Šiaulių Simono Daukanto gimnazija

Brandos darbo aprašas

IV klasės

**5 ašių roboto rankos – manipuliatoriaus valdymo algoritmo programavimas bei jai valdyti mobilaus telefono aplikacijos kūrimas.**

Dovydas Špukas

Ričardas Gečas. Informacinės technologijos - Mokytojas metodininkas

Šiauliai 2022

Turinys

[Įvadas 3](#_Toc96700596)

[Teorinis temos pagrindimas 4](#_Toc96700597)

[Roboto rankos - manipuliatoriaus projektavimas ir konstravimas 5](#_Toc96700598)

[Roboto rankos – manipuliatoriaus Arduino programos kūrimas 6](#_Toc96700599)

[Mobilaus telefono aplikacijos kūrimas valdyti 5 ašių roboto ranką – manipuliatorių 10](#_Toc96700600)

[Sukurtos mobilios aplikacijos naudojimas valdant 5 ašių roboto ranką – manipuliatorių 13](#_Toc96700601)

[Darbo išvados 14](#_Toc96700602)

[Informacijos Šaltiniai 15](#_Toc96700603)

# Įvadas

Darbų automatizavimas vis labiau aptariamas klausimas XXI amžiuje. Tačiau ne tik dabar šis klausimas iškilo, žmonės bandė automatizuoti darbus nuo senovės, paprasčiausias įrodymas – vandens ratas. Šis procesas raiškiausiai dabar ir matosi pramonės revoliucijos 4.0 metu. Atliekant automatizavimui aiškiai mažėja paprasčiausio lygio darbų kiekis, tačiau tai suteikia galimybės žmonės sutelkti jėgas į labiau esmines, aktualesnes problemas. Nors ne visi teigiamai vertina šį procesą, tai yra garantuotą ateitis darbo. Teisingas darbo automatizavimo supratimas stipriai padėtų vystytis visoms gyvenimo sritims: ekonomikai, mokslui, progreso spartai.

Šis projektas yra prieš tai minėtos idėjos įgyvendinimas. Kuriant darbo automatizavimui produktą, kuris būtų lengvai valdomas naudojant viena iš labiausiai paplitusių technologijos aparatų – išmanaus mobilaus telefono, būtų įmanoma įrodyti kaip paprastai galima valdyti ateities darbuotoją, pilnai automatizuotą, veikiantį 24/7. Šis projektas yra žinių panaudojimas sukuriant produktą, kuris gali būti naudojimas automatizuoti paprastus darbus.

Tikslas: Sukurti programą telefonui, kuri leistu valdyti roboto ranką

Uždaviniai:

1. Suprojektuoti ir sukonstruoti 5 ašių roboto rankos – manipuliatoriaus prototipą.
2. Suprogramuoti valdymo algoritmą roboto rankai - manipuliatoriui.
3. Sukurti programa telefonui, kuri leistu valdyti roboto ranką - manipuliatorių naudojant „Bluetooth“ ryšį.

# Teorinis temos pagrindimas

Daug darbo reikalaujančios daugiapakopės biologinės užduotys, pavyzdžiui, DNR molekulių konstravimas ir klonavimas, yra pagrindinės laboratorinio automatizavimo kandidatės. Lankstus ir biologijai pritaikyto robotų įrangos valdymas yra esminis dalykas siekiant sėkmingai integruoti juos į laboratorijas, o robotui valdyti reikalingos pastangos turi būti daug mažesnės nei alternatyvus rankinis laboratorinis darbas. Siekiant įgyvendinti darbų automatizavimo reikalaujama didelė pažanga robotų technologijoje.

Statyba yra viena iš pagrindinių pramonės šakų visame pasaulyje. Statybų pramonė reikalauja daug darbo jėgos ir vyksta pavojingomis sąlygomis, todėl statybų robotikos svarba sparčiai auga. Robotikos ir automatikos taikymas ir veikla šioje pramonės šakoje prasidėjo dešimtojo dešimtmečio pradžioje, siekiant optimizuoti įrangos darbą, pagerinti saugą, pagerinti darbo vietos suvokimą, be to, užtikrinti kokybišką aplinką pastato gyventojams. Tokiais tikslais yra siekiama vystyti robotų technologijas, kurios leistų sparčiai pagerinti visą statybų pramonę bei sumažinti žmonėms kylantį pavojų nuo jų.

Aiškūs du pavyzdžiai, kuriuose lengvai valdoma robotizuota įranga leidžia automatizuoti darbą, tobulinti jo kokybę. Naudojant roboto rankas - manipuliatorius yra įmanoma pagerinti darbo aplinką žmonės, patobulinti to pačio darbo spartą ir kokybę bei automatizavimas leistų žmonės sufokusuoti savo pastangas į sritis, kuriose reikia mąstymo, kūrybingumo lygio, kurio dabartiniai robotai negali pasiekti. Siekdami aukštesnio lygio automatizavimo žmonės sugebės pradėti vis didesnius, sudėtingesnius projektus. Pasikliaunat robotais yra ateitis, kuri leidžia žmonėms pasiekti kitą, kaip civilizacijos, etapą. Tad lengvas robotų valdymas yra vienas iš esminių žingsnių, kuriuos reikia įveikti. Tad spartesnė pažanga gerinant ir lengvinant robotų rankos valdymą leistų pasiekti naują darbingumo lygį. Tad pritaikant valdymo sistemą prie aparato, kurį didelė dalis žmonių jau gerai pažįsta, - mobilaus telefono leistų ženkliai suprastinti visą procesą. Taip pat matant, kas yra įmanoma pasiekti su viena iš ryškiausių sričių pramonės revoliucijos 4.0, žmonių susidomėjimas padidėtų. Atsirastų žmonių, kurie nežinodami, ką turėtų veikti, gautų tikslus, troškimą pagerinti, tobulinti šitas technologijų sritis, atsirastų žmonių, kurie norėtų finansuoti, studijuoti tai ir leistų jai klestėti Darbo automatizavimas naudojant robotus yra daug darbo sričių ateitis, leidžianti patobulintų tų sričių darbo kokybę, greitį, efektyvumą Tad ir kyla šio darbo aktualumas.

# Roboto rankos - manipuliatoriaus projektavimas ir konstravimas

Naudojant „AUTOCAD“ aplinka braižiau 5 ašių roboto rankos – manipuliatoriaus dalis. Su3D spausdintuvu suprojektuotas dalis atspausdinti pavyko. Po spausdinimo proceso reikėjo panaikinti visą „support“ medžiagą. Naudojant MG-966R servo variklius manipuliuoti roboto rankos – manipuliatoriaus juosmenį, petį bei alkūnę, o SG90 servo variklius manipuliuoti riešo rotaciją aplink savo ašį, riešo lenkimą vertikalia ašimi bei griebtuvo suspaudimą ir atleidimą sukonstravau 5 ašių roboto ranką – manipuliatorių. Toliau vyko elektroninės dalies sujungimas. Naudojant Arduino UNO mikrovaldiklį kaip roboto rankos – manipuliatoriaus smegenys prilitavau kiekviena servo variklį prie „digital“ jungčių(t.y nuo 5 iki 10 jungties), taip pat prijungti reikėjo teigiamo ir neigiamo poliaus laidus prie atitinkamų laidų 5V išorinio maitinimo šaltinio jungčių. Teigiamo ir neigiamo poliaus laidus prilitavau prie 5V ir GND atitinkamų jungčių Arduino plokštėje. Prijungiau HC-05 „Bluetooth“ modulį prie mikrovaldiklio, tačiau neprilitavau, nes keliant kodą į mikrovaldiklį kyla problemos jeigu bus nuolat modulis prijungtas prie mikrovaldiklio ( jungiant prie RX ir TX, atitinkamai 0 ir 1 jungtis, tos jungtys taip pat naudojamos komunikuojant su kompiuteriu per USB jungtį, dėl to jeigu bus prijungtas „Bluetooth“ modulis nepavyks įkeltį kodą.) Tačiau problema lengvai išsprendžiama prijungiant „Bluetooth“ modulį su laidais prie mikrovaldiklio. Konstravimas ilgai neužtruko, tačiau iš karto pasimatė problema, kad patik ranka neišstovi dėl laidų, kurie išsikiša iš po jos ir neleidžia stabiliai stovėti, tad reikėjo pasidaryti dėžutę. Ant dėžutės viršaus bus prisukta ranka.

# Roboto rankos – manipuliatoriaus Arduino programos kūrimas

Pradėjau programuoti nuo paprasčiausio duomenų siuntimo iš telefono į Arduino mikrovaldiklį. Šio etapo tikslas buvo išsiaiškinti kaip naudotis pačiu HC-05 moduliu bei Arduino galimybės su „Bluetooth“. Laikinas tikslas buvo įjungti ir išjungti LED lemputė esančia mikrovaldiklyje. Problemos kilo su „BAUD RATE“, tačiau atliekant bandymus, t.y. keisdamas tą vertę tarp galimų variantų pavyko nustatyti reikalinga skaičių (teisinga vertė, kuri leido sėkmingam susijungimui tarp telefono ir HC-05 modulio buvo 9600, tačiau pastebėjau, kad gali skirtis šis skaičius). Šis kodas išliks kaip pagrindas iki telefono programos kūrimo, kuris bus naudojamas, kaip „Bluetooth“ ryšio testas. Kuriant pagrindinę programa pradėjau nuo „SoftwareSerial.h“ bei „Servo.h“ bibliotekų įvedimą, toliau 6 Servo variklių paskelbimą taip pat pastovius kintamuosius, kurie nustato „char“ tipo masyvo dydį, bei vėliau kintamasis, kuris yra tiktais tarpas, kuris leis pravalyti „char“ tipo masyvą pavadinimu „Data“(toliau bus kreipiamasi į šį masyvą kaip Data). Taip pat sukuriami 12 kintamųjų, kurie parodys servo variklių pasukimo padėtį. 6 pastovūs kintamieji, kurie bus naudojami kaip rankos motorų pradinės pozicijos nustatymui(gale kintamųjų pavadinime „S“ raidė) ir bus 6 kintamieji, kurių vertės bus keičiamos atitinkamai pagal gauta informaciją. Toliau reikėjo deklaruoti, prie kurių jungčių prijungti servo varikliai, ir visi varikliai nustatomi į jų pradines pozicijas, kurios buvo ankščiau paskelbtos. Su šiuo veiksmu ir pasibaigė pirma („void setup“) programos dalis. Antra dalis(„void loop“) prasideda nuo sąlyginio sakinio, kuris tikrina ar yra atsiunčiama informacija. Jeigu yra atsiunčiama informacija, tada naudojant „for“ ciklą gauta informacija yra įrašoma į Data masyvą. Ir tuomet priklausant nuo informacijos yra keičiama servo varikliu pasukimo padėtis. Gauti duomenys yra dviejų tipu: arba dviejų raidžių kombinacija, kurioje pirma raidė parodo, kuris variklis yra valdomas, o antra raidė yra ką daryti su pačia verte, t.y. jeigu „p“ variklio motoro padėtis yra didinama, „n“ mažinama, „s“ variklis atstatomas į pradinę poziciją, jeigu gautos raidės yra „ra“ visa ranka gražinama į pradinę poziciją, jeigu atsiųsta informacija yra skaičius, tada jis nustato vertę, pagal kuria bus ar didinama, ar mažinama variklio padėties kintamieji. Naudojama servo varikliai, o jie gali tiktais pasisukti 180 laipsnių kampu, tad yra sąlyginis sakinys, kuris jeigu keičiant variklio motoro padėti tas skaičius tampa didesnis už 180 arba mažesnis už 0, tai tos reikšmės yra nustatomos atitinkamai 180 ir 0. Pavyzdys: gaunama iš pradžių skaičius 10, o vėliau raidės „wp“, tai reiškia, kad juosmuo( juosmuo, nes pirma raidė atitinka juosmenį) padidės(nes antra raidė p) per 10. „Void loop“ kodo dalis užsibaigia su Data masyvų pravalymų(t.y. visos Data masyvo reikšmės nustatomas į tarpą.)

Nors šitam projekte nėra šios galimybės, toliau tobulinant projektą yra padaryti pamatai, kurie leistu labai lengva pridėjimą į kodą funkcijos, kurios suteiktų galimybę atlikti iš anksto suplanuotus veiksmus. Pav. 1



pav. 1

Text

Description automatically generated

pav. 2

Text

Description automatically generated

pav. 3

Text

Description automatically generated

pav. 4

# Mobilaus telefono aplikacijos kūrimas valdyti 5 ašių roboto ranką – manipuliatorių

Pradedant programuoti pirmiausiai reikėjo nuspręsti, kokioje aplinkoje ir kokią kalbą naudosiu kuriant savo programą. Greitai paaiškėjo, kad turiu tris pasirinkimus. Programavimo kalbų, kurios turi platų naudojimą, priedų, kurios palengvintų visą procesą ir kt. kriterijai buvo atsižvelgiami renkant kalbą ir aplinką. Nors ir tarp trijų pasirinkimų: Java, Kotlin ir Python buvo daug pliusų naudoti kiekvieną kalbą, pavyzdžiui, Java buvo oficiali Android programėlių programavimo kalba ir turėjo daug aplinkų, kurios palengvintų darbą, taip pat labai platus naudojimas. Python, su savo labai paprasta sintaksė ir galimybė greitai išmokti dėl labai didelio ir įvairaus kiekio internete mokymosi šaltinių, knygų ir t.t. Taip pat būdama viena populiariausių, jeigu ne populiariausia programavimo kalba pasaulyje galėjo ženkliai palengvinti darbą. Trečioji kalba Kotlin gauna taip pat turi didelį kiekį pliusų. Pati kalba sukurta, kad patobulintu Java ir perimtu oficialia Android programėlių programavimo kalbos titulą. Taip pat Kotlin glaudžiai susijusi su Java, abi programavimo kalbos leidžiančios iškviesti ir naudoti kodą iš vienų kitų bibliotekų, galima teigti, kad Kotlin yra Java kalbos įpėdinė. Nors ir atrodo, kad Kotlin būtų geriausias pasirinkimas, tačiau būdama reliatyviai nauja kalba, yra ženkliai didesnis kiekis informacijos, darytų projektų ir kitų informacijos šaltinių apie Java, kad visus jos minusus užgožia šis faktas. Tad ir pasirinkau naudoti Java.

Pradėjau nuo programos pačios dizaino. Naudojant Android Studio ir joje jau integruotos XML galimybės, procesas nebuvo itin sunkus. Iš pradžių sukurti mygtukai prisijungti prie „Bluetooth“ modulio ir atsijungti, taip pat tekstas, kuris priklauso ir pasikeičia nuo tuo ar prijungtas įrenginys. Toliau buvo sukurti paprasti du mygtukai strėlės formos, kurie simbolizuoja roboto rankos – manipuliatoriaus servo varikliu pasukimą, bei išplečiamasis sąrašas, kuris leistų nustatyti variklių greitį. Taip pat yra išplečiamasis sąrašas, kuris leidžia pasirinkti, kuris variklis valdomas. Toliau pridėjau vaizdą, kuris parodytų, kuri dalis yra valdoma. Išsiunčiama informacija yra dviejų tipų: skaičiai, kuris parodo kokio varikliu pasukimo kampo dydys ir dvi raidės, kurios keičia atitinkamą variklio padėtį. Visas procesas užtruko didelį kiekį laiko, dėl Java sudėtingumo lygio, nes pradedančiajam mokytis kalbą palyginus su C++, kurą mokausi mokykloje, pasirodė ženkliai sunkesnė bei platesnė. Programa be labai didelių sunkumų, apart didelio laiko reikalavimo pavyko padaryti. Didelis kiekis sprendimų, kaip turėtų veikti pati programa, kokią informaciją išsiunčiu buvo padaryti rašant Arduino kodą. Taip pat yra pamatai, kaip ankščiau minėti, padaryti mygtukus, kurie galėtų išsiųsti informaciją, kad ranka atliktų iš anksto nustatytus veiksmus.

(Mobilios aplikacijos kodas bus pateiktas kodų „blokais“, kad lengviau butų suprasti Java kodą)

Timeline

Description automatically generated

pav. 5

Graphical user interface, application

Description automatically generated

pav. 6

Timeline

Description automatically generated

pav. 7

# Sukurtos mobilios aplikacijos naudojimas valdant 5 ašių roboto ranką – manipuliatorių

Pirmas telefono susiejimas su HC - 05 modulių, nors ir ryšys susidarė, buvo nesėkmingas, nes siunčiant informacija, Arduino mikrovaldiklis nereagavo į ją. Problema gavosi, kad įvyko dėl pačio mikrovaldiklio nes prijungiant prie naujo Arduino, viskas pradėjo vykti sklandžiai. Tačiau kilo problema, kad ranką pradėjo „drebėti“, t.y. motorai atrodė, kad pastoviai atsistatydavo į pradinę poziciją ir pajudėdavo į kitą, taip vyko, kol ranka buvo prijungta prie mobilaus telefono programos. Iš tikrųjų nepavyko išsiaiškinti kodėl taip vyko, tačiau pakeičiant vietą, kur Data masyvo pravalymas vyksta, sutvarkė problemą ranka „nebedrebėjo“. Sklandžiai pavyko valdyti sukonstruotą roboto ranką naudojant sukurta mobiliaus telefono aplikacija.

# Darbo išvados

1. Darbų automatizavimas naudojant robotiką yra pripažinta įvairiuose darbo aplinkose, kaip ateitis, kuri yra garantuota. Automatizuojant darbus drastiškai kyla darbingumas, produktyvumas. Tad paklausa robotų, kurie galėtų atlikti nuo paprastų ir pasikartojančių veiksmų iki kompleksinių užduočių kyla.
2. Siekiant darbų automatizavimui naudojant roboto rankas – manipuliatorius reikalingas prieigos taškas leidžiantis ją valdyti rankiniu būdu, tada šio prieigos taško suprastinimas yra esminis kriterijus didesnio kiekio darbų automatizavimui .
3. Atliekant projektinį darbą pavyko įgyvendinti visus užbrėžtus tikslus: 5 ašių roboto rankos – manipuliatoriaus konstravimas bei jos operacinio kodo kūrimas, mobilaus telefono aplikacijos kūrimą skirta valdyti sukonstruota roboto ranką – manipuliatorių, sukurtos mobilaus telefono aplikacijos pritaikymas valdant roboto ranką - manipuliatorių
4. Turėdami aukštesnį žinių lygį bei didesnį kiekį laiko bei žmonių kiekį, naudojant net ir santykinai paprastas darbo aplinkas kuriant šį projektą, jį tikrai įmanoma ištobulinti iki gatavo produkto, kurį būtų įmanoma parduoti, naudoti darbų automatizavimui, tačiau išlaikant roboto rankos – manipuliatoriaus valdymo paprastumą.
5. Iškilo problemos, kurios projekto eigoje neišsisprendė. Tai yra: po kažkiek laiko nenaudojimo Arduino nebereaguoja į gautą informaciją, tačiau problema lengvai galima apeiti paprasčiausiai atsijungiant nuo rankos ir išjungiant ją, kai yra nenaudojama arba atsijungiant nuo rankos ir perkraunant Arduino mikrovaldiklį su esančiu ant pačio mikrovaldiklio ir tokiai funkcijai skirtam mygtuku. Taip pat galima ženkliai patobulinti mobilią aplikaciją pritaikant Java kalbą aukštesniame lygyje. Taip pat yra atvejai jeigu Arduino mikrovaldiklis yra iš naujo paleidžiamas neatsijungiant nuo mobilaus telefono, dėl šio veiksmo pati aplikacija gali sutrikti ir yra naudotojas priversta iš naujo paleisti ją. Problemos kilmė nėra aiški, tačiau turint didesnį laiko kiekį, manau, kad tikrai galima ją išspręsti.
6. Atliekant patį projektą įgytas didelis kiekį praktinių žinių: mobilaus telefono aplikacijų kūrimą, didelė praktika sprendžiant problemas, įgyta patirtis dirbant su Arduino mikrovaldikliais bei jų programavimo aplinka, pagrindų įgijimas XML bei Java kalba (paliečiau Java kalbos galimybių patį paviršių, nes labai fokusavau į kalbos aspektą t.y. programėlės sukūrimą su „Bluetooth“ galimybėmis), projektavimo, laiko planavimo, litavimo įgūdžių. Taip pat įgyta patirtis praktiškai pritaikant įgytas žinias per pamokas(specifiškai elektros grandinės žinias) lituodamas elektroninę projekto dalį.

# Informacijos Šaltiniai

1. AUTOMATION AND ROBOTICS IN CONSTRUCTION:  
   OPPORTUNITIES AND CHALLENGES. S.M.S. Elattar. Prieiga internete: <https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.524.7909&rep=rep1&type=pdf>
2. PaR-PaR laboratorijų automatizavimo platforma. Gregory Linshiz, Nina Stawski, Sean Poust, Changhao Bi Jay D. Keasling, and Nathan J. Hilson Prieiga internete: <https://pubs.acs.org/doi/pdf/10.1021/sb300075t>
3. Arduino IDE, programavimo aplinkos gidas: <https://www.arduino.cc/reference/en/>
4. Android studio gidas: <https://developer.android.com/guide/>