

VILNIAUS UNIVERSITETAS
MATEMATIKOS IR INFORMATIKOS FAKULTETAS
INFORMACINIŲ SISTEMŲ INŽINERIJOS STUDIJŲ PROGRAMA

Povandeninio roboto sukūrimas

Building an Underwater Robot

Baigiamasis bakalauro darbas

Atliko: Eimantas Vinciūnas

VU el. p.: eimantas.vinciunas@mif.stud.vu.lt

Vadovas: jaunesnysis mokslo darbuotojas Linas Aidokas

Vilnius
2024

TURINYS

SANTRAUKA....5
SUMMARY....6
1. ĮVADAS....7
1.1. Temos aktualumas....7
1.2. Tikslas ir uždaviniai....7
1.3. Darbo struktūra....7
1.4. Sąvokos....7
2. ANALITINĖ DALIS....8
2.1. Povandeninių robotų apžvalga....8
2.2. Povandeninių robotų dizainai....8
2.2.1. Maksimalus dizainas....9
2.2.2. Vidutinis dizainas....9
2.2.3. Minimalus dizainas....9
2.2.4. Kuriamo dizaino išrinkimas....10
2.3. Vaizdo rinkimo metodų privalumai ir trūkumai....10
2.4. Vaizdo duomenų perdavimo metodų privalumai ir trūkumai....11
2.5. Komunikacijos metodai....12
2.6. Mikrovaldiklių ir mikrokompiuterių palyginimas....13
2.7. Variklių mechanizmų palyginimas....13
2.8. Elektros šaltinių palyginimai....14
2.9. Balasto mechanizmas ir pelekų mechanizmas....14
2.10. Propelerio savybės....14
2.11. Svorių atranka....15
3. METODINĖ DALIS....16
3.1. WebRTC....16
3.2. WebSocket....16
3.3. DHCP serveris....16
3.4. Node.js....17
3.5. UDP serveris....17
3.6. HTML, CSS ir JS....17
3.7. Google Flutter....17
3.8. MicroPython....17
3.9. Blender....18
3.10. Detalių generatoriai....18
3.11. Raspberry PI Pico W....18

3.12. Įtampos reguliatorius 19
3.13. DC variklis ir jo valdiklis 20
3.14. Servo variklis 21
3.15. Ličio jonų akumuliatorius ir apsaugos plokštė 22
3.16. Magnetai 23
3.17. Kitos dalys 23
4. ĮGYVENDINIMO DALIS 24
4.1. Duomenų perdavimo dalis 24
4.2. Elektronikos surinkimo dalis 25
4.3. Programų kūrimo dalis 27
4.3.1. Pagrindinis serveris 27
4.3.2. Valdymo pulto programėlė 29
4.3.3. Vaizdo perdavimo programėlė 30
4.3.4. Tinklo tilto programėlė 31
4.3.5. Mikrovaldiklio programėlė 32
4.4. Korpuso surinkimo dalis 33
4.4.1. Kabelio pravedimas 33
4.4.2. Detalių modeliavimas 34
4.4.3. Detalių surinkimas 37
4.5. Roboto testavimo dalis 41
5. REZULTATAI IR APIBENDRINIMAS 43
LITERATŪRA 44
PRIEDAI 46

SANTRAUKA

Pagrindinis baigiamojo darbo tikslas – sukurti povandeninį robotą. Darbo metu buvo palyginami skirtingi robotų dizainai, vaizdo perdavimo metodai ir ryšio užtikrinimo būdai. Buvo pasirinkta sukurti povandeninį robotą, kuris turi vieną propelerį gale, kuris stumia robotą į priekį ir pelekai, kurie leidžia robotui manevruoti vandenyje. Šie mechanizmai naudoja magnetinį sukibimą užtikrinti korpuso sandarumui. Toliau, aprašomas programinės įrangos veikimas ir roboto valdymo būdas. Robotas buvo surinktas iš pigių, lengvai prieinamų detalių. Taip pat buvo detalių, kurios buvo atspausdintos su 3D spausdintuvu. Galiausiai, šiame baigiamajame darbe yra aprašytas roboto testavimas ežere ir šio testo rezultatai.

Raktiniai žodžiai: povandeninis robotas, magnetinis sukibimas, tinklo ryšys

SUMMARY

The main objective of this thesis is to develop an underwater robot. During the thesis, different robot designs, image transmission methods and communication methods were compared. The choice was made to design an underwater robot, that has a single propeller at the back that pushes the robot forward and fins that allow the robot to maneuver in the water. These mechanisms use magnetic coupling to ensure the water tightness of the hull. Next, the functioning of the software and the way the robot is controlled are described. The robot has been assembled from low-cost, readily available parts. There were also parts that were printed with a 3D printer. Finally, the paper describes the testing of this robot in a lake and the results of this test.

Keywords: underwater robot, magnetic coupling, network connectivity

1. ĮVADAS

1.1. Temos aktualumas

Povandeniniai robotai leidžia individualiems žmonėms tyrinėti ežerų dugną ir augaliją bet kuriuo metų laiku. Įmontavus tinkamus jutiklius įmanoma ištirti ežerų temperatūrą ir vandens kokybę įvairiuose gyliuose. Taip pat, tokie robotai gali būti naudojami povandeninėse paieškose arba dugno valymo operacijose, jei yra įmontuotas šiukšlių rinkimo manipuliatorius.

1.2. Tikslas ir uždaviniai

Darbo tikslas – sukurti povandeninį robotą.

Darbo uždaviniai:

1. Realizuoti vaizdo perdavimą iš roboto į kompiuterį.
2. Realizuoti roboto valdymo galimybę iš kompiuterio.
3. Ištestuoti roboto veikimą vandens telkinyje.

1.3. Darbo struktūra

Analitinėje dalyje yra analizuojami metodai perduoti vaizdo duomenims iš roboto į kompiuterį ir būdai valdyti roboto mechaninius komponentus iš kompiuterio. Taip pat yra palyginami įvairūs robotų dizainai skirtinguose kainų rėžiuose ir aprašoma, kuris dizainas buvo kuriamas.

Metodinėje dalyje yra aprašomi visi įrankiai, dalys, elektronikos komponentai ir programinė įranga panaudoti kuriant robotą.

Įgyvendinimo dalyje aprašomas programinės įrangos kūrimas, roboto korpuso surinkimas, elektronikos sujungimas ir roboto bandymas vandens telkinyje. Taip pat, aptariama kokios problemos buvo iškilusios kuriant robotą ir kokie sprendimai buvo pritaikyti.

1.4. Sąvokos

- IR – infraraudona. Infraraudonieji spinduliai yra elektromagnetinio spinduliavimo dalis, kuri yra neįžvelgiama žmogaus akimi, nes jos bangų ilgiai yra ilgesni nei matomosios šviesos. Šie spinduliai dažniau yra žinomi kaip šilumos spinduliuotė.
- API – aplikacijų programavimo sąsaja (angl. Application Programming Interface). Tai yra programinės įrangos sąsaja, kuri leidžia skirtingoms programoms komunikuoti ir sąveikauti tarpusavyje. Tai reiškia, kad programuotojai gali naudoti API, kad gautų prieigą prie tam tikrų programų funkcijų ar duomenų, nesikreipdami tiesiogiai į šių programų šaltinį.
- IO – įvestis, išvestis (angl. Input Output). Tai yra trumpinys, kuris naudojamas apibrėžti įrenginį arba funkciją, leidžiančią duomenims ar signalams patekti į sistemą (įvestis) arba išeiti iš jos (išvestis).
- DC – nuolatinė srovė (angl. Direct Current). Tai yra elektros srovės tipas, kuri nuolat juda viena kryptimi ir nepakeičia polių.

2. ANALITINĖ DALIS

Šioje dalyje yra įvertinami įvairių roboto aspektų privalumai ir trūkumai. Atsižvelgus į šiuos duomenis toliau yra sudaryti galutiniai robotų dizainai ir aptariami jų privalumai ir trūkumai. Galiausiai, yra aptariama, kuris dizainas buvo išrinktas sukūrimui ir kodėl.

2.1. Povandeninių robotų apžvalga

Pirmame apžvelgtame straipsnyje aprašytas povandeninio stebėjimo dronas skirtas vandens kokybei vertinti ir povandeninei žvalgybai [1]. Dronas gali veikti savarankiškai, jame įrengti įvairūs jutikliai, kuriais matuojami pagrindiniai vandens kokybės parametrai, pavyzdžiui, pH, drumstumas ir temperatūra. Pagrindinis korpusas pagamintas iš PVC vamzdžių. Korpusas suprojektuotas su srieginiais galiniais dangteliais ir O-žiediniais sandarikliais. Judėjimo sistemą sudaro šeši bešepetėliniai DC varikliai. Keturi varikliai skirti vertikaliems judesiams, o du - judesiams pirmyn, atgal. Variklių greitį ir kryptį valdo "Raspberry Pi" mikrovaldiklis. Į droną įmontuotas kameros modulis, įrengtas permatomoje korpuso dalyje. Energiją tiekia ličio-polimero akumuliatorius.

Antrame straipsnyje aprašytas lydekos formos robotas [2]. Šis robotas pastatytas sekcijomis, kurios naudoja servo variklius su magnetiniu sukibimu judėjimui. Robotas juda BFC (angl. Body-Caudal Fin) plaukimo metodu. Robotas naudoja BLE (angl. Bluetooth Low Energy) modulius sudaryti ryšiui su kompiuteriu, kadangi bandymai vyksta laboratorijos sąlygose ir robotas plaukia paviršiuje, tai ryšys yra stabilus. Kiekviena sekcija naudoja Arduino Nano 33 BLE mikrovaldiklį. Galios šaltinis yra ličio jonų 18650 akumuliatoriai.

Trečiasis analizuojamas straipsnis aprašo penkis skirtingus povandeninių robotų judėjimo metodus [3]:

1. Klasikinis metodas – torpedos formos korpusas, gale propeleris, kryptis valdoma su vairu (angl. rudder).
2. Biomimetinis metodas – imituoja vandens gyvūnų judėjimą.
3. Vektorinių variklių metodas – naudojami propeleriai gali sukiotis ir traukti į visas puses.
4. Stacionarių vektorinių variklių metodas – naudojami propeleriai gali traukti į visas puses.
5. Sklandytuvo metodas – kaip klasikinis metodas, bet negali keisti krypties.

Šis straipsnis toliau apžvelgia magnetinio sukibimo metodus varomosioms sistemoms. Jie yra per daug sudėtingi atkartoti šiame darbe, bet paprasta magnetinio sukibimo versija buvo sukurta. Taip pat yra paminėtos vandens pompos skirtos valdyti roboto masei ir masės centrui. Jie naudojami plūduriavimo kontrolei. Atlikus šio straipsnio apžvalgą, nustatyta, kad pirmasis paminėtas „stebėjimo robotas“ naudoja stacionarių vektorinių variklių metodą, o antrasis – biomimetinį metodą. Šiame bakalauro darbe sukurtas robotas naudoja klasikinį metodą.

2.2. Povandeninių robotų dizainai

Šiame poskyriuje yra aprašomi sudaryti povandeninių robotų dizainai atsižvelgus į atliktą analizę analitinėje dalyje. Toliau bus palyginti šių dizainų pranašumai ir trūkumai ir aprašomas kodėl buvo parinktas tam tikras dizainas.

2.2.1. Maksimalus dizainas

Šis dizainas susidaro iš geriausių komponentų kurie buvo analizuojami. Kadangi tai yra prototipas, korpusas sudaromas iš 5 mm storio organinio stiklo, kurio kaina yra apie 40 eurų kvadratiniam metrui. Prie korpuso primontuoti du povandeniniai bešepetėliniai varikliai su propeleriais, kurių kaina yra apie 80 eurų. Vaizdo rinkimui naudojama Raspberry PI 5 su kamera, kainuoja apie 160 eurų. Vaizdo siuntimui naudojamas plūduras su maršrutizatoriumi, apie 40 eurų. Kadangi bešepetėliams varikliams reikia didelės srovės, tai yra naudojamas ličio polimerų akumuliatorius, kurio kaina yra apie 40 eurų ir įkroviklis, kainuojantis apie 30 eurų. Balasto mechanizmas kainuoja nuo 30 eurų iki 100 eurų. Kitos išlaidos susideda nuo 30 eurų iki 120 eurų (švino svoriai, jutikliai, laidai, klijai, šviestuvas ir t.t.). Iš viso kaina siekia nuo 450 eurų iki 610 eurų.

Šis povandeninis robotas gali panerti iki 50 m gylio, priklausomai nuo korpuso ir balasto pompos patvarumo. Gali perduoti aukštos kokybės vaizdą į paviršiuje esantį kompiuterį ir gali būti labai greitas ir manevringas.

2.2.2. Vidutinis dizainas

Korpusas sudaromas iš 5 mm storio organinio stiklo, kurio kaina yra apie 40 eurų kvadratiniam metrui. Naudojami du magnetinio sukibimo varikliai, vieno paskirtis stumti robotą į priekį, kito – pasukti robotą, kurių kaina yra apie 25 eurų. Vaizdo rinkimui naudojama Wi-Fi saugos kamera, kainuoja apie 30 eurų. Vaizdo siuntimui ir signalų perdavimui naudojamas tinklo kabelis nuo kompiuterio iki roboto, robote mažas maršrutizatorius, visa tai kainuoja apie 35 eurų. Balasto mechanizmas kainuoja nuo 30 eurų iki 100 eurų. Variklių valdymui naudojamas mikrovaldiklis, kainuoja apie 10 eurų. Naudoja ličio jonų akumuliatorius, kainuojančius apie 15 eurų. Kitos išlaidos susideda nuo 30 eurų iki 60 eurų (švino svoriai, jutikliai, laidai, klijai ir t.t.). Iš viso kaina siekia nuo 215 eurų iki 315 eurų.

Šis povandeninis robotas gali panerti iki 30m gylio, priklausomai nuo korpuso ir balasto pompos patvarumo, taip pat, nuo tinklo kabelio ilgio. Gali perduoti aukštos kokybės vaizdą į paviršiuje esantį kompiuterį ir gali būti greitas ir manevringas.

2.2.3. Minimalus dizainas

Šis dizainas pasižymi mažiausioms išlaidomis. Korpusas būtų 5 litrų stiklainis, kurio kaina yra apie 3 eurus. Pagrindinis variklis naudoja magnetinio sukibimo metodą, kurio kaina yra apie 15 eurų. Naudoja ličio jonų akumuliatorius, kainuojančius apie 15 eurų. Vaizdo ir signalų perdavimui naudojamas tinklo kabelis nuo kompiuterio iki roboto, kuris jungtųsi tiesiai į mobilųjį įrenginį naudojant adapterį, visa tai kainuoja 17 eurų. Variklių valdymui naudojamas mikrovaldiklis, kainuojantis apie 10 eurų. Vietoje balasto mechanizmo, būtų naudojamas pelekų mechanizmas, jam būtų naudojami trys servo varikliai, kurių kaina yra apie 12 eurų. Kitos išlaidos susideda nuo 30 eurų iki 50 eurų (švino svoriai, laidai, klijai, įtampos reguliatoriai ir t.t.). Iš viso kaina siekia nuo 102 eurų iki 122 eurų.

Šis povandeninis robotas gali panerti iki 2 m gylio, dėl pelekų mechanizmo, negali sustoti, kitaip pradėtų kilti aukštyn, nėra greitas ir nėra manevrus. Yra tikimybė, kad dizainas neveiks. Dėl stiklainio formos ir defektų vaizdo kokybė suprastėja.

2.2.4. Kuriamo dizaino išrinkimas

Kadangi šiam projektui finansavimas nebuvo suteiktas, tai buvo pasirinkta kurti minimalų dizainą. Taip pat, atsižvelgta, kad kuo daugiau projekto komponentų būtų galima pakartotinai panaudoti kituose projektuose.

2.3. Vaizdo rinkimo metodų privalumai ir trūkumai

Vaizdas gali būti renkamas su mobiliuoju įrenginiu. Šiuo atvejų vaizdo kokybė priklausys nuo mobilaus įrenginio specifikacijų.

Pateikiami mobilaus įrenginio privalumai:

- Mobilieji įrenginiai turi pakankamai galingus procesorius, kad vaizdas būtų suspaudžiamas ir išsiunčiamas.
- Mažesnė roboto kaina, nes nereikia specializuotų mikrokompiuterių.

Mobilaus įrenginio trūkumai:

- Mobilus įrenginys užima daug vietos, tai roboto korpusas turi tam būti pritaikytas ir išauga roboto korpuso išmatavimai.
- Roboto korpuso gedimo atveju yra rizika, kad mobilusis įrenginys bus prarastas arba sugadintas.
- Roboto elektronikos komponentų panaudojimas yra apribojamas, nes signalai turi būti perduodami pro Wi-Fi arba tinklą, neekonomiška naudoti radijo bangas.

Vaizdą galima rinkti su Raspberry PI kameromis. Šiems mikrokompiuteriams galima įsigyti IR spindulių kameras, kurios netinka nes po vandeniu IR spinduliai yra sugeriami [4]. Kameros be IR filtro – daugiau šviesos patenka į objektyvą, bet spalvos atvaizduojamos neteisingai. Taip pat, galima įsigyti plataus objektyvo kamerų, su kuriomis matomas platus laukas.

Raspberry PI kamerų privalumai:

- Yra didelis kamerų pasirinkimas.
- Kameros užima mažai vietos, tai neapsunkina korpuso dizaino.
- Plati kameros kontrolė su mikrokompiuteriu.
- Duomenų siuntimui galima naudoti ir tinklo galimybes ir radijo bangas.

Trūkumai:

- Didelė kaina, siekianti nuo 100 eurų iki 200 eurų.
- Didelės energijos sąnaudos, greitai išsektų baterijos.

Galima naudoti Wi-Fi apsaugos kameras.

Jų privalumai:

- Turi integruotas baterijas.
- Tiesiogiai transliuoja vaizdą pro Wi-Fi, nereikia mikrokompiuterio.
- Pigu – apie 30 eurų.
- Vaizdas yra automatiškai išsaugojamas.

Trūkumai:

- Nežinoma ar yra API, tai nežinoma kaip sudėtinga bus ši vaizdą naudoti savo aplikacijoje.

- Roboto elektronikos komponentų panaudojimas yra apribojamas, nes signalai turi būti perduodami pro Wi-Fi.

Dėl prieinamumo šiame projekte naudojamas mobilus įrenginys.

2.4. Vaizdo duomenų perdavimo metodų privalumai ir trūkumai

Vaizdą perduoti galima su tinklo kabeliu, kuris jungia povandeninį robotą su paviršiuje esančiu valdymo prietaisu, šiuo atveju – nešiojamu kompiuteriu.

Tinklo kabelio privalumai:

- Nėra trikdžių, didelis duomenų perdavimo greitis.
- Jei robotas nereaguoja į signalus, tai robotą galimą ištempti iš vandens su šiuo kabeliu.
- Su tinklu yra paprasta siųsti vaizdą. Naršyklės turi WebRTC (link i metodine dalį).
- Maža kaina, siekianti apie 15 eurų.

Trūkumai:

- Kabelis užima daug vietos, apsunkina roboto gabenimą.
- Kabelio masė gali apsunkinti roboto manevringumą.
- Kabelis apriboja judėjimo atstumą nuo valdymo vietos.
- Kabelio vidus yra pralaidus vandeniui, tai reikia papildomos izoliacijos.

Po vandeniu siųsti duomenis su Wi-Fi neįmanoma, nes vanduo stipriausiai sugeria elektromagnetines bangas diapazone, kuriame veikia Wi-Fi tinklai. Siųsti duomenis įmanoma su labai žemų dažnių radijo bangomis, kilohercų diapazone. Bet šis metodas turi labai daug trūkumų.

Žemo dažnio radijo bangų trūkumai:

- Labai žemų dažnių radijas negali perduoti didelio kiekio duomenų, tai vaizdas būtų labai žemos raiškos.
- Konvertuoti vaizdui į radijo signalus reikėtų dviejų mikrokompiuterių, kurie apdorotų radijo modulių signalus. Reikėtų kurti sudėtingą programinę įrangą, kuri pasirūpintų su kokia raiška siųsti vaizdą priklausomai nuo ryšio kokybės.
- Didelė kaina, siekianti nuo 200 iki 300 eurų.
- Nestandartinis radijo dažnių diapazonas gali būti apribotas įstatymų [5].
- Yra sudėtinga įsigyti žemų dažnių radijo modulį.

Šio metodo privalumas yra, kad nėra laidų. Bet dėl didelio kiekio trūkumų šis metodas nebus toliau vertinamas.

Kitas metodas perduoti duomenims, tai yra naudoti akustinius modemus. Šie prietaisai naudoja garso bangas, kad perduotų duomenis pro vandenį.

Privalumai:

- Labai didelis veikimo atstumas. Nuo 1 kilometro.

Trūkumai:

- Gaunama tik su specialiais užsakymais.
- Mažas duomenų perdavimo greitis. Apie 20 kbps.
- Labai didelės kainos. Apie 1000 eurų.

Dėl didelės kainos šis metodas nebus toliau vertinamas.

Metodas, kuriame naudojamas plūduras su antena, iš kurios perduodami duomenys su Wi-Fi.

Privalumai:

- Nėra apribotas mobilumas.

Trūkumai:

- Apribojamas maksimalus roboto plaukimo gylis.
- Antenos prijungimui reikėtų modifikuoti Raspberry PI mikrokompiuterį.
- Nenuspėjama ryšio kokybė.

Galima naudoti USB antenos adapterį ir prie jos prijungti anteną. Bet tai turi tas pačias problemas.

Toliau, metodas, kuris naudoja plūdurą, kuriame yra maršrutizatorius. Nuo šio maršrutizatoriaus yra nutiesiamas tinklo kabelis iki roboto.

Privalumai:

- Kokybiškas ryšys.
- Paprasta tinklo topologija. Paviršiuje kompiuteris gali prisijungti prie plūduro Wi-Fi tinklo. O robote – tinklo kabelis.
- Veikia su visais vaizdo rinkimo metodais, jei yra atitinkami adapteriai. Gali būti mobilus įrenginys, mikrokompiuteris su kamera arba Wi-Fi kamera.
- Didesnis galimas maksimalus atstumas nuo valdymo vietos, lyginant su tinklo kabeliu iki kompiuterio.

Trūkumai:

- Maksimalus plaukimo gylis yra apribotas kabelio ilgiu nuo plūduro.

Metodas, kuriame naudojamas plūduras su Wi-Fi ryšio stiprintuvu. Ryšio stiprintuvams dažnai reikia elektros lizdo, šį lizdą galima gauti naudojant nešiojamą įkroviklį su elektros lizdu, bet tokį turėti virš vandens yra pavojinga. Tai šis metodas nebus toliau vertinamas.

Naudoti USB kabelį nuo valdymo vietos iki roboto – šis metodas turi tuos pačius privalumus ir trūkumus kaip tinklo kabelis, bet sukurti programinę įrangą yra sudėtinga. Tai šis metodas nebus toliau vertinamas.

2.5. Komunikacijos metodai

Šiame poskyryje yra įvertinami metodai perduoti valdymo signalams į robotą. Dėl ekonomiškumo signalų perdavimo metodai yra apribojami, nuo pasirinkto vaizdo perdavimo metodo.

- Jei naudojamas tinklo kabelis galima naudoti asmeninį kompiuterį, kuris turi tinklui skirtą prievadą, arba sukonstruoti valdymo pultą pasinaudojant mikrokompiuteriu. Šis pultas turėtų turėti ekraną ir galimybę išsaugoti vaizdo medžiagą.
- Jei naudojamas Wi-Fi ryšys tai galima naudoti arba mobilų įrenginį arba asmeninį kompiuterį arba valdymo pultą.
- Jei naudojamos labai žemų dažnių radijo bangos, tai reikia arba sukonstruoti valdymo pultą arba sukonstruoti adapterį kompiuteriui. Bet šis metodas nebus naudojamas galutiniuose robotų dizainuose.

- Jei naudojamas USB kabelis, tai būtų naudojamas arba asmeninis kompiuteris arba valdymo pultas. Šis metodas, taip pat, nebuvo naudojamas sudarant galutinius robotų dizainus, kurie aprašyti 2.2 skyriuje.

Šiame projekte naudojamas tinklo kabelis, kuris prisijungia prie kompiuterio.

2.6. Mikrovaldiklių ir mikrokompiuterių palyginimas

Mikrokompiuteriai, tai yra visiškai funkcionalūs kompiuteriai, susidedantis iš mikroprocesoriaus, atminties, įvesties/įvesties įrenginių ir kitų reikalingų komponentų. Šie prietaisai turi pakankamai resursų įjungti operacinei sistemai, gali apdoroti vaizdą iš specializuotų kamerų. Taip pat, turi ribotą galimybę valdyti IO prietaisus.

Mikrovaldikliai, tai kompiuteriniai prietaisai skirti valdyti elektroninius modulius ar kitus įrenginius. Jie neturi pakankamai resursų apdoroti vaizdui iš kameros, bet gali apdoroti tinklo signalus ir valdyti IO prietaisus.

- Mikrokompiuteris Raspberry PI Zero W, kurio kaina yra apie 35 eurų, turi Wi-Fi modulį, bet dėl nedidelių kompiuterinių resursų, yra mažas kamerų pasirinkimas. Taip pat, praktiškai neišbandžius, nežinoma kaip efektyviai sugebėtų apdoroti ir išsiųsti vaizdo duomenis.
- Mikrokompiuteris Raspberry PI 4, kurio kaina yra nuo 70 eurų iki 120 eurų. Turi Wi-Fi modulį. Didelis kamerų pasirinkimas ir pakankamai kompiuterinių resursų, kad apdorotų ir išsiustų vaizdo duomenis.
- Mikrokompiuteris Raspberry PI 5, kurio kaina yra nuo 100 eurų iki 140 eurų. Turi Wi-Fi modulį. Turi daugiau kompiuterinių resursų už Raspberry PI 4.

Atsižvelgiant į įvertintus variantus, dizainuose kuriuose reikalingas mikrokompiuteris, yra naudojamas Raspberry PI 4 arba 5, nes nėra pigesnių variantų su pakankamomis specifikacijomis.

- Mikrovaldiklis Arduino UNO R3, kurio kaina yra apie 30 eurų. Nėra Wi-Fi modulio. IO naudoja 5V įtampą. Tinka DC variklių valdikliams ir servo varikliams
- Mikrovaldiklis Arduino Nano ESP32, kurio kaina yra apie 30 eurų. Turi Wi-Fi modulį. Kaip ir UNO R3 naudoja 5V įtampą.
- Mikrovaldiklis Raspberry PI Pico W, kaina siekia apie 10 eurų. Turi Wi-Fi modulį. Naudoja 3.3V įtampą, kurios gali neužtekti kai kuriems IO prietaisams.

Yra daug daugiau mikrovaldiklių, bet atsižvelgus, kad mikrovaldiklio reikės valdyti tik keliems varikliams, tai daugiau nebeanalizuojama. Roboto dizainams, kuriuose reikalingas mikrovaldiklis yra naudojamas Raspberry PI Pico W, nes jis turi Wi-Fi modulį, 3.3V įtampa yra pakankama variklių valdikliams ir šis prietaisas parduodamas už žemą kainą.

2.7. Variklių mechanizmų palyginimas

Šiame poskyriuje, yra palyginami skirtingi variklių mechanizmų tipai, ir nustatoma, kurie yra tinkamiausi kuriamam povandeniniui robotui.

Povandeniniai bešepetėliniai varikliai su propeleriais. Kainos prasideda nuo 40 eurų. Šie varikliai yra specifiškai pritaikyti būti po vandeniu, turi didelę galią, bet naudoja daug energijos. Dažnai turi montavimui skirtą pagrindą. Taip pat, būna propelerio sukimosi krypties pasirinkimas. Perkant du variklius su skirtingomis sukimosi kryptimis, išvengiama nenorimos roboto sukimosi reakcijos. Šiems varikliams reikia galingo energijos šaltinio, kuris aprašomas 2.8 skyriuje.

Kitas mechanizmas, naudoti magnetų sukibimą pro korpusą. Šioje situacijoje galima išsirinkti bet kokį variklį, kuris bus korpuso viduje ir sukurti mechanizmą, kuris perduos sukamą galią pro korpuso sieną pasinaudojant magnetų trauka. Šis mechanizmas yra pigesnis už specialiai skirtus povandeninius variklius, nes galima naudoti pigius DC variklius, kurių kaina yra mažiau nei 10 eurų. Magnetų kaina irgi siekia apie 10 eurų. Taip pat, nereikia tiek daug investuoti į energijos šaltinį. Bet korpuso medžiaga turi nereaguoti į magnetinę trauką.

2.8. Elektros šaltinių palyginimai

Šiame poskyriuje yra palyginami skirtingų rūšių akumuliatorių ir baterijų privalumai ir trūkumai.

Ličio jonų akumuliatoriai. Šie akumuliatoriai kainuoja nuo 3 eurų. Kiekviena akumuliatoriaus celė vidutiniškai turi 3.7V įtampos. Jei reikia didesnės įtampos – ličio jonų baterijų celes galima sujungti nuosekliai. Šie akumuliatoriai turi didelį energijos kiekį, nuo 2000mAh iki 3400mAh. Šiems akumuliatoriams reikia apsauginės plokštės, nuo trumpo jungimo, per didelio iškrovimo ir įkrovimo. Yra įkrovimo moduliai su apsauga, kurie pasirūpina, kad kiekvienas akumuliatorius turėtų vienodą įtampą įkraunant ir iškraunant. Įkrovimui aparatūra kainuoja nuo 5 eurų, kaina taip pat priklauso nuo to kiek akumuliatorių yra sujungtų nuosekliai.

Toliau yra ličio polimerų akumuliatoriai, šie turi mažesnį energijos tankį, negalima sujungti nuosekliai, bet yra prekyboje akumuliatorių nuo 3.7 V iki 22.2 V. Šie akumuliatoriai pasižymi gebėjimu saugiai tiekti labai dideles sroves, nuo 20 C iki 100 C. Kainuoja nuo 20 eurų. Įkrovimo aparatūra nuo 30 eurų.

Toliau galima naudoti AA baterijas. Jos nėra pakraunamos, jas reikėtų dažnai keisti, bet pradinė kaina maža, nereikia apsaugos modulių ir įkrovimo aparatūros.

2.9. Balasto mechanizmas ir pelekų mechanizmas

Povandeninių robotų balasto mechanizmas, naudojantis pompas, yra skirtas reguliuoti robotų plūdrumą ir taip užtikrinti jų vertikalią padėtį vandenyje. Šiame mechanizme naudojamos pompos, kurios į balasto talpas pumpuoja arba iš jų ištraukia vandenį. Keičiant vandens kiekį talpose, galima kontroliuoti roboto tankį, o tai leidžia jam kilti į viršų, leistis žemyn ar išlaikyti pastovią padėtį vandenyje.

Pelekų mechanizmas, priešingai nei balasto mechanizmas, nenaudoja plūdrumo reguliavimo. Vietoj to, pelekai, esantys povandeninio roboto konstrukcijoje, naudojami manevravimui vandenyje. Naudojant pelekus, robotas gali keisti kryptį, stabiliai plaukti tiesiai ar atlikti kitokius judesius. Tai leidžia geriau kontroliuoti roboto judėjimą ir poziciją. Trūkumas – jei robotas sustoja, jis negali išlaikyti vertikalios padėties.

2.10. Propelerio savybės

Šiame projekte yra naudojamas toroidinis propeleris [6]. Jis pasižymi tylesniu veikimu garso dažnių diapazone, kuriam žmonės yra jautriausi. Bet, šiame projekte toks propeleris buvo sugeneruotas su atviro kodo įrankiu ir 3D atspausdintas, tai forma gali neatitikti šaltinyje pavaizduotam propeleriui. Taip pat, 3D spausdinimo nelygumai gali pakenkti propelerio efektyvumui.

2.11. Svorių atranka

Kad sukurtas robotas galėtų panerti, bet ne skęsti, korpusas turi sverti tiek, kiek išstumia vandens tūrio [7]. Korpuse turi būti vietos įstatyti komponentams, tai svoriai turi būti tankesni už vandenį. Lentelėje yra palyginamos skirtingos, lengvai prieinamos, medžiagos ir jų tankiai (2.1 lentelė).

2.1 lentelė. Medžiagų tankiai ir kainos

Medžiaga	**Tankis, g/cm3**	**Kaina, €/kg**
Smėlis	1.4 - 1.5	-
Druska	2.17	0.5
Geležis	7.874	4
Švinas	11.35	8

Robote didžioji dalis masės yra švino svoriai, nes turi didžiausią tankį ir prekyboje yra nardymui skirtų švino svarmenų. Galutinai pasiekti plūdrumo pusiausvyrai yra naudojami maišeliai su druska.

3. METODINĖ DALIS

Šioje dalyje yra aprašomos technologijos, programos, įrankiai, dalys ir elektronikos komponentai panaudoti kuriant robotą.

3.1. WebRTC

WebRTC – realiojo laiko komunikacija tinklu (angl. Web Real-Time Communication), tai nemokamas, atviro kodo projektas, leidžiantis naršyklėms ir mobiliosioms programoms tiesiogiai bendrauti tarpusavyje per lygiarangių ryšį (angl. peer-to-peer communications, P2P) [8]. Tai leidžia perduoti garso, vaizdo ir duomenų srautus tarp įrenginių be tarpinių serverių ir užtikrina realaus laiko bendravimą.

WebRTC yra sudarytas iš kelių API ir protokolų, kuriuos palaiko dauguma šiuolaikinių naršyklių. Pagrindiniai WebRTC komponentai naudoti projekte yra:

- MediaStream API – leidžia naudoti ir valdyti garso bei vaizdo srautus, pvz., gaunamus iš mikrofono ar kameros.
- RTCPeerConnection API – leidžia tiesiogiai bendrauti su kitais įrenginiais, palaikančiais WebRTC, naudojant P2P ryšius.
- ICE (angl. Interactive Connectivity Establishment) protokolas, kuris padeda rasti geriausią kelią tarp dviejų įrenginių, kad būtų užmegztas ryšys.
- SDP (angl. Session Description Protocol) šis protokolas skirtas apibūdinti multimedijos sesijų parametrams.

Kuriant programinę įrangą skirta siųsti vaizdui iš roboto į kompiuterį, buvo pasiremta atviro kodo projektu, kuriame yra minimalus vaizdo siuntimo įgyvendinimas [9].

3.2. WebSocket

WebSocket yra protokolas, leidžiantis užmegzti dvipusį ryšį tarp kliento (dažniausiai naršyklės) ir serverio. Skirtingai nuo tradicinio HTTP protokolo, kuris yra skirtas vienakrypčiam ryšiui (klientas siunčia užklausą, serveris atsako), WebSocket leidžia abiem pusėms siųsti ir gauti duomenis bet kuriuo metu. Šis protokolas naudoja TCP (angl. Transmission Control Protocol) ryšį. WebSocket yra standartizuotas pagal RFC 6455 ir plačiai palaikomas šiuolaikinėse naršyklėse bei serverių technologijose [10].

3.3. DHCP serveris

DHCP (angl. Dynamic Host Configuration Protocol) yra tinklo protokolas, kuris automatiškai priskiria IP adresus ir kitus tinklo parametrus kompiuteriams ir įrenginiams tinkle, leidžiant jiems komunikuoti tarpusavyje. Tai supaprastina tinklo administravimą, nes nereikia rankiniu būdu konfigūruoti IP adresų kiekvienam įrenginiui.

Kai įrenginys (klientas) prisijungia prie tinklo, jis siunčia DHCP užklausą ieškodamas DHCP serverio. DHCP serveris, gavęs užklausą, atsako su pasiūlymu, kuris apima IP adresą ir kitus konfigūracinius parametrus, pvz., potinklio kaukę, numatytąjį šliuzą (angl. gateway) ir DNS (angl. Domain Name System) serverius. Klientas priima pasiūlymą siųsdamas žinutę, patvirtinančią, kad jis nori naudoti pasiūlytą IP adresą. DHCP serveris patvirtina adresų priskyrimą siųsdamas žinutę, kurioje yra galutiniai parametrai. Nuo to momento klientas gali

naudoti jam priskirtą IP adresą. Šiame projekte naudojamas Windows operacinei sistemai sukurtas DHCP serveris [11], šis projektas yra nemokamas akademiniais ir komerciniais tikslais.

3.4. Node.js

Node.js yra atvirojo kodo, tarpplatformė JavaScript vykdymo aplinka, leidžianti kurti serverio pusės programas naudojant JavaScript. Node.js yra pastatytas ant V8 JavaScript variklio, kurį sukūrė „Google“, ir leidžia vykdyti JavaScript kodą ne naršyklėje [12]. Šiame projekte naudota Node.js 21 versija.

3.5. UDP serveris

UDP (angl. User Datagram Protocol) serveris naudoja UDP protokolą, kad galėtų priimti ir siųsti duomenis. UDP yra vienas iš pagrindinių interneto protokolų, veikiantis tinklo lygmenyje ir užtikrinantis greitą duomenų perdavimą be ryšio nustatymo ir valdymo. UDP yra paprastesnis ir greitesnis nei TCP, nes jis neturi ryšio nustatymo, srauto valdymo ir klaidų taisymo mechanizmų. Šiame projekte naudojamas UDP serveris, kuris veikia Node.js aplinkoje [13].

3.6. HTML, CSS ir JS

HTML, CSS ir JavaScript yra trys pagrindinės programavimo kalbos naršyklėse, naudojamos kuriant interneto puslapius ir žiniatinklio programas. Kiekviena iš šių technologijų atlieka skirtingą vaidmenį tinklalapių veikime.

HTML (angl. HyperText Markup Language) yra pagrindinė žiniatinklio puslapių struktūrizavimo kalba. Ji naudoja žymeklius (angl. tags) ir atributus, kad apibrėžtų turinio elementus ir jų struktūrą puslapyje. HTML nurodo naršyklei, kaip turinys turėtų būti išdėstytas ir pateiktas.

CSS (angl. Cascading Style Sheets) yra stiliaus aprašymo kalba, naudojama nurodyti, kaip HTML elementai turėtų būti rodomi ekrane.

JS (angl. JavaScript) yra programavimo kalba, naudojama pridėti dinamiškumą ir interaktyvumą interneto puslapiams. JavaScript gali manipuliuoti HTML ir CSS, reaguoti į vartotojo veiksmus ir atlikti daugybę kitų funkcijų.

Naudojant šias naršyklės kalbas, šiame projekte yra sukurtas valdymo pultas ir vaizdo perdavimo programėlė.

3.7. Google Flutter

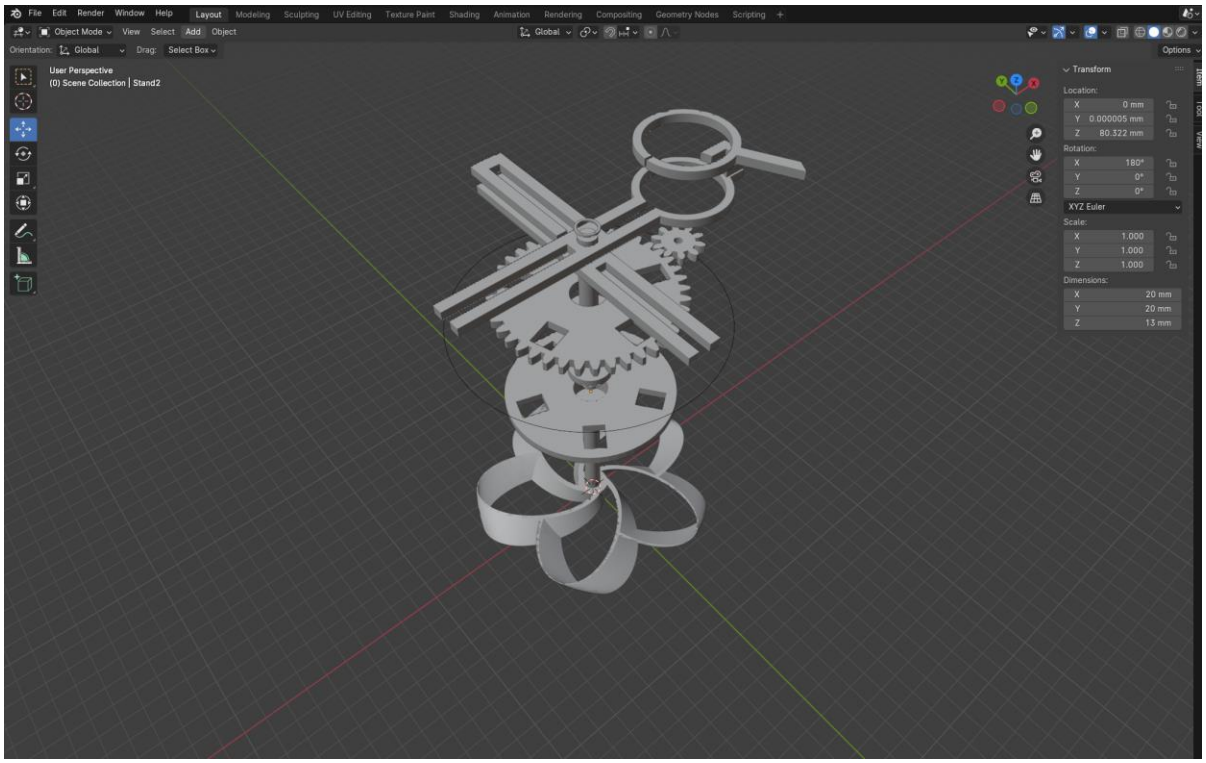
Google Flutter yra atviro kodo daugiaplatformis mobiliųjų programėlių kūrimo karkasas. Kuris leidžia kurti grafinias vartotojo sąsajas Android ir iOS įrenginiams. Flutter naudoja Dart programavimo kalbą, kurią taip pat sukūrė Google [14]. Šiame projekte naudojama 3.19.3 versija.

3.8. MicroPython

MicroPython yra „Python 3“ interpretatorius, skirtas mikrovaldikliams ir įterptinėms sistemoms. Tai mažas ir efektyvus Python programavimo kalbos variantas, pritaikytas įvairiems mažiems įrenginiams su ribotais resursais. Šiame projekte naudojamas MicroPython, kuris pritaikytas Raspberry Pi Pico W mikrovaldikliui [15].

3.9. Blender

Blender yra nemokama, atvirojo kodo 3D kūrimo programa, skirta modeliavimui, animacijai, vaizdų kūrimui ir daugeliui kitų projektų. Šiame projekte šis įrankis buvo naudojamas sukurti 3D modeliams skirtiems 3D spausdintuvui (3.1 pav.).



3.1 pav. Blender programos pavyzdys

3D spausdinimui tinkamiems modeliams kurti, buvo pakeista Blender scenos konfigūracija. Scenos matavimo vienetai pakeisti iš metrų į milimetrus ir vienetų skalė sumažina iš 1 į 0.001.

3.10. Detalių generatoriai

Krumpliaračiai ir propeleris buvo sugeneruoti pasinaudojant nemokamus generatorius. Krumpliaračiams generuoti buvo naudojamas internetinės programėlės įrankis „STLGears“ [16]. Propeleriui sugeneruoti buvo naudojamas atviro kodo įrankis „Ultimate Toroidal Propeller Generator“ [17].

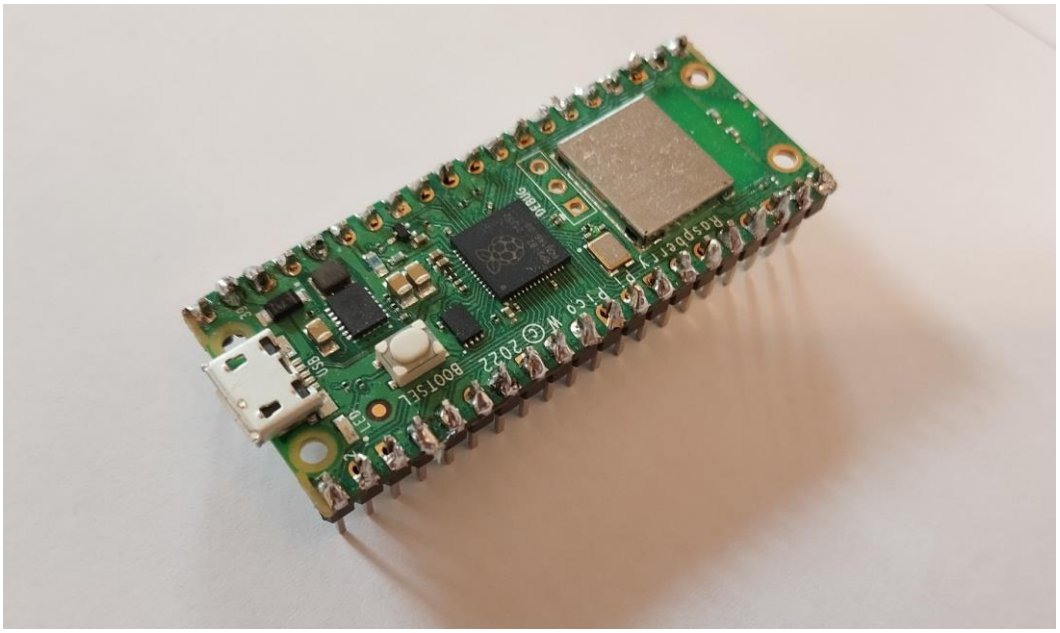
3.11. Raspberry PI Pico W

„Raspberry Pi Pico W“ yra mikrovaldiklio plokštė (3.2 pav.), sukurta „Raspberry Pi Foundation“. Ji yra „Raspberry Pi Pico“ plokštės variantas, papildytas Wi-Fi ryšiu, leidžiantis kurti projektus su belaidžiu ryšiu.

Pagrindinės „Raspberry Pi Pico W“ savybės [18]:

- Procesorius – Dual-core ARM Cortex-M0+, iki 133 MHz.
- Laikina atmintis – 264KB SRAM.
- Atmintis – 2MB.
- Wi-Fi – Infineon CYW43439 2.4GHz 802.11n.

- 26 daugiafunkciniai GPIO išvadai. Palaiko įvairius protokolus, tokius kaip I2C, SPI, UART, PWM, ADC ir kt.

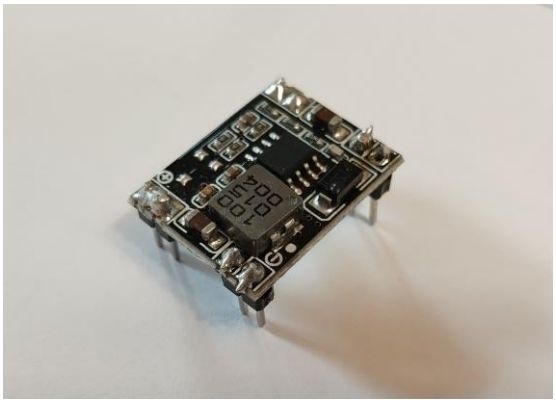


3.2 pav. Raspberry PI Pico W

Šiame prietaise buvo įdiegtas MicroPython interpretatorius.

3.12. Įtampos reguliatorius

Žeminamasis (angl. Step-down) įtampos reguliatorius (3.3 pav.) yra elektroninis prietaisas, skirtas sumažinti įeinančią įtampą į mažesnę. Šis reguliatorius naudojamas įvairiose elektroninėse sistemose, kur reikia stabilios ir žemesnės įtampos nei tiekiama pradiniame šaltinyje. Šie komponentai pasižymi dideliu efektyvumu ir mažu šilumos išskyrimu, o tai reiškia, mažai energijos prarandama šilumos pavidalu.



3.3 pav. Įtampos reguliatorius

Šie prietaisai yra įvairių formų, išvaizdų ir specifikacijų. Šiame projekte buvo naudojami keturi įtampos reguliatoriai:

1. 14 – 24 V į 12 V DC/DC žeminamasis įtampos reguliatorius.
2. 5.3 – 24 V į 3.3 V DC/DC žeminamasis įtampos reguliatorius.
3. Nuo 4.75 V – 23 V į 1 V – 17 V reguliuojamas žeminamasis įtampos reguliatorius.

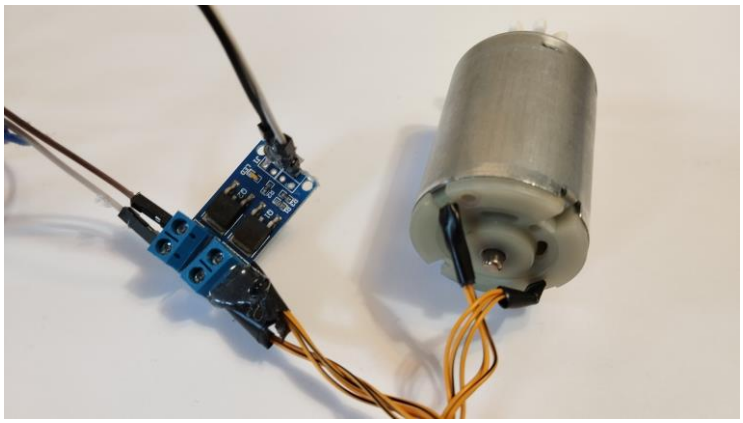
4. Nuo 3 V – 40 V į 1.5 V – 35 V reguliuojamas žeminamasis įtampos reguliatorius.

Žeminamiesiems įtampos reguliatoriams turi būti įvedama aukštesnės srovės įtampa negu išvedama. Jei tai reguliuojamas įtampos reguliatorius, tai galima pasirinkti kokios įtampos srovė bus išvedama.

3.13. DC variklis ir jo valdiklis

DC variklis, arba nuolatinės srovės variklis yra elektrinis variklis, veikia nuo nuolatinės srovės šaltinio. Šis, 3.4 paveiksle vaizduojamas šepetėlinis DC variklis, turi kolektorių ir šepetėlius, kurie padeda keisti srovės kryptį variklio vyniojimuose, kad variklis suktųsi.

PWM (angl. "Pulse Width Modulation", liet. "impulsų pločio moduliacija") valdiklis yra elektroninis prietaisas, naudojamas valdyti galios tiekimą į DC variklį (3.4 pav.). PWM metodas reguliuoja vidutinę tiekiamos galios vertę, kintant impulsų pločiui per tam tikrą periodą. Trumpai tariant, PWM valdiklis reguliuoja variklio greitį keisdamas impulso trukmę (darbo ciklą), o ne keisdamas tiekiamą įtampą ar srovę.



3.4 pav. DC variklis ir jo valdiklis

DC variklio specifikacijos:

- Įtampos diapazonas 6 - 12 V.
- Esant 12 V įtampai, srovė yra 1.95 A, o greitis – 18500 aps./min.
- Esant 6 V įtampai, srovė yra 1.3 A, o greitis – 4500 aps./min.
- Variklio ilgis: 50mm.
- Variklio skersmuo: 35.5 mm.
- Fiksuoto skylės atstumas: 25 mm.
- Ašies ilgis: 5 mm.
- Ašies skersmuo: 3.175 mm.
- Svoris: apie 161 gramų.

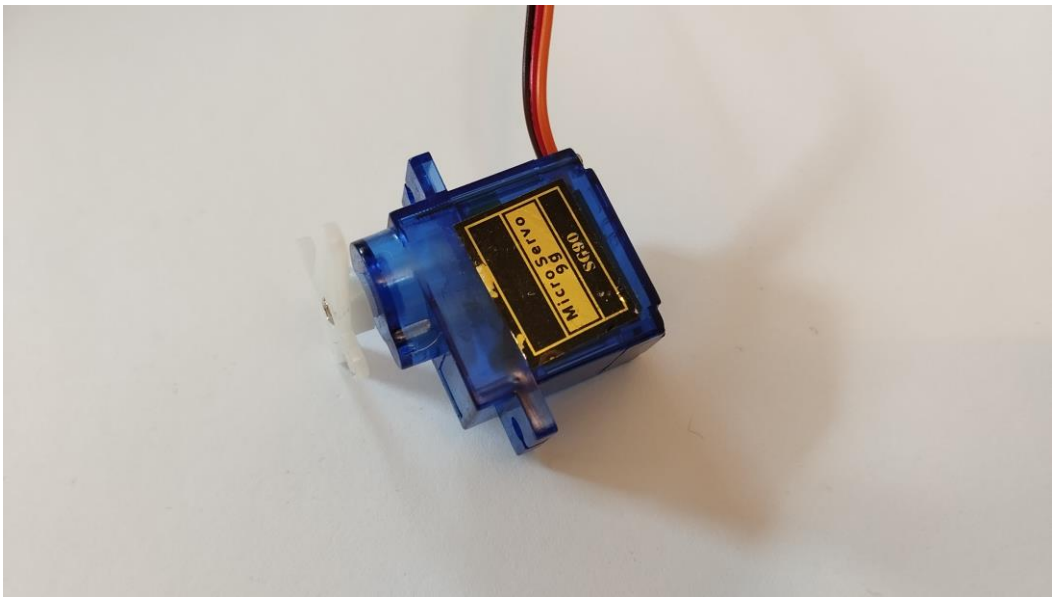
PWM valdiklio specifikacijos:

- Maksimali galia 400 W.
- Maksimali srovė 15 A.
- Maitinimo įtampa 5 - 36 V DC.
- PWM valdymo signalo amplitudė 2 - 20 V.
- Maksimalus PWM valdymo signalo dažnis – 20 KHz.
- PCB matmenys: 34 x 17 mm.

Šis PWM valdiklis neleidžia keisti sukimosi krypties, bet to ir nereikia šiame projekte.

3.14. Servo variklis

Servo variklis yra specializuotas elektrinis variklis, skirtas tiksliai valdyti kampinę padėtį [19]. Šiame projekte yra naudojamas SG90 modelis, kuris matomas 3.5 paveiksle.



3.5 pav. Servo variklis

Specifikacija:

- Darbinė įtampa: 4.8 V - 6.0 V
- Sukimo momentas: 1.6 kg/cm, kai 4.8 V
- Laido žymėjimas: raudona (+), ruda (-), geltona (signalas).
- PWM naudojamas pozicijos valdymui. PWM signalo ilgis yra nuo 1 ms (0°) iki 2 ms (180°), kai PWM periodas yra 20 ms.

Servo varikliai veikia naudodami PWM valdymo signalą nustatantį norimai veleno padėčiai. Potenciometras, prijungtas prie variklio veleno, matuoja esamą padėtį ir siunčia šią informaciją valdikliui. Valdiklis palygina gautą padėties informaciją su norimu gauti signalu ir, jei yra skirtumas, siunčia atitinkamą srovę varikliui, kad koreguotų jo padėtį. Variklis sukasi, kol pasiekia norimą padėtį, taip užtikrinamas tikslus valdymas.

3.15. Ličio jonų akumuliatorius ir apsaugos plokštė

Ličio jonų akumuliatorius 18650 (3.6 pav.) yra cilindrinės formos įkraunama baterija, kuri plačiai naudojama įvairiuose elektronikos įrenginiuose, tokiuose kaip nešiojamieji kompiuteriai, žibintuvėliai, elektriniai įrankiai ir taip toliau. Pavadinimas "18650" reiškia baterijos matmenis: 18 mm skersmens ir 65 mm ilgio. Tipinė vardinė įtampa yra 3.6 – 3.7 voltai. Baterijos talpa paprastai svyruoja nuo 2000 iki 3500 mAh (miliampervalandžių), priklausomai nuo gamintojo ir modelio. Šios baterijos turi aukštą energijos tankį ir gali būti įkraunamos bei iškraunamos daug kartų, tačiau joms būtina atitinkama apsauga nuo didelių įkrovimo ir iškrovimo srovių.

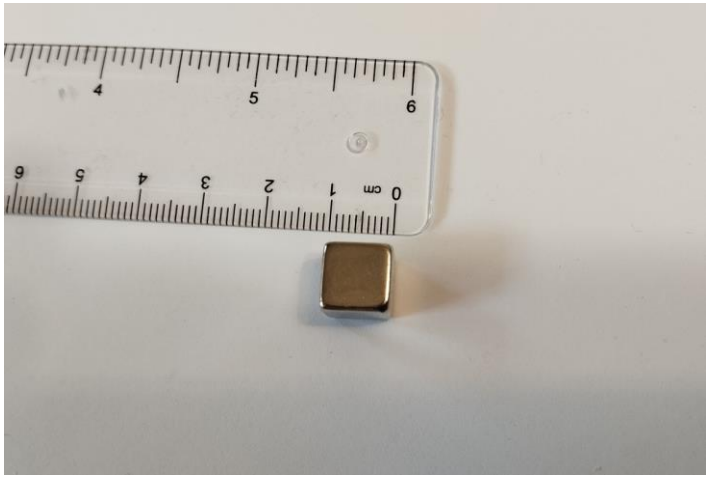


3.6 pav. Ličio jonų akumuliatorius ir apsaugos plokštė

Šiems akumuliatoriams naudojamas baterijų įkrovimo modulis su apsauga, kuris skirtas keturiems akumuliatoriams nuosekliai (3.6 pav.). Šis modulis neturi įtampos balansavimo gebėjimo, tai po kiekvieno įkrovimo iškrovimo ciklo pradės skirtis individualių akumuliatorių įtampos. Yra įtampos viršijimo, per didelio įkrovimo ir iškrovimo ir trumpojo jungimo apsauga.

3.16. Magnetai

Magnetinio sukibimo mechanizmui naudojami 10mm x 10mm x 10mm N42 neodimio magnetai (3.9 pav.).



3.9 pav. Neodimio magnetas

Specifikacijos:

- Medžiaga: NdFeB.
- Išmatavimai: 10x10x10 mm.
- Padengimas: Ni-Cu-Ni.
- Įmagnetinimas: N42 per storį 10mm.
- Stiprumas: apie 7 kg.
- Maks. darbinė temp.: 80°C.
- Svoris: 7.4 g.

Su internetiniu skaičiuotuvu [20] buvo apskaičiuota, kad magnetų trauka 10 mm atstume yra apie 2 kg. Vieneto kaina yra 0.80 eurų.

3.17. Kitos dalys

Šiame skyriuje yra aprašomos dalys, kurios nebuvo aprašytos praeituose skyriuose.

- Tinklo kabelis „Patch cord category 5e flooded shell“, kuris yra 50 metrų ilgio.
- „Esperanza ENA102“ adapteris RJ-45 - USB-C.
- F688ZZ guolis su flanšu 8x16x5mm.
- Guminiam nardymo diržui skirti švino svoriai, trys po 1kg.
- Maketavimo plokštės. Šios plokštes skirtos lengvai sujungti elektronikos komponentus jų nelituojant.
- Jungės laidai. Šie laidai specifiškai pritaikyti maketavimo plokštėms.
- Rezistoriai. Tai pasyvūs elektronikos komponentai skirti apriboti elektros srovei.
- Kondensatoriai. Tai pasyvūs elektronikos komponentai skirti kaupti energijai, filtravimui, signalų apdorojimui ir įvairių elektroninių grandinių stabilizavimui.
- M5 varžtai (14 mm), poveržlės ir veržlės.
- 5 l stiklainis.
- Įkroviklio kištukas. Skirtas nešiojamo kompiuterio įkrovikliui.

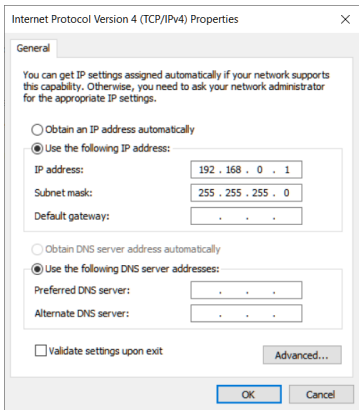
Šių detalių nuotraukos yra 1 PRIEDE.

4. ĮGYVENDINIMO DALIS

Šiame skyriuje aprašomas vaizdo perdavimas, signalų perdavimas, elektronikos ir roboto surinkimas ir roboto testas ežere.

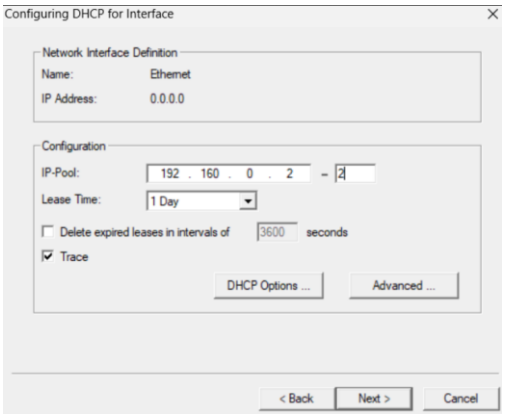
4.1. Duomenų perdavimo dalis

Vaizdo siuntimui ir valdymo signalų perdavimu yra naudojamas nešiojamas kompiuteris, kuris yra prijungtas prie mobilaus įrenginio su tinklo kabeliu ir RJ-45 – USB-C adapteriu (4.3 pav). Kad ryšys būtų užmegztas tarp kompiuterio ir mobilaus įrenginio, kompiuteryje buvo pakeista eterneto plokštės konfigūracija – (TCP/IPv4) sąrankoje nustatytas IP adresas į 192.168.0.1 (4.1 pav.).



4.1 pav. Eterneto konfigūracija

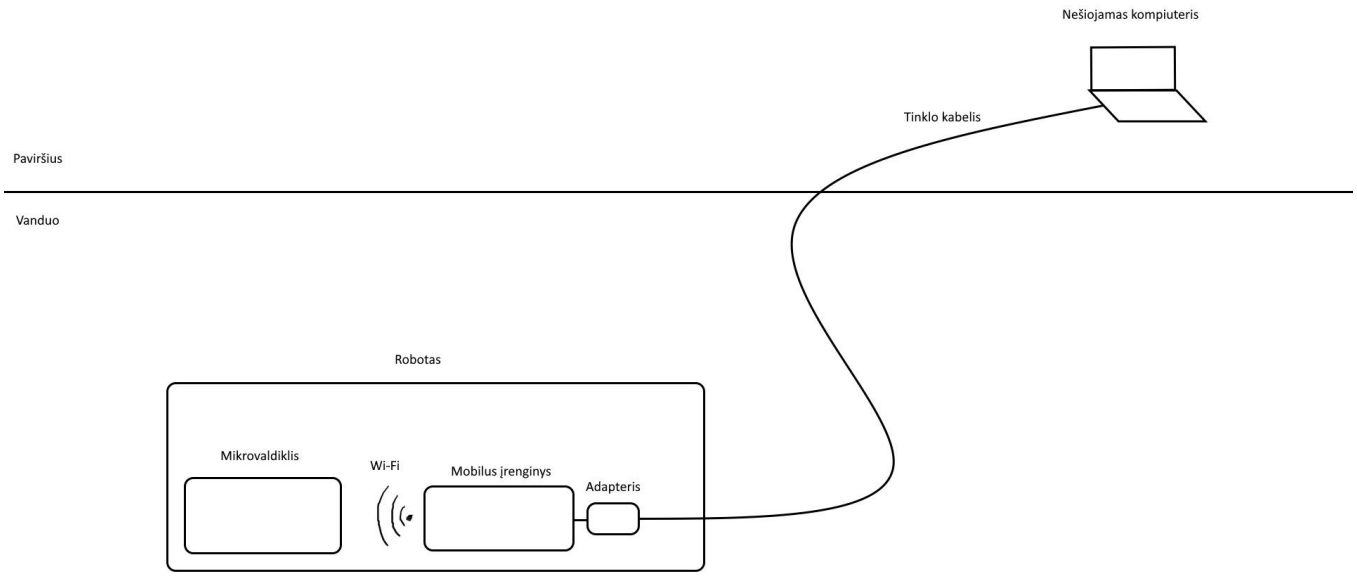
Kadangi mobiliajame įrenginyje negalima keisti tinklo konfigūracijos, tai kompiuteryje buvo įjungtas DHCP serveris (3.3 skyrius), kuris sukonfigūruotas suteikti vieną IP adresą, kuris yra 192.168.0.2 (4.2 pav.).



4.2 pav. DHCP serverio konfigūracija

Atlikus šiuos veiksmus buvo sudarytas vietinis tinklas tarp kompiuterio ir mobilaus įrenginio.

Toliau, turi būti galimybė siusti signalus iš kompiuterio į mikrovaldiklį. Kadangi mikrovaldiklis neturi galimybės tiesiogiai komunikuoti su kompiuteriu (2.4 skyrius), tai mobilusis įrenginys turi veikti kaip tarpininkas. Tai atliekama mobiliajame įrenginyje įjungiant interneto prieigos tašką, prie kurio mikrovaldiklis prisijungia pasinaudodamas Wi-Fi ryšiu. Tai atlikus yra du tinklai, vienas tarp kompiuterio ir mobilaus įrenginio, o kitas tarp mobilaus įrenginio ir mikrovaldiklio (4.3 pav.).



4.3 pav. Tinklo sujungimo pavyzdys

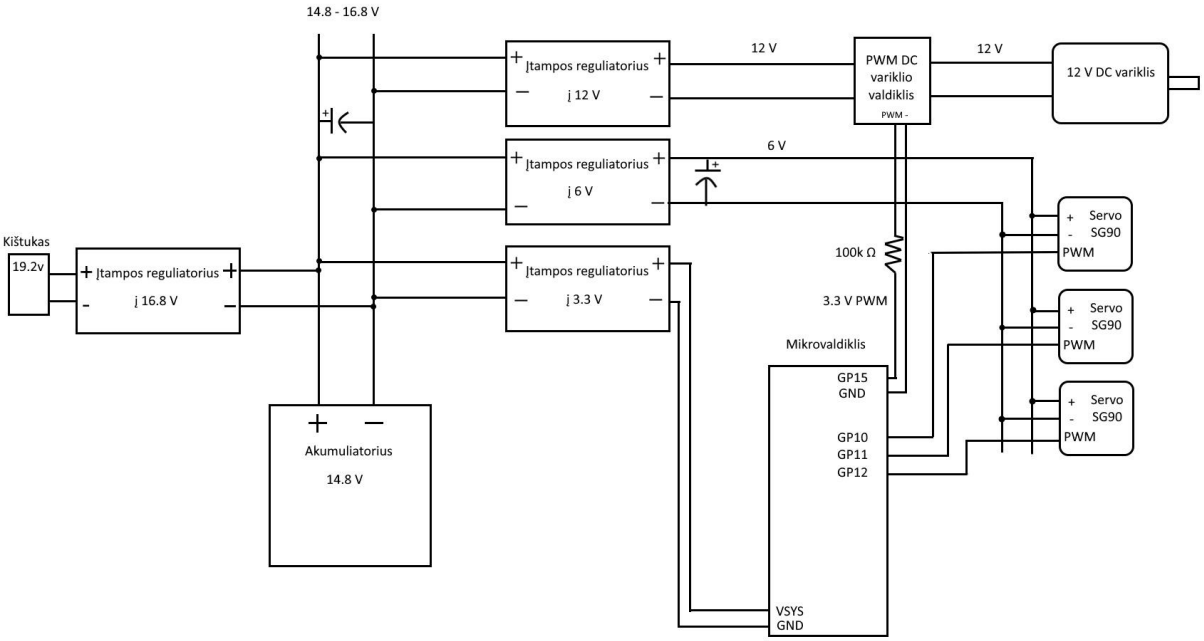
Sudaryti tinklų tiltui mobiliajame įrenginyje nėra galimybės, tai naudojama mobili programėlė (4.3.4 skyrius), kuri gauna roboto valdymo signalus iš kompiuterio ir juos perduoda į mikrovaldiklį, taip sudaromas ryšys tarp kompiuterio ir mikrovaldiklio.

4.2. Elektronikos surinkimo dalis

Elektronikos komponentų paskirtis yra valdyti variklius pagal kompiuterio siunčiamus signalus. Šioje diagramoje (4.4 pav.) yra pavaizduota, kaip sujungti laidai ir komponentai. Diagramoje matoma, kad yra naudojamos keturios skirtingos įtampos. Kairėje į kištuką jungiasi įkroviklis (3.17 skyrius), šio įkroviklio įtampa yra sumažinama į 16.8 V, kuri yra saugi akumuliatoriaus įkrovimui. Toliau, akumuliatorius tiekia elektros srove, kurios įtampa yra tarp 14.8 V ir 16.8 V priklausomai nuo pakrovimo lygio, trims įtampos reguliatoriams:

- Pirmasis reguliatorius teikia 12 V įtampos srovę DC varikliui.
- Antrasis – teikia 6 V įtampos srovę servo varikliams.
- Trečiasis – teikia 3.3 V įtampos srovę mikrovaldikliui.

Grandinėje yra pastatyti kondensatoriai, jie yra skirti išlyginti elektros šuoliams, kurie atsiranda jeigu staigiai sustoja varikliai. Taip pat yra varža skirta apsaugoti mikrovaldiklį nuo srovės šuolių.



4.4 pav. Elektrinė principinė schema

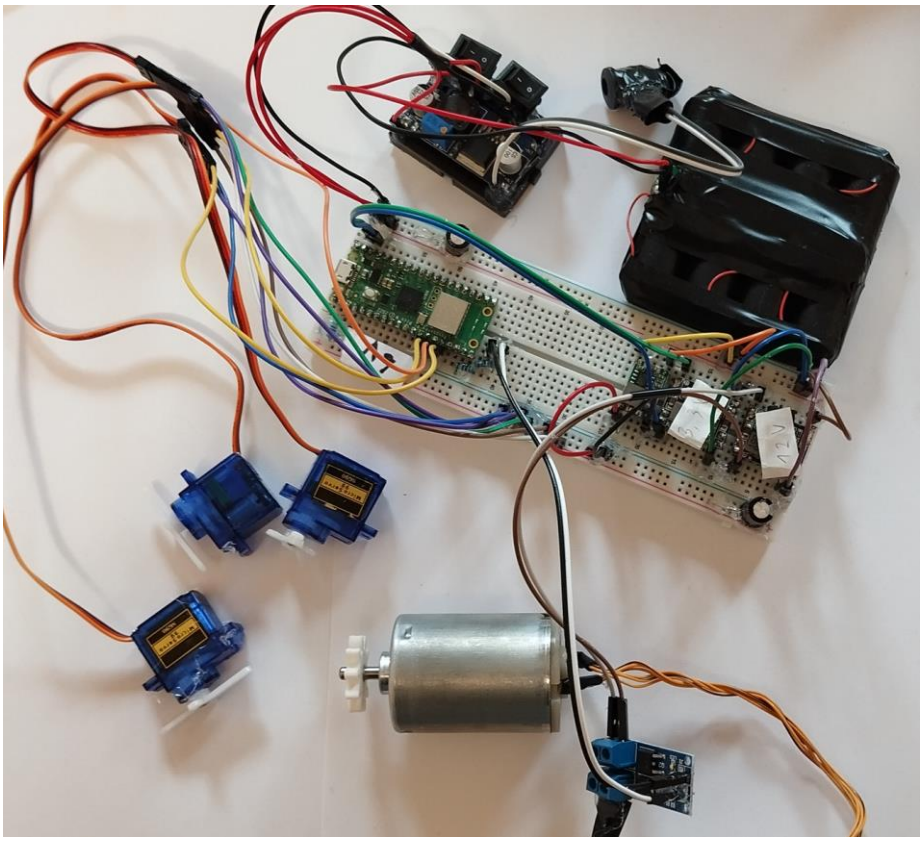
Mikrovaldiklis ir įtampos reguliatoriai buvo įstatyti į maketavimo plokštę. Šie komponentai neturėjo kontaktų skirtų maketavimo plokštei, tai jie buvo prilituoti. Toliau, su jungės laidais (angl. jumper wire) buvo sujungti komponentai, taip kaip pavaizduota diagramoje (4.4 pav.). Šie laidai ir komponentai buvo priklijuoti su karštų klijų pistoletu prie maketavimo plokštės, kad neatsijungtų nuo fizinių trikdžių.

Kitoje, mažesnėje maketavimo plokštėje, yra prijungtas kištukas, du jungikliai ir įtampos reguliatorius į 16.8 V. Ši plokštė skirta roboto įkrovimui, įjungimui ar išjungimui. Vienas jungiklis atjungia mikrovaldiklio plokštę nuo grandinės, kitas atjungia akumuliatorių nuo grandinės. Tai daroma, kad akumuliatoriaus krovimo metu mikrovaldiklis nebūtu įjungtas.

Toliau, prie DC variklio buvo prilituoti laidai, kurie prijungti prie PWM valdiklio. PWM valdiklis ir servo varikliai yra sujungti taip, kaip pavaizduota diagramoje.

Ličio jonų baterijos yra sujungtos nuosekliai ir prilituotos prie apsaugos plokštės.

Galutinai surinkti elektros komponentai matomi 4.5 paveiksle.



4.5 pav. Elektronikos diagrama

Laidų ilgiai nuo plokštės iki variklių yra apie 30 cm.

4.3. Programų kūrimo dalis

Programinė įranga susideda iš šių dalių:

- Serveris, kuris priglobia (angl. hosting) naršyklės programėles ir valdo duomenų judėjimą.
- Kompiuteriui skirta valdymo pulto naršyklės programėlė.
- Mobiliam įrenginiui skirta vaizdo siuntimo naršyklės programėlė.
- Mobiliam įrenginiui skirta tinklo tilto programėlė.
- Mikrovaldiklio programa.

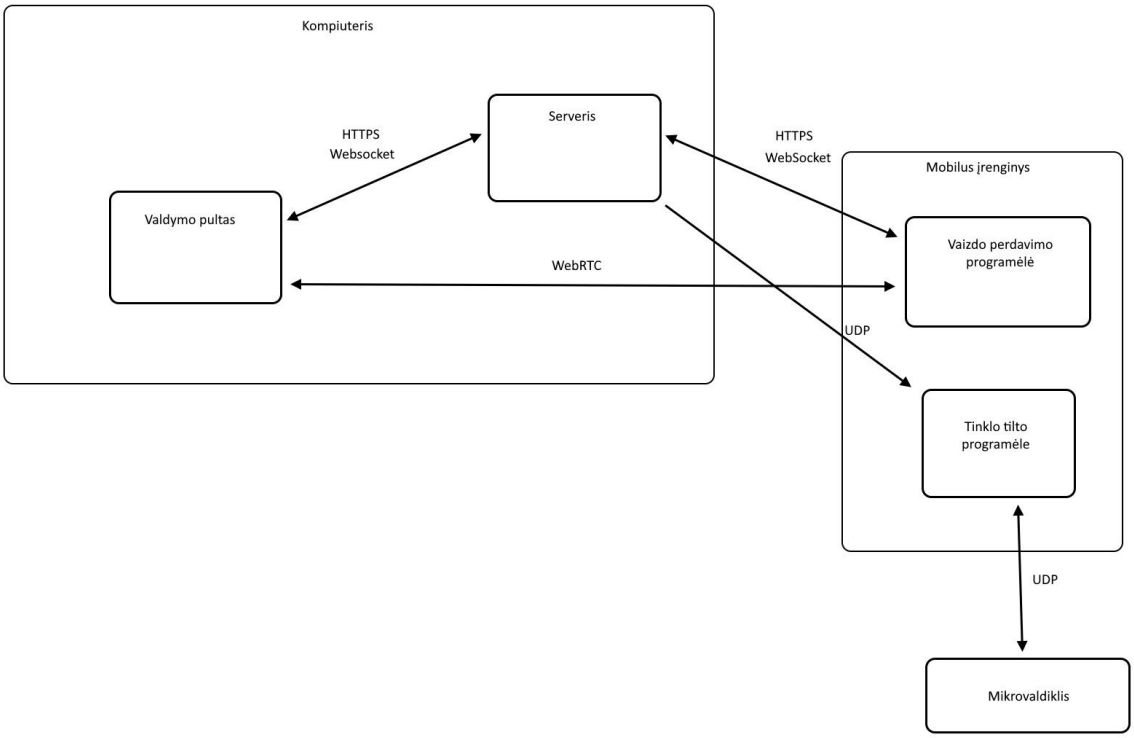
Šių programų veikimui pavaizduoti buvo sukurtos srauto diagramos (angl. flowchart).

4.3.1. Pagrindinis serveris

Pagrindinis serveris veikia kompiuteryje, Node.js aplinkoje. Susideda iš keturių dalių:

- HTTPS serverio.
- WebSocket serverio.
- UDP serverio.
- Ryšio užtikrinimo funkcijos.

Kaip duomenys yra perduodami sistemoje yra pavaizduota 4.6 paveiksle.



4.6 pav. Ryšių diagrama

Ryšio užtikrinimo funkcija terminale įvykdo `ping 192.168.0.2` komandą ir pagal atsakymą sužinoma ar mobilus įrenginys yra sėkmingai prijungtas prie kompiuterio.

HTTPS protokolas yra naudojamas, nes naršyklės neleidžia naudotis kameromis, kai yra HTTP ryšys. HTTPS serverio sukūrimui reikėjo sugeneruoti privatų raktą ir sertifikatą. Tai buvo atlikta pasinaudojant OpenSSL terminalo įrankiu.

```
openssl req -x509 -newkey rsa:4096 -keyout key.pem -out cert.pem -days 3650 -nodes
```

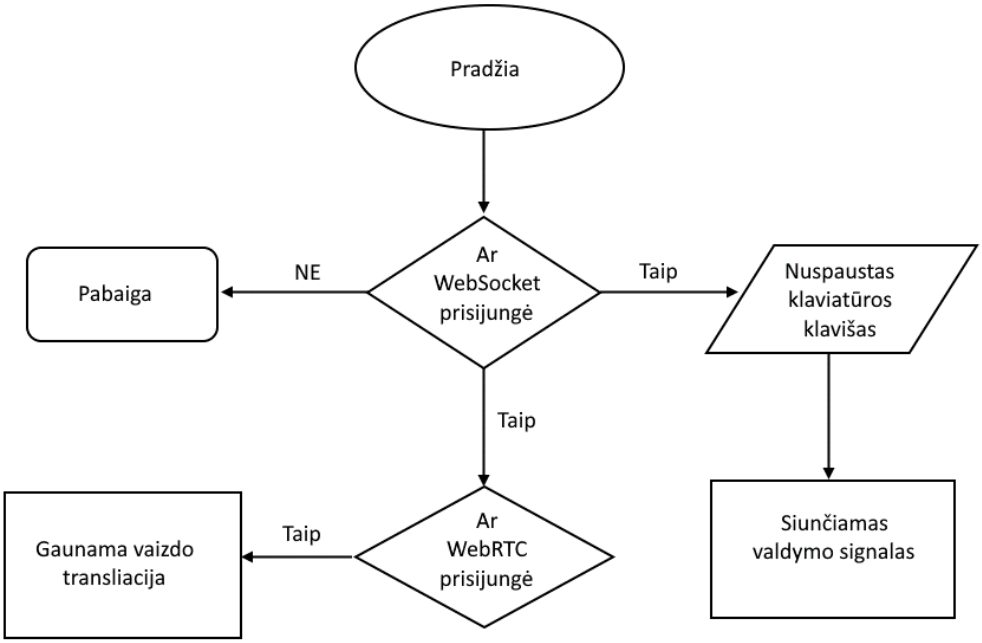
HTTPS serveriui yra naudojamas „express“ įskiepis, kuris supaprastina failų prieglobą. Šis serveris leidžia naršyklėse atverti programėles skirtas roboto valdymui (4.3.2 skyrius) su https://192.168.0.1:3000 URL adresu.

WebSocket serveris naudoja tą patį sertifikatą, nes modernios naršyklės draudžia prisijungti iš saugaus tinklalapio į nesaugų serverį. Šis serveris skirtas užmegzti WebRTC ryšiui tarp valdymo pulto programėles kompiuteryje ir vaizdo siuntimo programėles mobiliajame įrenginyje. Taip pat, šis serveris skirtas perduoti valdymo signalams iš pulto programėlės į UDP serverį, nes naršyklės neturi galimybės siųsti UDP žinučių.

UDP serveris gauna JSON žinutes iš WebSocket serverio ir persiunčia UDP žinutes į tilto programėle mobiliajame įrenginyje. Tam naudoja šį IP adresą: 192.168.0.2:12345

4.3.2. Valdymo pulto programėlė

Ši programėlė veikia kompiuterio naršyklėje, ji iš karto bando prisijungti prie WebSocket serverio. Jei sėkmingai prisijungia, tada laukia, kol serveris atsius WebRTC SDP ir ICE žinutes, su kuriomis naršyklė galės užmegzti P2P ryšį ir gauti vaizdo transliaciją iš mobilaus įrenginio.



4.7 pav. Valdymo pulto programos srauto diagrama

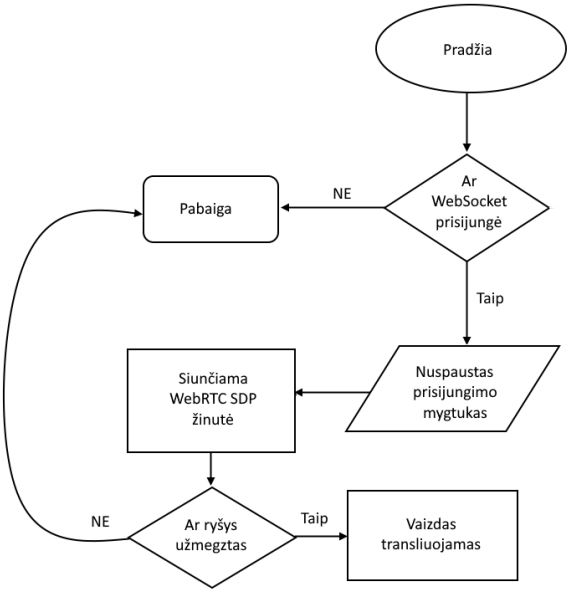
Taip pat, nuspaudus tam tikrus klaviatūros klavišus išsiunčiami valdymo signalai į WebSocket serverį:

„QWASDE“ – šie mygtukai išsiunčia tokią žinutę: „s {A} {B} {C}“, kur A, B ir C gali būti skaičiai tarp 3700 ir 6300. Pvz.: „s 4950 4950 4950“. Šie skaičiai nusako kiekvieno servo variklio pasisukimo kampą.

„Tarpas“ – nuspaudus šį klavišą yra išsiunčiama „on“ žinutė – ji nusako, kad DC variklis turėtų suktis. Atleidus šį klavišą išsiunčiama „off“ žinutė, kuri išjungia variklį.

4.3.3. Vaizdo perdavimo programėlė

Ši programėlė veikia mobilaus įrenginio naršyklėje. Joje matomas vienas mygtukas, kurį nuspaudus bandoma užmegzti WebRTC ryšį (4.8 pav.).

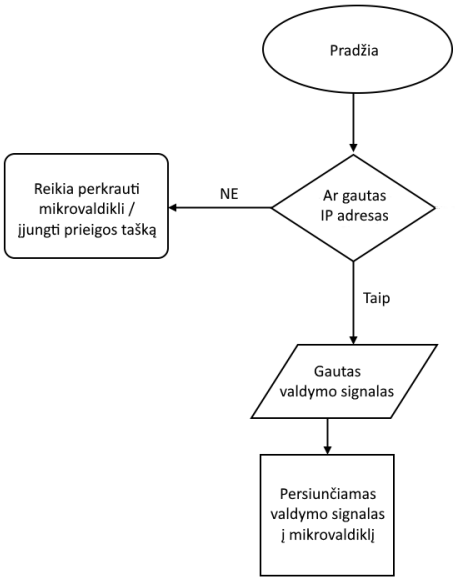


4.8 pav. Vaizdo perdavimo programos schema

Ryšiui užmegzti yra išsiunčiama SDP žinutė į pulto programėlę naudojant WebSocket serverį. Šią žinutę sugeneruoja RTCPeerConnection API. Pulto programėlė gavusi šią žinutę išsiunčia atsakymą atgal. Taip abu klientai žino kokius duomenis vienas kitam nori perduoti pro P2P ryšį, šiuo atvejų vienpusis vaizdo siuntimas iš mobilaus įrenginio į kompiuterį. Tada abu klientai generuoja ICE kandidatus (3.1 skyrius), kuriuos vienas kitam nusiunčia. Galiausiai, abu klientai išsirenka geriausią ICE kandidatą ir pradedamas P2P ryšis.

4.3.4. Tinklo tilto programėlė

Ši programėlė yra sukurta su Google Flutter karkasu, ji veikia mobiliajame įrenginyje, jos tikslas yra perduoti UDP valdymo signalus iš kompiuterio į mikrovaldiklį. Pradžioje ši programa gauna mikrovaldiklio IP adresą. Mikrovaldiklis prisijungęs prie prieigos taško gauna IP adresą, kurį išsiunčia į mobilųjį įrenginį su UDP žinute. Ši programėlė taip gauna mikrovaldiklio IP adresą. Toliau, visos ateinančios UDP žinutės yra persiunčiamos į mikrovaldiklį (4.9 pav.).

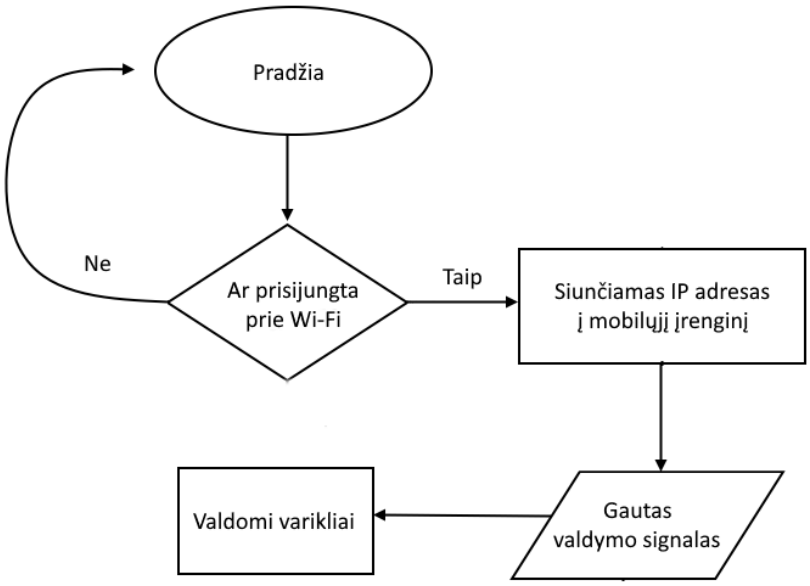


4.9 pav. Tinklo tilto programos schema

Pradiniame plane ši programėlė nebuvo numatyta, vaizdo perdavimo programėlė turėjo galimybę perduoti valdymo signalus naudojant HTTP protokolą. Bet Raspberry PI Pico W yra per lėtas, kad galėtų realiu laiku apdoroti šias užklausas. Geriausiu atvejų – vieną užklausą per 0.5 s. Bet HTTP naudoja TCP protokolą, tai reiškia, kad jei susiduriama su ryšio problemomis, tai užsiblokuoja ryšis ir užtrunka kol atsiblokuoja.

4.3.5. Mikrovaldiklio programėlė

Ši programėlė veikia mikrovaldiklyje, sukurta su MicroPython, jos tikslas yra interpretuoti valdymo signalus į variklių judesius. Pradžioje programos bandoma prisijungti prie mobilaus įrenginio prieigos taško. Kai mikrovaldiklis prisijungia prie Wi-Fi, tada išsiunčia savo IP adresą atgal į mobilųjį įrenginį. Programa toliau laukia UDP žinučių, kai gaunamas valdymo signalas jis yra interpretuojamas į variklių judesius (4.10 pav.).



4.10 pav. Mikrovaldiklio programos schema

Valdymo signalai yra trys: „on“, „off“ ir „s {uint16} {uint16} {uint16}“.

„on“ ir „off“ signalai valdo DC variklio valdiklį, kuris valdo DC variklį.

„s {uint16} {uint16} {uint16}“, „s“ reiškia servo, toliau trys skaičiai nusako PWM signalo darbo ciklą. Mikrovaldiklis PWM signalui naudoja 16 bitų skaičių (0 - 65535). Bandymais nustatyta, kad minimalus PWM darbo ciklas į kurį servo variklis reaguoja yra 1140, maks. – 8620. Per vidurį – 4950.

Pastebėta, kad SG90 servo varikliai gali daugiau pasisukti negu 180°, tai bandymais nustatyta, kad 0° darbo ciklas yra 1700, 180° – 8260.

4.4. Korpuso surinkimo dalis

Šiame skyriuje yra aprašomas 3D modeliavimo procesas ir roboto surinkimo procesas.

4.4.1. Kabelio pravedimas

Šiame projekte yra naudojamas 5 l stiklainis, tinklo kabelis buvo pravestas pro stiklainio dangtį. Tai atlikti buvo išpjauta „X“ formos skylę dangčio viduryje, pro ją pravestas 15 cm kabelio ilgis ir užsandarinta skylė su klijų sluoksniais. Vidinis sluoksnis buvo užsandarinamas su momentiniais klijais, išorinis su karštais klijais. Taip pat, vietoje, kurioje kabelis kontaktuoja su dangčiu, buvo įšvirkščiami momentiniai klijai į kabelio vidų. Tai atlikus buvo sudarytas sandarus laido pravedimas (4.11 pav.).



4.11 pav. Kabelio pravedimas pro dangtį

Buvo naudojami du sluoksniai klijų, nes, kol džiūsta momentiniai klijai, karšti klijai išlaiko klijuojamas dalis toje pačioje vietoje.

4.4.2. Detalių modeliavimas

Pirma, buvo išmatuotai stiklainio matmenys:

- Vidinis skersmuo apie 16.5 cm.
- Kaklelio vidinis skersmuo apie 8 cm.

Toliau sugeneruoti krumpliaračiai pasinaudojant „STLGears“ įrankiu (3.10 skyrius). Šie krumpliaračiai skirti perduoti variklio sukimuisi į magnetų sukibimo mechanizmą. Didesnio krumpliaračio specifikacija:

- Dantų skaičius – 40.
- Modulis – 2 mm.
- Aukštis – 4 mm.
- Kampas – 20°.
- Centrinės skylės spindulys – 8 mm.

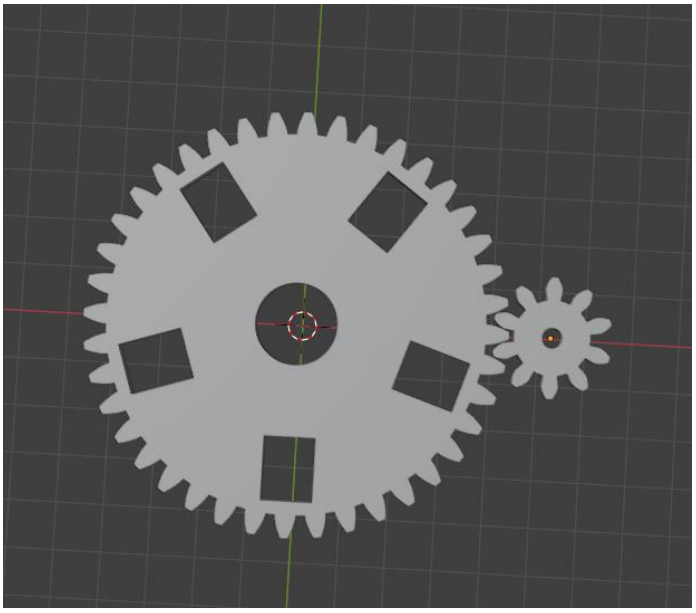
Mažesnio krumpliaračio specifikacija:

- Dantų skaičius – 10.
- Modulis – 2 mm.
- Aukštis – 4 mm.
- Kampas – 20°.
- Centrinės skylės spindulys – 1.75 mm.

Taip sudaromas 4:1 santykis tarp magnetų mechanizmo ir variklio. DC Varikliai dirba efektyviau, kai jie greitai sukasi.

Propelerio generavimas buvo atliktas su „Ultimate Toroidal Propeller Generator“ įrankiu, jo konfigūracija yra 2 PRIEDE.

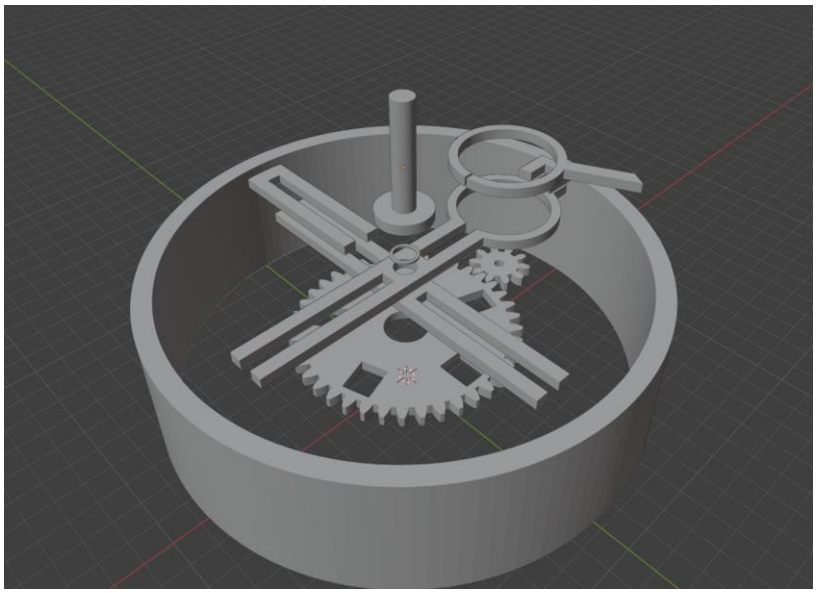
Toliau, didesnis krumpliaratis buvo modifikuotas Blender programoje (4.12 pav.), kad turėtų vietas į kurias galima įstatyti magnetus. Buvo išpjauti penki 10 mm pločio tarpai aplink sukimosi ašį.



4.12 pav. Krumpliaračiai

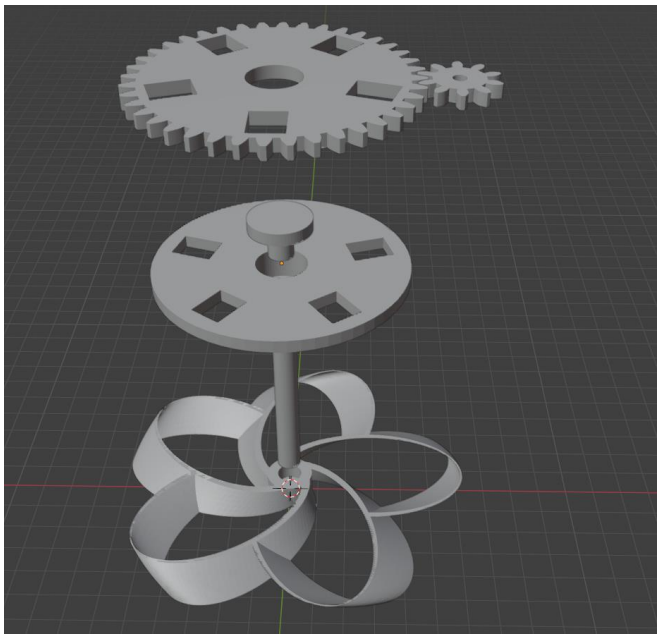
Toliau kuriami komponentai skirti išlaikyti šį mechanizmą tinkamoje vietoje.

Blender programoje buvo sukurta ašis su pagrindu, ašies skersmuo yra 8 mm, skirtas guoliui. Toliau sukurti įtvarai skirti įtvirtinti ašį prie stiklo, šie įtvarai susideda iš dviejų dalių: „H“ formos įtvaro ir „U“ formos galūnės, jie turi 5 mm tarpą per vidurį, kad juos butų galima suveržti su M5 tipo varžtais. Toliau yra ašies įtvaras, su varikliui skirtu laikikliu (matoma virš mažojo krumpliaračio). Aukščiau jo yra varikliui skirtas įtvaras. Galiausiai, sukurta 1 mm aukščio tarpinė, skirta ašiai. Visa tai matoma 4.13 pav.



4.13 pav. Visų vidinių detalių modeliai

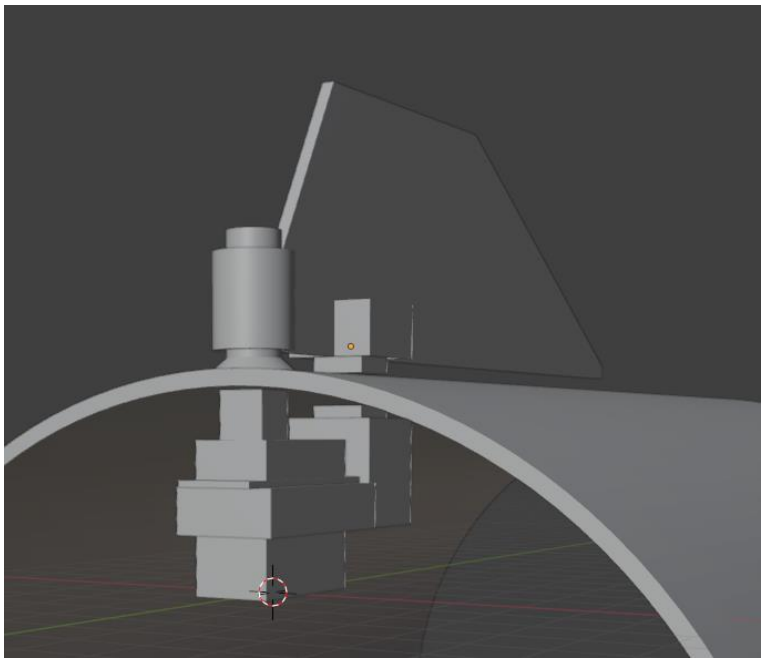
Toliau buvo sukurtos išorinės dalys skirtos sukti propeleriui naudojant magnetinį sukibimą. Matoma magnetams skirta platforma su ašimi, prie kurios gali prisijungti sugeneruotas propeleris. Į centrinį tarpa įsistato guolis. Magnetų tarpai yra tokie patys kaip ir didžiajame krumpliaratyje (4.14 pav.).



4.14 pav. Visų išorinių detalių modeliai

Virš magnetų platformos yra neilga ašis su pagrindu, skirta guoliui.

Toliau, buvo sumodeliuoti pelekai, jie skirti robotui suteikti gebėjimą manevruoti vandenyje. Susideda iš trijų detalių: peleko, peleko ašies ir magneto laikiklio. Pelekas turi 10 mm pločio dėžutę, kuri skirta neodimio magnetui. Pelekas įsistato ant ašies, kuri yra pastatoma tiesiai virš servo variklio sukimosi ašies. Toliau, servo varikliui sukurtas magneto laikiklis, kuriame magnetas įsistato tiesiai po peleko magnetu (4.15 pav.).



4.15 pav. Peleko modelis

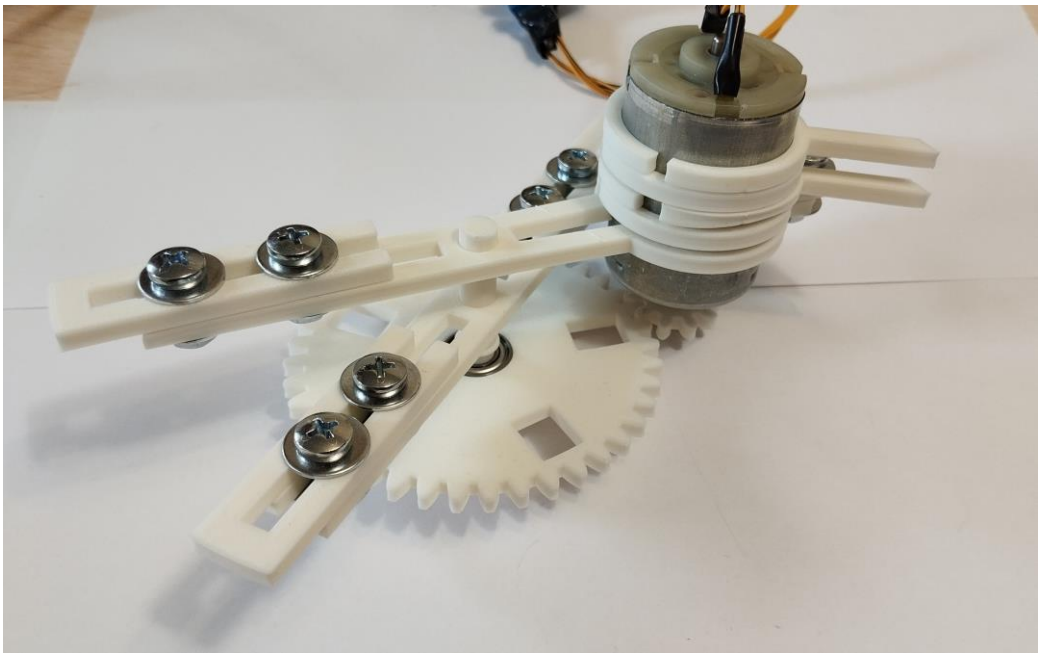
Kadangi stiklas yra išlenktas, tai magnetų laikikliai sukurti taip, kad magnetai turėtų gebėjimą laisvai judėti vertikalia ašimi. Taip magnetai visada bus prigludę prie stiklo ir trauks maksimalia trauka. Visos šios detalės buvo atspausdintos su 3D spausdintuvu. Buvo atspausdinti šie detalių kiekiai:

- Po tris pelekus ir jų priedus.
- Vienas propeleris, magnetų platformą ir magnetų platformos ašis.
- Vienas didelis ir mažas krumpliaratis.
- Viena vidinė ašis.
- Vienas „H“ įtvaras.
- Trys „U“ jungtys.
- Vienas „H“ variklio įtvaras.
- Keturi „Q“ formos variklio įtvarai.
- Dešimt 1 mm ašies tarpinių.

Modeliai yra pasiekiami programinio kodo kataloge. Toliau, aprašomas detalių surinkimo procesas.

4.4.3. Detalių surinkimas

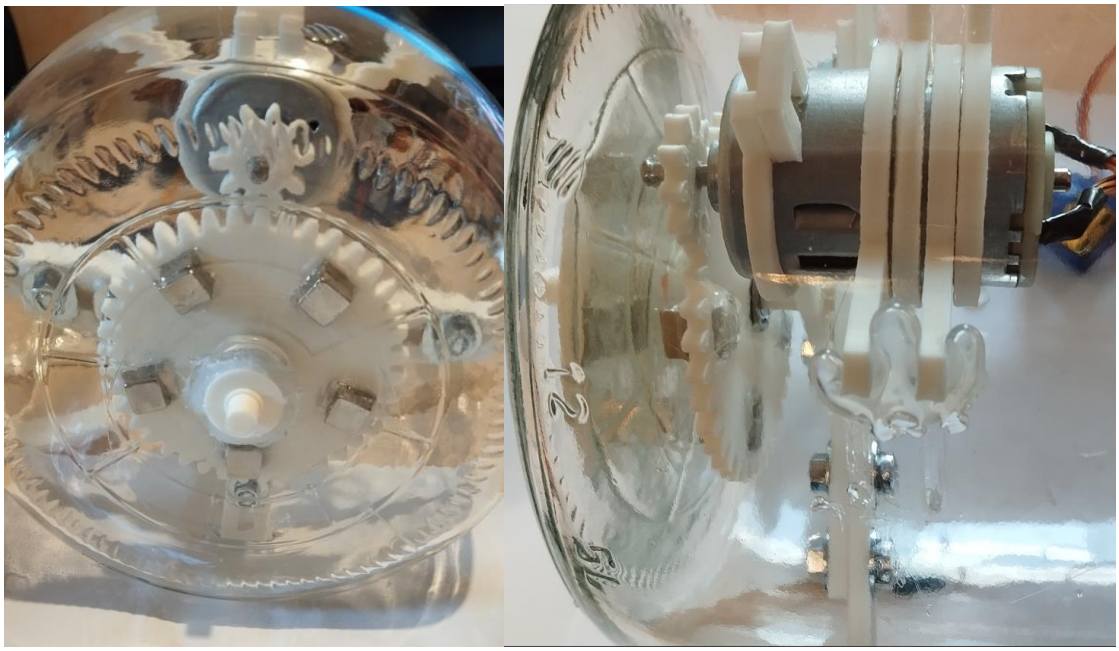
Atspausdintos detalės buvo surinktos (4.16 pav.) patikrinti suderinamumui – viskas tvarkingai susidėliojo. Į didįjį krumpliaratį buvo įstatytas guolis ir krumpliaratis užmautas ant ašies. Toliau, prie „H“ įtvarų buvo priveržtos „U“ jungtys, šis dizainas leidžia reguliuoti įtvaro ilgį, tai daroma, nes stiklainio vidaus išmatavimai nėra vientisi. Ant variklio ašies buvo užmautas mažasis krumpliaratis. Toliau, ant variklio užmauti „Q“ laikikliai. Galiausiai, variklis įstatomas į „H“ įtvarą ir šis įtvaras užmaunamas ant ašies.



4.16 pav. Visos vidinės detalės

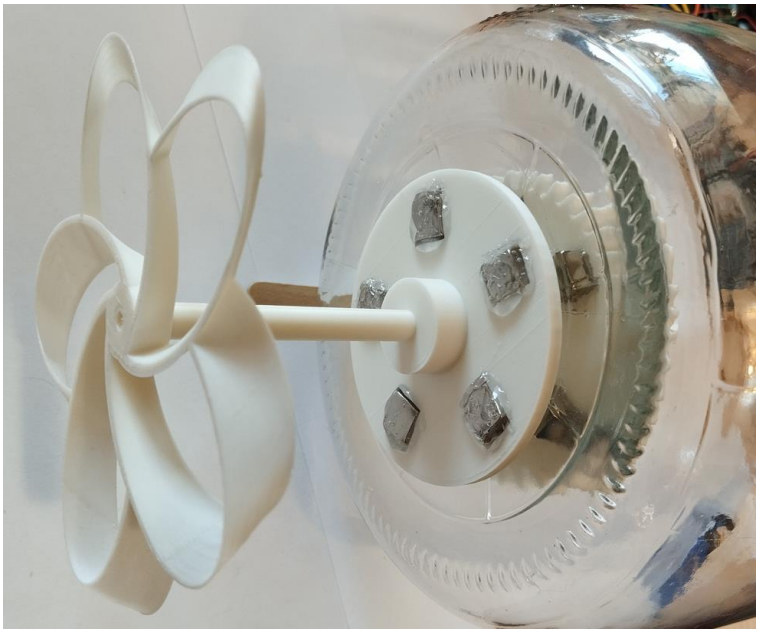
Toliau šis mechanizmas buvo išardytas ir perstatytas stiklainio viduje (4.17 pav.).

Pirmiausia, ašis buvo priklijuota stiklainio pagrindo centre su momentiniais ir kaštais klijais. Toliau, ant ašies buvo užmautas didysis krumpliaratis, į jį sudėti neodimio magnetai taip, kad liestu stiklą, pasirūpinta, kad vienodas magnetų polis būtų nukreiptas į stiklą, šie magnetai buvo priklijuoti su karštais klijais. Toliau, krumpliaratis buvo laikinai išimtas, ant ašies užmauta viena 1 mm tarpinė, tada vėl užmautas krumpliaratis. Taip sudaromas 1 mm tarpas tarp magnetų ir stiklo ir taip pašalinama trintis. Toliau buvo sudėti įtvarai ir variklis, įtvarai buvo priklijuoti prie stiklo su karštais klijais. Rezultatas matomas 4.17 pav.



4.17 pav. Vidinės detalės stiklainyje

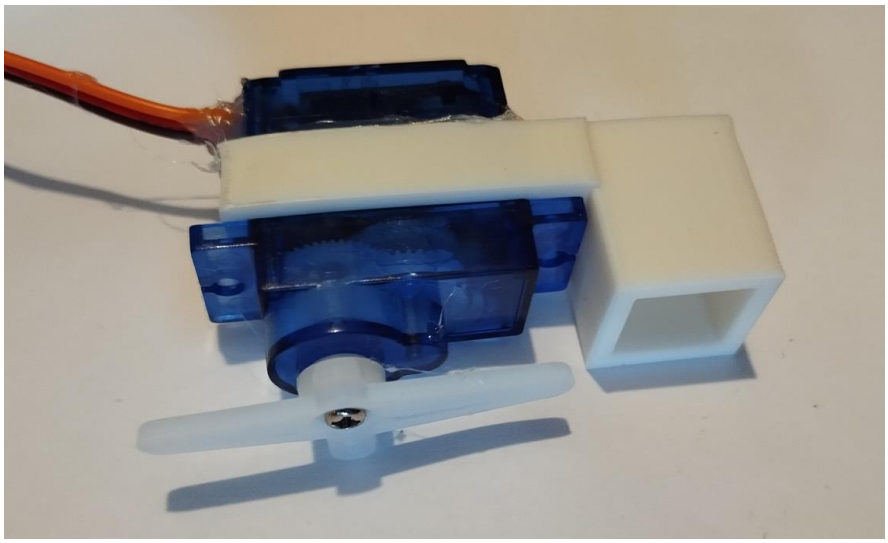
Toks pat procesas buvo atliekamas išorinėje dalyje (4.18 pav.). Priklijuota ašis prie stiklainio pagrindo, uždėtas guolis, priklijuoti magnetai (poliai nukreipti, kad trauktų magnetus) ir priklijuotas propeleris, tada užmauta 1 mm tarpine sudaryti tarpui tarp magnetų ir stiklo.



4.18 pav. Išorinės detalės

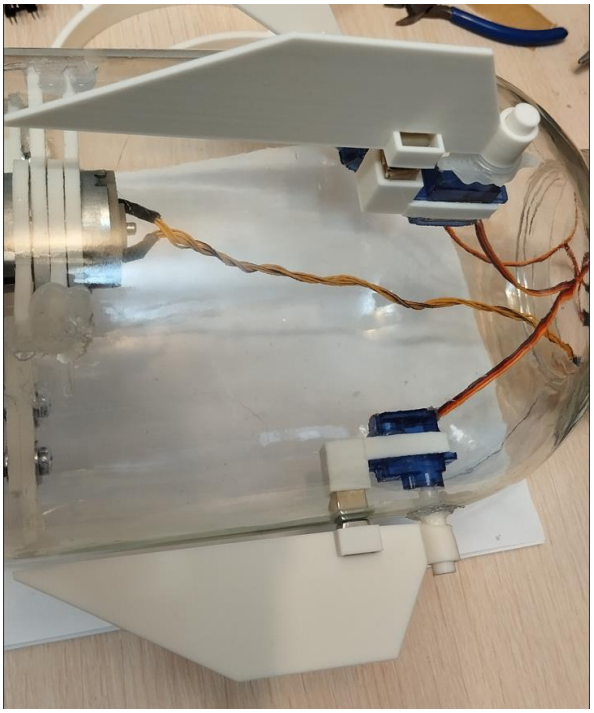
Tai atlikus buvo užbaigtas magnetinio sukibimo mechanizmas.

Toliau, aprašomas pelekų mechanizmo sustatymas. Buvo atspausdinti trys pelekai, ašys ir laikikliai. Magneto laikikliai buvo užmauti ant servo variklių taip kaip pavaizduota 4.19 pav.



4.19 pav. Magneto laikiklis ant servo variklio

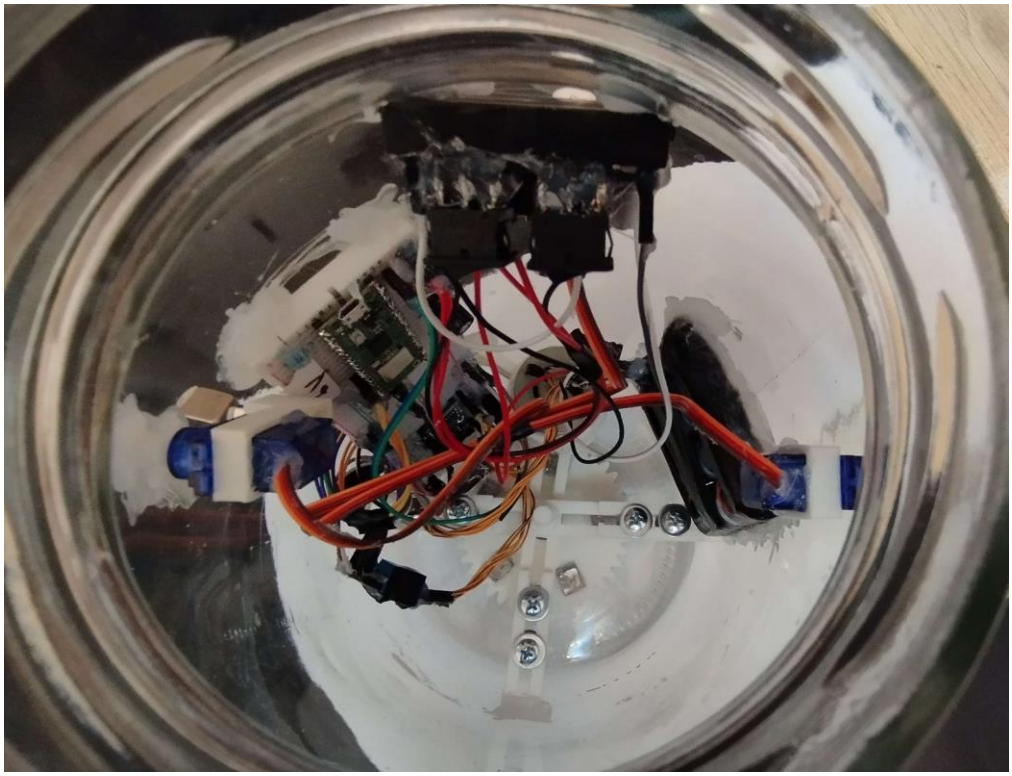
Kitame žingsnyje, šie varikliai buvo priklijuoti stiklainio kairėje, dešinėje ir viršuje (4.20 pav.). Išorinėje dalyje ašys buvo priklijuotos su momentiniais klijais virš variklio sukimosi centro. Ant šių ašių buvo užmauti pelekai ir buvo įstatyti magnetai į jiems skirtas vietas. Galiausiai, 1 mm tarpinės buvo užmautos ir priklijuotos ant ašių, kad pelekai neiškristu plaukiant.



4.20 pav. Peleko mechanizmai

Toliau, aprašoma kaip elektronikos komponentai buvo pritvirtinti prie korpuso.

Maketavimo plokštės ir akumuliatorius buvo priklijuoti viršutinėje korpuso dalyje su karštais klijais (4.21 pav.). Taip atlaisvinama vieta apačioje, kurioje dedasi mobilus įrenginys ir švino svoriai. Mažoji maketavimo plokštė su jungikliais buvo pastatyta arti kaklelio, kad jungikliai būtų lengvai pasiekiami.



4.21 pav. Elektronika esanti korpuse

Toliau, 4.22 paveiksle pavaizduojama kaip gali būti įstatomi švino svoriai.



4.22 pav. Mobilus įrenginys ir švino svoriai

Pastačius svorius apačioje roboto masės centras taip pat būna apačioje. Nuo to roboto posvyrio padėtis būna stabili vandenyje. Svoriai pritvirtinami su dvipuse lipnia juosta.

4.5. Roboto testavimo dalis

Šiame skyriuje aprašoma roboto paleidimo seka ir rezultatai. Iš rezultatų išvedamos išvados apie roboto efektyvumą.

Paleidimo eiga:

- Nešiojame kompiuteryje įjungiamas DHCP serveris.
- Prijungiamas mobilus įrenginys prie kompiuterio su tinklo kabeliu.
- Nešiojame kompiuteryje įjungiamas pagrindinis serveris „`node ./main.js`“ su administratoriaus teisėmis.
- Mobiliajame įrenginyje įdiegiama UDP tilto programėlė (jei jos nėra).
- Mobiliajame įrenginyje įjungiamas prieigos taškas.
- Mobiliajame įrenginyje įjungiama naršyklė su URL https://192.168.0.1:3000, tada paspaudžiamas „Robotas“ mygtukas. Toliau, vienu metu, padalintame ekrane įjungiama UDP tilto programėlė.
- Įjungiamas mikrovaldiklis ir laukiama kol prisijungs prie prieigos taško.
- Kompiuteryje įjungiama naršyklė su URL https://192.168.0.1:3000 ir paspaudžiamas „Pultas“ mygtukas.
- Mobiliajame įrenginyje paspaudžiamas „Send Video“ mygtukas. Vaizdas turėtų būti transliuojamas į kompiuterį.
- Mobilus įrenginys įstatomas į korpusą, pagrindinė kamera turi žiūrėti į dugną.
- Įstatomi svoriai, pamatuojama, kad robotas vos kiltų vandenyje.
- Tvirtai užsukamas dangtis.

Tai atlikus robotas yra paruoštas veikimui vandenyje. Programų dizainų pavyzdžiai yra 3 PRIEDE.

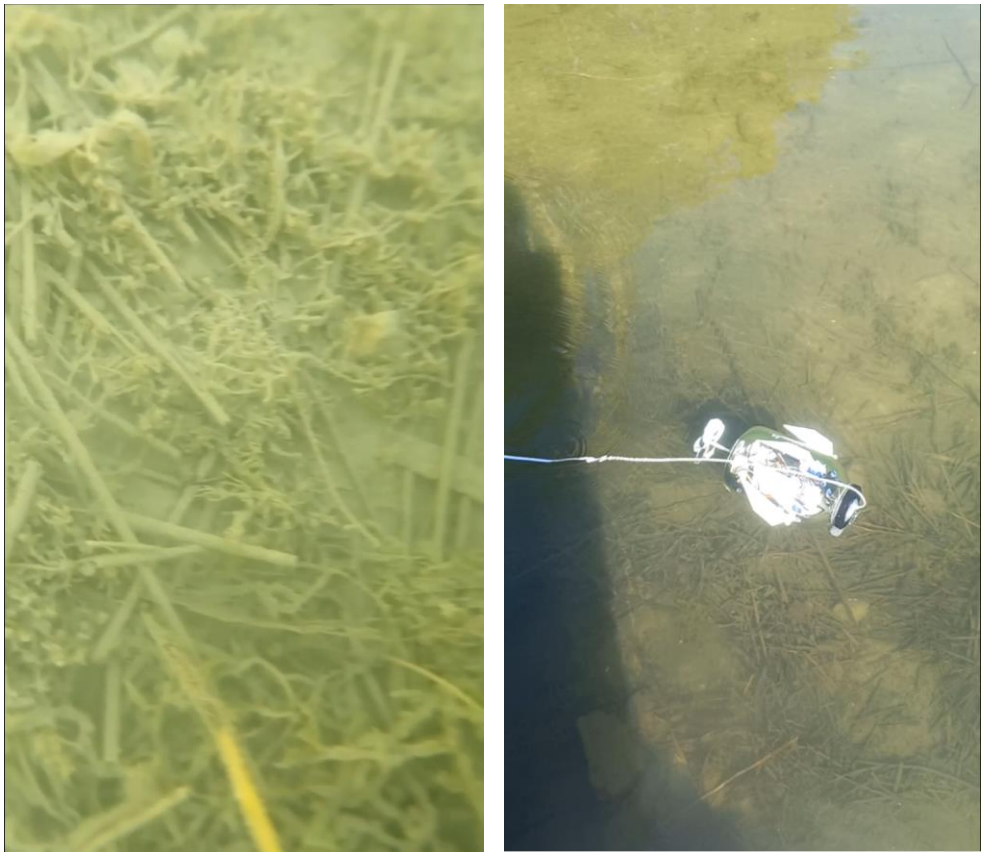
Galutinis roboto dizainas matomas 4.23 pav. Galutinė kaina – apie 110 eurų.



4.23 pav. Galutinis roboto dizainas

Toliau aprašomas roboto bandymas.

Robotas buvo išbandytas 2024-05-13 Galvės ežere, Trakuose. Kairioje nuotraukoje matomas vaizdas iš roboto, dešinėje – robotas, kuris po vandeniu (4.24 pav.).



4.24 pav. Roboto bandymas ežere

Robotas plaukia stebėtinai greitai, apie 0.5 m/s greičiu. Vanduo neprasiskverbė į vidų 1 m gylyje. Vaizdo kokybė buvo geresnė negu tikėtasi. Bet servo varikliai nėra pakankamai galingi, kad robotas galėtų prasmingai manevruoti vandenyje. Roboto variklis per daug sulėtėja vandenyje, nuo to kaista variklis ir greičiau išsikrauna akumuliatorius.

Kitoje roboto iteracijoje bus naudojami MG995 servo varikliai, jų sukimosi momentas yra 15 kg/cm, kuris yra 10 kartų didesnis už SG90 servo variklių sukimosi momentą. Taip pat, bus atspausdinti didesni pelekai, kurie naudos daugiau magnetų. Toliau, bus naudojamas mažesnis propeleris, kad variklis greičiau suktųsi.

5. REZULTATAI IR APIBENDRINIMAS

Šiame darbe buvo įgyvendintas darbo tikslas ir atlikti uždaviniai.

Vaizdo perdavimas iš roboto į kompiuteri buvo atliktas pasinaudojant mobiliuoju įrenginiu, kuris įstatomas į robotą. Ryšys yra sudaromas su tinklo kabeliu, kuris jungia kompiuterį su mobiliu įrenginiu. Vaizdo duomenys yra perduodami pasitelkiant WebRTC projekto galimybėmis.

Roboto valdymas iš kompiuterio buvo realizuotas. Valdymo signalai yra perduodami iš kompiuterio į mobilųjį įrenginį ir iš mobilaus įrenginio į mikrovaldiklį su UDP protokolu. Robotas valdomas su klaviatūros klavišais.

Robotas buvo sėkmingai išbandytas vandens telkinyje. Jokie komponentai nebuvo sugadinti, vaizdo medžiaga buvo įrašyta. Robotas juda greitai, bet nėra manevrus.

LITERATŪRA

[1] H. Choudhari, S. Gangshettiwar, S. Fale, V. Mohite, „Design and Fabrication of Underwater Surveillance Drone“, International Research Journal of Engineering and Technology (IRJET), vol. 08, no. 04, pp. 3799, Apr. 2021. [Tinkle]. Available: https://www.academia.edu/51593910/Design_and_Fabrication_of_Underwater_Surveillance_Drone.

[2] M. Wright, Q. Xiao, S. Dai, M. Post, H. Yue, B. Sarkar, „Design and development of modular magnetic bio-inspired autonomous underwater robot – MMBAUV“, Ocean Engineering, vol. 273, p. 113968, 2023, doi: 10.1016/j.oceaneng.2023.113968. [Tinkle]. Available: https://www.sciencedirect.com/science/article/pii/S0029801823003529.

[3] H. F. Gasparoto, O. Chocron, M. Benbouzid, P. S. Meirelles, „Advances in Reconfigurable Vectorial Thrusters for Adaptive Underwater Robots“, J. Mar. Sci. Eng., vol. 9, p. 170, 2021, doi: 10.3390/jmse9020170. [Tinkle]. Available: https://www.mdpi.com/2077-1312/9/2/170.

[4] P. Lunkenheimer, S. Emmert, R. Gulich, M. Köhler, M. Wolf, M. Schwab, A. Loidl, „Electromagnetic-radiation absorption of water“, arXiv:1612.01457 [cond-mat.soft], 2016. [Tinkle]. Available: https://arxiv.org/abs/1612.01457.

[5] Lietuvos Respublikos ryšių reguliavimo tarnyba, „Radijo dažnių skyrimas ir naudojimas,“. [Tinkle]. Available: https://www.rrt.lt/radijo-spektras/radijo-dazniu-valdymas/skyrimas-ir-naudojimas/. [Žiūrėta 2024-05-20].

[6] MIT Lincoln Laboratory, „Toroidal propeller“ TVO Technology Highlight 41, Sep. 2022. [Tikle]. Available: https://www.ll.mit.edu/sites/default/files/other/doc/2022-09/TVO_Technology_Highlight_41_Toroidal_Propeller.pdf. [Žiūrėta 2024-05-20].

[7] Visuotinę lietuvių enciklopediją, „Archimedo dėsnis“. [Tinkle]. Available: https://www.vle.lt/straipsnis/archimedo-desnis/. [Žiūrėta 2024-05-20].

[8] Google for Developers, „Real-time communication for the web“. [Tinkle]. Available: https://webrtc.org/. [Žiūrėta 2024-05-20].

[9] Paweł Fertyk, „WebRTC: a working example“. [Tinkle]. Available: https://github.com/pfertyk/webrtc-working-example. [Žiūrėta 2024-05-20].

[10] MDN, „The WebSocket API“. [Tinkle]. Available: https://developer.mozilla.org/en-US/docs/Web/API/WebSockets_API. [Žiūrėta 2024-05-20].

[11] „DHCP Server for Windows“. [Tinkle]. Available: https://www.dhcpserver.de/cms/ . [Žiūrėta 2024-05-20].

[12] „Node.js runtime“. [Tinkle]. Available: https://nodejs.org/en. [Žiūrėta 2024-05-20].

[13] „UDP/datagram sockets“. [Tinkle]. Available: https://nodejs.org/api/dgram.html. [Žiūrėta 2024-05-20].

[14] Google, „Flutter“. [Tinkle]. Available: https://flutter.dev/. [Žiūrėta 2024-05-20].

[15] Raspberry Pi Ltd, „Raspberry Pi Pico Python SDK“. [Tinkle]. Available: https://datasheets.raspberrypi.com/pico/raspberry-pi-pico-python-sdk.pdf. [Žiūrėta 2024-05-20].

[16] „STL gears generator“. [Tinkle]. Avaialble: https://www.stlgears.com/generators/3dprint. [Žiūrėta 2024-05-20].

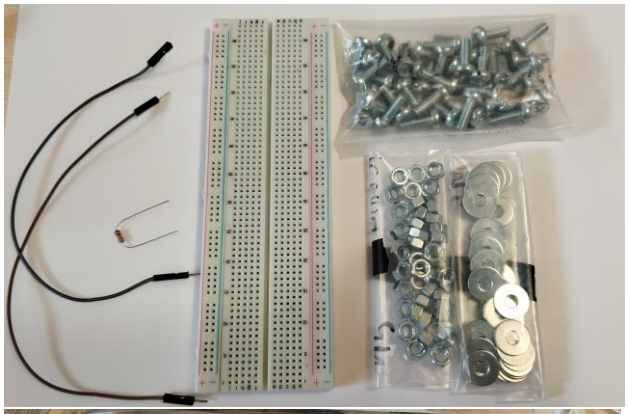
[17] Raúl Bejarano, „Ultimate Toroidal Propeller Generator“. [Tinkle]. Available: https://github.com/RaulBejarano/Ultimate-Toroidal-Propeller-Generator?tab=readme-ov-file. [Žiūrėta 2024-05-20].

[18] Raspberry Pi Ltd, „Raspberry Pi Documentation“. [Tinkle]. Avaialble: https://www.raspberrypi.com/documentation/microcontrollers/raspberry-pi-pico.html. [Žiūrėta 2024-05-20].

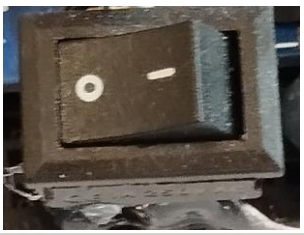
[19] „Servo & PWM Guide: SG90 9g Micro Servo“. [Tinkle]. Avaialble: https://opencoursehub.cs.sfu.ca/bfraser/grav-cms/cmpt433/links/files/2022-student-howtos/ServoPWMGuide_SG90_9gMicroServo.pdf. [Žiūrėta 2024-05-20].

[20] „Magnet Pull Force Calculator“. [Tinkle]. Avaialble: https://www.kjmagnetics.com/calculator.asp. [Žiūrėta 2024-05-20].

1 PRIEDAS













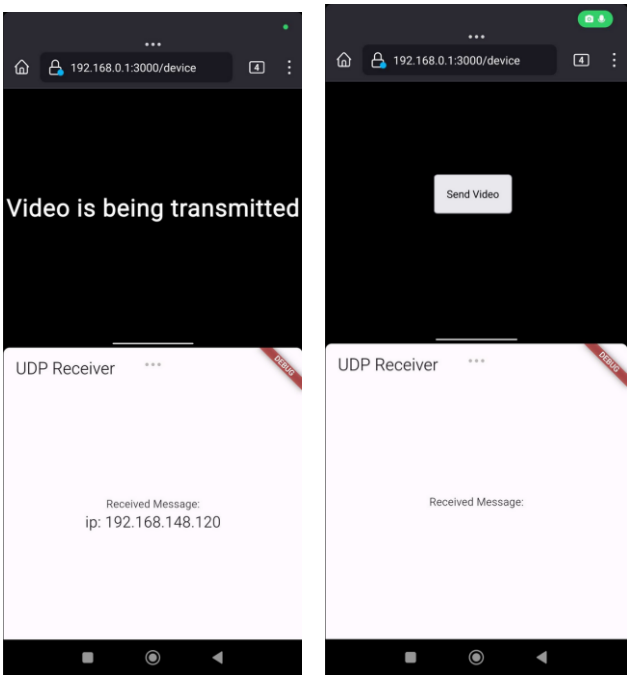
Roboto surinkimui naudoti komponentai ir detalės

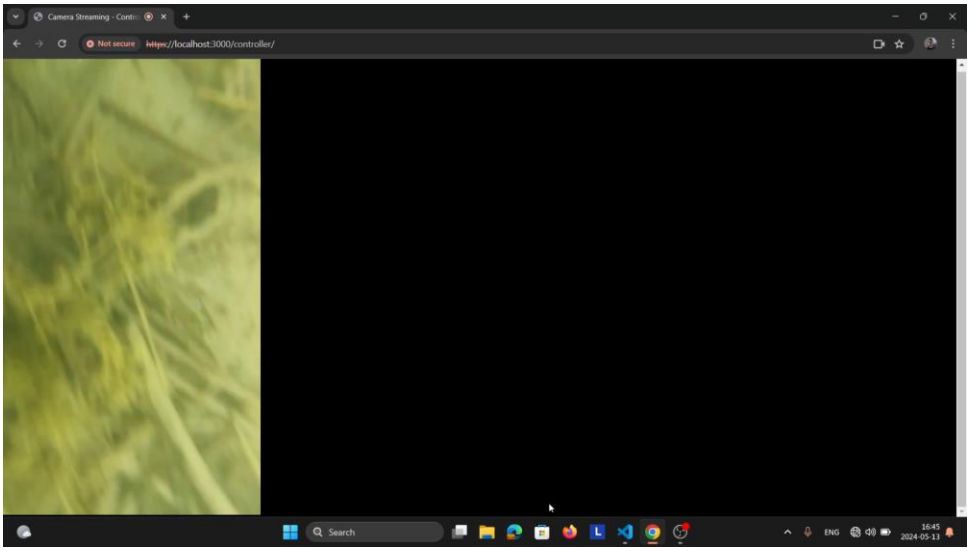
2 PRIEDAS

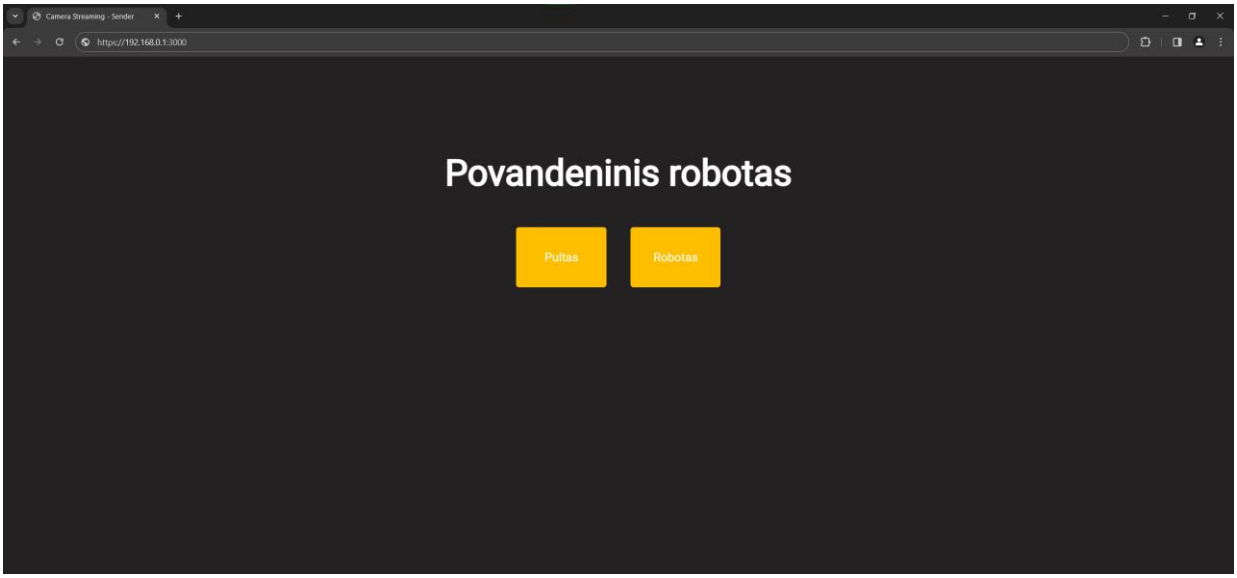
```
use <src/toroidal_propeller.scad>
$fn = 1000;                       // how polligonall you want the model

toroidal_propeller(
    blades = 5,                     // number of blades | Default(3)
    height = 8,                     // height | Default(6)
    blade_length = 68,              // blade length | Default(68)
    blade_width = 42,               // blade width | Default(42)
    blade_thickness = 4,            // blade thickness | Default(4)
    blade_hole_offset = 1.4,        // blade hole offset | Default(1.4)
    blade_attack_angle = 35,        // blade attack angle | Default(35)
    blade_offset = -6,              // blade distance from propeller axis | Default(-
6)
    blade_safe_direction = "PREV",  // indicates if a blade must delete itself from
getting into the previous (PREV) or the next blade (NEXT) | Default("PREV")
    hub_height = 8,                 // Hub height | Default(6)
    hub_d = 16,                     // hub diameter | Default(16)
    hub_screw_d = 4,              // hub screw diameter | Default(5.5)
    hub_notch_height = 6,           // height for the notch | Default(0 = [No
support])
    hub_notch_d = 8.3                 // diameter for the notch | Default(0 = [No
support])
);
```

3 PRIEDAS







Programų dizainai