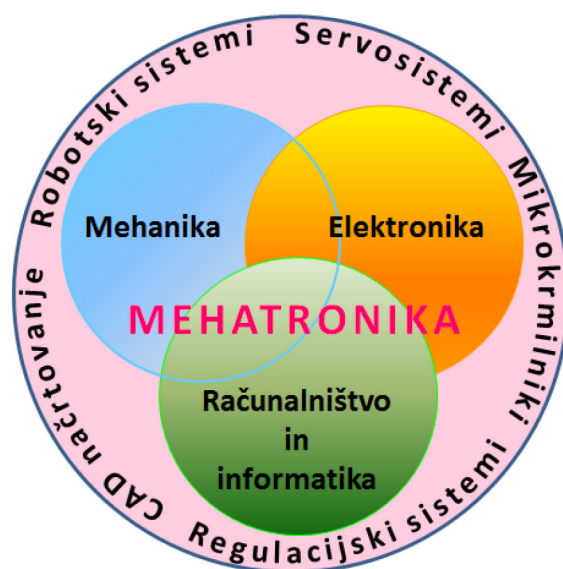


UNIVERZA V MARIBORU  
FAKULTETA ZA ELEKTROTEHNIKO, RAČUNALNIŠTVO IN INFORMATIKO  
FAKULTETA ZA STROJNIŠTVO

# 10. LETNA KONFERENCA MEHATRONIKE 2021

## ZBORNİK POVZETKOV ŠTUDENSKIH PROJEKTOV



Uredniki:  
**Janez Pogorelc**  
**Aleš Hacı**  
**Uroš Župerl**



Univerzitetna založba  
Univerze v Mariboru







Univerza v Mariboru

Fakulteta za elektrotehniko,  
računalništvo in informatiko

Fakulteta za strojništvo

## **10. LETNA KONFERENCA MEHATRONIKE 2021**

Zbornik povzetkov študentskih projektov

Uredniki

**Janez Pogorelc**

**Aleš Hace**

**Uroš Župerl**

Maribor, avgust 2021

- Naslov** **10. letna konferenca mehatronike 2021**  
*Title* *10<sup>th</sup> Annual Conference of Mechatronics 2021*
- Podnaslov** **Zbornik povzetkov študentskih projektov**  
*Subtitle* *Book of Abstracts, Student Projects*
- Uredniki** Janez Pogorelc  
*Editors* (Univerza v Mariboru, Fakulteta za elektrotehniko, računalništvo in informatiko)
- Aleš Hace  
(Univerza v Mariboru, Fakulteta za elektrotehniko, računalništvo in informatiko)
- Uroš Župerl  
(Univerza v Mariboru, Fakulteta za strojništvo)
- Tehnična urednika** Janez Pogorelc  
*Technical editors* (Univerza v Mariboru, Fakulteta za elektrotehniko, računalništvo in informatiko)
- Jan Perša  
(Univerza v Mariboru, Univerzitetna založba)
- Oblikovanje ovitka** Uroš Župerl  
*Cover designer* (Univerza v Mariboru, Fakulteta za strojništvo)
- Grafika na ovitku** UM FS in UM FERI (logotip)  
*Cover graphics*
- Grafične priloge** Avtorji prispevkov  
*Graphics material*
- Konferenca** **10. letna konferenca mehatronike 2021**  
*Conference*
- Datum in kraj** 23. 6. 2021, Maribor, Slovenija  
*Date and place*
- Organizacijski odbor** Aleš Hace (Univerza v Mariboru, Fakulteta za elektrotehniko, računalništvo in informatiko), Janez Pogorelc (Univerza v Mariboru, Fakulteta za elektrotehniko, računalništvo in informatiko), Uroš Župerl (Univerza v Mariboru, Fakulteta za strojništvo), Miro Milanovič (Univerza v Mariboru, Fakulteta za elektrotehniko, računalništvo in informatiko), Karl Gotlih (Univerza v Mariboru, Fakulteta za strojništvo), Miran Rodič (Univerza v Mariboru, Fakulteta za elektrotehniko, računalništvo in informatiko), Darko Hercog (Univerza v Mariboru, Fakulteta za elektrotehniko, računalništvo in informatiko) in Darko Lovrec (Univerza v Mariboru, Fakulteta za strojništvo).  
*Organizing committee*
- Založnik** **Univerza v Mariboru**  
*Published by* **Univerzitetna založba**  
Slomškovo trg 15, 2000 Maribor, Slovenija  
<https://press.um.si>, [zalozba@um.si](mailto:zalozba@um.si)
- Izdajatelj** **Univerza v Mariboru**  
*Co-published by* **Fakulteta za elektrotehniko, računalništvo in informatiko**  
Koroška cesta 46, 2000 Maribor, Slovenija  
<http://www.feri.um.si>, [feri@um.si](mailto:feri@um.si)
- Izdajatelj** **Univerza v Mariboru**  
*Co-published by* **Fakulteta za strojništvo**  
Smetanova ulica 17, 2000 Maribor, Slovenija  
<https://fs.um.si>, [fs@um.si](mailto:fs@um.si)

<b>Izdaja</b> <i>Edition</i>	Prva izdaja
<b>Vrste izdaje</b> <i>Publication type</i>	E-knjiga
<b>Izid</b> <i>Published</i>	Maribor, avgust 2021
<b>Dostopno na</b> <i>Availabe at</i>	<a href="https://press.um.si/index.php/ump/catalog/book/596">https://press.um.si/index.php/ump/catalog/book/596</a>



© Univerza v Mariboru, Univerzitetna založba  
/ *University of Maribor, University Press*

**Tekst / Text** © Pogorelc, Hace, Župerl in avtorji, 2021

To delo je objavljeno pod licenco Creative Commons Priznanje avtorstva 4.0 Mednarodna.  
/ *This work is licensed under the Creative Commons Attribution 4.0 International License.*

Uporabnikom je dovoljeno tako nekomercialno kot tudi komercialno reproduciranje, distribuiranje, dajanje v najem, javna priobčitev in predelava avtorskega dela, pod pogojem, da navedejo avtorja izvirnega dela.

Vsa gradiva tretjih oseb v tej knjigi so objavljena pod licenco Creative Commons, razen če to ni navedeno drugače. Če želite ponovno uporabiti gradivo tretjih oseb, ki ni zajeto v licenci Creative Commons, boste morali pridobiti dovoljenje neposredno od imetnika avtorskih pravic.

<https://creativecommons.org/licenses/by/4.0/>

CIP - Kataložni zapis o publikaciji  
Univerzitetna knjižnica Maribor

007.5:681.5(0.034.2)

LETNA konferenca Mehatronike (10 ; 2021 ; Maribor)  
10. letna konferenca mehatronike 2021 [Elektronski vir] : zbornik  
povzetkov študentskih projektov / uredniki Janez Pogorelc, Aleš Hace, Uroš  
Župerl. - 1. izd. - E-knjiga. - Maribor : Univerzitetna založba Univerze, 2021

Način dostopa (URL) : <https://press.um.si/index.php/ump/catalog/book/596>  
ISBN 978-961-286-504-7 (PDF)  
doi: 10.18690/978-961-286-504-7  
1. Drugi var. nasl. 2. Pogorelc, Janez  
COBISS.SI-ID 72068099

**ISBN** 978-961-286-504-7 (pdf)

**DOI** <https://doi.org/10.18690/978-961-286-504-7>

**Cena**  
*Price* Brezplačni izvod

**Odgovorna oseba**  
**založnika**  
*For publisher* prof. dr. Zdravko Kačič,  
rektor Univerze v Mariboru

**Citiranje**  
*Attribution* Pogorelc, J., Hace, A. in Župerl, U. (ur.) (2021). *10. letna konferenca mehatronike 2021: zbornik povzetkov študentskih projektov*. Maribor: Univerzitetna založba. doi: 10.18690/978-961-286-504-7



## Kazalo

<b>VS MEHATRONIKA</b>	<b>1</b>
<b>Obnova tračnega transporterja</b> Jaka Černec, Denis Marš, Jure Golavšek, Matic Prah, Nik Vitez in Andraž Požar	<b>3</b>
<b>RobotStudio – Demo Exhaust Pipe</b> Blaž Grobelnik, Patrik Jakopiček, Tilen Račnik in Kristjan Mihael Reisman	<b>5</b>
<b>RobotStudio – Demo Solar Panel</b> Darjan Štajnmeč in Blaž Pečenik	<b>7</b>
<b>Vodenje virtualnih industrijskih procesov – vrtalni stroj</b> Avgust Sakelšek in Žiga Štern	<b>9</b>
<b>Vodenje virtualnih industrijskih procesov – razvrstitev zabojev</b> Jan Soto Vargas in Rok Pulko	<b>11</b>
<b>Vodenje virtualnih industrijskih procesov - Sortiranje škatel</b> Anej Loboda, Žiga Markovič in Viktorija Hriberšek Pocajt	<b>13</b>
<b>Laserska gravirna naprava</b> Anže Krevh, Anej Rupnik in Nik Senekovič	<b>15</b>
<b>Motoriziran sledilni mehanizem za solarne panele</b> Aleksander Breznik, Dejan Rezec in Matej Mayer	<b>17</b>
<b>Prometni semafor za pešce in kolesarje z animacijo gibanja</b> Anej Loboda, Žiga Markovič in Viktorija Hriberšek Pocajt	<b>19</b>
<b>Brezdotikalno proženje semaforja za pešce in kolesarje</b> Jan Soto Vargas, Rok Pulko in Matic Prah	<b>21</b>
<b>Razvoj in izdelava krmiljenja avtomatizirane sklopke na Formuli Student</b> Denis Marš, Jure Golavšek in Jaka Černec	<b>23</b>
<b>RobotStudio – Fizično modeliranje kablov</b> Kristjan Mihael Reisman, Nik Senekovič in Anže Krevh	<b>25</b>
<b>RobotStudio – Vgrajen sistem optične zaznave objektov</b> Blaž Grobelnik, Patrik Jakopiček in Tilen Račnik	<b>27</b>

---

<b>Uporaba programa Scilab/Matlab pri Regulacijski tehniki 1</b> Žan Šilak in Darjan Štajnmec	29
<b>Pospeškometer in krmiljenje tekočega traku</b> Avgust Sakelšek, Blaž Pečenik in Matej Mayer	31
<b>Izdelava merilnika mikro razdalje na osnovi nizkokoherenčnega Michelsonovega interferometra</b> Nik Vitez, Andraž Požar in Dejan Rezec	33
<b>Naprava za nadzorovano polaganje optičnega vlakna</b> Aleksander Breznik, Anej Rupnik in Žiga Štern	35
<b>UN MEHATRONIKA</b>	37
<b>Izgradnja in vodenje mobilnega robota</b> Žan Janežič, Rok Knupleš, Štefan Jakob Štrucl in Žiga Lorger	39
<b>Graviranje z laserskim izvorom na koordinatni mizi</b> Zoran Uran in Jaka Štampar	41
<b>CNC rezkalnik</b> Andrej Walter in Jakob Horvat	43
<b>RobotStudio-IRB 1200: WorldZones</b> Matic Brunec, Urban Kolman in Matic Smogavec	45
<b>RobotStudio-IRB 1200 Soft Servo</b> Urban Naveršnik, Liza Škulj in Luka Šipek	47
<b>Nadgradnja proizvodnje linije s phoenixcontact krmilnikom</b> Taja Pec, Zala Slapnik in Jure Stošič	49
<b>Razvoj naprave za mikro erozijo stekla</b> Matic Markovič in Ana Gregor	51
<b>Realizacija delovne postaje Mitsubishi Electric</b> Matic Zelenik, Miha Vunderl in Antonio Hulec	53
<b>Razvoj in izdelava variabilne cevi na agregatu Formula Student</b> Miha Hržič in Žan Čakš	55
<b>Modeli letal in dronov v orodju AVL cruiseM</b> David Vide in Aleksa Todorović	57
<b>Razvoj mikro-fluidnih motorjev na osnovi lončka in stebrička</b> Rudolf-Leon Filip, Gregor Korže in Luka Sakač	59
<b>Integracija naprednih sistemov vožnje v avtonomni kmetijski robot Farmbeast</b> Gregor Popič in Domen Toš	61

---

<b>Industrijski robot in ToF kamera</b> Primož Kobale in Žiga Pajtler	63
<b>Intuitivno programiranje robota s pomočjo ToF kamere</b> Miha Artič in Žiga Štern	65
<b>Optično zajemanje površine kosov s kolaborativnim robotom</b> Anže Ros	67
<b>Nadgradnja proizvodnje linije z Beckhoff krmilnikom</b> Dominik Hraštnik	69
<b>MAG MEHATRONIKA</b>	71
<b>Robot NAO - Učenje prepoznavanja objektov in obrazov</b> Martin Cokan in Mihael Taužič	73
<b>Vodenje brezpilotnega zrakoplova</b> Kristijan Polovič	75
<b>Trajnostno vzdržljivostno testiranje elektro hidravličnega aktuatorja</b> Aleks Petrovič in Mihael Janežič	77
<b>Razvoj telemetrije krmilne enote MOTEC in komunikacije z voznikom Formule Student</b> Tadej Mauko, Tine Masilo in Nick Drobec	79
<b>Snovanje krmilnika sistema gorivnih celic v električnem vozilu</b> Goran Mundar, Filip Mehmetović in Žiga Balajc	81
<b>Verification of electric vehicles fuel cell controller models</b> Essam Hashem	83
<b>Razvoj sistema mikro-fluidnega motorja: Vpliv parametrov mehanskega ojačevalnika na ojačenje pomikov</b> Matjaž Malok in Marko Dukarič	85
<b>Kooperacija robot-robot s kolaborativnimi roboti</b> Anton Mataj, Vid Ostroško in Dimitrije Prelević	87
<b>Učenje kolaborativnega robota z demonstracijo preko kamere Kinect</b> Tomo Pšeničnik in Nejc Tovornik	89
<b>Preučevanje delovanja HASEL umetne mišice in možnost realizacije</b> Matija Ramšak, Gregor Oset in Kristjan Fišer	91
<b>Razvoj sistema za interaktivno upravljanje procesov montaže</b> Uroš Vincetič	93





VS

**MEHATRONIKA**



# OBNOVA TRAČNEGA TRANSPORTERJA

JAKA ČERNEC, DENIS MARŠ, JURE GOLAVŠEK, MATIC PRAH, NIK VITEZ IN  
ANDRAŽ POŽAR  
*2. letnik, Projekt I*

*Mentor: izr. prof. dr. Karl Gotlib  
(Univerza v Mariboru, Fakulteta za strojništvo)*

## Povzetek

Pri projektu I smo bili zadolženi za obnovo in nadgradnjo tračnega transporterja v Laboratoriju za robotiko. Naš uspeh je temeljil na izbiri ustreznih komponent in osvojitve programa TIA-Portal.

Projekt smo začeli z določitvijo frekvenčnega pretvornika, ki je bil kompatibilen z našim elektromotorjem in s krmilnikom. Izbrali smo frekvenčni pretvornik **DANFOSS-FC 302**, ki je imel možnost povezave s krmilnikom preko **ProfiNET** komunikacije. Univerza nam je prispevala Siemensov krmilnik, ki predstavlja možgane našega sistema. Ta uporablja programsko opremo **TIA-Portal**, ki nam omogoča preko **ProfiNET**-a virtualno povezan sistem. Ko smo imeli vse komponente v programu povezane, smo lahko začeli programiranje. Sekvenca programa je naslednja: če je obdelovanec na začetnem položaju (zalogovnik), ga robotska roka prime in prenese v CNC obdelovalni stroj. Po končani obdelavi se robotska roka ponovno pomakne za prijem obdelovanca ter ga prenese na transportni trak. Senzor na transportnem traku to zazna in požene tekoči trak. Na koncu traka izdelek pade v embalažo in potuje na nadaljnjo obdelavo.

Zaradi trenutnih razmer nam ni uspelo v resnici izvesti projekta, smo pa sestavili dobro podlago, na kateri bodo lahko prihodnje generacije dalje razvijale ta sistem. V poročilu smo navedli vse potrebne podatke o posameznih komponentah, električno shemo celotnega sistema z opisom, osnove programiranja v TIA-Portal-u in opis delovanja našega programa.

**Ključne besede:** TIA-Portal, frekvenčni pretvornik, krmilnik



# ROBOTSTUDIO – DEMO EXHAUST PIPE

BLAŽ GROBELNIK, PATRIK JAKOPIČEK, TILEN RAČNIK IN  
KRISTJAN MIHAEL REISMAN  
*2. letnik, Projekt I*

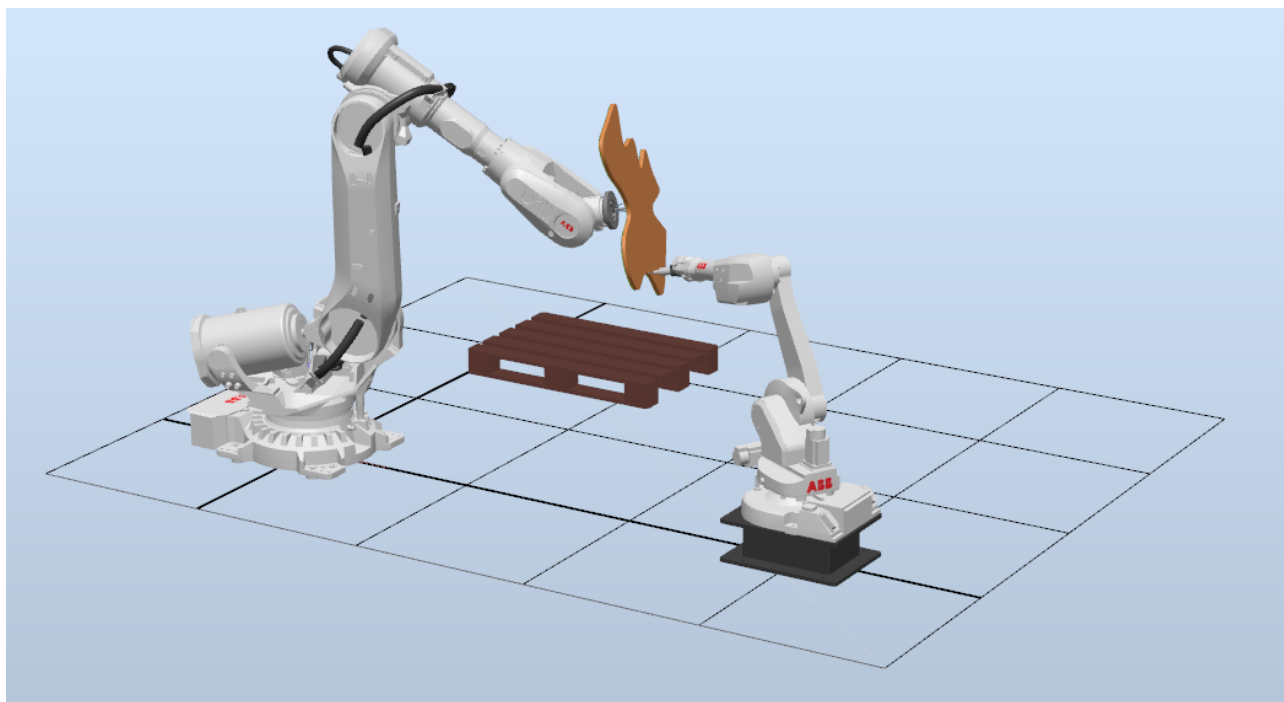
*Mentorja: izr. prof. dr. Karl Gotlih, asist. dr. Timi Karner  
(Univerza v Mariboru, Fakulteta za strojništvo)*

## **Povzetek**

Cilj našega projekta je bila izdelava robotske celice, znotraj katere bi uporabili dva robota, ki bi med sabo sodelovala. Naša naloga je bila, da dokumentiramo postopek izdelave robotske celice in napišemo navodila, kako se uporabljajo ukazi za programiranje več robotov, ki med seboj sodelujejo.

Projekta smo se lotili s preučevanjem že izdelane demo robotske celice »Demo Exhaust Pipe«, ki je na voljo znotraj programskega paketa RobotStudio. Iz demo celice smo ugotovili, da se ukazi ne pišejo ročno, ampak se generirajo samodejno, potem ko uporabimo želene funkcije. Za naš primer smo uporabili funkcijo »Multi Move«, ki smo jo morali naknadno prenesti. Izbrali smo si dva robota in ju vstavili v celico.

Odločili smo se, da bomo robota uporabili v aplikaciji varjenja kompleksnejšega obdelovanca in tako je eden izmed robotov varil, drugi pa je obračal obdelovanec. Delovanje robotov smo preizkušali v simulaciji in parametre funkcije »Multi Move« nastavljali tako dolgo, da smo dobili zadovoljiv rezultat. Celoten postopek kreiranja takšne celice in težave, na katere lahko naletimo, smo na koncu opisali v navodilih.



**Slika 1: Simulacija robotske celice**  
(vir: lasten)

**Ključne besede:** RobotStudio, robotika, celica, sinhrono gibanje

# ROBOTSTUDIO – DEMO SOLAR PANEL

DARJAN ŠTAJNMEC IN BLAŽ PEČENIK

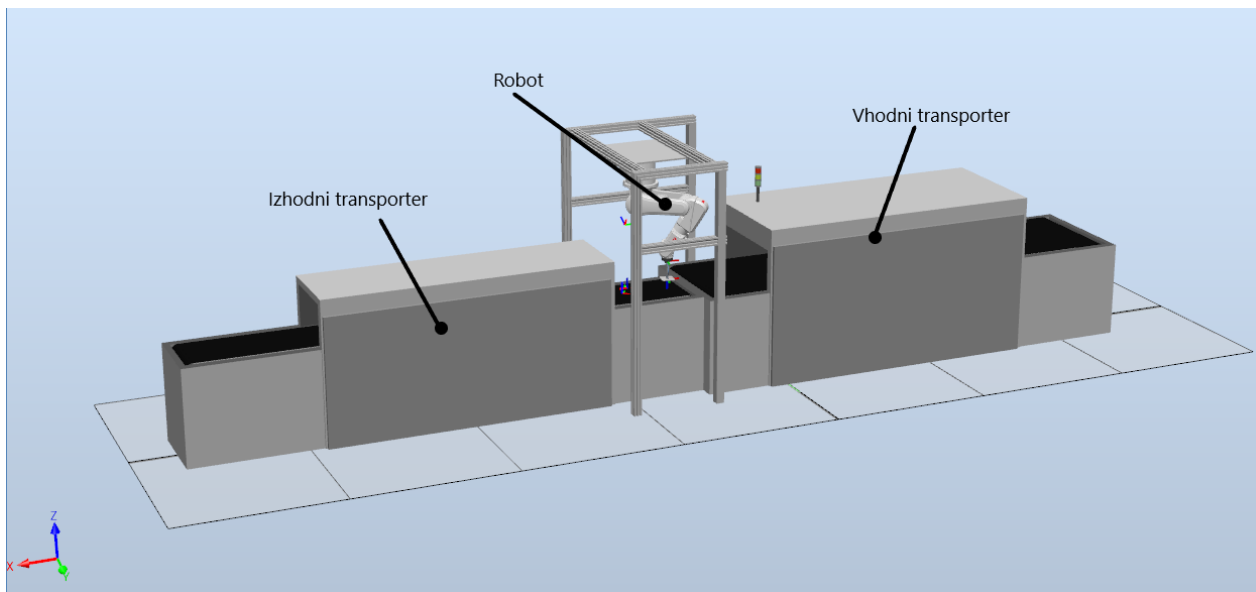
*2. letnik, Projekt I*

*Mentorja: izr. prof. dr. Karl Gotlib, asist. dr. Timi Karner  
(Univerza v Mariboru, Fakulteta za Strojništvo)*

## Povzetek

Pri Projektu I sva morala zgraditi robotsko celico z dvema transporterjema in eno robotsko roko, ki je prenašala solarno celico iz vhodnega na izhodni transporter. To pa ni bilo tako enostavno, saj so se solarne celice generirale v vhodnem transporterju z naključno orientacijo med  $\pm 45^\circ$  in naključnim položajem v y-smeri med stranicama vhodnega transporterja. Na začetku sva se lotila naloge tako, da sva vse komponente (robotska roka, vhodni/izhodni transporter) postavila na delovno mesto. Pri tem sva naletela na težavo, saj nisva vedela, kako morava pravilno obrniti robota, kar sva rešila s poizkušanjem. Ker so se solarne celice generirale naključno, sva tudi tukaj naletela na oviro, vendar sva jo rešila tako, da sva napisala dva programa; eden izmed njiju je bil »main« oz. glavni program, v katerem sva določila koordinate vseh točk in napisala del programa za pobiranje in odlaganje panel na izhodni transporter iz vhodnega. Vzporedno pa se je izvajal drugi program, s katerim sva na podlagi generiranega položaja panela dobila vrednosti orientacije in lego položaja po y-osi, ki sva jo lahko pobrala in prenesla na izhodni transporter. Ta dva programa sta ves čas delovala vzporedno, saj drugače ne bi pobiralo izdelkov. Z x-osjo nisva imela težav, saj se je vhodni transporter ustavil, ko je prišel izdelek do sensorja, kjer ga je robotska roka tudi pobrala in ga odnesla na izhodni transporter. Na izhodni transporter sva dala tri sensorje, enega po širini, da zazna kos, drugega po dolžini transporterja, da se je izdelek lahko peljal do konca in tretjega na koncu, kjer sva imela posebno funkcijo vezano na senzor. Tej funkciji se reče »sink«, ki omogoča, da komponente izginejo; v proizvodnji bi to upravljala druga robotska roka, ali pa delavec. Nič od naštetega pa ne bi delovalo, če ne bi pravilno povezala logike, to so pravilno povezani vhodi oz. izhodi na krmilniku, ki jih dobimo iz sensorjev ali kakšne druge funkcije. Ustvariti sva morala sedem novih signalov in jih pravilno povezati, ki so omogočali pravilno delovanje med vhodnim transporterjem, robotsko roko in

vakuumskim prijemalom, saj ti deli neposredno sodelujejo drug z drugim. Logiko za izhodni transporter sva naredila posebej, ker je bolj enostavno in hitreje. Za ta del nisva ustvarjala novih signalov, vendar sva senzorje, ki sva jih dodala, neposredno vključila v izhodni transporter in nato samo pravilno povezala. Pri tem sva prej omenjeno funkcijo »sink« povezala s senzorjem, da izdelki izginjajo, ko pridejo do določene lege po x-osi. Seveda ni vse delovalo že v prvem poizkusu, zato sva porabila veliko časa za testiranje programske kode, logike krmilnika in pravilne nastavitve senzorjev.



**Slika 1: Slika robotske celice**

Vir: lasten

**Ključne besede:** RobotStudio, robotska celica, paralelno izvajanje programa, logika



# VODENJE VIRTUALNIH INDUSTRIJSKIH PROCESOV – VRTALNI STROJ

AVGUST SAKELŠEK IN ŽIGA ŠTERN

*2. letnik, Projekt I*

*Mentor: mag. Janez Pogorelc*

*(Univerza v Mariboru, Fakulteta za elektrotehniko, računalništvo in informatiko)*

## Povzetek

Glavni namen naloge je bil izvesti in zgraditi program za avtomatsko vrtnanje vrtalnika v obdelovanec. Pri tem sva imela na voljo računalnik z oddaljenim dostopom, na katerega je bil priključen industrijski PLC krmilnik **Simatic S7-1200**, kakor je tudi nameščen program **TIA Portal** za upravljanje s tem krmilnikom. PLC krmilnik je bil povezan s **Phidget Interface Kit** pomožno kartico preko digitalnih vhodov in izhodov, katerih stanja so se sproti prikazovala na programu **Machines Simulator**, ki je bil že vnaprej pripravljen. Stanja in potek so se prikazovala na grafičnem zaslonu v obliki 3D animirane simulacije. Program smo kodirali v koračnem načinu in v skladu z zahtevami posamezne naloge.

Zagon sistema se začne s tipko »**Start**«, takrat se iz zalogovnika delov program spusti prvi obdelovanec na delovno površino. Ko je del na delovni površini, ga zazna prva fotocelica, ki sproži pomik prvega cilindra, ta pa opravi premik obdelovanca pod sam vrtalni stroj. Cilinder ima dva končna stikala, z drugim zazna, ko se cilinder do konca pomakne in ga nato vrne v prvotni položaj, s prvim prekine premik cilindra nazaj.

Nato sledi postopek vrtnanja v obdelovanec. Ko je enkrat obdelovanec pod samim vrtalnim strojem, ga zazna druga fotocelica, ki sproži vrtnenje vrtalnika in njegov pomik do obdelovanca, ter tudi skozi obdelovanec. Tudi vrtalnik ima dve končni stikali. Drugo končno zazna, ko je vrtalnik v svoji najnižji točki, torej ko je že naredil izvrtino, da ga vrne v prvotni položaj. Prvo končno stikalo pa služi za izklop samega vrtenja na vrtalnem stroju in hkrati sproži pomik drugega cilindra.

Drugi cilindar se nato premakne in izpodrine obdelovanec iz delovne površine v linijo, kjer se kopičijo obdelovanci drug za drugim. Do pet obdelovancev se lahko nakopiči na tej liniji, šesti obdelovanec pa potem že pade v zaboj s končanimi obdelovanci. Tako kot tudi prvi cilindar za pomik ima tudi drugi cilindar dve končni stikali. Drugo služi za izklop pomika cilindra in vrnitev v njegov prvotni položaj. Prvo pa služi za ponovno aktiviranje programa, torej za spust novega dela iz zalogovnika delov. Tako se postopek ponovi.

S tipko »**Stop**« se lahko program prekine v kateri koli fazi. Tako se vsi elementi vrnejo na svoja prvotna mesta in se potem sprožijo ob ponovnem pritisku tipke »**Start**«. Program se je tako nadaljeval izvajati tam, ko je ostal del oz. obdelovanec.

Naloga je bila v celoti uspešno izvedena s pomočjo oddaljenega dostopa na računalnik v laboratoriju, saj nam epidemiološke razmere niso dopuščale obiska fakultete in izvedbe naloge v živo.

**Ključne besede:** vrtalni stroj, TIA Portal, Machines Simulator, Simatic S7-1200, izvajanje programa, sprožitelj dejanj, koračni način

# VODENJE VIRTUALNIH INDUSTRIJSKIH PROCESOV – RAZVRSTITEV ZABOJEV

JAN SOTO VARGAS IN ROK PULKO

*2. letnik, Projekt I*

*Mentor: mag. Janez Pogorelc*

*(Univerza v Mariboru, Fakulteta za elektrotehniko, računalništvo in informatiko)*

## Povzetek

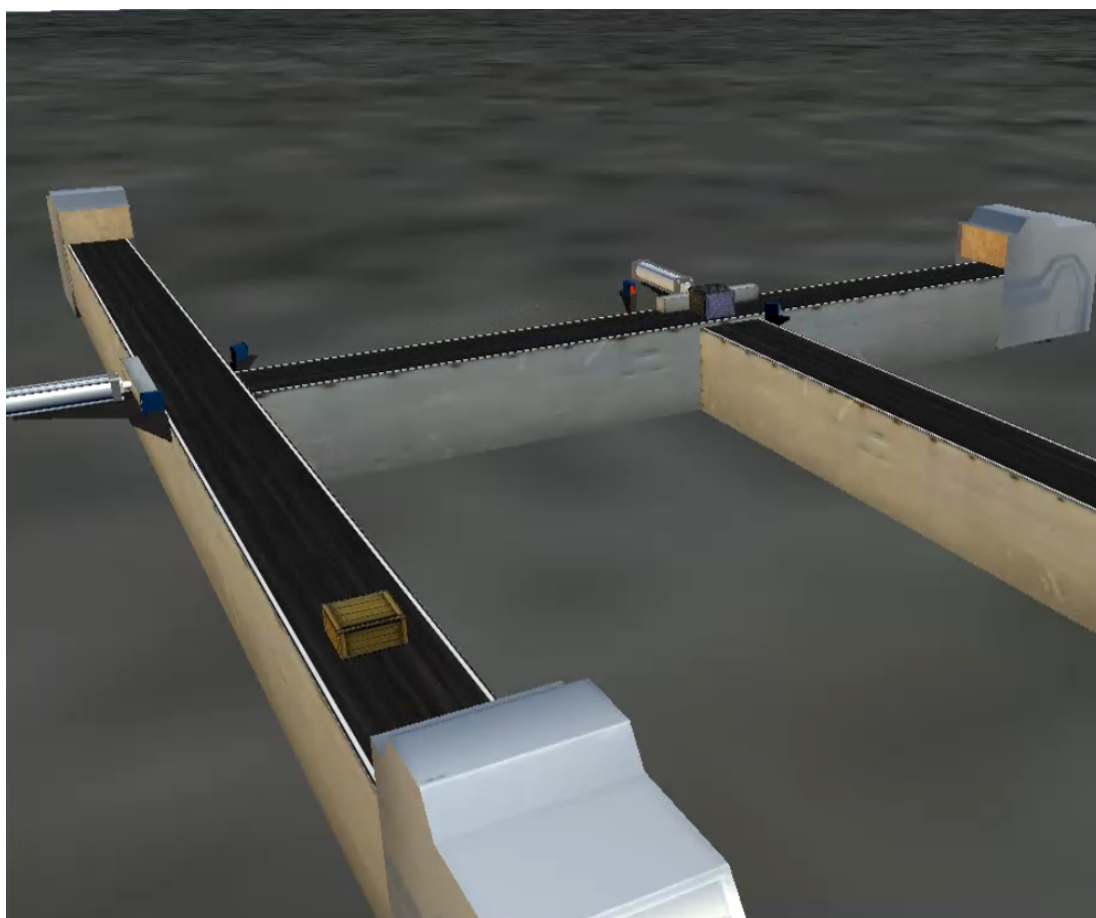
Potrebno je bilo avtomatizirati razvrščanje treh zabojev različnih velikosti v ustrezne zbiralnike glede na njihovo velikost.

Program sva razvijala v programskem okolju **TIA Portal** s pomočjo industrijskega PLK krmilnika Siemens **Simatic S7-1200**. Zaradi dela na daljavo sva za testiranje algoritma krmilja uporabila **PhidgetInterfaceKit** vmesno kartico in program **Machines Simulator**, v katerem se je animirana 3D simulacija izvajala v realnem času, kar nama je omogočilo uspešno nadomestitev procesa. Program je prikazoval tudi stanja vhodov in izhodov.

Glede na podane zahteve so se ob pritisku na tipko »Start« vključili vsi tekoči trakovi. Po prvem traku so začeli prihajati zaboji različnih velikosti. Ko so zaboji prispeli do senzorjev, so jih ti zaznali in glede na njihovo velikost spustili naprej po prvem traku (mali zaboji) do zbiralnika malih zabojev ali jih s pomočjo pehala potisnili na drugi trak (srednji in veliki zaboji). Zaboji, ki so bili prestavljeni na drugi trak, so potovali do naslednjih senzorjev, ki so zaznali razliko v velikosti med velikimi in srednjimi zaboji ter s pomočjo pehala potisnili velike zaboje na tretji trak, ki vodi do zbiralnika velikih zabojev. Srednji zaboji so potovali po drugem traku v zbiralnik srednjih zabojev. Proces lahko kadarkoli ustavimo s »Stop« tipko.

Kot dodatek osnovnim zahtevam sva izvedla še štetje zabojev, ki je potekalo s pomočjo števecv CTU. S pomočjo informacij tretjega senzorja sva štela velike in srednje zaboje, s četrtem samo velike zaboje ter s prvim vse zaboje. Z dodatnimi izračuni sva pridobila informacijo o številu ostalih zabojev.

Z rezultatom naloge sva zadovoljna, saj sva dosegla, da avtomatizacija procesa deluje v skladu s podanimi zahtevami. Posebej sva ponosna na najin dodatek - štetje zabojev. Zaradi razmer sva morala projekt izdelati na oddaljeni način, preko dostopa do računalnikov z ustrezno opremo v laboratoriju.



**Slika 1: Razvrstitev zabojev**

Vir: lasten

**Ključne besede:** Simatic S7-1200, Tia Portal, Machines Simulator, PhidgetInterfaceKit, tekoči trak, senzorji, razvrščanje

# VODENJE VIRTUALNIH INDUSTRIJSKIH PROCESOV - SORTIRANJE ŠKATEL

ANEJ LOBODA, ŽIGA MARKOVIČ IN VIKTORIJA HRIBERŠEK POCAJT

*2. letnik, Projekt I*

*Mentor: mag. Janez Pogorelc*

*(Univerza v Mariboru, Fakulteta za elektrotehniko, računalništvo in informatiko)*

## Povzetek

Cilj naloge je avtomatsko sortiranje treh tipov škatel, ki se ločijo po barvi: rumena, črna in siva. Na voljo smo imeli računalnik z oddaljenim dostopom, na katerega ja bil priključen industrijski PLC krmilnik **Simatic S7-1200** in nameščen program **TIA Portal**. PLC krmilnik je bil preko digitalnih vhodov in izhodov povezan s **Phidget Interface Kit** pomožno kartico, katere stanja so se sproti prikazovala s pomočjo vnaprej pripravljene aplikacije programa **Machines Simulator** na grafičnem zaslonu v obliki 3D animirane simulacije. V skladu s podanimi zahtevami smo kodirali PLC program v načini »**Ladder**« in ga interaktivno testirali s programom **Machines Simulator**.

Ob zagonu sistema, se ob pritisku na tipko »**Start**«, na glavnem traku pojavi škatla. Tip škatle določi kamera pri prvem sensorju. S določitvijo tipa škatle se nato sistem pomika v skladu z algoritmom programa.

Algoritem kamere izvede optično obdelavo, pri čemer zazna za kakšen tip škatle gre, čitalnik pa aktivira dva izhoda (Camera Out 0 in Camera Out 1). Če gre za rumeno škatlo se aktivira izhod »**Camera Out 0**«, če gre za črno »**Camera Out 1**«, za ter tretji tip škatle, pa sta oba izhoda aktivna.

V primeru, da gre za rumeno škatlo, ki se prva sortira na traku, jo mora zaznati drugi senzor, ki je postavljen vzporedno z glavnim in stranskim trakom. Ko senzor zazna škatlo, pošlje ukaz za izvlek prvega cilindra, ki škatlo potisne na stranski trak. Takoj po izvleku, se ta spet povrne in pošlje ukaz za prižig spodnjih trakov, ki nato škatlo dokončno

sortirajo. Podoben postopek je za črno škatlo oz. škatlo, ki se druga sortira. Ko škatlo zazna 3. senzor, se izvleče drugi cilindar in postopek se spet ponovi kot pri rumeni škatli.

Postopek za sortiranje sive škatle je nekoliko drugačen, saj škatla potuje ves čas po glavnem traku, dokler ne pade na spodnji sortirni trak.

Celoten sistem lahko kadar koli zaustavimo s pritiskom na tipko »**Stop**«. Ker je naloga od nas zahtevala tudi uporabo svetlobnih indikatorjev, smo si zamislili, da rdeča luč sveti tako dolgo, dokler škatla ne pade na ustrezen spodnji sortirni trak. Ko se škatla »pravilno« sortira, se prižge zelena luč. Če se škatla pojavi na napačnem sortirnem traku, se vklopi sirena in rotacijska luč.

Naloga je bila uspešno izvedena z oddaljenim dostopom do računalnika v laboratoriju, saj nam epidemiološke razmere niso dopuščale obiska fakultete in izvedbe nalog v živo.

**Ključne besede:** sortiranje, škatle, logične operacije, sortirni traki, Simatic S7-1200, TIA Portal, Machines Simulator

# LASERSKA GRAVIRNA NAPRAVA

ANŽE KREVIH, ANEJ RUPNIK IN NIK SENEKOVIČ  
*2. letnik, Projekt I*

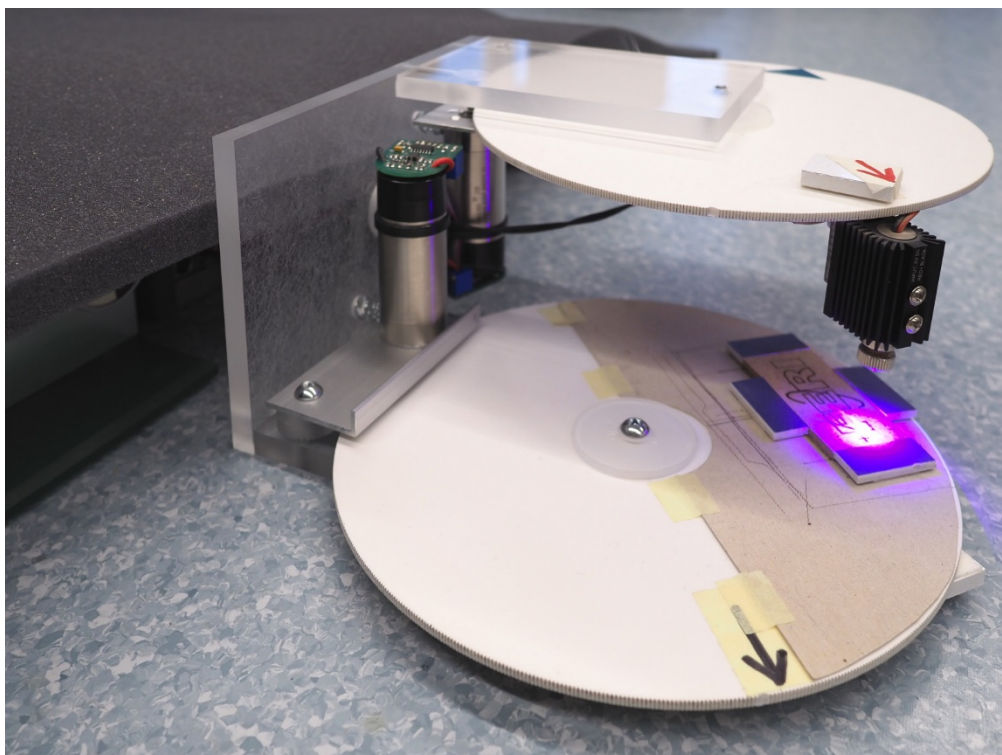
*Mentorja: mag. Marijan Španer, mag. Janez Pogorelc  
(Univerza v Mariboru, Fakulteta za elektrotehniko, računalništvo in informatiko)*

## Povzetek

Pri projektnejem delu smo si skupina treh študentov za izziv pri predmetu Projekt I izbrali izdelavo laserske gravirne naprave. Zasnovali smo inovativni mehanizem, ki uporablja dve rotacijski gibanji; prva rotacijska os premika obdelovanec, druga os pa v prečni smeri premika vir laserske svetlobe (slika). Čeprav sta gibanji po obeh oseh rotacijski in posledično nelinearni, je s pomočjo kinematične transformacije izvedeno gibanje mehanizma v zunanjih koordinatah, oziroma v kartezičnem koordinatnem sistemu. Generiranje trajektorij gibanja, položajno vodenje obeh servomotorjev in sprotno preračunavanje kinematične transformacije je izvedeno z industrijskim krmilnikom gibanja (**Motion Controller TRIO**).

Pri načrtovanju zasnove mehanizma smo se ravnali po načelih: preprosto, inovativno, učinkovito ter preprosto za izdelavo. Obe osi sta identični in izvedeni na osnovi rotacijske plošče premera 200 mm z zunanjim cikloidnim ozobljenjem ( $m=0,3$ ), ki jo preko drugega zobnika poganja servomotor. Za zmanjšanje neželenih vplivov zračnosti in trenja je pogonski motor vpet premično, potrebno kontaktno silo med zobnikoma pa določa pritisk spiralne vzmeti.

V času projektne delo pri predmetu smo pridobili mnogo novih znanj in veščin: spoznali smo postopek načrtovanja kompleksne naprave, delovanja električnih servopogonov, regulacijske tehnike pri pozicionirnih napravah, uporabe kinematične transformacije SCARA robota, uporabe polprevodniškega laserja za namen graviranja, kot tudi uspešne integracije vseh komponent v delujočo napravo ter oblikovanja in same izdelave produkta.



**Laserska gravirna naprava**

Vir: lasten

**Ključne besede:** dvoosni mehanizem, servopogon, kinematika, laser, graviranje



# MOTORIZIRAN SLEDILNI MEHANIZEM ZA SOLARNE PANELE

ALEKSANDER BREZNIK, DEJAN REZEC IN MATEJ MAYER

*2. letnik, Projekt I*

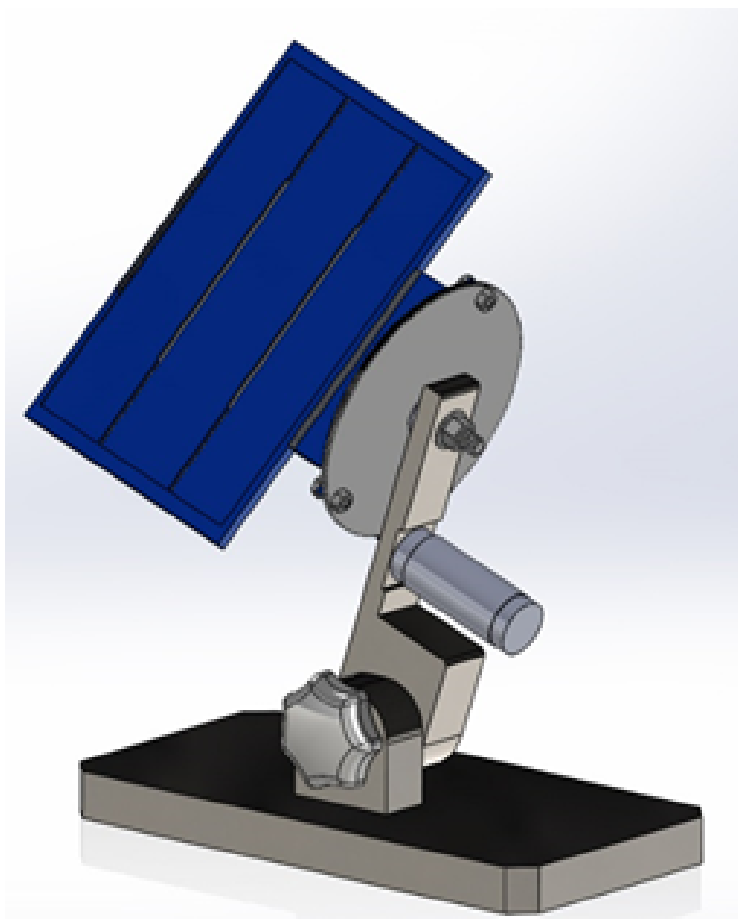
*Mentorja: mag. Marijan Španer, mag. Janez Pogorelc  
(Univerza v Mariboru, Fakulteta za elektrotehniko, računalništvo in informatiko)*

## Povzetek

Ob začetku naše naloge smo morali izvedeti nekaj več o gibanju Zemlje okoli Sonca in okoli svoje osi ter kakšno je navidezno premikanje sonca po našem nebu. Ugotovili smo, da je urni kot ob sončnem poldnevu vedno  $0^\circ$  in se vsako uro spremeni za  $15^\circ$ . Tako smo se odločili, da bomo sledilni mehanizem pripravili tako, da ne sledi soncu vsak trenutek, ampak se bo vsakih 15 minut premaknil na najbolj optimalno pozicijo. Na ta način smo dosegli optimalno razmerje med izkoristkom in porabo električne energije.

Na osnovi podatkov s spleta smo naredili tabelo za vsak mesec in izračunali povprečno uro sončnega vzhoda in zahoda, kar smo uporabili pri programu algoritma v krmilniku **TRIO MC 403**. Le-ta opravlja korekcije panela glede na mesec, dan in uro tako, da doseže najboljši izkoristek. Na krmilnik smo se z računalnikom povezali s pomočjo povezave Ethernet in napisali program v **Motion Perfect 4**, ki je prilagojen za programiranje TRIO krmilnikov. Na krmilnik smo še priklopili sistem s servo pogonom, ga preizkusili, in glede na naš "model" kalibrirali. Celotno konstrukcijo in simulacijo delovanja smo izvedli v programu Solidworks. Na žalost zaradi razmer epidemije projekta nismo naredili do te faze, da bi panel lahko testirali fizično na konstrukciji.

Pri projektu smo uporabili in dopolnili znanja iz nekaterih šolskih predmetov, ki so: Računalniško podprto inženirstvo, Gradnja mehatronskih sistemov, Regulacijska tehnika in Računalništvo.



**Slika 1: 3D model**

Vir: lasten

**Ključne besede:** solarni sledilnik, enosni solarni sledilnik, 3D model, krmilnik, obnovljivi viri energije, sončna elektrarna

# PROMETNI SEMAFOR ZA PEŠCE IN KOLESARJE Z ANIMACIJO GIBANJA

ANEJ LOBODA, ŽIGA MARKOVIČ IN VIKTORIJA HRIBERŠEK POCAJT  
*2. letnik, Projekt II*

*Mentorja: red. prof. dr. Darko Lovrec (Univerza v Mariboru, Fakulteta za strojništvo), mag. Janez Pogorelc (Univerza v Mariboru, Fakulteta za elektrotehniko, računalništvo in informatiko)*

## Povzetek

Cilj projekta je zasnovati »bolj svežo« in učinkovitejšo signalizacijo, ki bi omogočala animacijo gibanja pešca. Prav tako smo poiskali možnost, kako to realizirati samo z eno lučjo namesto dveh, ter vključili način prilagajanja svetilnosti luči jakosti dnevne svetlobe. Semafor mora imeti enak premer luči kot sedaj uporabljeni semafor v signaliziranem križišču.

Pri projektu smo si pomagali s programskim okoljem **Arduino IDE** za zapis programa in **SolidWorks** za izris LED diod na plošči. Glede programskega dela oz. mikrokrmilnika, pa smo uporabili **Arduino UNO**.

Za izdelavo podlage semaforja smo uporabili leseno ploščo, ki smo jo s pomočjo laserskega rezalnika izrezali krog premera **210 mm**.

Za luč oz. osvetlitev semaforja smo uporabili **RGB LED trak**, z oznako **WS2812**, kar pomeni, da je programirljiva vsaka **LED dioda**.

Prav tako pa smo za napajanje celotnega sistema potrebovali **napajalnik**, in sicer **Alitove 5 V, 20 A, 100 W, AC 110/220V DC**, z oznako **WS2812**, kar pomeni, da z njim lahko napajamo **LED trak**, ki smo ga naročili.

Na leseno ploščo smo sprva namestili **LED diode**, da smo dobili simetričen izgled oz. razpodelitev **LED diod**. Te smo kasneje z lepilom tudi zalepili na leseno ploščo.

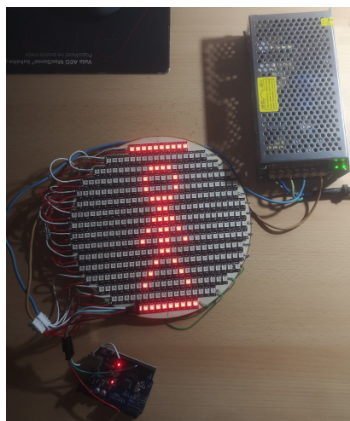
Sledila je vezava **LED diod, Arduina in napajalnika**, kjer smo iz **Arduino UNO** mikrokontrolerja povezali tri žice (+, - in signal) na **LED trak**, prav tako pa smo priključili napajanje **LED traku** na napajalnik, ki je vezan neposredno na omrežje. Vsako vrstico **LED diod** smo vezali zaporedno, pri čemer je vsaka dioda imela +, - in signal vodnik.

**Arduino** program smo razdelili na več delov in sicer; »**Stop**«, »**Test**«, »**Zelena luč**«, »**Premik**« in »**Celoten program**«. Program deluje tako, da se ob prižigu semaforja vklopi zelena luč, silhueta se premika iz desne proti levi strani, nad in pod silhueto pa je zelena črta. Proces traja eno minuto. Sledi menjava luči, v rdečo, pri čemer silhueta stoji pri miru, nad in pod silhueto pa je rdeča črta.

Spoznali smo številne vrste semaforjev po svetu, sodobne prikaze semaforjev za pešce in kolesarje, kot tudi za vozila. **Animirani prikazi silhuet** na semaforjih postajajo vse bolj popularen, saj privlačijo pozornost pešcev in kolesarjev, zmanjšajo nestrpnost med čakanjem na zeleno luč in s tem pa tudi zmanjšajo potencialne nesreče v prometu.

Pri predmetu Projekt II smo spoznali veliko novosti in pridobili znanje na področju programiranja, električnih vezav, dela s svetlobnimi elementi, itd. ... S predznanjem, ki smo ga nadgradili, in s skupinskim delom, je projekt uspel tako, kot smo si ga zamislili.

Naloga je bila uspešno izvedena z oddaljenim dostopom do računalnika v laboratoriju, saj nam epidemiološke razmere niso dopuščale obiska fakultete in izvedbe nalog v živo.



Slika 1: Prometni semafor za pešce in kolesarje z animacijo gibanja

Vir: lasten

**Ključne besede:** semafor, animacija, silhueta, Arduino, LED diode/trak, pešci in kolesarji

# BREZDOTIKALNO PROŽENJE SEMAFORJA ZA PEŠCE IN KOLESARJE

JAN SOTO VARGAS, ROK PULKO IN MATIC PRAH  
*2. letnik, Projekt II*

*Mentor: red. prof. dr. Darko Lovrec  
(Univerza v Mariboru, Fakulteta za strojništvo)*

## Povzetek

Potrebno je bilo izdelati sistem za brezdotikalno proženje semaforja za pešce in kolesarje. Nanizali smo si tri različne izvedbe in izbrali eno izmed njih. Izvedbe so bile: s kapacitivnim senzorjem, ultrazvočnim senzorjem in izvedba proženja semaforja s tipko integrirano v tla. Izbrali smo izvedbo z ultrazvočnim senzorjem.

Za začetek smo izbrali vse potrebne elemente za izvedbo. To je bil **Arduino UNO**, **ultrazvočni senzor**, ter **matrični prikazni modul**.

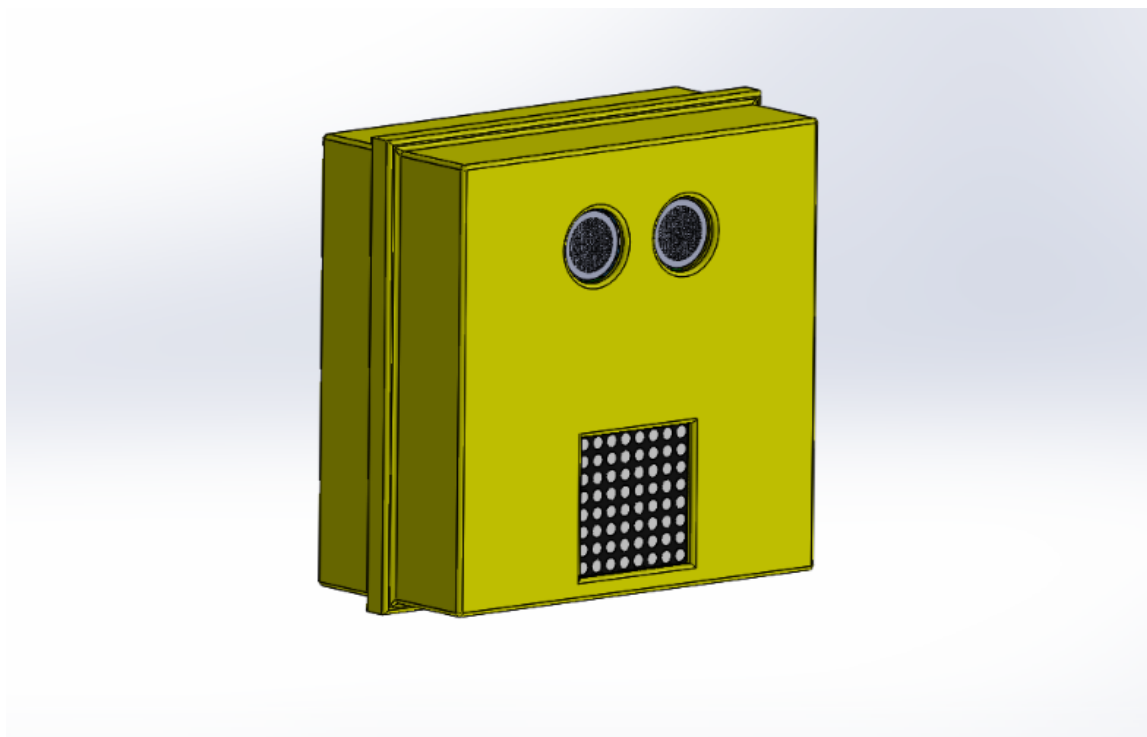
Program za celoten sistem smo razvijali v programskem okolju **Arduino IDE**, preko katerega smo sprogramirali delovanje senzorja in matričnega prikazovalnega modula.

Izdelali smo ohišje, ki je bilo modelirano v programu **SolidWorks**, in 3D tiskano. Za lažje razumevanje in uporabo pešcem in kolesarjem smo sistem izpopolnili s **piktogrami**.

Program oziroma celoten sistem deluje tako, da ultrazvočni senzor pošilja signal časa potovanja zvočnega vala, katerega smo spremenili v razdaljo s pomočjo enačbe. Izračunana razdalja se ves čas izpisuje na serijskem monitorju. Glavni del delovanja sistema je zaznavanje pešca oziroma kolesarja. To poteka na razdalji 40 – 50 cm od senzorja. Ultrazvočni senzor mora zaznavati pešca 2 s, da se program izvede. Ko je pešec zaznan se začne odštevanje časa na matričnem prikazovalnem modulu do zelene luči oziroma veselega »smeškota« ob katerem ima pešec prost prehod čez prehod za pešce. Po 10 s se prižge rdeča luč na semaforju in žalosten »smeško« na matričnem prikazovalnem

modulu, kar pomeni, da pešec ne sme prečkati prehoda za pešce. Ob pristnosti pešca oziroma kolesarja se program začne ponavljati.

Z rezultatom naloge smo zadovoljni, saj smo dosegli zastavljen cilj. Možne bi bile mnoge izboljšave, ampak glede na minimalne stroške izdelave in materiala je to idealna izvedba, primerna za uporabo tudi v praksi.



**Slika 1: Ohišje s komponentami**

Vir: Lasten

**Ključne besede:** ultrazvočni senzor, Arduino UNO, Matrični prikazni modul, Piktogrami, SolidWorks

# RAZVOJ IN IZDELAVA KRMILJENJA AVTOMATIZIRANE SKLOPKE NA FORMULI STUDENT

DENIS MARŠ, JURE GOLAVŠEK IN JAKA ČERNEC  
*2. letnik, Projekt II*

*Mentorja: izr. prof. dr. Vito Tič, red. prof. dr. Riko Šafarič  
(Univerza v Mariboru, Fakulteta za strojništvo)*

## **Povzetek**

Pri projektu II smo si izbrali razvoj in izdelavo avtomatske sklopke, ki bi jo implementirali na Formulo Student Maribor. Ob pričetku projekta smo se povezali s člani ekipe Formule Student, ki so nam dali navodila in napotke o delovanju sklopke.

Projekt smo razdelili na dva dela, pri čemer smo se v prvem semestru lotili načrtovanja strojnega dela, v drugem semestru pa električnega. Za načrtovanje strojnega dela smo uporabili CAD modeliranje v Solidworks in Catia programskem okolju. Zasnovali smo veliko prototipov ter možnosti izvedb, a smo se na koncu s pomočjo mentorja in ekipe Formula Student odločili za izvedbo z ekscentričnim krogom, pritrjenim na servo motor, kateri bo aktiviral sklopko. Po izboru servo motorja smo na podlagi njegovih dimenzij ter oblike prožnega bata, zmodelirali celotno ohišje z aktivatorjem. Nato smo se lotili električnega dela projekta, pri čemer smo sodelovali z delom ekipe, zadolžene za elektro sklop Formule Student, ki nam je zagotovila potrebno programsko opremo za izdelavo tiskanega vezja in krmilnega programa. Najprej smo se lotili tiskanega vezja, kjer smo morali sestaviti primerno vezje, ki bi lahko napajalo ter vodilo naš servo motor, nato smo pa še zasnovali program, ki bi ob pritisku gumba aktiviral sklopko.

Zaradi epidemioloških razmer smo imeli določene omejitve, saj nismo imeli fizičnega stika z ekipo Formule ter samo formulo. Zato smo projekt izpeljali do te mere, da je pripravljen za testiranje in optimizacijo delovanja.

**Ključne besede:** Formula student, sklopka, krmiljenje



# ROBOTSTUDIO – FIZIČNO MODELIRANJE

## KABLOV

KRISTJAN MIHAEL REISMAN, NIK SENEKOVIČ IN ANŽE KREVIH

*2. letnik, Projekt II*

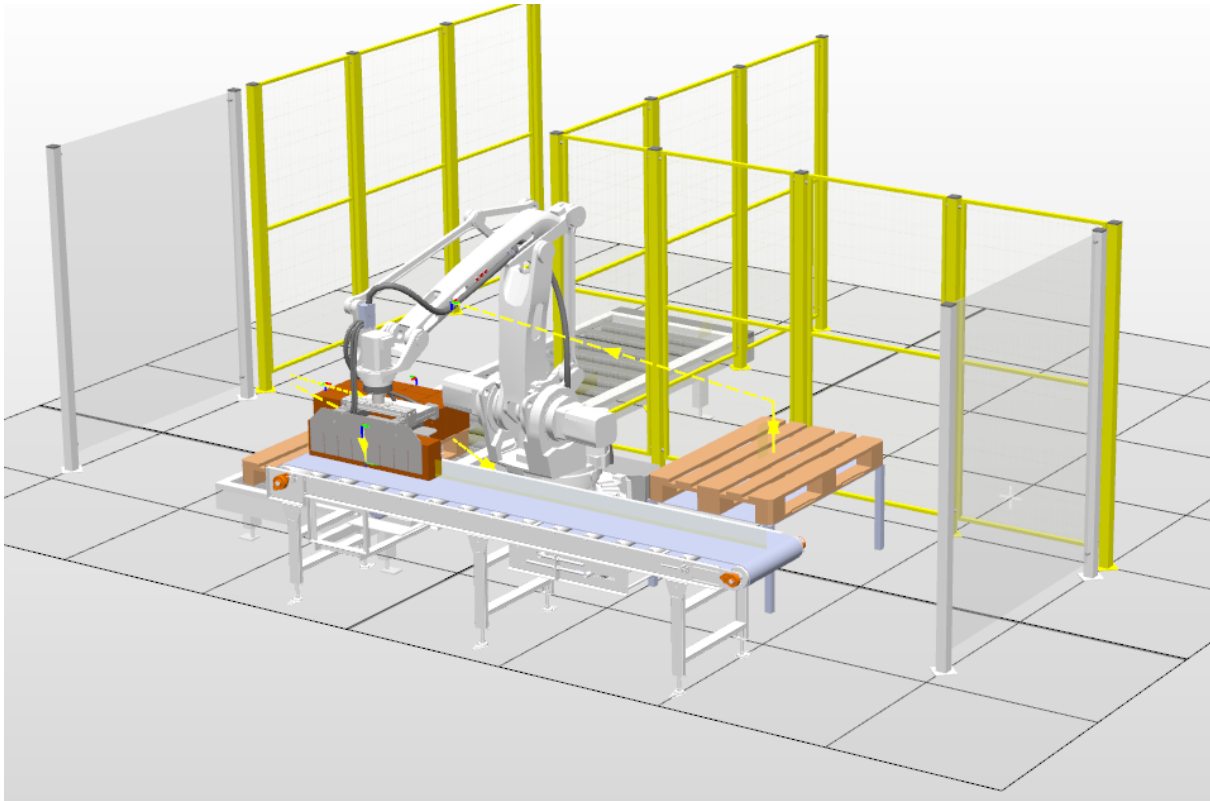
*Mentorja: izr. prof. dr. Karl Gotlib, asist. dr. Timi Karner (Univerza v Mariboru, Fakulteta za strojništvo)*

### **Povzetek**

Omenjeno nalogo smo izbrali, saj je med vsemi razpisanimi nalogami v nas vzbudila največ zanimanja. Želeli smo se namreč nekoliko bolje spoznati s programiranjem robotov.

Cilj projekta je bila izdelava fizičnih kablov oz. cevi na robotu, jim določiti primerne dimenzije, dodati komponente, ki bi v resničnem svetu služile za elektro-pnevmatično krmiljenje uporabljenega robota tipa IRB 460 in v vnaprej podani robotski celici s tekočim trakom in dvema paletama ustvariti program za pobiranje in nakladanje obdelovancev.

Projekta smo se lotili z uvozom CAD modela poprej omenjenega modela robota v program SolidWorks za 3D modeliranje, na katerem smo izdelali dva nosilca za glavno cev in škatlo v katero smo nato namestili CAD modela uporabljenih elektro-pnevmatičnih ventilov (5/2). Izdelana škatla ima en vhod za glavno cev (za električno in pnevmatično napajanje ventilov) in štiri izhode za pnevmatično krmiljenje prijemala. Po izdelavi smo vseh potrebnih komponent smo sam model robota tako rekoč skrili in v podano simulacijo v RobotStudio-u vnesli izdelane komponente in jih pritrdili na robota. Sledila je izdelava kablov oz. cevi s funkcijo »Cables«, čigar dolžino smo ustrezno prilagodili po izvedbi, ki je na paleti odlagal obdelovance, ki so bili v njegovo delovno območje pripeljani po tekočem traku.



**Slika 1 Simulacija v robotski celici**

Vir: lasten

**Ključne besede:** RobotStudio, robotika, celica, fizične cevi

# ROBOTSTUDIO – VGRAJEN SISTEM OPTIČNE ZAZNAVE OBJEKTOV

BLAŽ GROBELNIK, PATRIK JAKOPIČEK IN TILEN RAČNIK  
*2. letnik, Projekt II*

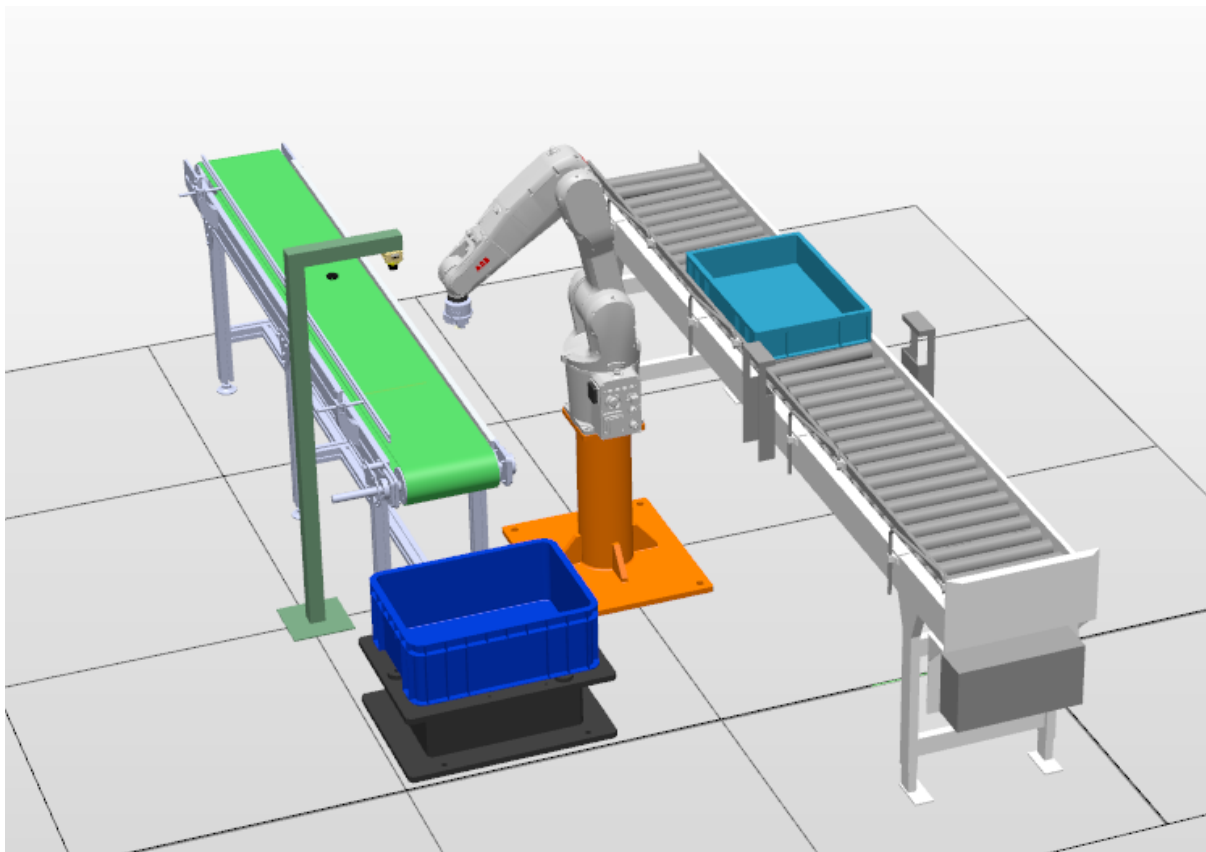
*Mentorja: izr. prof. dr. Karl Gotlib, asist. dr. Timi Karner  
(Univerza v Mariboru, Fakulteta za elektrotehniko, računalništvo in informatiko)*

## **Povzetek**

V projektu II smo želeli nadgraditi znanje pridobljeno pri projektu I, zato smo se odločili za to projektno nalogo.

Naloga je od nas zahtevala, da s pomočjo strojnega vida ločujemo med dobrimi in slabimi obdelovanci (v našem primeru zobniki). Ker je izvajanje projekta potekalo na daljavo, smo se reševanja naloge lotili s pomočjo simulacije. Zaradi takšnega načina izvajanja naloge smo se soočili s problemi, ki sicer v fizičnem izvajanju ne bi nastali. Z nekaj improvizacije smo te težave zaobšli in simulacijo uspešno izvedli.

Rezultat našega dela je simulirana robotska celica, v kateri s pomočjo dveh transportnih trakov, kamere in industrijskega robota ločujemo dobre in slabe zobnike. Proces poteka tako, da se po prvem transportnem traku pripelje zobnik, in ko ta pride v območje zaznavanja kamere, se transportni trak ustavi in kamera posname sliko zobnika. Krmilnik na podlagi te slike določi nadaljnji potek izvajanja procesa. Če je zobnik dovolj podoben sliki referenčnega zobnika, krmilnik pošlje signal robotu, ta pa zobnik pobere in ga zloži v škatlo na drugem transportnem traku. V primeru, da je zobnik poškodovan ali se kakorkoli ne zadostno ujema z referenčnim, pa robot ostane na svojem mestu, prvi transportni trak pa izloči zobnik v škatlo za izmet.



**Slika 1: Kompozicija simulirane robotske celice**  
Vir: lasten

**Ključne besede:** RobotStudio, robotika, simulacija, optična zaznava, Cognex

# UPORABA PROGRAMA SCILAB/MATLAB PRI REGULACIJSKI TEHNIKI 1

ŽAN ŠILAK IN DARJAN ŠTAJNMEC  
*2. letnik, Projekt II*

*Mentorja: doc. dr. Suzana Uran, doc. dr. Božidar Bratina  
(Univerza v Mariboru, Fakulteta za elektrotehniko, računalništvo in informatiko)*

## **Povzetek**

Pri predmetu Regulacijska Tehnika I so domače naloge sestavni del sprotne oblike učenja. Študentje domače naloge rešijo pred vsakim kolokvijem s pomočjo programov, kot sta Scilab ali MATLAB. In rešitve domačih nalog pred pisanjem kolokvija oddajo. Nato domače naloge profesor pregleda in oceni. Obliko sprotnega učenja s pomočjo domačih nalog smo želeli izboljšati, tako da bi z izboljšavo pridobili tako študentje, kot profesor. Študentje bi dobili po oddaji rešitev domačih nalog hitrejši odziv na to ali so nalogo pravilno rešili in bi jo lahko rešili večkrat z različnimi podatki. Profesor pa bi bil razbremenjen pregledovanja velikega števila domačih nalog, ki jih mora pregledati v zelo kratkem času. Za reševanje domačih nalog pri Regulacijski Tehniki I lahko uporabimo programa Scilab (prosto dostopen) ali MATLAB. Za uporabo programa MATLAB smo se odločili zaradi nedavne uvedbe programa MATLAB grader-ja, ki omogoča enostaven prenos nalog v učno okolje Moodle Univerze v Mariboru.

Dela smo se lotili tako, da smo najprej pretvorili domače naloge pri predmetu Regulacijska tehnika 1 v program Matlab, kasneje pa smo jih prenesli v program Matlab grader. Glavna prednost programa ni samo avtomatsko popravljanje in ocenjevanje rešenih domačih nalog. Med reševanjem nalog lahko študent preveri rezultat, program pa mu v primeru napake izpiše navodilo ali pomoč. Program omogoča nastavitve časovnih oken reševanja nalog, kar spodbudi sprotno delo, prav tako pa ima prednost v kreiranju, saj lahko prilagaja podatke posamezne naloge in rezultate. Na ta način ima vsak študent drugačne podatke, hkrati pa dobi takojšen odgovor na rešeno nalogo in lahko ima omejeno ali neomejeno število poizkusov preverjanja rezultatov.

Reševanja smo se lotili tako, da smo najprej določili potek reševanja in posamezne rezultate, ki jih želimo preverjati med reševanjem. Pri tem smo si pomagali s programom Matlab. Kodo smo nato prenesli v okolje Matlab Grader. Preverjamo lahko vrednosti rezultata ali pa podobnost zapisa rezultata (npr. prenosne funkcije). Na koncu smo morali še zapisati podatke, navodila in rezultate, ki mora študent izračunati in zapisati. Med delom smo naleteli na nekaj težav, predvsem na kak način pretvoriti posamezno nalogo, da bo primerna za vnos v spletno učilnico in na kak način lahko preverjamo posamezne rezultate. Pri nekaterih nalogah smo morali upoštevati tolerance odgovorov, saj lahko pri branju iz diagrama dobimo pravilne rezultate z različnimi decimalnimi vrednostmi. Pri tem projektu smo bolje spoznali program Matlab in Matlab Grader in njune možnosti uporabe. Na tak način bi lahko izboljšali postopek reševanja in učenja pri predmetu Regulačijska tehnika 1.

REGULACIJSKA TEHNIKA I

Domov / Učne enote / REGULACIJSKA TEHNIKA I (9056) / Splošno / 5.a Naloga

5.a Naloga

**5.a Naloga** My Solutions >

Rešite naslednjo nalogo!

a) Za vezje na sliki 2 napišite prenosno funkcijo v p-prostoru (s povratno zanko).

**Script** Save Reset MATLAB Documentation

```

1 %Napiši prenosno funkcijo s povratno zanko v Laplaceovem prostoru.
2 %-----PODATKI-----
3 %Štejevalnik: Kc1, Ts=0.01
4 %Spraga: R1=10, R2=20, C=10
5 %-----
6 s = tf('s');
7 F=
8

```

Run Script

Slika 1: Primer naloge v spletni učilnici Moodle

Vir: lasten

**Ključne besede:** Matlab, Matlab Grader, Regulačijska tehnika 1, Moodle

# POSPEŠKOMETER IN KRMILJENJE TEKOČEGA TRAKU

AVGUST SAKELŠEK, BLAŽ PEČENIK IN MATEJ MAYER  
*2. letnik, Projekt II*

*Mentorja: doc.dr. Suzana Uran, doc. dr. Božidar Bratina  
(Univerza v Mariboru, Fakulteta za elektrotehniko, računalništvo in informatiko)*

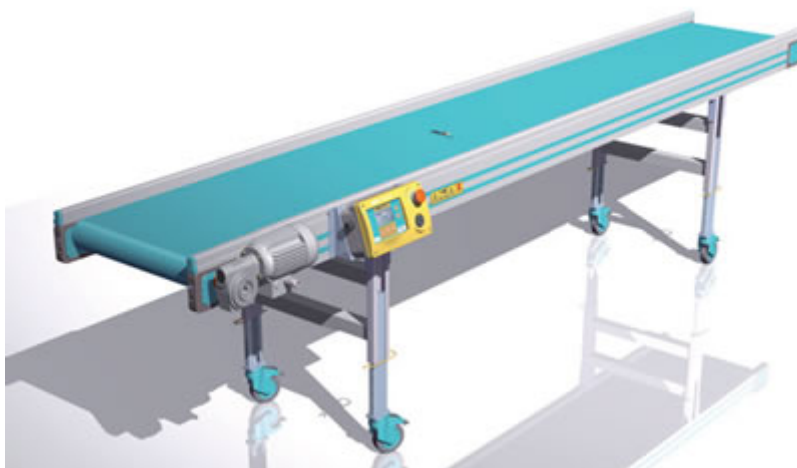
## **Povzetek**

Na začetku naše naloge smo morali najprej izvedeti več o krmilnih sistemih, elektromotorjih, krmilnikih in pospeškometrih ter o komunikaciji med njimi. Naša naloga je bila, da krmilimo pospešek na tekočem traku s pomočjo pospeškometra z uporabo mikrokrmilnika in frekvenčnega pretvornika. Uporabili smo 3 različne pospeškometre (ADXL 335, ADXL 345, MMA 8451) in 3 različne mikrokrmilnike (Arduino, Raspberry PI in PIC 18F25K22).

Projekta smo se lotili tako, da smo najprej pospeškometer priključili na krmilnik in vzpostavili komunikacijo s krmilnikom. Ko smo na računalniku dobili prave in realne podatke, smo se lotili vzpostavitve komunikacije RS485 med krmilnikom in frekvenčnim pretvornikom, ki pa nam zaradi pokvarjenega frekvenčnega pretvornika žal ni uspela.

Na žalost celotnega projekta nismo izvedli, saj smo imeli najprej ogromno zaostanka zaradi epidemije, ko pa so se ukrepi sprostili, pa je bilo že zelo pozno, da bi lahko projekt izvedli do konca.

Pri projektu smo uporabili in dopolnili znanja iz nekaterih šolskih predmetov kot so: Računalništvo, Osnove elektrotehnike, Osnove regulacij in naprednejša znanja iz elektronike.



**Slika 1: Simbolična slika tekočega traku**

Vir: lasten

**Ključne besede:** tekoči trak, pospeškometer, krmilnik, računalnik, frekvenčni pretvornik



# IZDELAVA MERILNIKA MIKRO RAZDALJE NA OSNOVI NIZKOKOHERENČNEGA MICHELSONOVEGA INTERFEROMETRA

NIK VITEZ, ANDRAŽ POŽAR IN DEJAN REZEC

*2. letnik, Projekt II*

*Mentorji: doc. dr. Simon Pevec, red. prof. dr. Denis Donlagić, izr. prof. dr. Uroš Župerl  
(Univerza v Mariboru, Fakulteta za elektrotehniko, računalništvo in informatiko)*

## **Povzetek**

V sklopu projekta II smo želeli izdelati merilnik mikro razdalje na osnovi Michelsonovega interferometra.

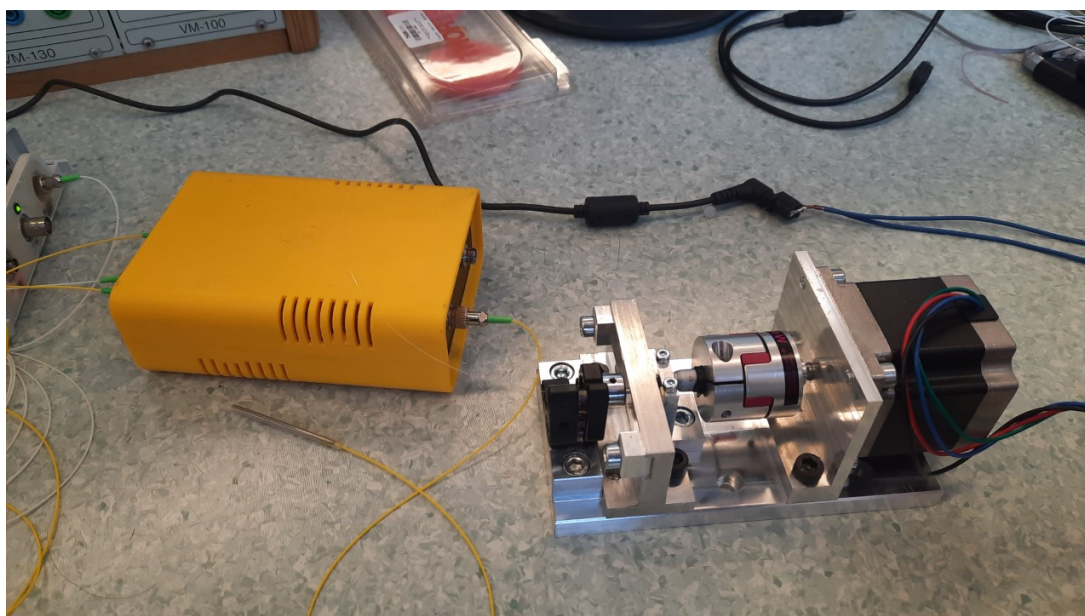
Projekt smo pričeli z raziskavo principa delovanja interferometra ter na podlagi tega podali vezalno shemo z zahtevanimi komponentami. V prvi fazi smo si za cilj zastavili načrtovanje in izdelavo mehanskega dela, ki je obsegal fino nastavitvev zračne reže med kolimatorjem in vzmetnim trakom, in kolimacijo optičnega snopa za potrebe skeniranja optične razdalje. Princip delovanja sloni na finih pomikih vzmetnega elementa, ki ga osvetljujemo z nizkokoherenčnim optičnim virom. Med pomembne naloge sodi izbiranje ustreznih komponent za izvedbo sistema. Med te komponente sodijo koračni motor z maksimalnim številom korakov na obrat, gonilnik za koračne motorje z možnostjo fine delitve korakov in nastavitvijo toka, linearni translator z mikro vijakom, parkljasta sklopka za mehanski sklop med gredjo motorja in mikro vijakom, vzmetni trak ter z vlaknom sklopljen kolimator. V programskem paketu SOLIDWORKS smo izrisali tridimenzionalni model celotnega aktuatorskega sistema. Za krmiljenje koračnega motorja je bil uporabljen gonilnik, ki je bil povezan z razvojno ploščico NUCLEO-F334R8, na kateri smo imeli tri tipke. S pritiskom na prvo tipko se je motor zavrtel v levo za en korak, s pritiskom na drugo tipko pa v desno za en korak. S pritiskom na tretjo tipko smo izvedli cikel, v katerem dosežemo deformacijo traka za 1 mm in se vrnemo v začetno točko. Pred vzmetni trak, ki ga deformira mikro-vijak, smo postavili optični kolimator, s katerim smo

zajemali optično valovanje ter ga nato preko svetlobno/napetostnega pretvornika prikazali v obliki interferograma na osciloskopu. V prvi fazi smo uspeli pokazati, da lahko s fino nastavitvijo razdalje med kolimatorjem in vzmetnim trakom izmerimo koherenčno razdaljo nizkokoherentnega svetlobnega vira, kar je dobra osnova za nadaljnji razvoj Michelsonovega skenirnega sistema.



**Slika 1: Prikaz vseh komponent merilnika**

Vir: lasten



**Slika 2: Nazornejši prikaz merilnika razdalje**

Vir: lasten

**Ključne besede:** SOLIDWORKS, koherenca, optično vlakno, SLED

# NAPRAVA ZA NADZOROVANO POLAGANJE OPTIČNEGA VLAKNA

ALEKSANDER BREZNIK, ANEJ RUPNIK IN ŽIGA ŠTERN  
*2. letnik, Projekt II*

*Mentorji: red. prof. ddr. Denis Donlagič, doc. dr. Matej Njegovec, red. prof. dr. Nenad Gubelnjak  
(Univerza v Mariboru, Fakulteta za elektrotehniko, računalništvo in informatiko)*

## **Povzetek**

Pri projektne delu na študijski smeri Mehatronika VS smo si kot skupina treh študentov pri predmetu Projekt 2 izbrali izdelavo projekta: Naprava za nadzorovano polaganje optičnega vlakna (slika 1). Izdelava naprave obsega znanje strojništva (projektiranje), elektronike (povezovanje elektronskih komponent) in programiranja (Mbed). Naprava deluje na principu odvijanja optičnega vlakna iz standardno navitega koluta na navijalni kolut, da je vlakno navito natančno med navijalni in iztočni kolut postavimo vmes linearno vodilo, na katerem se nahaja podstavek z ferulo (cevka katera ima notranji premer cca. 0,25mm). Potrebno je nastaviti čas premikanja linearnega vodila glede na premikanje koračnega motorja navijalnega koluta za natančnost navitih vlaken. Za napetost vlakna smo uporabili programsko vključitev zavore prvega motorja, ki poganja (vrti) iztočni kolut. Cilj projekta je naviti vlakno na kolut tesno drugo ob drugem, ter vlakno pripraviti na enostavno namestitev na ravno površino (podstavek) v poljubni obliki (elipsa), ki se bo nato uporabilo kot optični senzor za merjenje raztezka z veliko točnostjo. Več kot imamo navitih steklenih vlaken večjo ločljivost in točnost lahko dosežemo. Pri projektu smo pridobili znanje iz tehnologije obdelave, programiranja in predvsem znanja o uporabi senzorjev na osnovi steklenih optičnih vlaken.



Slika 1: Naprava za nadzorovano polaganje optičnega vlakna

Vir: lasten

**Ključne besede:** steklena optična vlakna, navijalni kolut, iztočni kolut, merjenje raztezka, optični senzor

UN

MEHATRONIKA





# IZGRADNJA IN VODENJE MOBILNEGA ROBOTA

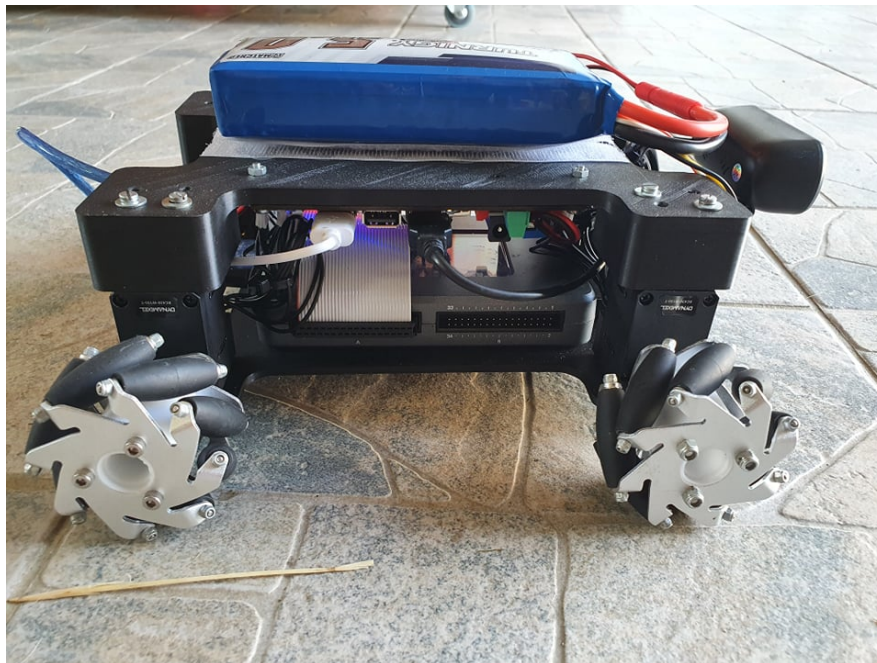
ŽAN JANEŽIČ, ROK KNUPLEŠ, ŠTEFAN JAKOB ŠTRUCL IN ŽIGA LORGER  
*2. letnik, Projekt I*

*Mentor: doc. dr. Darko Hercog  
(Univerza v Mariboru, Fakulteta za Elektrotehniko, računalništvo in informatiko)*

## Povzetek

V okviru našega projekta smo s pomočjo profesorja izdelali mobilnega robota. Delo je obsegalo idejno zasnovo, izbiro in naročanje komponent, izdelavo razširitvene ploščice, izdelavo mobilne platforme in izvedbo komunikacije s senzorji in aktuatorji, ter zasnova osnovnega vodenja robota. Delo smo v začetku razdelili na štiri dele tako, da je bil vsak izmed študentov zadolžen za svoj del. Najprej smo na spletu poiskali vse komponente ter pripadajoče podatkovne liste, ki so bili ključni za nadaljnji potek dela. Osnova robota je mikrokrmilnik MyRIO, s katerim smo izvedli komunikacijo s senzorji in osnovno vodenje robota. Mobilna platforma je zgrajena iz zgornje in spodnje plošče ter nastavkov za montažo Mecanum koles. Poleg 4WD platforme smo predvideli in pripravili tudi 2WD platformo, ki je iz enega kosa. Zasnova razširitvene ploščice je obsegala načrtovanje porabe energije in prilagoditev različnih nivojev napetosti posameznih komponent ter modulov, ki robotu omogočajo opravljati zelene funkcije gibanja v odvisnosti od podatkov pridobljenih iz USB kamere in RPLIDAR-ja. Zasnovana in izdelana tiskanina omogoča povezave do posameznih motorčkov in površino za pritrditev modulov U2D2, USB hub in DC/DC pretvornika. Delo z krmilnikom MyRIO smo razdelili na dva dela, delo s spletno kamero in osnovno vodenje robota s pomočjo Android aplikacije. Za ročno vodenje robota smo sprogramirali Android aplikacijo v razvojnem okolju MIT App Inventor. Izdelana aplikacija pošiljala referenčne vrednosti hitrosti z uporabo TCP/IP protokola v krmilnik MyRIO, le-ta pa na podlagi prejetih podatkov in izbrane konfiguracije robota (2WD, 4WD, Mecanum) izračuna referenčne hitrosti posameznih motorjev ter jih posreduje krmilnikom motorjev. Konfiguracija spletne kamere je obsegala povezavo na myRIO krmilnik. Povezava je lahko izvedena preko USB kabla ali brezžične dostopne točke, vendar je slednja nezanesljiva in neuporabna za aplikacije realnega časa

(Real-Time), zaradi preslabega prenosa podatkov. Ko smo konfigurirali osnovo ploščo smo dodali še detektor robov, ki se poveže s programom krmiljenja robota, tako da se robot izogne zaznamim oviram.



**Slika 1: Izdelan mobilni robot**  
Vir: lasten

**Ključne besede:** MyRIO, Mecanum, tiskanina, LabVIEW, USB kamera



# GRAVIRANJE Z LASERSKIM IZVOROM NA KOORDINATNI MIZI

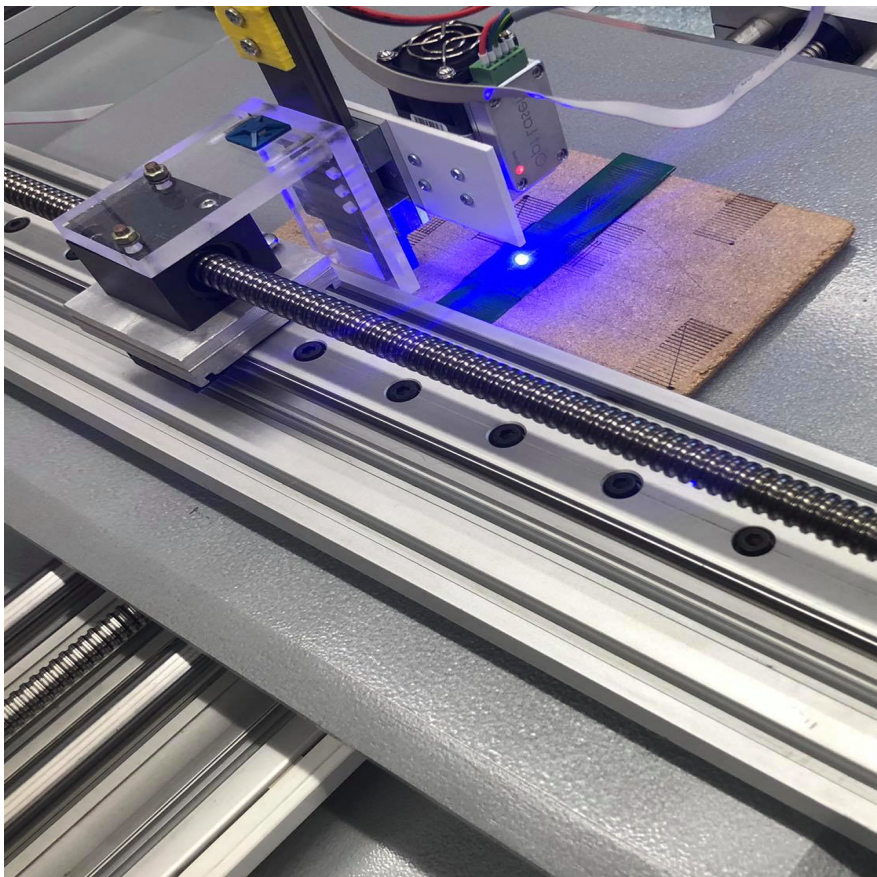
ZORAN URAN IN JAKA ŠTAMPAR  
*2. letnik, Projekt I*

*Mentorja: doc. dr. Darko Hercog, mag. Marijan Španer  
(Univerza v Mariboru, Fakulteta za elektrotehniko računalništvo in informatiko)*

## **Povzetek**

V projektni nalogi sva si s kolegom zastavila cilj izdelave tiskanega vezja z metodo laserskega graviranja. Izdelavna tehnologija tiskanja je odstranjevanje (odparjanje) barve, ki je nanešena na pobakreno plast. Odstranjevanje barve je bilo izvedeno z laserskim žarkom, ki barvo sežge (upari) in zato na tem mestu ostane čisti baker, ki se kasneje odstrani z jedkanjem s pomočjo kisline. Za uspešno izdelavo tiskanega vezja je bila uporabljena laserska glava moči 6 W, ki je zmontirana na dvoosno koordinatno mizo. Kasneje smo se skonstruirali še dodatno linearno Z-os, s katero je bilo možno nastavljanje gorišče laserskega snopa. Sestavne elemente dodatne linearne osi smo tudi 3D natisnili.

Za programiranje koordinatne mize smo uporabljali programsko opremo Motion Perfect 2, ki podpira TRIO krmilnike. V tem programskem okoljem smo testirali vse potrebne funkcije koordinatne mize z laserjem ločeno, nastavljeni so bili potrebni parametri (hitrost, pospešek, programske omejitve, ...) in napisani potrebni podporni programi (npr. iskanje referenčne točke, iskanje gorišča laserskega snopa). Glavni program za risanje tiska je bil računalniško generiran s pomočjo računalniško modeliranega vezja v programu Eagle. Potrebna programska koda za krmilnik TRIO v Basic formatu se iz Eagle formata generira posredno, preko ustreznih programov.



**Slika 1: Lasersko graviranje tiskanine**

Vir: lasten

**Ključne besede:** Trio krmilnik gibanja, koordinatna miza, laser, lasersko odstranjevanje barve, tiskana vezja

# CNC REZKALNIK

ANDREJ WALTER IN JAKOB HORVAT

*2. letnik, Projekt I*

*Mentorja: doc. dr. Darko Hercog, mag. Marijan Španer  
(Univerza v Mariboru, Fakulteta za elektrotehniko računalništvo in informatiko)*

## Povzetek

Osnovni cilj projekta je bil, da se seznanimo in naučimo uporabljati krmilnik Trio ter njegov programski jezik Trio Basic v programskem okolju Motion Perfect. Najprej smo za vajo krmilili en sam elektromotor (1 os). Ko smo ga obvladovali, smo si zadali, da želimo na CNC rezkalniku zrezkati nekakšen predmet ter tako svoje znanje uporabiti. Odločili smo se za model teniškega loparja.

Na začetku smo se dobro seznanili z CNC rezkalnikom, ki smo ga imeli na voljo. Tokrat smo imeli na voljo 3 elektromotorje, saj smo delali v 3D prostoru (x-, y-, z-os). Izvesti smo morali program, da bo rezkalnik s pomočjo referenčnih stikal sam poiskal referenčne točke. S tem si zagotovimo, da začnemo vedno na enakem mestu, kar služi ustreznemu delovanju programa v nadaljevanju in varnostnim lastnostim. Za doseganje varnostnih lastnosti smo si omejili tudi delovno območje stroja. Za optimalno premikanje po treh oseh smo morali sprogramirati oziroma pravilno nastaviti regulator (P, I, D parametre) za naše elektromotorje. Eden izmed pomembnejših vhodov krmilnika je še izklop v sili, ki smo uporabili v podprogramu, ki je bil v zagonu sočasno z glavnim programom. Zeleno tipko smo sprogramirali tako, da služi za začetek delovanja programa. Na izhodih krmilnika smo pa torej krmilili motorje za premikanje po oseh in motor vretena, ki poganja rezkar, s katerim rezkamo obdelovanec.

Pomagali smo si tudi s programom CAD2Motion, kamor smo najprej uvozili DXF format predhodno izbrane preproste slike teniškega loparja. Ta format smo nato še urejali v programu in nato iz tega programa prekopirali sliki ustrezno kodo, oziroma sekvence gibov, ki smo jih vnesli v naš končni program. Tam smo še program ustrezno preuredili

do končne oblike. Kljub nekaj vmesnih zapletov, smo uspešno izdelali majhne, a lepe modele teniškega loparja.



**Slika 1: CNC stroj med delovanjem**

Vir: lasten



**Slika 1: Končen izdelek**

Vir: lasten

**Ključne besede:** CNC, rezkalnik, Trio Motion, rezkanje, servomotor, elektromotor



# ROBOTSTUDIO-IRB 1200: WORLDZONES

MATIC BRUNEC, URBAN KOLMAN IN MATIC SMOGAVEC

*2. letnik, Projekt I*

*Mentorja: asist. dr. Timi Karner (Univerza v Mariboru, Fakulteta za strojništvo),  
izr. prof. dr. Karl Gotlib (Univerza v Mariboru, Fakulteta za strojništvo)*

## Povzetek

Pri predmetu Projekt 1 smo dobili nalogo natančneje spoznati funkcijo WorldZone ob simulaciji delovnega procesa obdelave komadov na CNC stružnici v programu RobotStudio. V ta namen smo preučili posamične ukaze te funkcije in jih uporabili na primerih, ki bi se lahko zgodili v vsakodnevni praksi ter to prikazali tudi na simulaciji.

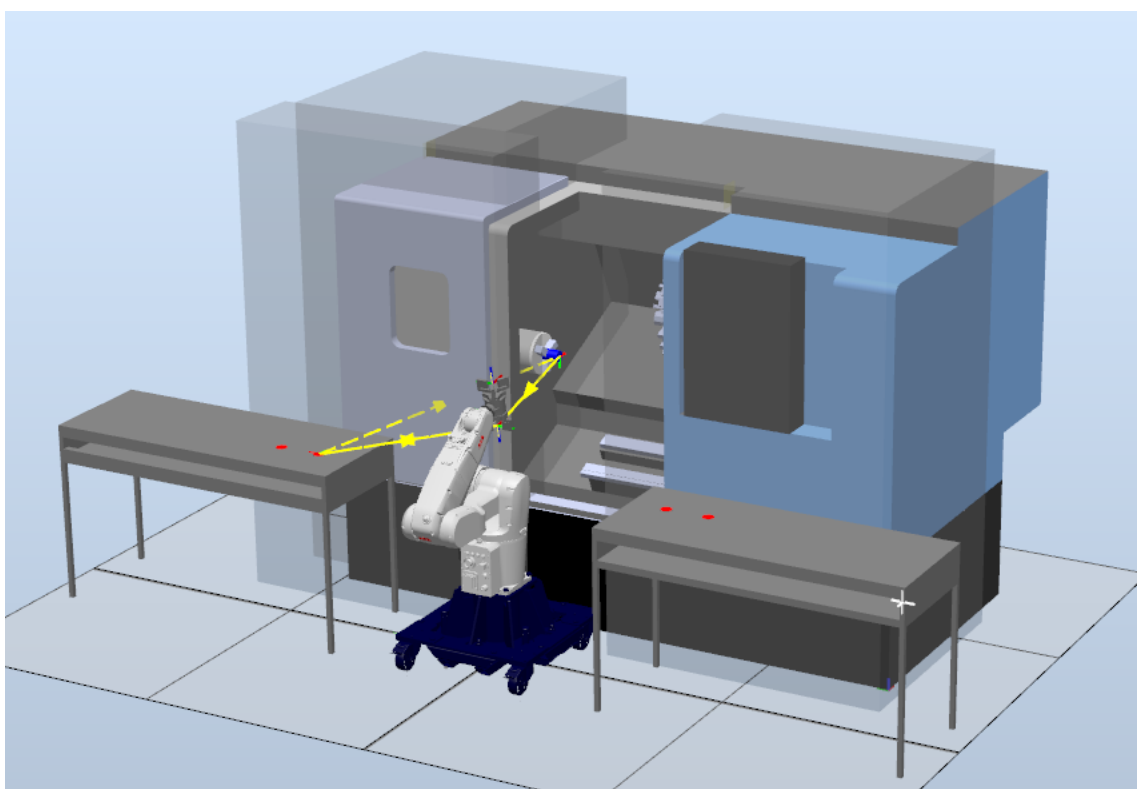
Kot prvi del projekta smo ustvarili torej simulacijo, pri kateri je robot pobiral in nato zlagal oz. predajal surovce stroju in jih po obdelavi prevzel in zložil na prevzemno mesto. Pri tem smo želeli poustvariti klasičen primer avtomatizacije v velikoserijski proizvodnji, da bi s tem lažje prikazali uporabo funkcije Worldzone. Namreč sodobni roboti so sposobni narediti gibe z hitrostjo 6m/s, kar rezultira v hitrost približno 20km/h. Če pri tem upoštevamo še veliko togost robota in maso, ki jo večji roboti prenašajo (tudi do 1 tono), spoznamo, da morebitno trčenje robota z drugim objektom lahko povzroči veliko materialno škodo. Tudi dogodki iz preteklosti nam povedo, da se lahko trčenje robota v operaterja oz. človeka konča celo s smrtnim izidom. Zato je uporaba funkcije Worldzone iz vidika varnosti še kako pomembna.

Funkcija namreč omogoča, da definiramo v bližini robota polje, v katerega robot ne sme vstopiti ali pa ga ne sme zapustiti. Zelo priporočljivo je slednje narediti pred začetkom "učenja" robota kakršnih koli gibov. S tem si operater zagotovi, da se bo robot ustavil, v primeru da na trajektoriji nekega nepredvidenega giba robota stoji objekt. S tem tudi zmanjša ogroženost sebe in nevarnost povzročanja morebitne materialne škode. Do nepredvidenih gibov prihaja zelo pogosto, večinoma pri programiranju robotov. Ko robota programiramo, dejansko ves čas popravljamo in dopolnjujemo program in dokler

programa ne zaženemo in testiramo so gibi robota le predvidevanje, saj vedno obstaja možnost, da smo se pri pisanju instrukcije zmotili. Če se to zgodi pri simulaciji, je treba samo popraviti instrukcijo giba, medtem ko je takšen gib v realnem primeru zelo tvegan. Slednjega žal nismo morali spoznati zaradi situacije s Covid-19.

Kaj se bo zgodilo z robotom, ko bo vstopil v prepovedano območje, lahko upravljamo z dvema ukazoma. Prvi ima okrajšavo WZDOSet (WorldZone Digital Output Set) in nam postavi izbran signal na visok ali nizek nivo. Ta signal lahko naprej uporabimo, da izvedemo nadaljnji gib robota (na primer povežemo program z Trap rutino, v kateri se robot sam vrne v domač položaj). Lahko pa uporabimo ukaz WZLimSup (World Zone Limit Supervision), ki lahko le ustavi robota, ko ta zaide v prepovedano območje.

Na koncu smo prišli do ugotovitve, da je uporaba ukaza WZLimSup iz vidika varnosti primernejša, ker ustavi robota tudi, kadar ga vodimo v ročnem režimu, med tem ko pa je ukaz WZDOSet v ročnem režimu neuporaben.



**Slika 1: Celica s prikazanim delovnim območjem**

Vir: lasten

**Ključne besede:** RobotStudio, IRB-1200, WorldZone

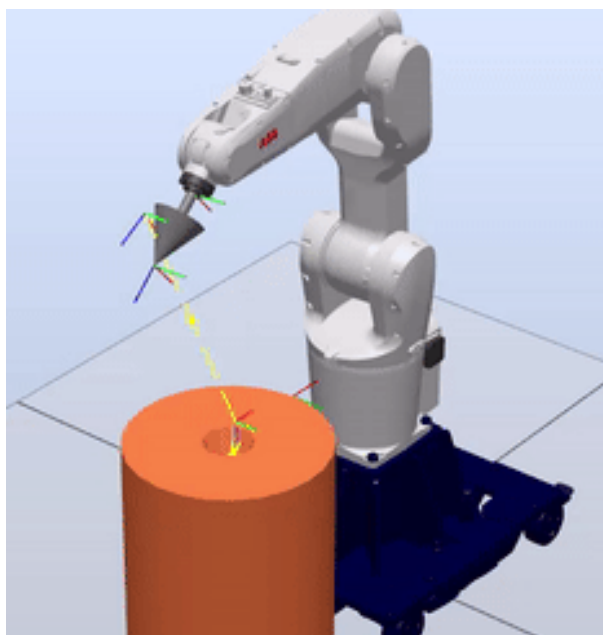
# ROBOTSTUDIO-IRB 1200 SOFT SERVO

URBAN NAVERŠNIK, LIZA ŠKULJ IN LUKA ŠIPEK  
*2. letnik, Projekt I*

*Mentorja: izr. prof. dr. Karl Gotlib in asist. dr. Timi Karner  
(Univerza v Mariboru, Fakulteta za strojništvo)*

## Povzetek

Tema projektne naloge je bila raziskava funkcij, ki omogočajo Soft Servo vodenje robota. Kot končni cilj smo si zastavili ustvariti programsko kodo, ki bi omogočila robotski roki, da se kljub dislokaciji centra izvrtine in orodja poravna in natančno usede v izvrtino. S pomočjo aplikacijskega priročnika smo ustvarili programsko kodo, za katero smo prepričani, da bi morala delovati, ampak program RobotStudio simulacijo takšnih funkcij ter takšnih interakcij med geometrijo ne omogoča. Projekt smo zaključili z dokončano kodo, vendar te kode nismo uspeli realno testirati.



**Slika 1: Simulacija končnega programa**

Vir: lasten

**Ključne besede:** RobotStudio, RobotStudio-IRB 1200, SoftServo, SoftMove, ABB, industrijski roboti, simulacijsko okolje



# NADGRADNJA PROIZVODNJE LINIJE S PHOENIXCONTACT KRMILNIKOM

TAJA PEC, ZALA SLAPNIK IN JURE STOŠIČ

*2. letnik, Projekt I*

*Mentorja: izr. prof. dr. Karl Gotlih, asist. dr. Timi Karner  
(Univerza v Mariboru, Fakulteta za strojništvo)*

## **Povzetek**

Pri projektu smo se ukvarjali s posodobitvijo programske opreme simulacijskega modela proizvodne linije podjetja Staudinger GmbH. S pomočjo pomanjšanega modela proizvodne linije, ki vključuje tekoče trakove, senzorje in obdelovalne stroje smo spoznali vodenje krmilnega sistema z industrijskim programirljivim logičnim krmilnikom (PLK). Naša naloga je bila, da zastarelo programsko okolje Multiprog 4.8 nadomestimo s sodobnejšim programom PC WORX. Prav tako smo morali v sodobnem okolju zapisati program, ki bo vodil obdelovani objekt v krogu po tekočih trakovih.

Najprej smo se spoznati s simulacijskim modelom. Razumeti smo morali vse njegove izhodne in vhodne spremenljivke, da smo lahko iz njih izločili le tiste, ki smo jih potrebovali za zapis našega programa. Izrisali smo skice, časovne diagrame in koračni diagram, s katerimi smo si pomagali pri programiranju. Nato smo vnesli vhodne in izhodne spremenljivke v PC WORX. Spoznali smo se s konstrukcijo lestvičnega diagrama in zapisali naš program. Po zaključenem zapisu programa je PC WORX javil napako in izbrisal vse vhodne in izhodne spremenljivke ter ohranil program. Te težave nam ni uspelo rešiti. Predvidevali smo, da je težava nastala, ker smo uporabljali demo različico programa. Po napotkih profesorja smo menjali programsko okolje iz PC WORX v Tia Portal V16 proizvajalca Siemens. Pridobili smo licenco za ta program. Na začetku programiranja v Tia Portal smo morali izbrati krmilnik ter njegove vhodne in izhodne enote. Izbrali smo takšne, ki jih uporabljamo na fakulteti. Ponovno smo morali vnesti vse vhodne in izhodne

spremenljivke in zapisati program. Program smo nato simulirali z okoljem S7-PLCSIM V16. Tako smo preverili njegovo delovanje.

Ugotovili smo, da nadomestitev programskega okolja Multiprog 4.8 ni mogoča z demo različico PC WORX. Licence za ta program niso bile na voljo, zato smo zamenjali programsko okolje. Programsko okolje Tia Portal je v primerjavi s PC WORX zahtevnejše, nudi več funkcij, vendar je kljub temu primerno za začetnike. Prav tako smo ugotovili, da je za razumevanje in konstrukcijo krmilnega programa potrebno izdelati več diagramov za nedvoumno razumevanje krmilja. Prav tako je pomembno, da pred pričetkom razvijanja krmilne logike sistematično in po sklopih v tabele razporedimo vhodne in izhodne spremenljivke.

**Ključne besede:** model proizvodne linije, Staudinger GmbH, PLK, Multiprog, PC WORX, Tia Portal V16, lestvični diagram, PLCSIM, krmilni program

# RAZVOJ NAPRAVE ZA MIKRO EROZIJO STEKLA

MATIC MARKOVIČ IN ANA GREGOR

*3. letnik, Projekt II/III*

*Mentorja: doc. dr. Suzana Uran, doc. dr. Božidar Bratina  
(Univerza v Mariboru, Fakulteta za elektrotehniko, računalništvo in informatiko)*

## Povzetek

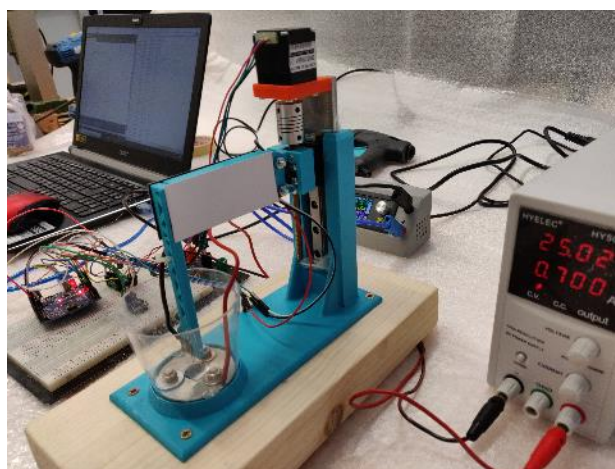
V prvi ideji projekta smo želeli razviti napravo za mikro erozijo stekla z uporabo nanorobota. Zaradi svetovnih razmer je bil dostop do nanorobota onemogočen, zato smo se usmerili v razvoj in izdelavo lastne naprave za mikro erozijo stekla. Želeli smo razviti napravo, ki bo dovolj precizna, da bomo vrtali luknje v steklo s premerom manjšim od enega milimetra. V nadaljevanju dela lahko na nanorobota prenesemo izdelani elastični nosilec s sistemom za merjenje pritiskne sile ter kozarec – potopni bazen.

Mikro erozija je postopek odrezovanja stekla z električnimi iskrami, ki se pojavijo ob elektrodi orodja, kadar ta pritiska na obdelovanec (steklo) s silo nadzorovano z elastičnim nosilcem. Nalogo nasprotne elektrode v našem primeru opravlja podložka iz nerjavnega jekla potopljena na dno elektrolita (Natrijev hidroksid). Na elektrodi dovajamo konstantno enosmerno napetost. Ko z višanjem napetosti dosežemo kritično vrednost napetosti nekje med 20 V in 30 V in gostoto toka  $1 \text{ A/mm}^2$  se prične elektroliza. Pojavlja se vedno več mehurčkov in pričnejo se pojavljati iskre, ki zadenejo stekleno površino in jo erodirajo – pojavijo se mikro kraterji. Na tem mestu elektrolit razjeda površino in odzema material.

Sestavni deli naprave za mikro erozijo stekla so zaradi svojih specifičnih zahtev namensko modelirani v programu SolidWorks in natisnjeni s 3D tiskalnikom. Vsak natisnjen del ima svojo nalogo in je prilagojen drsni mizi, koračnemu motorju in kozarcu – potopnemu bazenu.

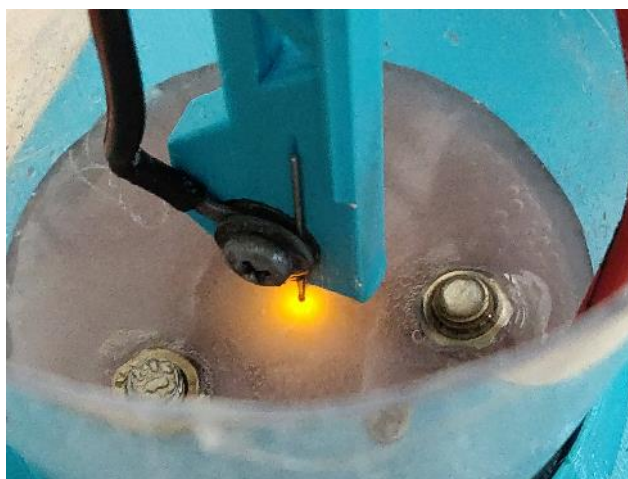
Uporabili smo avtomatsko vodenje bipolarnega koračnega motorja z mikrokrmilnikom Arduino UNO. Na podlagi izmerjenih vrednosti pomika in sile iz senzorja odboja svetlobe QTR-1A, motor obdelovalno iglo pomika navzdol, ko je vrednost pritiskne sile manjša od 0,2 N.

Izvedli smo več uspešnih poskusov vrtnanja lukenj v laboratorijsko steklo z iglo premera 0,4 mm ter paladijevo žičko premera 50  $\mu\text{m}$ . Potopni bazen smo napolnili z elektrolitom ravno toliko, da je obdelovanec – laboratorijsko steklo potopljeno.



**Slika 1: Izdelana naprava za mikro erozijo stekla**

Vir: lasten



**Slika 2: Pojav isker**

Vir: lasten

**Ključne besede:** mikro erozija stekla, elektrolit, koračni motor, elastični nosilec, 3D tiskanje, merjenje pritiskne sile

# REALIZACIJA DELOVNE POSTAJE MITSUBISHI ELECTRIC

MATIC ZELENIK, MIHA VUNDERL IN ANTONIO HULEC

*3. letnik, Projekt II/III*

*Mentorja: izr. prof. dr. Karl Gotlib, asist. dr. Timi Karner  
(Univerza v Mariboru, Fakulteta za strojništvo)*

## **Povzetek**

Projekt II smo nadaljevali v Projektu III ter smo uspešno dokončali realizacijo delovne postaje Mitsubishi Electric, kar je bil naš prvotni cilj. Drugi cilj je bil, da strojno opremo medsebojno povežemo glede na narisane električne sheme in ustvarimo komunikacijsko povezavo preko Ethernet protokola, za katero skrbi industrijsko stikalo. Pri tem delu smo imeli manjše težave glede medsebojne komunikacije, oz. pošiljanja in sprejemanja podatkov med PLK-jem in ostalo strojno opremo. Navedeno težavo smo uspešno odpravili s pravilno vezavo serijske komunikacije.

Naš tretji in zadnji cilj je bil narediti krmiljenje za posamezne enote. Prva enota je predstavljala gretje vode z grelcem ter merjenje temperature s Pt100 senzorjem. Druga enota je predstavljala krmiljenje asinhronskega motorja s pomočjo inverterja ter nastavljanje frekvence motorja s potenciometrom. Tretja enota pa je predstavljala pozicioniranje servo motorja na točno določene pozicije ter nastavljanje parametrov pospeška, hitrosti in pojemka s servo ojačevalnikom. Vse parametre ter krmiljenje strojne opreme lahko nastavljamo preko HMI zaslona, ki poskrbi za vizualno in grafično komunikacijo med operaterjem in delovno postajo.



Slika 1: Končana delovna postaja

Vir: lasten

**Ključne besede:** Mitsubishi, Mitsubishi Electric, PLC

# RAZVOJ IN IZDELAVA VARIABILNE CEVI NA AGREGATU FORMULA STUDENT

MIHA HRŽIČ IN ŽAN ČAKŠ  
*3. letnik, Projekt II/III*

*Mentorji: red. prof. dr. Darko Lovrec, doc. dr. Vito Tič, red. prof. dr. Riko Šafarič  
(Univerza v Mariboru, Fakulteta za strojništvo in Fakulteta za elektrotehniko računalništvo in informatiko)*

## **Povzetek**

Cilj projekta je bil razviti in izdelati del sesalnega voda, ki bi med vožnjo formule prilagajal dolžino sesalnega voda. Delo sva si razdelila tako, da je en študent oblikoval PCB ploščo v programu Altium, drugi pa je dopolnil koncept variabilne sesalne cevi v programu Catia.

Za krmiljenje variabilne sesalne cevi smo se odločili, da ga izvedemo s pomočjo PCB plošče. Posebej za namen je bilo treba v programu Altium zasnovati posamezne enote plošče in te narisati. Zanj pa v knjižnico ekipe Formula student vnesti vse manjkajoče elemente.

Ploščo smo nato oblikovali, komponente povezali ter plošči dodati tako imenovane poligone, ki po plasteh plošče povezujejo elemente med sabo. Ploščo je bilo nato treba izdelati in nanj pritrditi elemente. V programu Catia pa se je med tem dodelalo končni koncept ter se ga natisnilo s 3D tiskalnikom za preverjanje.

Ker je projekt zelo obsežen, ga bova nadgradila v diplomski nalogi. Izdelali bomo sesalno cev, ki ji bomo dodali mehanizem za pomik.

**Ključne besede:** sesalna cev, PCB, elektronski elementi





# MODELI LETAL IN DRONOV V ORODJU AVL CRUISEM

DAVID VIDE IN ALEKSA TODOROVIĆ

*3. letnik, Projekt II/III*

*Mentorji: doc. dr. Miran Rodič (Univerza v Mariboru, Fakulteta za elektrotehniko, računalništvo in informatiko), izr. prof. dr. Uroš Župerl (Univerza v Mariboru, Fakulteta za strojništvo), Andrej Cvikl (AVL-AST d.o.o. Slovenija)*

## **Povzetek**

V okviru projekta smo želeli modelirati model letala, določiti cikle letenja za ta model ter ugotoviti porabo energije v teh voznih ciklih. Projekt je nadaljevanje projekta iz zimskega semestra, v katerem smo izdelali model letala v orodju CruiseM. Model letala smo nadgradili tako, da smo dodali model propelerja, ki služi kot pogon letala. Ta model tako kot model letala uporablja več formul, ki smo jih uporabili v funkcijskem bloku v orodju CruiseM. Na začetku smo zapisali dve formuli, ki računata silo ter navor propelerja. Propelerju pod zavihkom Parametri smo dodali dolžino listov ter maso, letalu pa dodali maso ter površino kril. V modelu propelerja potrebujemo še površino kril letala ter začetno vrednost kotne hitrosti, ki smo jo natakili s pomočjo funkcije realTime. V tem modelu smo uporabili veliko konstant, ki jih je treba pazljivo izbrati, saj bo v primeru uporabe napačnih podatkov propeler narobe deloval. Uporabili smo štiri regulatorje v dveh kaskadnih regulacijah. Želena vrednost kotne hitrosti dobimo iz tabele, dejanska vrednost regulatorja pa je hitrost letala (zunanja zanka). Želena vrednost regulatorja pogonske sile je izhod iz regulatorja hitrosti, dejanska vrednost pa je kotna hitrost propelerja, ki jo dobimo iz modela propelerja. Za reguliranje višine smo uporabili želeno višino iz tabele ter dejansko višino iz modela letala. Izhod zunanje zanke regulatorja je kot dvigovanja letala, v notranji zanki pa nastavljamo kot višinskih krmil letala. Ko propeler, kot pogon letala smo preizkusili, smo ugotavljali še porabo energije letala. Za to uporabimo blok baterije v orodju CruiseM. Poiskali smo baterijo, ki jo je uporabljalo realno letalo. Njene vrednosti smo vpisali v blok. Ta blok že vsebuje napetost, tok ter stanje napolnjenosti

baterije kot izhode, zato smo navor povezali na breme baterije. V blok baterije smo vpisali napetosti pri različnih stanjih napolnjenosti baterije. Vozne cikle letala smo definirali v tabelah za želeno hitrost ter višino. Nato smo vključili izhode baterije v simulacijo, v kateri smo lahko videli porabo baterije oz. napolnjenost baterije med simulacijo leta.

**Ključne besede:** letalo, poraba, propeler, simulacija, CruiseM

# RAZVOJ MIKRO-FLUIDNIH MOTORJEV NA OSNOVI LONČKA IN STEBRIČKA

RUDOLF-LEON FILIP, GREGOR KORŽE IN LUKA SAKAČ  
*3. letnik, Projekt II/III*

*Mentorji: red. prof. dr. Riko Šafarič, doc. dr. Božidar Bratina, red. prof. dr. Karl Gotlib  
(Univerza v Mariboru, Fakulteta za strojništvo in Fakulteta za elektrotehniko računalništvo in  
informatiko)*

## Povzetek

Cilj našega projekta je bil razviti več različic mikro-fluidnih motorjev in doseči čim hitrejšo/bolj ugodno vrtenje vode. Pri projektu 2 smo večino časa izboljševali pogoje za zajemanje nadaljnjih meritev.

Za začetek smo z dvema led diodama izboljšali osvetlitev mikroskopa. Led diodi sta na mikroskopu postavljeni tako, da ena osvetljuje opazovani objekt od zgoraj, druga pa od spodaj, ter tako pridobili večjo fleksibilnost osvetlitve. Načrtovali in izdelali smo tudi držalo za pametni telefon in kamero GoPro, ki se pritrdi na okular. Nastavek nam bo omogočal boljše zajemanje fotografij in posnetkov, kot do sedaj uporabljana mikroskopska kamera. Iz polysmoot karbona pa smo 3D natisnili mehanski ojačevalnik za piezoelement.

Zaradi globalne pandemije in prepovedi obiskovanja laboratorija se je glavni del projekta malce zakasnil, zato pa smo izkoristili čas ter si izboljšali pogoje za nadaljnje delo.

**Ključne besede:** osvetlitev, držalo, piezoelement, polysmoot karbon



# INTEGRACIJA NAPREDNIH SISTEMOV VOŽNJE V AVTONOMNI KMETIJSKI ROBOT FARMBEAST

GREGOR POPIČ IN DOMEN TOŠ

*3. letnik, Projekt III*

*Mentorji: doc. dr. Jurij Rakun (Univerza v Mariboru, Fakulteta za kmetijstvo in biosistemske vede),  
doc. dr. Suzana Uran (Univerza v Mariboru, Fakulteta za elektrotehniko, računalništvo in  
informatiko), doc. dr. Aleš Belšak (Univerza v Mariboru, Fakulteta za strojništvo)*

## **Povzetek**

V študijskem letu 2020/2021 sva s sošolcem nadaljevala z delom z avtonomnim kmetijskim robotom FarmBeast. Po opravljenem drugem delu projekta sva se v okviru projekta 3 lotila priprav na mednarodno tekmovanje FieldRobotEvent. Zadolžena sva bila za pripravo algoritmov za prvo in drugo kategorijo tekmovanja.

Začela sva na preprostem primeru TurtleSim. TurtleSim je 2D simulacija, kjer sva spoznala, kako manipulirati z linearnimi in kotnimi hitrostmi. Kasneje sva se preselila v simulacijsko okolje Gazebo, v katerem je bil robot in v ravnih vrstah postavljene palice. V Rviz-u sva lahko videla, kako senzor na robotu zaznava ovire in na podlagi slednjega sva začela s pisanjem algoritma za vožnjo robota. Kar hitro sva se preselila v novo simulacijo, kjer je oviro predstavljala koruza. Glede na prejšnji algoritem sva morala prilagoditi območje zaznave senzorja, saj so pred robotom viseli listi in so motili algoritem vožnje. Nato sva dodala del kode, da se robot na koncu vrste zmeraj obrne in hkrati postavi pred novo vrsto. Pomembno je bilo tudi to, da se robot vsakič obrne v drugo stran, saj bi se v nasprotnem primeru vozil le v dveh vrstah. Končna simulacija je postavila visok cilj. Ravne vrste koruze so zamenjale krivuljaste vrste, ravna tla so nadomestila grbinasta tla z zemljo in kamenjem.

Organizatorji so poskrbeli tudi, da so kolesa robota včasih na kakšni grbini preprosto zdrsnila in nam na takšen način otežila uporabo odometrije. Zaradi neravnih vrst sva naredila štiri področja, kjer večja kot je napaka odklon poti od sredine vrste, večja bo kotna hitrost in manjša bo linearna hitrost. Tako sva dosegla, da je robot v zavojih nekoliko upočasnjen in se tako uspel poravnati v vrsto. Kdaj pa kdaj se je tudi zgodilo, da so v vrstah manjkale koruze, zato sva s pomočjo Rviz-a opazovala zaznavo in prilagodila program tako, da je senzor vedno "gledal" dovolj naprej.

V drugi kategoriji pa je bila posebnost to, da se je robot na koncu vrste, da se je moral robot zapeljati v točno določeno vrsto. Na tekmovanju smo pogledali tekstovno datoteko organizatorja, v kateri je bila vpisana načrtovana pot, ki sicer ni bila znana vnaprej. Zato sva prilagodila algoritem za ravno vožnjo ob vrstah in dodala del za branje tekstovne datoteke.

V drugi kategoriji smo med dosegli 3. mesto.

**Ključne besede:** FarmBeast, FieldRobotEvent, ROS, Rviz, Gazebo

# INDUSTRIJSKI ROBOT IN TOF KAMERA

PRIMOŽ KOBAL IN ŽIGA PAJTLER

*3. letnik, Projekt III*

*Mentorji: izr. prof. dr. Aleš Hace, Rok Pučko (Univerza v Mariboru, Fakulteta za elektrotehniko, računalništvo in informatiko), izr. prof. dr. Karl Gotlib (Univerza v Mariboru, Fakulteta za strojništvo)*

## **Povzetek**

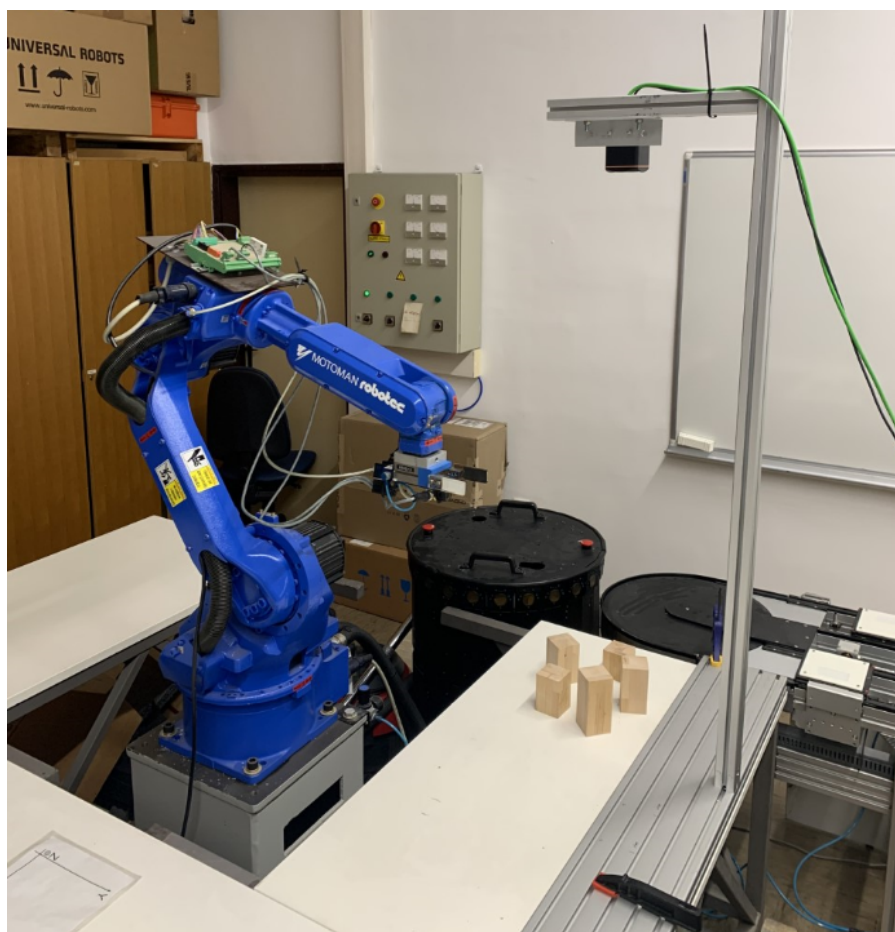
Naloga projekta je prepoznati nek v naprej določen objekt v delovnem prostoru robota na delovni površini s pomočjo ToF kamere IFM O3D303, ter nato z industrijsko 6-osno robotsko roko Motoman HP6 izvesti manipulacijo prepoznane objekta. V laboratorijski izvedbi smo se omejili na objekte kvadraste oblike različnih dimenzij, primernih za prijemanje z dvo-prstnim prijemalom in za zaznavanje z uporabljeno 3D kamero ter robotsko nalogo, ki izvede sortiranje objektov po velikosti.

Prepoznavanje objektov je izvedeno s pomočjo Matlab programskega okolja, kjer so že pripravljene preproste rutine za delo z oblakom točk, ki ga pridobimo s pomočjo 3D kamere. Algoritem v Matlabu iz oblaka točk naprej poišče vse kvadraste oblike in nato izlušči podatke o položaju, orientaciji ter dimenzijah kvadrov ter na podlagi teh določi točko prijemanja za vsak prepoznani objekt. Za prenos točke prijemanja na robota, je treba točko prijemanja pretvoriti iz koordinatnega sistema kamere na koordinatni sistem robota.

Komunikacija med računalnikom in robotom je izvedena preko ethernet povezave s pomočjo MotoCom SDK knjižnice, ki nam omogoča prenos podatkov v/iz robota. MotoCom knjižnica je pripravljena za uporabo z okoljem VisualStudio na Windows platformi, kjer smo zgradili preprosti program za prenos detektiranih točk prijemanja iz Matlab algoritma.

V simulacijskem okolju MotoSim (Yaskawa) smo izvedli simulacijo celotne laboratorijske robotske naloge. V simulacijo smo vključili CAD rekonstruirano okolje iz laboratorija.

Kalibracija koordinatnih sistemov robota in 3D kamere bo izvedena s pomočjo podobnostne transformacije s sedmimi parametri (Helmert transformation). Sistem deluje na podlagi skupnih veznih točk. Na robota bomo pritrdili tarčo okrogle oblike pri čemer bomo TCP točko robota nastavili na središče krogle. Tarčo bomo s pomočjo robota premaknili v več različnih točk in pri vsaki odčitali položaj TCP točke robota ter pripadajoč položaj središča krogle iz zajete slike s pomočjo 3D kamere (pri tem bomo za iskanje središča tarče uporabili algoritem, ki smo ga pripravili v Matlabu). Iz veznih točk bomo nato preračunali translacijo in rotacijo koordinatnega sistema kamere glede na koordinatni sistem robota.



Slika 1: Laboratorijsko okolje (Robot Motoman HP6 in Kamera IFM O3D303)

Vir: lasten

**Ključne besede:** ToF kamera O3D303, Oblak točk, Matlab, Motosim, prepoznavanje objektov, Motoman HP6



# INTUITIVNO PROGRAMIRANJE ROBOTA S POMOČJO TOF KAMERE

MIHA ARTIČ IN ŽIGA ŠTERN  
*3. letnik, Projekt II/III*

*Mentorja: izr. prof. dr. Aleš Hace, Rok Pučko (Univerza v Mariboru, Fakulteta za elektrotehniko, računalništvo in informatiko) izr. prof. dr. Uroš Župerl (Univerza v Mariboru, Fakulteta za strojništvo)*

## **Povzetek**

Za robotizacijo neke naloge in kasnejše prilagajanje in modifikacije robotskih programov potrebujemo visoko usposobljen kader. Izvajanje očitnih, manjših in nezahtevnih modifikacij obstoječih robotskih programov bi lahko izvedel kar operater neke robotizirane naloge sam, vendar v splošnem predpostavimo, da ni usposobljen za uporabo zahtevnih in kompleksnih robotskih vmesnikov.

Cilj projekta je zgraditi laboratorijski testni vmesnik, ki bo za človeka omogočal naravno, učinkovito in intuitivno komunikacijo med človekom in robotom.

V projektu smo uporabili globinsko kamero za prepoznavanje in sledenje človeškim dlanem LEAP Motion Controller, ki samodejno zaznava položaj človeške dlani, prepoznava preproste geste in tudi iztegnjene ali skrčene prste roke ter kolaborativnega robota UR3, ki nudi možnost programiranja, izvajanja programov in komunikacije v virtualnem okolju URSim.

Podatki se v realnem času iz kamere pošiljajo na računalnik, kjer jih s pomočjo programa v okolju VisualStudio obdelamo in posredujemo do robota. Program v VisualStudio in UR3 robot komunicirata preko TCP/IP povezave.

Kamera podaja položaj dlani na podlagi merjenja časa preleta IR svetlobe. Položaj dlani je določena s točko s kartezičnimi koordinatami z središčem v težišču dlani. Te vrednosti se prenesejo do robota, kjer so definirane v obliki točke, kateri robot sledi. Ta je podana kot matrika s položajem in rotacijo TCP -ja, ki je konstantna. S premikom dlani vplivamo na spremembo položaja središča dlani in posledično na položaja točke, ki ji robot sledi med gibanjem. Projekt se bo nadaljeval v okviru dveh diplomskih nalog, v katerih želimo nadgraditi aplikacijo do te mere, da bomo lahko robota samostojno programirali z gestami in opisanim postopkom vodenja.



Slika 1: Delovanje aplikacije

Vir: lasten

**Ključne besede:** robotizacija, kolaborativni robot, UR Polyscope, LEAP Motion Controller, Visual Studio

# OPTIČNO ZAJEMANJE POVRŠINE KOSOV S KOLABORATIVNIM ROBOTOM

ANŽE ROS

*3. letnik, Projekt II/III*

*Mentorja: izr. prof. dr. Aleš Hace, Mitja Golob*

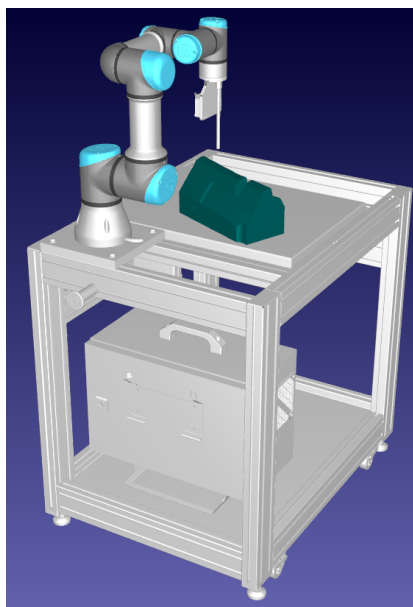
*(Univerza v Mariboru, Fakulteta za elektrotehniko, računalništvo in informatiko)*

## **Povzetek**

V primeru izvedbe procesnih nalog s kolaborativnim robotom, kot so npr.: mikrokovanje, brušenje, poliranje itd. je potrebno poznati natančno lego obdelovanca. Natančno poznavanje lege obdelovanca v delovnem prostoru robota je eden izmed ključnih faktorjev za uspešno in varno izvedbo robotskih nalog. Za določitev lege obdelovanca obstaja več različnih metod. Trenutno uporabljen, klasični način se izvaja z fizičnim odtipom potrebnih kalibracijskih točk na predmetu oz. obdelovancu, medtem ko se pri novejših postopkih vse več uporablja optična tehnologija zajemanja točk ali površine. V okviru tega projekta je bil v laboratoriju za industrijsko robotiko vzpostavljen testni sistem, ki omogoča izvedbo kalibracije obdelovanca s pomočjo laserskega merilnika razdalje. Sistem sestavljajo kolaborativni robot UR3-cb, robotski krmilnik in laserski merilnik razdalje SICK OD2. Robot UR3-cb in robotski krmilnik sta nameščena na ročni mobilni robotski platformi, ki je zgrajena iz ALU profilov. Na prirobnici robotske roke se je namestil nosilec, ki je omogočil namestitev laserskega merilnika razdalje na prirobnico. V okviru projekta je bila izvedena namestitev laserskega merilnika razdalje na prirobnico kolaborativnega robota, implementacija algoritmov za kalibracijo TCP točke laserskega sensorja razdalje in kalibracijo obdelovanca na robotski krmilnik kolaborativnega robota. Algoritma za kalibracijo TCP točke laserskega sensorja razdalje s implementirano 4-točkovno metodo in kalibracijo obdelovanca s implementirano 3-točkovno metodo smo za potrebe implementacije na robotski krmilnik pripravili v ustreznem URScript programskem jeziku. Programski jezik URScript je namenjen za programiranje kolaborativnih robotov UR. Algoritmi so se predhodno preizkusili v simulacijskem

programskem okolju URSim, ki predstavlja simulacijsko okolje za 'offline' programiranje in simulacijo izvajanja robotskega programa.

Dobljeni rezultati kalibracije obdelovanca z laserskim senzorjem so bili primerjani s klasičnim pristopom (kalibracijska konica). Za grafično predstavitev rezultatov smo uporabili programsko orodje RoboDK, ki je namenjeno za programiranje in simuliranje delovanja različnih industrijskih robotov. Okolje RoboDK ima že integrirani metodi za kalibracijo orodja in obdelovanca, zato so rezultati teh dveh metod predstavljali referenčne podatke. Po opravljenih primerjavah lahko potrdimo, da optična metoda zajemanja točk in kalibracije obdelovanca zagotavlja dobre, stabilne in ponovljive rezultate, treba pa upoštevati, da ima slednja tudi nekatere slabosti. Eno od slabosti predstavlja težavno odtipanje točk na robovih obdelovancev, temu sledi, da je točnost izvedene kalibracije obdelovanca manjša. Pojavlja se tudi problematika reflektivnih površin, kjer pri odtipanju prihaja do odbojev svetlobe, ti pa lahko povzročijo neustrezno izmerjene rezultate.



**Slika 1: Ročna mobilna robotska platforma**

Vir: lasten

**Ključne besede:** kalibracija, kalibracijska konica, laserski merilnik razdalje, obdelovanec, kolaborativni robot

# NADGRADNJA PROIZVODNJE LINIJE Z BECKHOFF KRMILNIKOM

DOMINIK HRASTNIK  
*absolvent, Projekt II/III*

*Mentorji: izr. prof. dr. Karel Gotlib, dr. Timi Karner, doc. dr. Janez Gotlib (Univerza v Mariboru, Fakulteta za strojništvo), izr. prof. dr. Aleš Hacc (Univerza v Mariboru, Fakulteta za elektrotehniko, računalništvo in informatiko)*

## **Povzetek**

Namen projekta *Nadgradnja proizvodnje linije z Beckhoff krmilnikom* je posodobiti programsko opremo simulacijskega modela Staudinger GmbH z gradniki Ficher Technik. Simulacijski model sestavljajo štiri multifunkcijski modularni moduli oz. plošče. Pri projektu 2/3 smo uporabljali ploščo 1. Linijo smo krmilili s krmilnikom Multiprog 4.0, ki je zastarel. Z zamenjavo obstoječega krmilnika s krmilnikom Beckhoff bomo dosegli zanesljivejše delovanje proizvodne linije. Projekt smo začeli z branjem dokumentacije o komunikacijskem vmesniku Phoenix Contact Fl IL 24BK-PAC, ki se nahaja med fizičnimi procesnimi veličinami (vhodi in izhodi) ter virtualnim krmilnikom, ki je nameščen na osebem računalniku s programom TwinCAT proizvajalca Beckhoff. Ta omogoča, da osebni računalnik spremenimo v virtualni krmilnik, s pomočjo katerega nato komuniciramo do komunikacijskega vmesnika. Projekt 3 smo v obliki wordovega zapisa nadaljevali s spremembo dobljenih vrednosti, ki smo jih s pomočjo že izdelanega funkcijskega bloka FB\_KL85xxWordTo16Bit dobili v bitih. To je zelo poenostavilo in ustvarilo boljši pregled nad glavnim programom pisanja ukazov. Zaradi komunikacije potrebujemo posebna bloka, ki sta opisana v projektu 2. Nato smo ustvarili lastni funkcijski blok, v katerega smo zapakirali komunikacijo in pretvorbo vrednosti za branje vhodov ter postavitev izhodov. Tu je prišlo do težav, saj smo komunikacijo izvedli samo enkrat. Da bi to rešili, je bilo treba ustvariti pulzni signal, z uporabo dveh vzporednih časovnikov. Tako je komunikacija delovala brez težav. Sedaj smo lahko začeli sestavljati pregledni glavni program, ki bo z izvajanjem določenega vhodnega bita prožil samo en bit

izhodnega signala. Tako smo na plošči 1 napisali program za premik izdelka preko štirih tekočih trakov, ki so se ob zaznavi prejšnjega sensorja začeli vrteti, ob zaznavi naslednjega pa izklopili. S tem smo prišli do konca projekta 3; uspešno smo naredili funkcijska bloka in ju vstavili v program. Prav tako smo uspeli ustaviti program za premik izdelka po plošči 1.



**Slika 1: Simulacijski model Staudinger GmbH**

Vir: lasten

**Ključne besede:** proizvodnja linija, TwinCAT, Beckhoff, funkcijski blok, nadgradnja krmilnika

# MAG

MEHATRONIKA





# ROBOT NAO - UČENJE PREPOZNAVANJA OBJEKTOV IN OBRAZOV

MARTIN COKAN IN MIHAEL TAUŽIČ

*2. letnik, Projekt*

*Mentorji: izr. prof. dr. Karl Gotlib, asist. dr. Timi Karner (Univerza v Mariboru, Fakulteta za strojništvo), izr. prof. dr. Aleš Hec (Univerza v Mariboru, Fakulteta za elektrotehniko, računalništvo in informatiko)*

## Povzetek

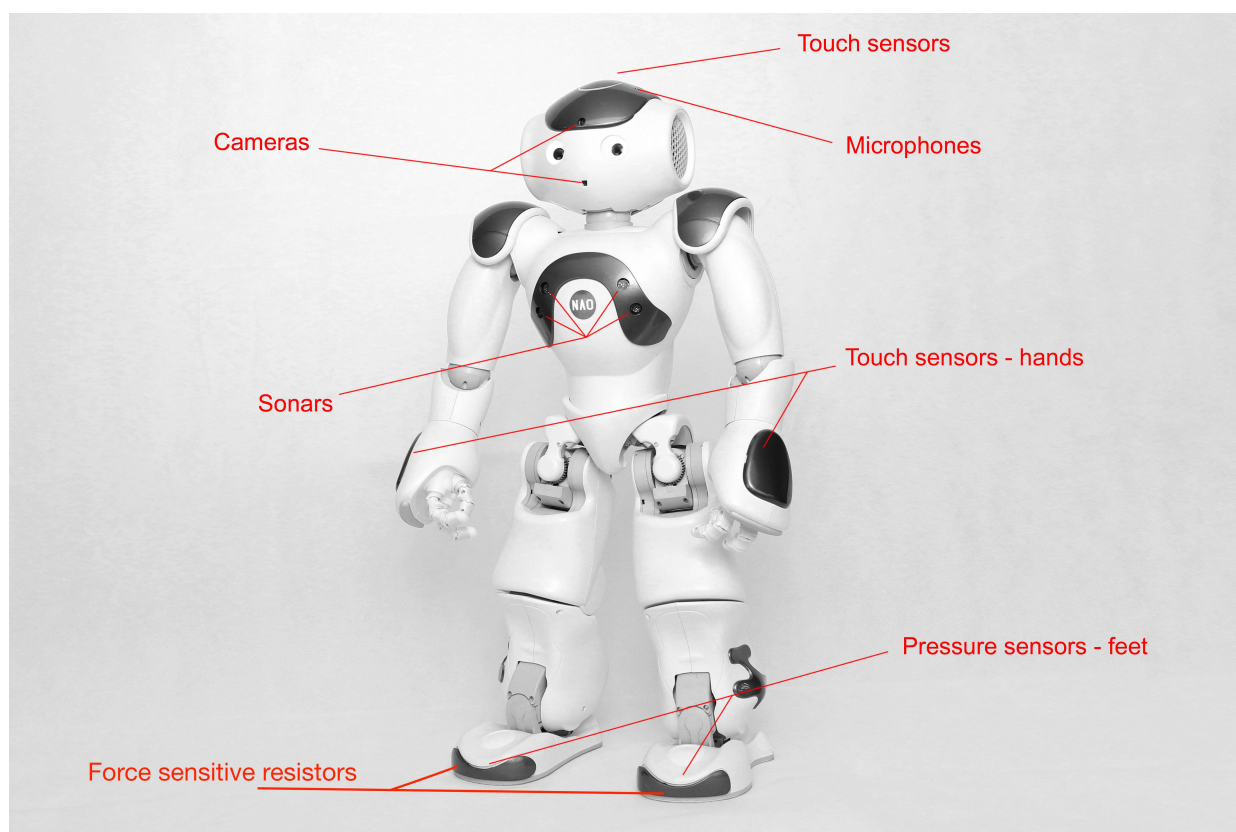
Cilj projekta je bil spoznati delovanje humanoidnega robota **NAO** in napisati program za prepoznavo objektov in obrazov. Kot je razvidno iz spodnje slike, humanoidni robot NAO izgleda kot človek. Posnema lahko funkcije človeka, saj ima roke, noge in glavo ter vsebuje številne senzorje za posnemanje človeških funkcij. Glava robota **NAO** vsebuje 3 senzorje dotika na samem vrhu, dve kameri, mikrofona, zvočnike in se vrti v vratu. Naslednja komponenta je prsni del, ki vsebuje 4 ultrazvočne senzorje razdalje, s katerimi meri odmike od zidov, ljudi oz. objektov. Robot ima dve roki, na vsaki po 3 prste. Na vsaki roki ima tudi po tri taktilne senzorje dotika. Spodnji del robota, se pravi dve nogi vsebujeta na sprednjem delu stopala tlačne senzorje, ki jih imenujemo tudi odbijači in na spodnjem delu stopala prav tako tlačne senzorje, ki služijo temu, da vemo ali robot stoji na nogah.

Robota **NAO** programiramo preko programskega okolja **Choregraphe**, ki ga ponuja proizvajalec, ali pa preko **Python** programskega jezika. Sama sva uporabila programsko okolje **Choregraphe**, ki nam ponuja že nekatere v naprej pripravljene bloke in je tako delo lažje in hitrejše. Robot se z računalnikom poveže preko wi-fija in svojega IP naslova.

Sam projekt sva razdelila na dva dela. V prvem delu projekta sva napisala program za prepoznavo obrazov. Robot se je sposoben naučiti različne obraze, jih zaznati in nato prepoznati ter ločiti. Robot lahko izvaja programsko kodo po naši želji, torej se z osebo pogovarja, jo nagovori ali izvaja določeno funkcijo. V našem primeru smo uporabili 5

različnih obrazov, ki smo jih naložili v knjižnico robota, ki jih je nato zaznal in prepoznal brez napake.

Drugi del projekta je služil za prepoznavanje objektov. Z robotom smo prepoznali različne objekte, za katere smo ga pred tem naučili, da si jih zapomni. Na koncu sva se osredotočila na žogico in izpisala program, da jo zazna pred seboj in se premakne do nje. Projekt nadaljujeva v smeri, da bi robot žogico pobral in jo prinesel do koša, vendar zaradi časovne stiske nisva prepričana, da bova to dokončala v danem roku.



**Slika 1: Robot NAO in položaji senzorjev.**

Vir: eBook NAO6 - The basic learn.

**Ključne besede:** robot, humanoidni robot NAO, prepoznavanje obrazov, prepoznavanje objektov, program

# VODENJE BREZPILOTNEGA ZRAKOPLOVA

KRISTIJAN POLOVIČ

*2. letnik, Projekt*

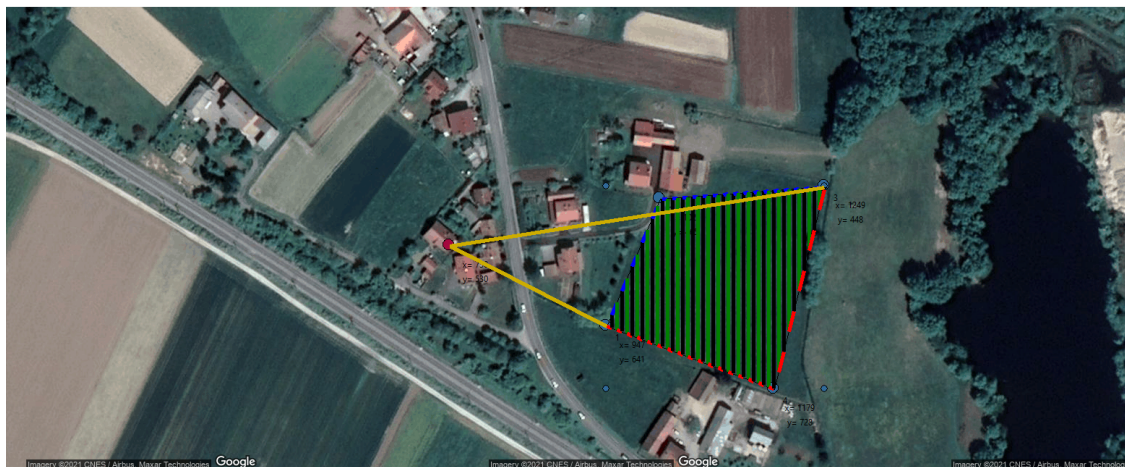
*Mentorja: red. prof. dr. Dušan Gleich (Univerza v Mariboru, Fakulteta za elektrotehniko, računalništvo in informatiko), doc. dr. Simon Klančnik (Univerza v Mariboru, Fakulteta za strojništvo)*

## **Povzetek**

V projektni nalogi je predstavljen RTK sistem s centimetrsko položajno natančnostjo in namenska aplikacija, ki omogoča označevanje poljubnega območja na zemljevidu in generiranja najkrajše poti za prelet brezpilotnega zrakoplova.

Prvi del projekta je zajemal vzpostavitev RTK sistem na brezpilotnem zrakoplovu v treh različnih načinih delovanja. V načinu »Rover«, kjer naprava simpleRTK2B daje trenutne koordinate neposredno preko USB komunikacije v računalnik, za korekcijo koordinat pa uporablja zemeljske baze. V načinu »Rover-Base« (enako kot v predhodnem načinu) koordinate neposredno pošilja v računalnik, korekcijske koordinate pa prejme od naše simpleRTK2B baze. V zadnjem načinu »Rover position in Base« pa lahko trenutno lokacijo brezpilotnega zrakoplova spremljamo v realnem času, če smo povezani na simpleRTK2b bazo z USB kablom, zrakoplov pa je lahko od baze oddaljen tudi za več kot deset kilometrov.

Drugi del projekta zajema izdelavo aplikacije, preko katere imamo dostop do Google Maps satelitskih slik, omogočeno nam je označevanje preletnega območja, generiranja najkrajše poti in prikazovanje trenutnih koordinat, na katerih se zrakoplov nahaja. Aplikacija, ki smo jo poimenoval SmartPath, je v celoti napisana v programskem jeziku Python in se uporablja v paketu Tkinter.



Ustvari pot preleta
Poveži
Briši
Prekini povezavo
X(mm) Y(pixel) ZOOM
3000 1 18
Start
Parametri
4_1_2020_Size640_ZOOM18.png 4_1_2020_Size640_ZOOM19.png 4_1_2020_Size640_ZOOM20.png C_B_C_Size640_ZOOM18.png D
COM18 - USB Serial Device
Trenutna lokacija
stop
Posodobi
Lat 45.9286326
Lon 15.5804708

Slika 1: SmartPath aplikacija

Vir: lasten

**Ključne besede:** RTK, simpleRTK2B, brezpilotni zrakoplov

# TRAJNOSTNO VZDRŽLJIVOSTNO TESTIRANJE ELEKTRO HIDRAVLIČNEGA AKTUATORJA

ALEKS PETROVIČ IN MIHAEL JANEŽIČ

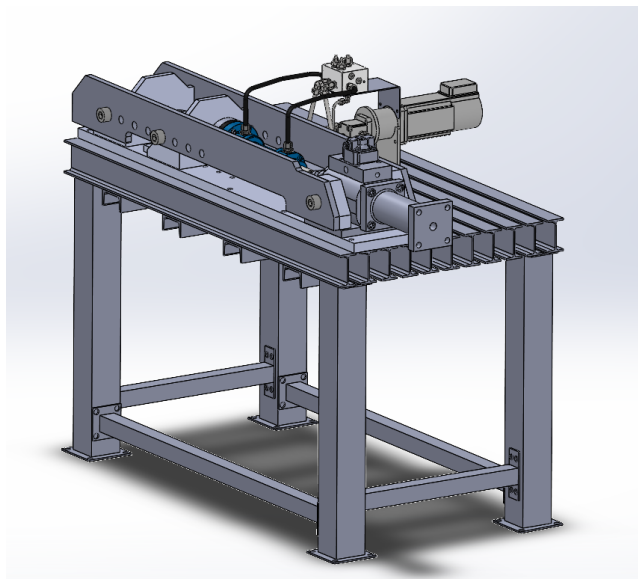
*2. letnik, Projekt*

*Mentorja: red. prof. dr. Darko Lovrec (Univerza v Mariboru, Fakulteta za strojništvo),  
red. prof. dr. Riko Šafarič (Univerza v Mariboru, Fakulteta za elektrotehniko, računalništvo  
in informatiko)*

## Povzetek

Direktno vodenje hidravličnih valjev brez krmilnega ventila neposredno s črpalko, poenostavljeno **EHA** (Elektro Hidravlični Aktuator), predstavlja novejši koncept energetske varčnega vodenja valjev brez dušilnih izgub. Kljub določenim prednostim pa takšen način krmiljenja valja predstavlja svojevrstne izzive, med katere spadajo predvsem majhna količina hidravlične tekočine, nagnjenost k pregrevanju sistema ter slabša filtracija tekočine. Da bi podrobneje spoznali problematiko trajnega obratovanja takšnih sistemov, je smiselno izvesti trajno testiranje, ki bo posnemalo delovanje **EHA** sistema v realnosti.

V ta namen smo zasnovali preskuševališče elektro hidravličnih aktuatorjev brez krmilnega ventila, ki temelji na trajnem obratovanju oz. pomikanju in obremenjevanju z regulirano silo. Bremensko enoto predstavlja hidravlični valj, krmiljen s servo ventilom, ki deluje v zaprto zračni regulaciji sile, ki jo zaznavamo z merilno celico sile. Sistem je zasnovan na osnovi krmilnika **Beckhoff** ter programskega paketa **TWINCAT 3**, ki nam omogoča izvedbo realno-časovnega dela programa, kakor tudi izvedbo vmesnika človek-stroj (HMI) na isti enoti.



**Slika 1: Model naprave za trajnostno vzdržljivostno testiranje**

Vir: lasten

**Ključne besede:** trajnostno vzdržljivostno testiranje, Bremenska enota, krmilnik Beckhoff, HMI, uporabniški vmesnik, Elektro – hidravlični aktuator



# RAZVOJ TELEMETRIJE KRMILNE ENOTE MOTEC IN KOMUNIKACIJE Z VOZNIKOM FORMULE STUDENT

TADEJ MAUKO, TINE MASILO IN NICK DROBEC

*2. letnik, Projekt*

*Mentorji: red. prof. dr. Darko Lovrec, doc. dr. Vito Tič (Univerza v Mariboru, Fakulteta za strojništvo) in red. prof. dr. Riko Šafarič (Univerza v Mariboru, Fakulteta za elektrotehniko, računalništvo in informatiko)*

## **Povzetek**

Cilj projektne naloge je razvoj telemetrije za prenos podatkov senzorjev nameščenih na vozilu Formula Student. S tem omogočimo sprotno spremljanje delovanje motorja in obnašanja avta na stezi preko računalnika. Zraven že nujno nameščenim senzorji na motorju, potrebnih za njegovo pravilno obratovanje, smo namestili še dodatne senzorje za boljše razumevanje dinamike vozila med vožnjo. Podatki, zajeti med vožnjo, kot so: kot zasuka volana, temperaturna razlika po širini pnevmatike, pomik vzmetenja, sile skozi obese in navor v volanski gredi, ne služijo le za boljše razumevanje delovanja vozila in optimizacijo nastavitev podvozja za določeno progo, ampak tudi kot osnova za nadaljnji razvoj vozila.

Za zajemanje in boljši pregled nad poslanimi podatki smo razvili tudi uporabniški vmesnik. V vmesniku se preko sprejemnika in oddajnika z MOTEC-om povežemo preko serijske komunikacije RS232. Zaradi omejitve gibanja smo izvedli testno povezavo komunikacije in uspešno testirali prenos podatkov in njihov prikaz v uporabniškem vmesniku. Realen test telemetrije na vozilu ob delovanju motorja bo izveden po sprostitvi ukrepov pandemije.

**Ključne besede:** Formula Student, telemetrija, merilni sistemi, uporabniški vmesnik





# SNOVANJE KRMILNIKA SISTEMA GORIVNIH CELIC V ELEKTRIČNEM VOZILU

GORAN MUNĐAR, FILIP MEHMETOVIĆ IN ŽIGA BALAJC  
*2. letnik, Projekt*

*Mentorji: izr. prof. dr. Miran Rodič (Univerza v Mariboru, Fakulteta za Elektrotehniko, Računalništvo in informatiko), izr. prof. dr. Uroš Župerl (Univerza v Mariboru, Fakulteta za Strojništvo), Dražhen Baketarič (AVL – AST)*

## **Povzetek**

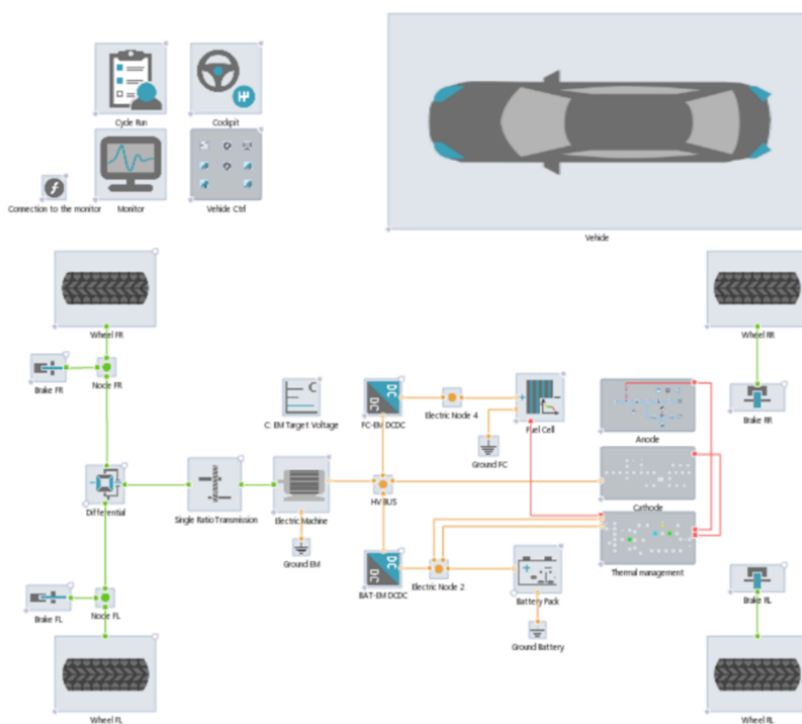
Za vozila z večjo maso in/ali dosegom (tovornjaki, letala, avtobusi na dolgih progah) se gorivne celice pojavljajo kot zanimiv alternativni pogon. Cena in masa baterij bosta namreč za tovrstna vozila tudi v prihodnosti predstavljali pomembno omejitev in jih postavljali v manj konkurenčen položaj v primerjavi z vozili na motorje z notranjim zgorevanjem. Gorivna celica kot kompleksen sistem zahteva kompleksen krmilnik, s katerim krmilimo delovanje celice in pripadajočih pomožnih komponent (ventili, črpalke, ...) ter razdelitev potrebne moči za pogon med gorivno celico in baterijo.

S programsko opremo podjetja AVL (AVL CRUISE™ M) smo zasnovali in izdelali celovit model krmilnika vozila s pogonom na gorivne celice. Krmilnik je namenjen učinkovitemu vodenju vozila pri različnih voznih režimih (ciklih). Razviti model omogoča testiranje postopkov vodenja s stališča vodljivosti in energijske učinkovitosti (vključena je tudi poraba goriva – vodika).

Uporabljen je bil obstoječi model gorivnih celic podjetja AVL, ki je bil nadgrajen z dodatnimi modeli pomožnih komponent ter pripadajočih senzorjev in aktuatorjev. Delo je bilo razdeljeno na tri dele: električni, temperaturni in tlačni del. Vsak del smo razvijali posebej, na koncu pa vse skupaj združili v celoten model.

Predvideli smo tudi potrebne vozne cikle za simulacijo porabe energije (NEDC in podobne). Delovanje je bilo preizkušeno z uporabo verificiranih simulacijskih orodij

podjetja AVL. V okviru projekta smo sodelovali s podjetjem AVL, podružnico v Mariboru.



Slika 1: Model električnega vozila na gorivne celice v programu Cruise M

Vir: lasten

**Ključne besede:** gorivne celice, PEMFC, modeliranje, FCEV, močnostna elektronika, tlak, termika, optimizacija

# VERIFICATION OF ELECTRIC VEHICLES FUEL CELL CONTROLLER MODELS

ESSAM HASHEM  
*2<sup>nd</sup> grade, Projekt*

*Mentors: doc. dr. Miran Rodič (University of Maribor, Faculty of Electrical Engineering, Computer Science and Informatics), izr. prof. dr. Uroš Župerl (University of Maribor, Faculty of Mechanical Engineering)*

## Abstract

In the frame of the project my work was focused on modelling and simulation of the hydrogen fuel cell system. Static and dynamic model Of the fuel cell were applied, simulated and validated. The hydrogen fuel cell (PEMFC) dynamic model from the Matlab Simulink library was used. The purpose of the dynamic model is to simulate the control of the DC-link voltage of the hybrid system consisting of fuel cell, battery and an induction motor serving as load. The fuel cell is connected to the DC-link with the unidirectional boost converter, and the fuel cell output current is controlled. The battery is connected to DC-link with the bidirectional synchronous DC-DC converter, output current of the battery is controlled, and the desired current is the output of the DC-link voltage controller. All controllers are of the PI type. The idea is to provide the average value of the current from the fuel cell, whereas the variations of load power are covered from the battery. In the case of higher load current consumed by the load (motor) required current is provided from the battery and in the case of low load power the fuel cell excess current is used to store the energy into the battery. In the simulation of the dynamic model, the output voltage and current of the fuel cell can be observed. It is also possible to monitor the state of the charge, voltage and current of the battery. Additionally, a static model of the fuel cell was created using the same parameters as the ones used in the dynamic model of the hydrogen fuel cell.

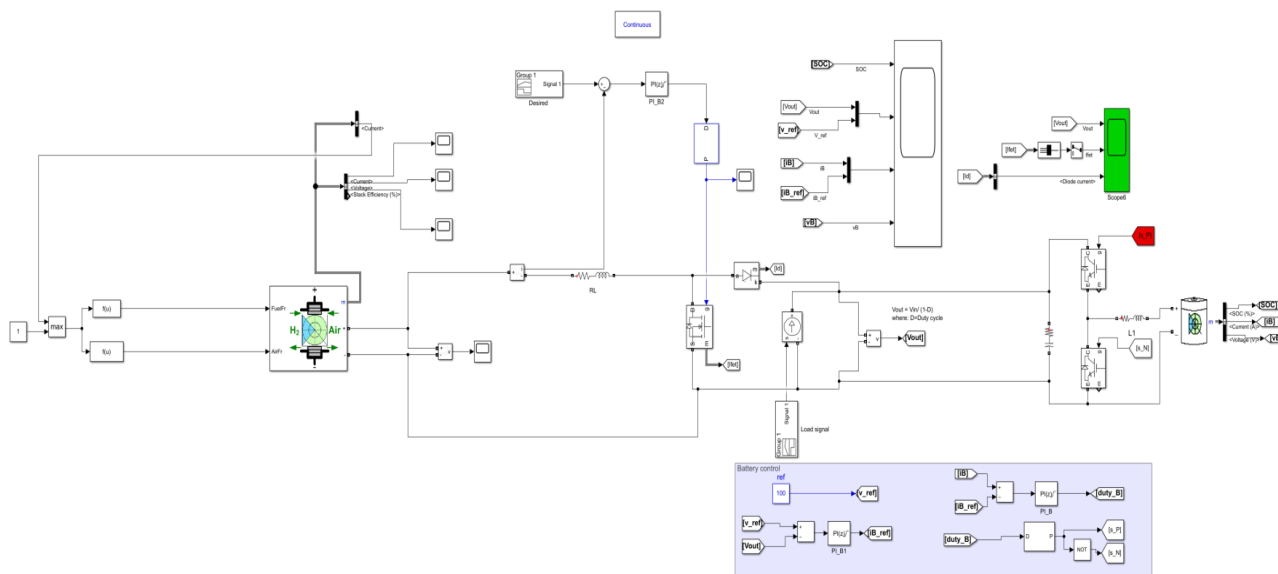


Figure 1: Dynamic hydrogen fuel cell model

Source: own

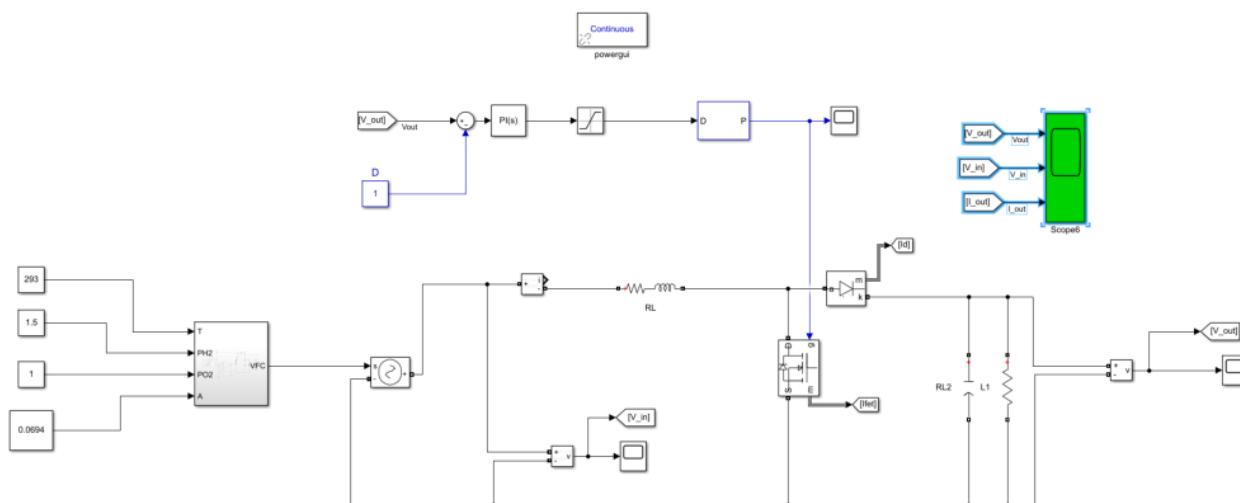


Figure 2: Static hydrogen fuel cell model

Source: own

**Keywords:** fuel cell, model, model-based, control, Matlab/Simulink

# RAZVOJ SISTEMA MIKRO-FLUIDNEGA MOTORJA: VPLIV PARAMETROV MEHANSKEGA OJAČEVALNIKA NA OJAČENJE POMIKOV

MATJAŽ MALOK IN MARKO DUKARIČ

*2. letnik, Projekt*

*Mentorja: red. prof. dr. Riko Šafarič (Univerza v Mariboru, Fakulteta za elektrotehniko, računalništvo in informatiko), red. prof. dr. Darko Lovrec (Univerza v Mariboru, Fakulteta za strojništvo).*

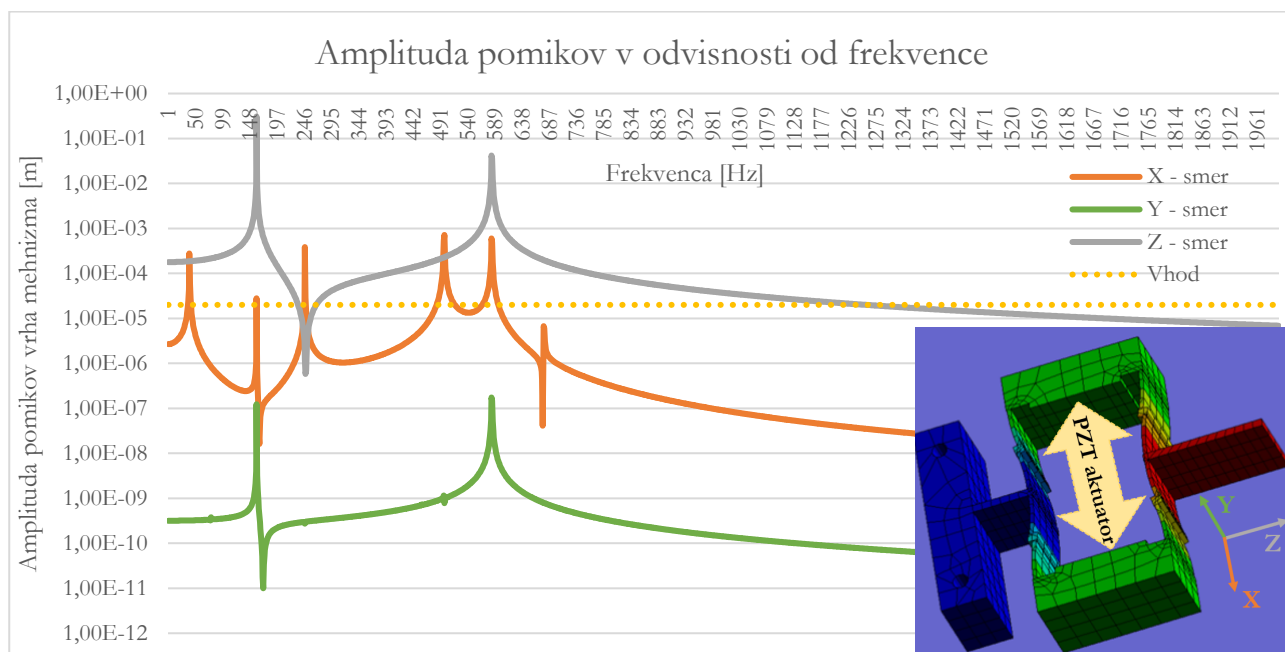
## **Povzetek**

Gradnja mikrofluidnega motorja se izvaja na objektnem stekelcu, ki je pritrjeno na vibracijsko napravo. Vibracijsko napravo sestavljata dva pravokotno pritrjena mehanska ojačevalnika, ki ju aktiviramo z dvema piezoelektričnima (PZT) aktuatorjema. Mehanske ojačevalnike moramo uporabiti, saj so amplitude pomikov, ki jih zagotavljajo piezoelektrični aktuatorji, brez mehanskih ojačevalnikov premajhne za delovanje mikrofluidnega motorčka, za delovanje katerega potrebujemo čim večjo amplitudo pomikov ob čim višji frekvenci. Tako smo v okviru projekta preučili vpliv parametrov mehanskega ojačevalnika na izhodne pomike mehanskega ojačevalnika.

Simulacije smo izvajali v programu ANSYS. S statično analizo smo določili statično ojačenje amplitude pomikov. Nato smo poiskali lastne frekvence mehanizma ter pogledali njegove lastne oblike nihanja. Pri tem smo ugotovili, da ne prihaja do ojačenj pomikov le v Z smeri, kakor smo želeli, ampak tudi v X smeri in manj izrazito v Y smeri (slika 1). Potek oblik lastnih nihanj in njihovo zaporedje je enako ob spreminjanju dimenzij posameznih parametrov. Simulacije smo izvedli z različnimi parametri posameznih dimenzij mehanskega ojačevalnika (slika 2). Pri tem smo ugotovili, da z večjo širino dobimo manjše statično ojačenje in nižje lastne frekvence. Z večanjem dolžine mehanizma se povečuje statično ojačenje, a se drastično znižujejo lastne frekvence. Višina mehanizma ter odmik povezav od roba mehanizma nimata vpliva na delovanje mehanizma. Z

večanjem širine dela za PZT aktuatorjem dobimo večje statično ojačenje, a nižje in širše lastne frekvence. Z debelejšimi povezavami dobimo nižjo ojačenje, a so lastne frekvence mnogo višje. Optimalna dolžina dela med povezavama mehanizma je 6 mm, nad oz. pod to dolžino vrednost ojačenje upada. Z večanjem te dolžine dobivamo višje resonančne frekvence. Dolžini l1 in l3 morata biti enaki, da dobimo največje statično ojačenje. Ko je zamik med povezavama 0,3 mm dobimo največje statično ojačenje, ko pa 0,55 mm pa dobimo najvišje lastne frekvence. Z uporabo materiala lahko močno vplivamo na statično ojačenje in lastne frekvence. Najmanjše statično ojačenje in najnižje lastne frekvence dobimo z uporabo plastik, večje z uporabo kovin in pa največje z uporabo stekla.

Tako smo ugotovili, da lahko s spreminjanjem posameznih dimenzij mehanskega ojačevalnika dosežemo manjše spremembe ojačenj in lastnih frekvenc naprav izboljšavam, ki jih je možno doseči s spremembo materiala.



**Slika:** Amplituda izhodnih pomikov pri referenčnem mehanizmu v odvisnosti od frekvence ter rezultat iz simulacije z označenimi koordinatnimi osmi. Vhod predstavlja pomike v X smeri generirane s piezoelektričnim (PZT) aktuatorjem.

Vir: lasten

**Ključne besede:** mehanski ojačevalnik, ojačenje pomikov, dinamična analiza, frekvenčna analiza, lastne frekvence, optimizacija parametrov

# KOOPERACIJA ROBOT-ROBOT S KOLABORATIVNIMI ROBOTI

ANTON MATAJ, VID OSTROŠKO IN DIMITRIJE PRELEVIČ

*2. letnik, Projekt*

*Mentorja: izr. prof. dr. Aleš Hace, asist. Rok Pučko (Univerza v Mariboru, Fakulteta za elektrotehniko računalništvo in informatiko), izr. prof. dr. Karl Gotlib (Univerza v Mariboru, Fakulteta za strojništvo)*

## **Povzetek**

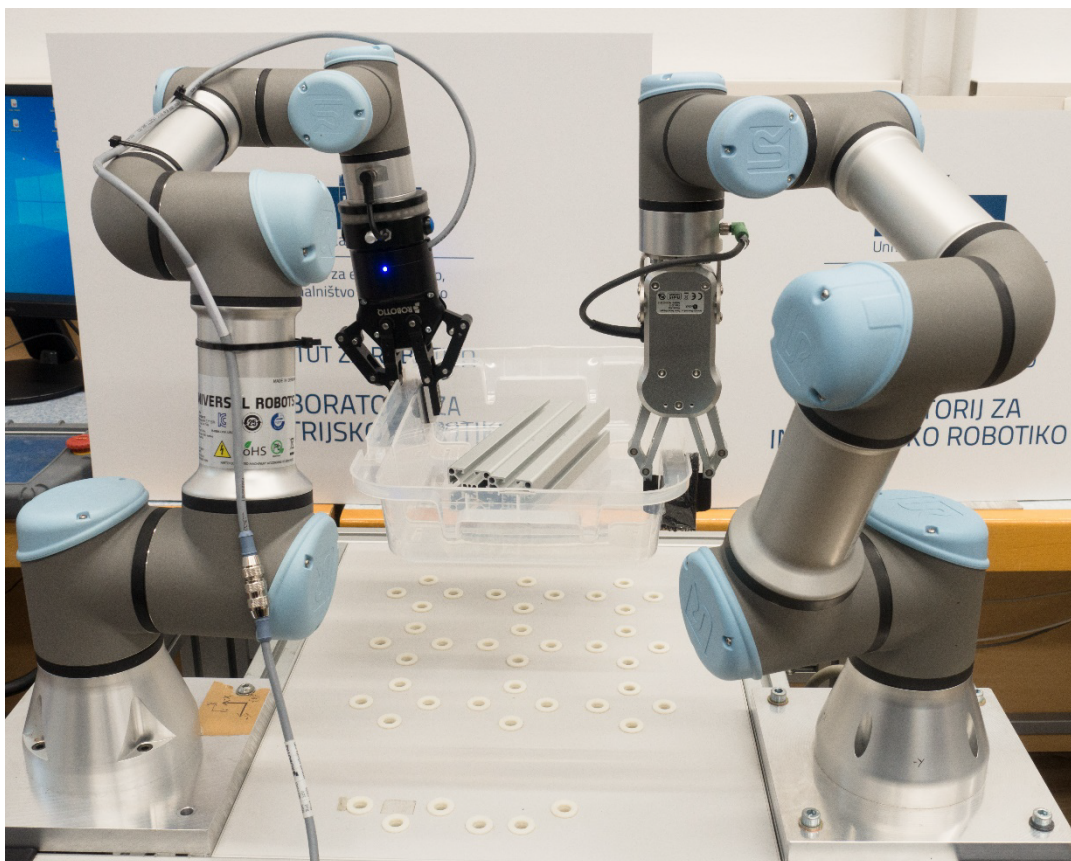
Robotizirani procesi v industriji občasno zahtevajo sodelovanje dveh robotskih manipulatorjev, pri čemer se lahko uporabijo dvoročni roboti (ang. bimanual robot) ali pa kooperacija dveh ločenih robotov. V okviru projekta smo z uporabo dveh UR3 kolaborativnih robotov demonstrirali kooperacijo med robotoma pri operaciji skupnega prenašanja tovora. Naloga robotov je bila kooperativna manipulacija objekta, ki ga sicer en sam robot ne zmore, zaradi prevelike mase ali pa neustreznih dimenzij.

Prijemanje je izvedeno tako, da se teža objekta enakomerno razdeli na oba robota ter, da s prijemanjem ne omejimo skupnega manipulacijskega prostora. Uporabljena je konfiguracija »gospodar-suženj« (ang. Master-slave), pri kateri je en od robotov izbran kot vodilni, drugi pa mu sledi. Vodilni robot narekuje gibanje objekta in s tem tudi sledilnega robota, pri čemer je vodilni togo voden po položaju, sledilni pa mora svoje gibanje ustrezno prilagajati in hkrati zadostiti zahtevam naloge.

S pomočjo virtualnega okolja robotskega simulatorja CoppeliaSim smo načrtovali sočasno manipulacijo objekta z dvema robotoma, pri čemer smo trajektorijo gibanja programsko določili le vodilnemu robotu, sledilni pa se je prilagajal. V simulacijskem okolju sta bila oba robota položajno vodena, česar zaradi prevelikih odstopanj ni bilo mogoče uresničiti s fizičnima robotoma.



Na realnem sistemu kooperacije dveh UR3 robotov je sledilni robot voden po sili s funkcijo »Force Mode«. Togost objekta, ki ga robota prenašata, predstavlja povezavo med robotoma in močno vpliva na kakovost sledenja robota. Izvedli smo več eksperimentov z objekti različne togosti, prilagajanjem parametrov vodenja po sili sledilnega robota ter lastnosti trajektorije gibanja vodilnega robota.



Slika 1 Prikaz kooperacije med dvema UR3 robotoma na mobilni platformi v LIR

Vir: lasten

**Ključne besede:** vodilni robot, sledilni robot, kooperacija, položajno vodenje, vodenje po sili



# UČENJE KOLABORATIVNEGA ROBOTA Z DEMONSTRACIJO PREKO KAMERE KINECT

TOMO PŠENIČNIK, NEJC TOVORNIK

*2. letnik, Projekt*

*Mentorji: izr. prof. dr. Aleš Hace, asist. Rok Pučko,  
Saša Stradovnik (Univerza v Mariboru, Fakulteta za elektrotehniko, računalništvo in informatiko),  
izr. prof. dr. Uroš Župerl (Univerza v Mariboru, Fakulteta za strojništvo)*

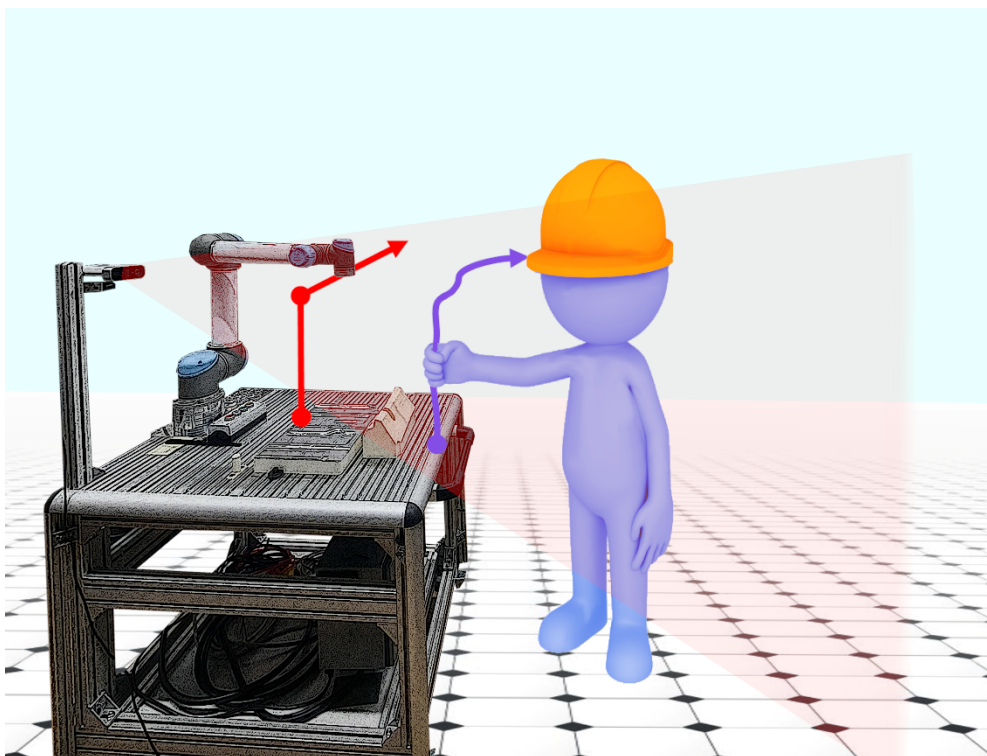
## **Povzetek**

Z razvojem kolaborativnih robotov se razvijajo tudi nove oblike programiranja, ki želijo približati uporabo robotov širši množici ljudi. V ta namen novi načini programiranja temeljijo predvsem na uporabi človeku naravnih oblik komuniciranja, kot so kretnje, fizična interakcija, glasovni ukazi itd., za kar pa je potrebno vključiti tehnologije, ki omogočajo zajem le-teh.

Cilj projekta je bil zasnovati sistem učenja, kjer s kretnjami rok demonstriramo enostavnejše robotske naloge pobiranja in odlaganja. Za prepoznavo uporabnika in njegovih kretenj smo uporabili kamero Kinect Azure proizvajalca Microsoft. Uporabnik z dvigom in spustom leve roke sproži ter zaustavi zajemanje trajektorije desne roke, ki kaže željeno pot premika predmeta. Ob zajemanju koordinat desne roke prav tako izvajamo računanje hitrosti premika, ki nam omogoči določitev glavnih vmesnih točk, med katerimi se bo gibal robot. Prepoznavo uporabnika, zajemanje točk, sprotno glajenje poti in shranjevanje podatkov trajektorije v datoteko smo izvedli v programu VisualStudio, naknadno obdelavo in simulacijo robota pa smo izvedli s programoma Matlab in CoppeliaSim.

V programu Matlab smo izdelali aplikacijo, ki uvozi datoteko, dodatno filtrira in na podlagi pragovne vrednosti določi glavne vmesne točke, po katerih se bo premikal robot. Uporabnik ima znotraj aplikacije možnost ročno popraviti pragovno vrednost in preveriti, ali se bo robot varno premikal med točkami. Po končanem pregledu se izvede izračun

inverzne kinematike, določi tip giba med točkami in izvede simulacija v programu CoppeliaSim.



**Slika 1: Učenje robota z demonstracijo**

Vir: lastni

**Ključne besede:** Kinect Azure, kolaborativni robot, Visual Studio, Matlab, CoppeliaSim, programiranje

# Preučevanje delovanja HASEL umetne mišice in možnost realizacije

MATIJA RAMŠAK, GREGOR OSET IN KRISTJAN FIŠER

*2. letnik, Projekt*

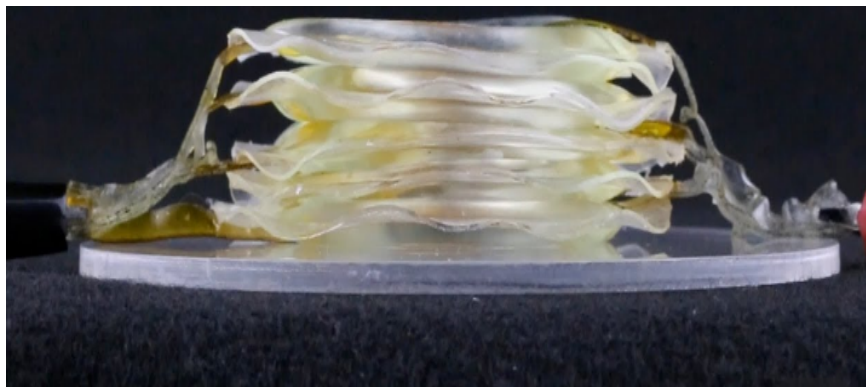
*Mentorji: izr. prof. dr. Karl Gotlib, asist. dr. Timi Karner (Univerza v Mariboru, Fakulteta za Strojništvo), izr. prof. dr. Aleš Hacı (Univerza v Mariboru, Fakulteta elektrotehniko, računalništvo in informatiko)*

## **Povzetek**

V okviru projekta smo si zadali cilj izdelati umetno mišico. Idejo smo dobili ob prebiranju že obstoječih člankov na spletu, ki govorijo o izdelavi in delovanju umetne mišice HASEL. Umetna mišica se uporablja v mehki robotiki za manipuliranje izdelkov, kjer ni potrebna velika sila in natančnost. S povezovanjem več mišic v verigo oziroma s pravilno geometrijo umetne mišice lahko dosežemo dokaj velike sile (5 kg) in pa tudi premike. V okviru projekta smo se soočili z raziskovanjem delovanja in izdelavo lastnega prototipa mišice. Izbirali smo med materiali, ki so lahko dostopni na trgu. Poskusili smo razne folije, na koncu pa uporabili skrčno folijo. Poizkusili smo tudi z različnimi olji, kot so parafinsko olje in navadno kuhinjsko olje. Raziskovali smo tudi razne postopke izdelave mišice (varjenje z vročo žico, varjenje z modelom, varjenje z vročim zrakom). Svojo mišico smo testirali, vendar odziv ni bil takšen kot smo ga želeli. Zaradi prenizke napetosti se mišica ni popolnoma skrčila, kot smo pričakovali, ampak je bil viden le majhen premik mišice.

Slika 1 prikazuje umetno mišico ki smo jo zasledili v članku.

Slika 2 prikazuje umetno mišico ki smo jo izdelali v okviru projekta.



**Slika 1: Hasel mišica**

Vir: <https://onlinelibrary.wiley.com/doi/full/10.1002/adma.202003375>



**Slika 2: Lasten prototip mišic**

Vir: Lasten

**Ključne besede:** HASEL mišica, folija, elektrode, izolator, varjenje

# RAZVOJ SISTEMA ZA INTERAKTIVNO UPRAVLJANJE PROCESOV MONTAŽE

UROŠ VINCETIČ  
*2. letnik, Projekt*

*Mentorja: red. prof. dr. Tone Lerber (Univerza v Mariboru, Fakulteta za logistiko), doc. dr. Darko Hercog (Univerza v Mariboru, Fakulteta elektrotehniko, računalništvo in informatiko)*

## Povzetek

V okviru projektne delo smo si zastavili cilj, da izboljšamo obstoječi sistem za upravljanje montaže, ki je bil razvit in izdelan v okviru lastnega diplomskega dela z naslovom »Razvoj in izvedba prototipnega sistema za vizualno komisioniranje«. Ta sistem je sestavljen iz: delovne mize (sestavljene iz standardnih aluminijastih profilov, delovne površine in dve polici z zabojčki s potrebnimi sestavnimi deli), »pick-to-light« modulov (krmilniki z zaslonom na dotik, povezani prek USB kabla na računalnik, vse vgrajeno v 3D tiskano ohišje) ter aplikacije na osebem računalniku (izdelane v razvojnem okolju LabVIEW), ki vodi ves proces montaže in služi kot vizualna predstavitev sestave.

Ta sistem smo nadgradili s sledečimi spremembami: drugi krmilnik brez zaslona na dotik (bolj robusten, primernejši za industrijsko okolje), tipko za možnost potrditve vzetih delov, tiskano vezje za povezavo električnih vodnikov za napajanje krmilnika, tipke in samega krmilnika (tako smo pridobili na kompaktnosti modula, ter zagotovili dobre kontakte med povezovalnimi žicami) ter tem spremembam prilagojeno 3D tiskano ohišje. Nato smo sistem nadgradili še z zmožnostjo brezžične posodobitve strojno-programске opreme (firmware), ter z implementacijo detekcije objektov.

Brezžičen prenos programa smo omogočili z implementacijo algoritma OTA (ang. Over The Air), ki skrbi, da je krmilnik ves čas pripravljen, da se prek internetne povezave nanj poveže uporabnik in naloži nov program iz razvojnega okolja na lokalnem računalniku ali prek spletnega brskalnika. Detekcija objektov pa služi preverjanju ali je delavec vzel pravi sestavni del, kot mu je aplikacija narekovala. Ta podsistem je sestavljen iz kamere in

algoritma na računalniku, ki v živo preverja ali se izbran del ujema s pričakovanim., ki ga narekuje LabVIEW aplikacija.

**Ključne besede:** montaža, interaktivno, vizualno, krmilnik, detekcija

# 10. LETNA KONFERENCA

## MEHATRONIKE 2021:

### ZBORNİK POVZETKOV ŠTUDENTSKIH

### PROJEKTOV

JANEZ POGORELC<sup>1</sup>, ALEŠ HACE<sup>1</sup> IN UROŠ ŽUPERL<sup>2</sup>

<sup>1</sup> Univerza v Mariboru, Fakulteta za elektrotehniko, računalništvo in informatiko, Maribor, Slovenija

E-pošta: janez.pogorelc@um.si, ales.hace@um.si

<sup>2</sup> Univerza v Mariboru, Fakulteta za strojništvo, Maribor, Slovenija

E-pošta: uros.zuperl@um.si

**Povzetek** Fakulteta za elektrotehniko, računalništvo in informatiko (FERI) in Fakulteta za strojništvo (FS) Univerze v Mariboru (UM) sta edini v Sloveniji, ki izvajata kakovostne samostojne študijske programe Mehatronike na dodiplomski univerzitetni in visokošolski strokovni 1. stopnji ter na podiplomski magistrski 2. stopnji študija. Diplomirani inženir Mehatronike je v domači industriji, še bolj pa v naši soseščini, izredno in vedno bolj iskan profil, strokovnjaki na tem področju pa sodelujejo v proizvodnji in razvoju najsodobnejših mehatronskih izdelkov. Študijski programi Mehatronike na Univerzi v Mariboru se odlikujejo s projektno orientiranim načinom izobraževanja, kjer študenti delajo v skupinah na različnih praktičnih mehatronskih problemih. Rezultate svojega projektne dela predstavijo študenti javno konec zimskega semestra in konec študijskega leta na Letni konferenci Mehatronike. Tako so letos na konferenci predstavili (organizirani v dveh delih 18. 2. 2021 in 23. 6. 2021) 44 projektov, od tega 17 projektov študenti visokošolskega strokovnega programa, 16 projektov študenti univerzitetnega dodiplomskega študijskega programa in 11 projektov študenti podiplomskega magistrskega študijskega programa. Povzetke teh projektov smo zbrali v pričujoči zbornik, kjer so predstavljene osnovne informacije, več podrobnosti pa so ekipe študentov predstavile na javni konferenci. Vsi projekti so zanimivi, zato vas v imenu organizatorjev FERI in FS vabimo, da se udeležite tudi naslednje Letne konference Mehatronike!

**Ključne besede:**

mehatronika,  
robotika,  
avtomatika,  
industrija,  
študentski  
projekti



# 10<sup>TH</sup> ANNUAL CONFERENCE OF MECHATRONICS 2021: BOOK OF ABSTRACTS, STUDENT PROJECTS

JANEZ POGORELC<sup>1</sup>, ALEŠ HACE<sup>1</sup> & UROŠ ŽUPERL<sup>2</sup>

<sup>1</sup> University of Maribor, Faculty of Electrical Engineering, Computer Science and Informatics,  
Maribor, Slovenia

E-mail: janez.pogorelc@um.si, ales.hace@um.si

<sup>2</sup> University of Maribor, Faculty of Mechanical Engineering, Maribor, Slovenia  
E-mail: uros.zuperl@um.si

**Abstract** Faculty of Electrical Engineering and Computer Science (FERI) and Faculty of Mechanical Engineering (FS) of University of Maribor (UM) are the only ones in Slovenia to conduct quality independent study programmes of Mechatronics at the undergraduate academic, higher professional and at the postgraduate master level. The study programs of Mechatronics at the University of Maribor are distinguished by the project-oriented method of education, where students work in groups on various practical mechatronic problems. The results of their project work are presented at the end of the study semesters at the Mechatronics Annual Conference. Thus, this year, our students at the conference organized (in two parts on February 18 and on June 23, 2021) represent a total of 44 projects. 17 projects will be represented by students of a professional study programme, 16 projects by students of an academic undergraduate study program, and 11 projects by students of the postgraduate master study programme.

**Keywords:** mechatronics, robotics, automatization, industry, student projects

Summaries of these projects have been gathered in the present Proceedings where basic information is shown. More details you can find out at the conference. All of the projects are interesting and therefore we invite you on behalf of the FERI and FS organizers to take part in next year's Annual Mechatronic Conference!

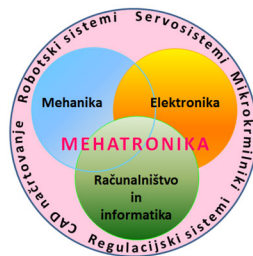






Univerza v Mariboru

Fakulteta za elektrotehniko,  
računalništvo in informatiko  
Fakulteta za strojništvo



10. letna konferenca mehatronike 2021  
Maribor, 23.6.2021