

Universiteti i Prishtinës "Hasan Prishtina"

Fakulteti i Inxhinerisë Mekanike

Drejtimi: Mekatronikë



PUNIM SEMINARIK

Lënda: Bazat e teknologjisë së softuerëve në prodhimin automatik

Tema: Pajisja për hulumtim dhe eksplorim

Punuan:

Sadri Shala

Lendrit Krasniqi

Florent Hoti

Albion Qovanaj

Profesori:

Dr.Sc. Shaban Buza

Prishtinë, 2020

Përmbajtja

1.	Abstrakti.....	2
2.	Hyrja	2
3.	Softueri i përdorur për konstruktiv dhe simulim.....	3
4.	Modelimi i pjesëve të Roverit.....	3
5.	Simulimi i lëvizjes së motorëve	8
6.	Moduli L293D	10
7.	Radio Moduli NRF 24L01 Pa+Lna.....	10
8.	Joystick Moduli	11
9.	Referencat	142

1. Abstrakti

Qëllimi i këtij projekti është që të marrim njohuri reth funksionimit të sistemeve mekatronike duke përdorur softuer për konstruktiv, dizajnim dhe simulim të këtyre sistemeve. Metoda e realizimit të këtij projekti është bazuar në literaturën profesionale të lëndës dhe literaturë shtesë nga interneti, ndërsa simulimi është realizuar duke përdorur softuer të avancuar kompjuterik. Gjatë konstruktimit dhe simulimit ne arritëm të marrim njohuri të mira për përdorimin e softuerve të cilët janë softuer ky në prodhim automatik.

2. Hyrja

Roveri eksplorues ka të bëjë me hulumtimin e teritoreve të vështira të cilat janë të rrezikshme ose të pamundura që njeriu ti eksplorojë. Me anë të kësaj pajisje, nga largësia mund të eksplorojnë teritore të cilat mund të jenë të banuara nga kafshë të rrezikshme apo teritore të vështira ose të ngushta që njeriu nuk mund të hyjë. Ky rover është i pajisur me modul i cili komandohet në largësi 500m deri në 1km identifikimin e zonave të vështira. Pasi që kjo pajisje është e pajisur me 6 motor dhe 6 rrota, e bën tejet të fuqishëm, dhe mënyra se si vozitet ky rover, e bën të lehtë për të që ti kalojë vështirësitë që i hasen gjatë eksplorimit.

3. Softueri i përdorur për konstruktim dhe simulim

SolidWorks është një program kompjuterik shumë i përdorur për modelime tre-dimensionale dhe montimin e pjesëve në një tersi të vetme që ne e quajmë makinë.

Parametrat i referohen kufizimeve, vlerat e të cilave përcaktojnë formën ose gjeometrinë e modelit ose montimit. Parametrat mund të jenë ose parametra numerikë, të tilla si gjatësia e vijës ose diametri i rrethit, ose parametrat gjeometrik, të tilla si tangjent, paralel, koncentrik, horizontal ose vertikal, etj. Parametrat numerik mund të shoqërohen me njëri-tjetrin përmes përdorimit të marrëdhënieve, gjë që u lejon atyre për të kapur qëllimin e projektimit.

Synimi i dizajnit është se si dizajneri i pjesës dëshiron që ajo t'i përgjigjet ndryshimeve dhe azhurnimeve.

Karakteristikat i referohen blloqeve ndërtimore të pjesës. Ato janë format dhe operacionet që ndërtojnë pjesën. Karakteristikat e bazuara në forma zakonisht fillojnë me një skicë 2D të cilat pastaj modelohen në 3D.

4. Modelimi i pjesëve të Roverit

Modeli i Roverit bazohet në roverët e përdorur nga SpaceX dhe Nasa për eksplorimin e Marsit. Roveri gjithsej përmban njëzet pjesë, të gjitha të dizajnuara në SolidWorks, të cilat në fund montohen dhe e ndërtojnë tërësinë e plotë të tij.

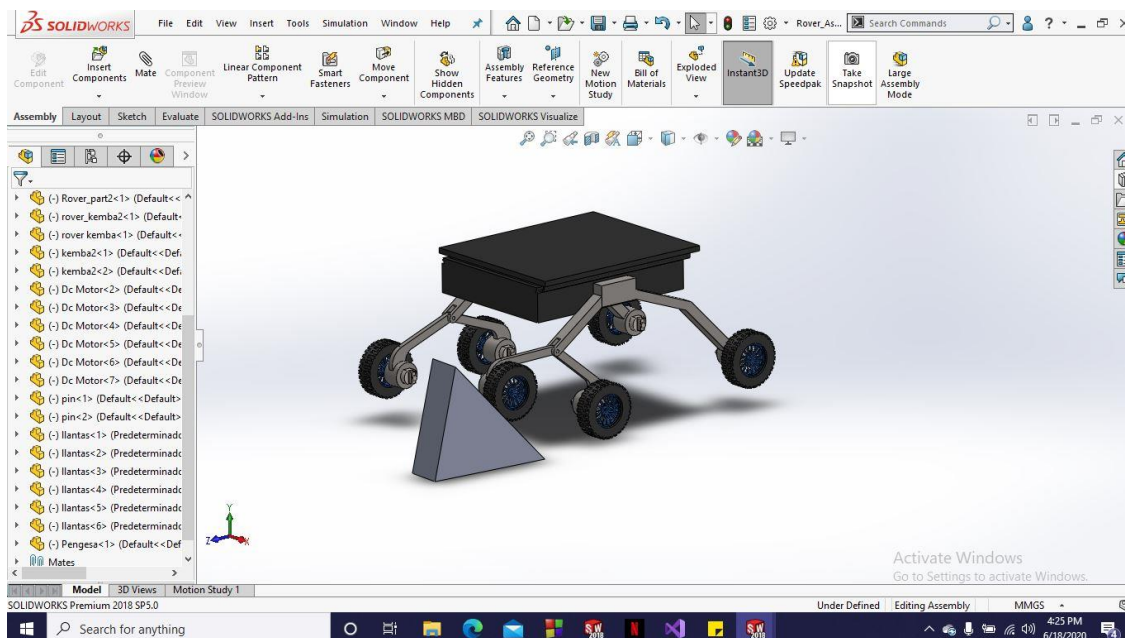


Fig 1. Roveri duke kaluar pengesën

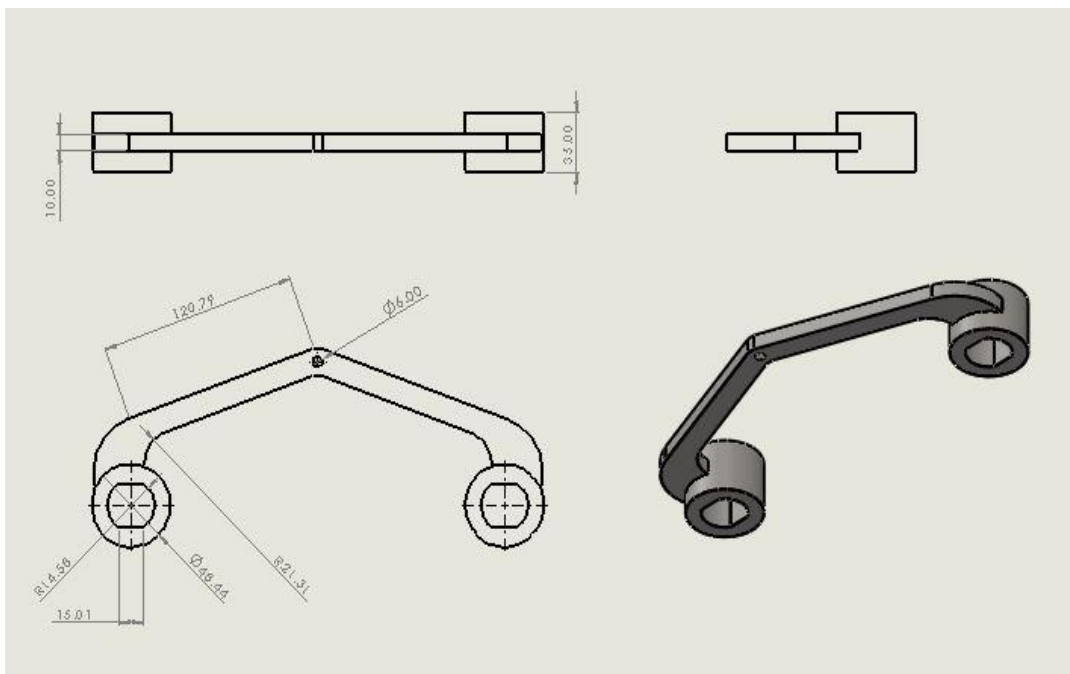


Fig 3. Mbajtësja e rrotave të përparpme

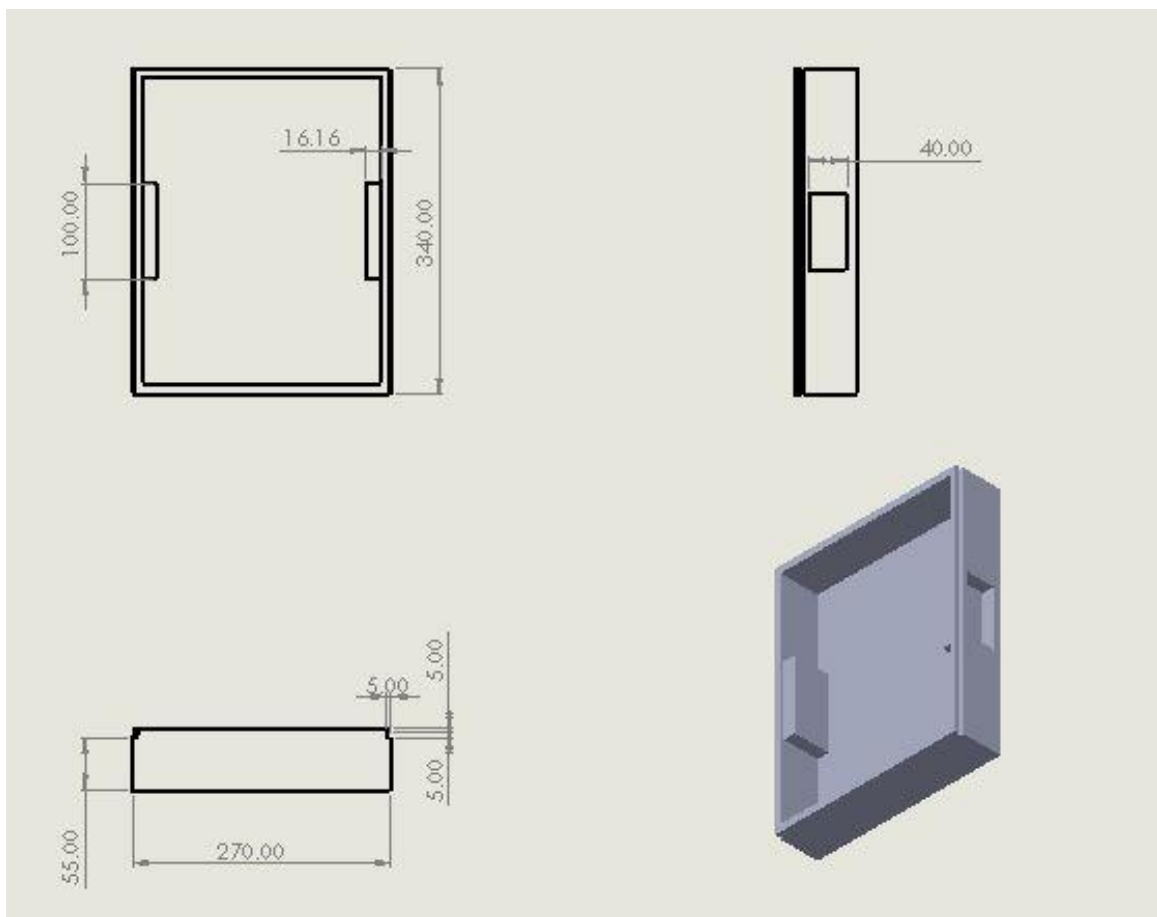


Fig 4. Mbajtësja e pjesëve elektrike

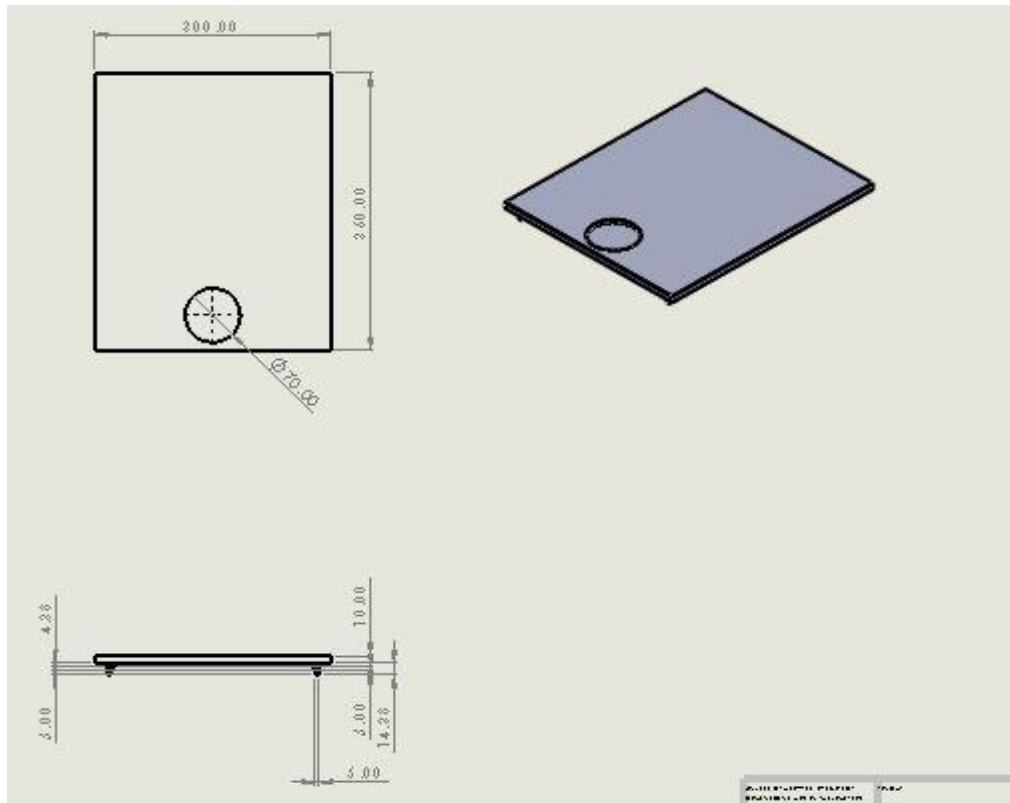


Fig 5. Mbyllësja e kutisë

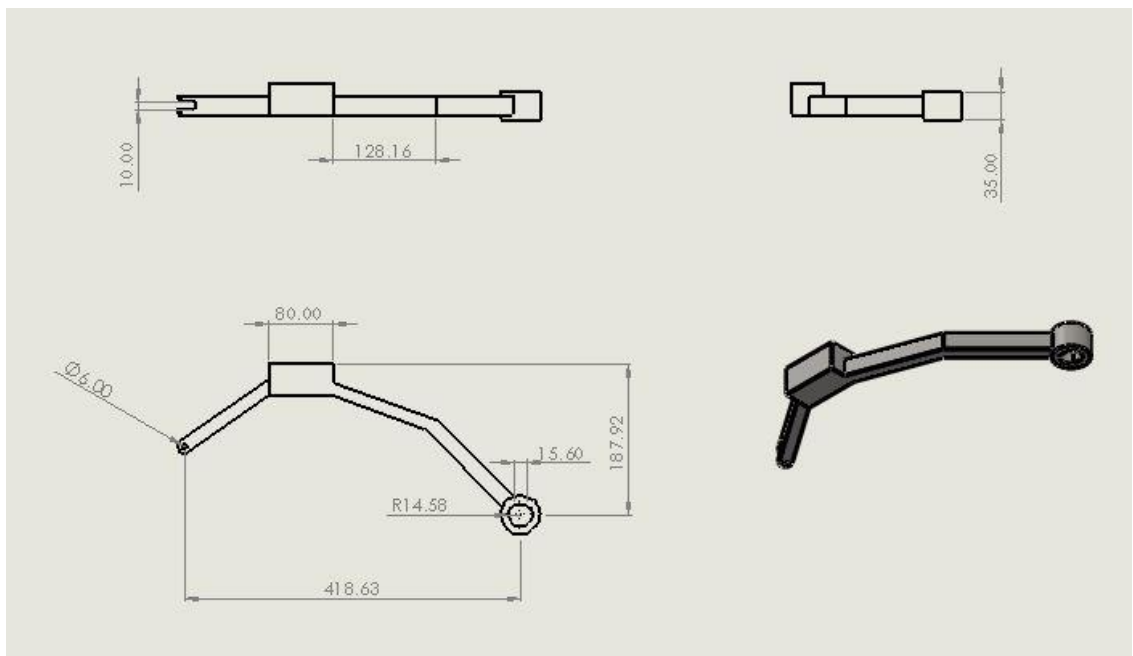


Fig 6. Mbajtësja e rrotës së prapme

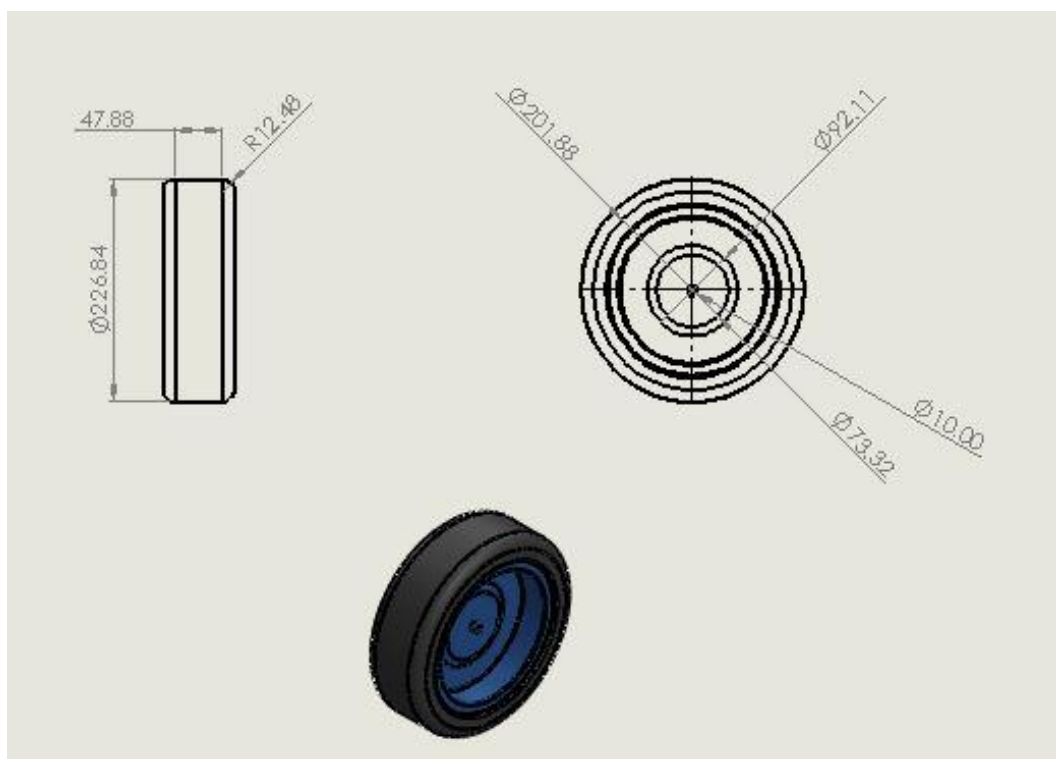


Fig 7. Rrota dhe dimensionimi

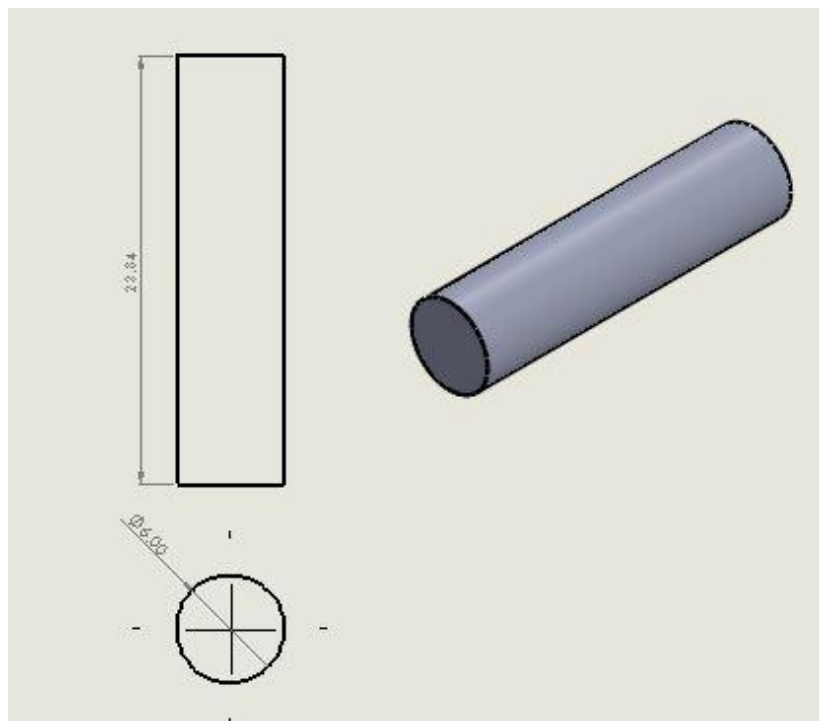


Fig 2. Lidhësja

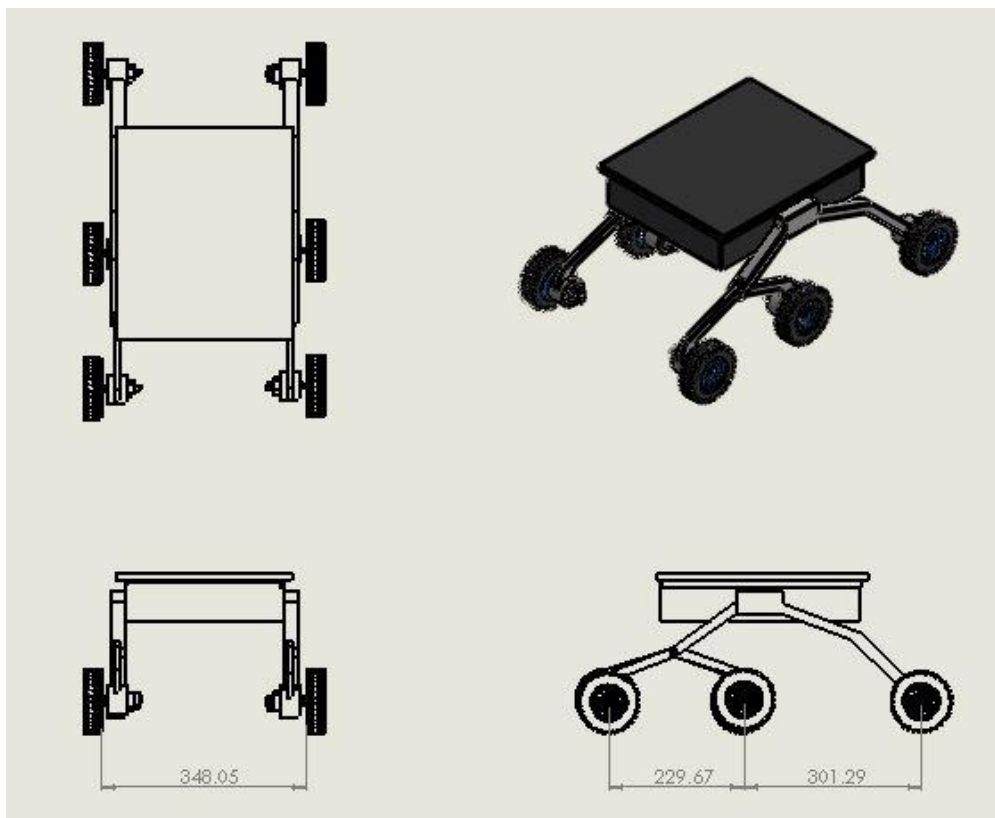


Fig 8. Pamja e tërësishme e robotit

Në pjesën e sipërme të Roverit është vendosur një kamerë e tipit SAMSUNG në të cilën përmes një aplikacioni mobil mund të kontrollohen levizjet e kameres dhe të eksplorohej terreni përreth Roverit.

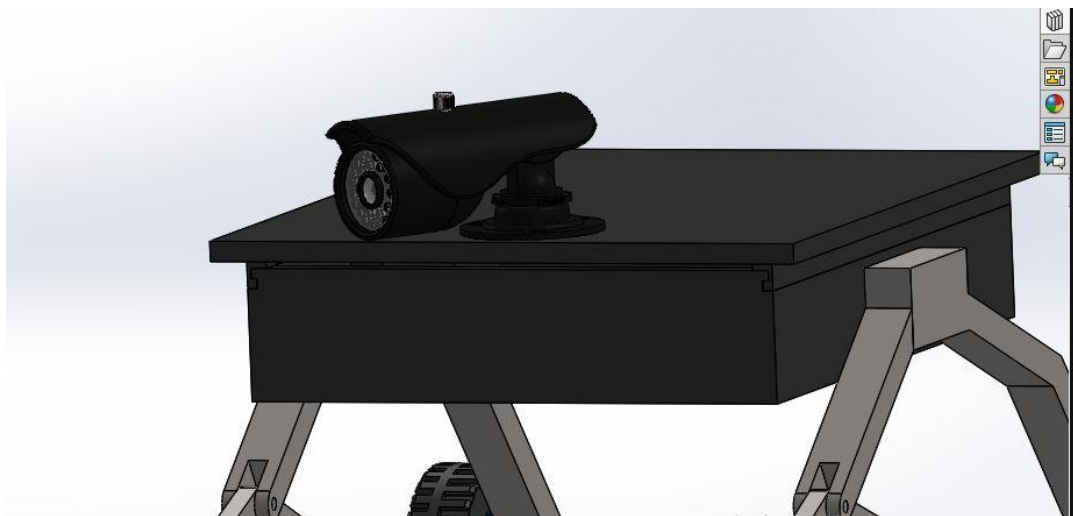


Fig 9. Kamera e vendosur në robot



Fig 10. Pamja finale e robotit në tërsi dhe në explode view

5. Simulimi i lëvizjes së motorëve

Për simulimin e lëvizjes së motorëve, është përdorur programi Autodesk TinkerCad i cili mundëson realizimin e simulimit me anë të kompajllërit të integruar të njëjtë si Arduino. Në anën e djathtë në menyn Components zgjedhen komponentët e nevojshëm, kurse nën menynë Code, zgjidhet opsioni Text dhe shkruhet kodi i nevojshëm për simulim, i bazuar në kodin Arduino.

Pasiqë është kompletuar projekti, me anë të butonit Start Simulating, simulohet i tërë projekti. Në figurat më poshtë janë paraqitur menyatë për zgjedhjen e komponentëve, për shkrimin e kodit dhe për simulimin e programit.

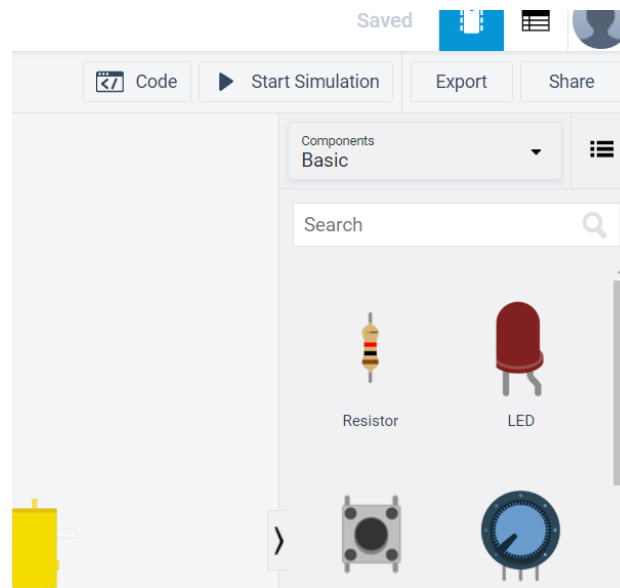


Fig 10. Komponentët

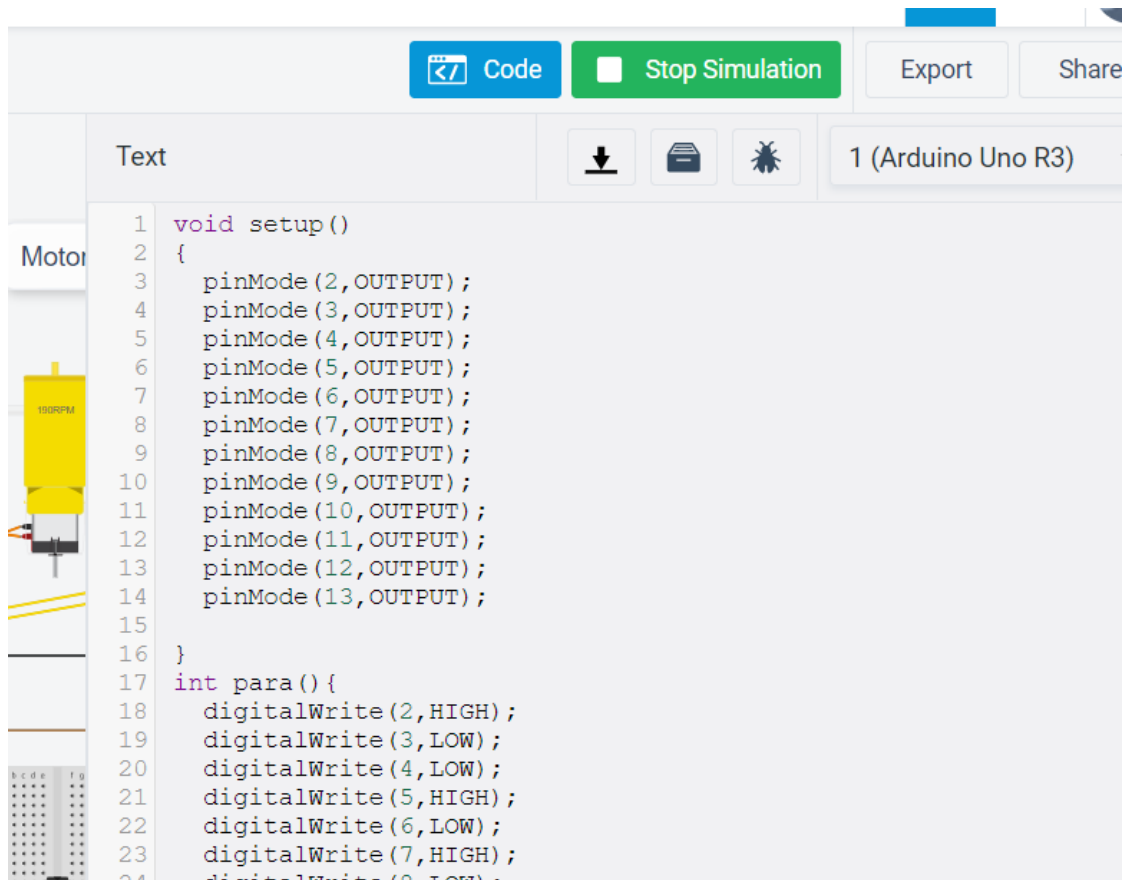


Figura 11. Programi dhe simulimi

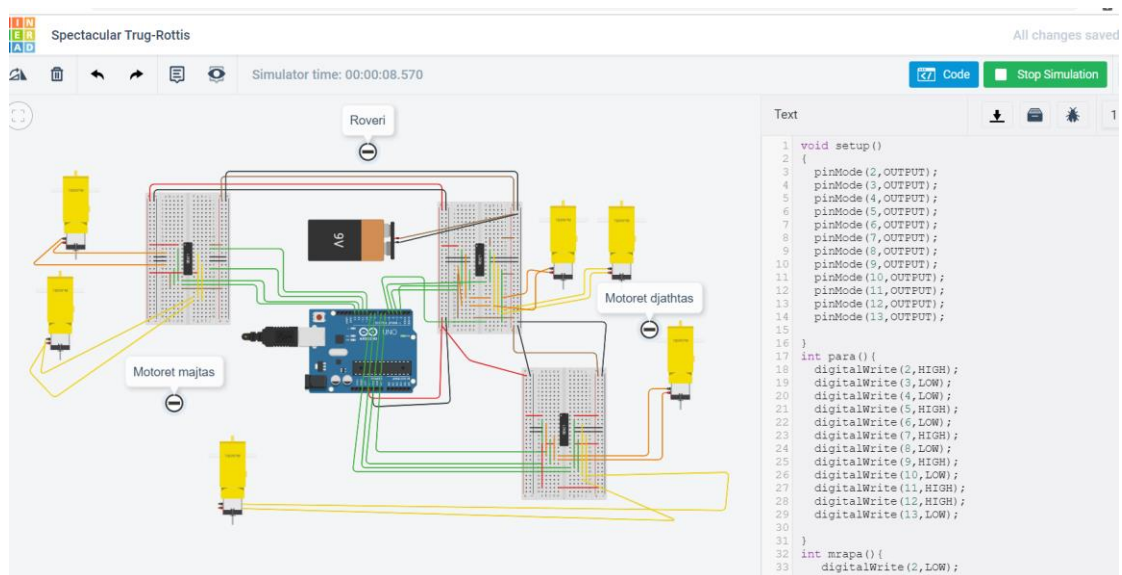


Figura 12. Simulimi i Roverit

6. Moduli L293D

L293D është një ngasës motorik i cili zakonisht është një qark i integruar, dhe përdoret për të kontrolluar motorët në robotë autonom. Moduli L293D vepron si një ndërlidhje mes arduinos dhe motorëve. L293D përbëhet nga Ura-H që është qarku më i thjeshtë për kontrollimin e kahjes së motorëve në rrymë të ulët.

Në skemën e mëposhtme janë treguar lidhjet e modudit, ku Enable 1 dhe 2 lidhen me 5V Arduino, Hyrjet e modudit lidhen në Digital 4, 5, 6, 7 Arduino, e cila jep informat rreth kahjes për rrotullim. Daljet e tij lidhen me Motorët përkatësisht me pjesën pozitive dhe negative të motorëve, ku në skemë janë të paraqitura si 1Y, 2Y, 3Y dhe 4Y.

Furnizimi i motorëve vjen nga bateria 9V, në skemë e paraqitur si VCC1 dhe VCC2 për motorët djathtas dhe majtas roverit.

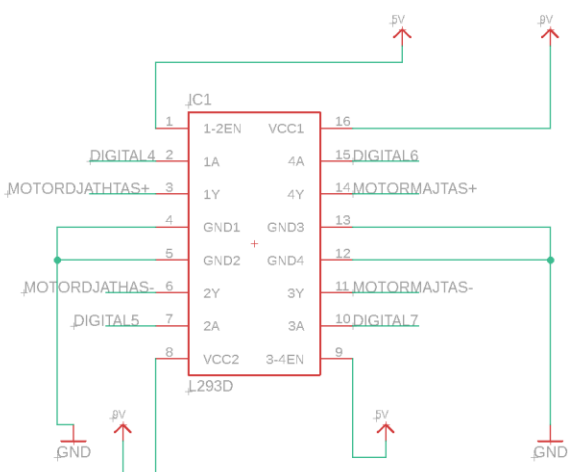


Fig 13. Skema e qarkut kontrollues L293D

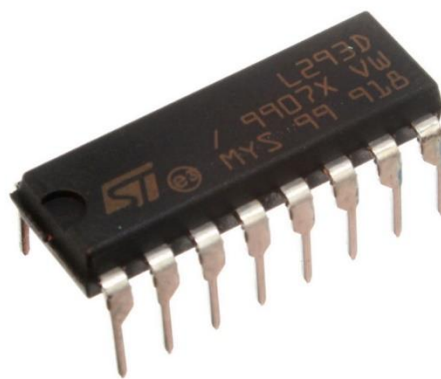


Fig 14. Pamja fizike e modudit

7. Radio Moduli NRF 24L01 Pa+Lna

Ky radio modul mundëson komunikimin ndërmjet dy apo më shumë Arduinove në një distancë mjaft të largët. Ky modul punon në parimin e dërgimit dhe pranimi të valëve elektromagnetike. Është i dizajnuar të operoj në frekuencë 2.4GHz, dhe përdor modulimin GFSK për transmetim të të dhënave. Shkalla e transmetimit të të dhënave mund të jetë 250Kbps, 1Mbps dhe 2Mbps. Në projekt janë përdorur 2 module, secila e lidhur me nga një arduino, ku njëra do të marrë të dhënat nga personi që e kontrollon roverin, dhe tjetra do të përdor ato të dhëna për komandim të motorëve, ku në konceptin e mekatronikës, njëra përdoret si Master (urdhëruesi) dhe tjetra si Slave (vepruesi).

Skema e përdorur më poshtë paraqet lidhjet e njëjta për dy radio modulet.

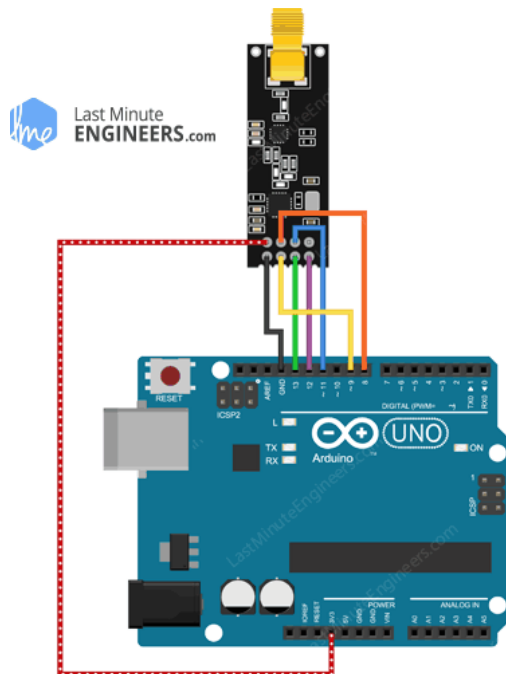


Fig 15. Lidhja e NRF 24L01 me Arduino Uno

8. Joystick Module

Ky modul mundëson kontrollin e X, Y dimensioneve, ku llojet e këtyre dimensioneve përdoren si hyrje analoge, të cilat ndryshojnë brenda me ndryshim të rezistencës.

Parimi i punës është i njëjtë me potenciometrin, pasi që ky vetë përmban 2 potenciometra, duke mundësuar kontrollim të dy dimensioneve në të njëjtën kohë.

Mënyra se si ne e kemi lidhur dhe koduar është se ne i kemi shëndrruar këto hyrje analoge të cilat nga Arduino lexohen në vlera deri në 1024, dhe vlerat të cilat janë nën apo mbi 300 i kemi kthyer si 1 apo -1, kurse vlerat të cilat janë nën këto vlera janë vlera 0, dhe kjo mënyrë e leximit nga joysticku është përdorur për kontrollim të kahjes së rrotullimit të këtyre motorëve.

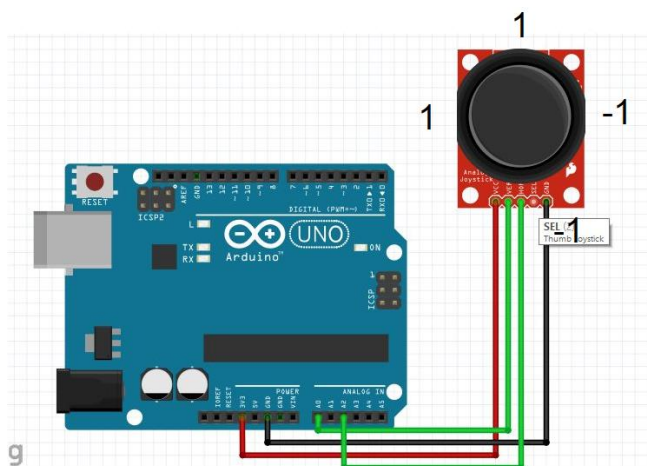


Fig 16. Skema e lidhjes se Joystick modulit me Arduino

9. Kodi

```
Joysticki
#include <RF24Network.h>
#include <RF24.h>
#include <SPI.h>
RF24 radio(7, 8);           // nRF24L01 (CE,CSN)
RF24Network network(radio);
const uint16_t node01 = 01;
const uint16_t node00 = 00;
//Joystick Analoget fillohen ne 0
int a1 = 0;
int a2 = 0;
void setup() {
    SPI.begin();
    radio.begin();
    network.begin(90, node00);
    radio.setDataRate(RF24_2MBPS);
}

void loop() {
    a1 = analogRead(A0);
    a2 = analogRead(A1);
    if (a1>300) {
        a1 = 1;
    }
    else if (-300<a1<300) {
        a1 = 0;
    }
    else if (a1<-300) {
        a1 = -1;
    }
    if (a2>300) {
        a2 = 1;
    }
    else if (-300<a2<300) {
        a2 = 0;
    }
    else if (a2<-300) {
        a2 = -1;
    }
    network.update();
    RF24NetworkHeader header1(node00);    // Adressa ku te dhenat shkojn
    bool ok3 = network.write(header1, &a1, sizeof(a1)); // Dergo te dhena
    bool ok1 = network.write(header1, &a2, sizeof(a2));
}
```

Fig 17. Arduino dhe Joystick

```

Roveri
#include <RF24Network.h>
#include <RF24.h>
#include <SPI.h>
RF24 radio(7, 8);          // nRF24L01 (CE,CSN)
RF24Network network(radio);
const uint16_t this_node = 00;
const uint16_t node01 = 01;
int motordjathtas1 = 4;
int motordjathtas2 = 5;
int motormajtas1 = 6;
int motormajtas2 = 7;

void setup() {
  SPI.begin();
  radio.begin();
  network.begin(90, this_node);
  radio.setDataRate(RF24_2MBPS);
  pinMode(motordjathtas1, OUTPUT);
  pinMode(motormajtas1, OUTPUT);
  pinMode(motordjathtas2, OUTPUT);
  pinMode(motormajtas2, OUTPUT);
}

void loop() {
  network.update();
  //Pranimi
  while ( network.available() ) {
    RF24NetworkHeader header;
    unsigned long gjendja1;
    unsigned long gjendja2;
    network.read(header, &gjendja1, sizeof(gjendja1));
    network.read(header, &gjendja2, sizeof(gjendja2));

    if (gjendja1 == 0 && gjendja2 == 0){
      digitalWrite(motordjathtas1, LOW);
      digitalWrite(motordjathtas2, LOW);
      digitalWrite(motormajtas1, LOW);
      digitalWrite(motormajtas2, LOW);
    }
    else if (gjendja1 == 1 && gjendja2 == 0) {
      digitalWrite(motordjathtas1, HIGH);
      digitalWrite(motordjathtas2, LOW);
      digitalWrite(motormajtas1, HIGH);
      digitalWrite(motormajtas2, LOW);
    }
    else if (gjendja1 == 0 && gjendja2 == 1){
      digitalWrite(motordjathtas1, HIGH);
      digitalWrite(motordjathtas2, LOW);
      digitalWrite(motormajtas1, LOW);
      digitalWrite(motormajtas2, HIGH);
    }
    else if (gjendja1 == 0 && gjendja2 == -1) {
      digitalWrite(motordjathtas1, LOW);
      digitalWrite(motordjathtas2, HIGH);
      digitalWrite(motormajtas1, HIGH);
      digitalWrite(motormajtas2, LOW);}}}

```

Fig 18. Kontrollimi i Roverit nga distanca

10. Referencat

- [1]. https://www.pinterest.com/pin/621285711076649191/?nic_v1=1awHf5m03MWxV00os5IMjQgLsTJBuOdqeWx%2BBf2wJJL%2B9CrckhgA2lFKnmTK5hQNHT
- [2]. <https://lastminuteengineers.com/>