, Shakey the robot, Technical report, SRI International, 1984.](#)
- [2] [I. Karachevtseva, J. Oberst, F. Scholten, A. Konopikhin, K. Shingareva, E.](#)
- [3] [Lunokhod 2](#)
- [4] [Viking project](#)
- [5] [Standford Cart](#)
- [6] [E.D. Dickmanns, Dynamic Vision for Perception and Control of Motion, 2007](#)
- [7] [NASA - Mars Pathfinder and Sojourner](#)
- [8] [Robotul chirurgical da Vinci](#)
- [9] [Istoria iRobot](#)
- [10] [NASA - Mars exploration rovers](#)
- [11] [Istoria ROS](#)
- [12] [Google self-driving Toyota Prius](#)
- [13] [VisLab Challenge](#)
- [14] [NASA Mars Science Laboratory - Curiosity](#)
- [15] [Google self-driving car prototype](#)
- [16] [KUKA](#)

- [17] [Standard Wheel](#)
- [18] [Roată omnidirectională - Wikipedia](#)
- [19] [Roată pivotantă](#)
- [20] [Roată sferică](#)
- [21] [Salto robot](#)
- [22] [Boston Dynamics](#)
- [23] [Nasa R5 Robot](#)
- [24] [Tank steering systems - Wikipedia](#)
- [25] [NASA Grover](#)
- [26] [Pizza Drone](#)
- [27] [Dynamixel](#)
- [28] [Senzor ultrasonic](#)
- [29] [Senzor ultrasonic - datasheet](#)
- [30] [U2D2 Driver](#)
- [31] [U2D2 Power HUB](#)
- [32] [Gens ace battery](#)
- [33] [A Guide to Understanding Battery Specifications MIT 2008](#)
- [34] [MP1584 Datasheet](#)
- [35] [MP1584 Step Down](#)
- [36] [RPI High Quality Camera Datasheet](#)
- [37] [RPI High Quality Camera Product Page](#)
- [38] [GPK032 Tracked Robot](#)
- [39] [Platforme mobile autonome](#)
- [40] [Divizor de tensiune](#)
- [41] [Raspberry PI PINOUT](#)
- [42] [Raspberry PI](#)