



Naprava za avtomatsko rezanje žice

Raziskovalna naloga

PODROČJE: elekrotehnika, elektronika

Mentor: Kramer Gregor

Avtor: Alen Jakopič

Mestna občina Celje, Mladi za Celje

Celje, 2021

KAZALO VSEBINE

KAZALO VSEBINE	2
KAZALO SLIK	4
1 POVZETEK	5
2 UVOD	6
2.1 Predstavitev raziskovalnega problema	6
2.2 Delovanje naprave	7
2.4 Hipoteze	7
2.5 Opis raziskovalnih metod	7
3 NAČRTOVANJE	8
4 IZDELAVA ELEKTRIČNEGA DELA	9
4.1 Načrt za električno vezje	9
4.2 Izbera elementov in testiranje	10
4.2.1 Arduino in dodatki	11
4.2.2 Motorja	11
4.2.3 Napajanje	11
4.3 Vezanje elementov	12
5 ELEKTRO ELEMENTI	17
5.1 Arduino MEGA 2560	17
5.2 L298N modul	17
5.3 Koračni motor NEMA 17	18
5.4 LCD ekran 2004	19
5.5 Tipkovnica 4 x 4	19
5.6 Napajalnik DC 12V 5A	20
5.7 LM 2596 DC-DC pretvornik	20
5.8 XL 4005 DC-DC pretvornik	21
6 IZDELAVA PROGRAMA	22
7 STROJNA IZVEDBA	29
7.1 Potek strojne izvedbe	29
7.2 Izdelava posameznih elementov	30
7.2.1 Izdelava plošče	30
7.2.2 Izdelava drsnika za zgornje kolo	31
7.2.3 Izdelava nosilcev za koračna motorja	33
7.3 Izdelava elementov na stružnici	35
7.3.1 Izdelava pogonskega kolesa	35
7.3.2 Izdelava kolesa za drsnik	39
7.4 Izdelava škarij in vzdoda	40

NAPRAVA ZA AVTOMATSKO REZANJE ŽICE

7.5 Sestava vseh elementov.....	43
8 VARNOST	44
9 NAVODILA ZA UPORABO	45
9.1 Ročno	45
9.2 Elektro	45
9.3 Programsко	45
10 RAZPRAVA	47
11 ZAKLJUČEK.....	48
12 VIRI IN LITERATURA.....	49

KAZALO SLIK

<u>Slika 1: Naprava za rezanje žice</u>	6
<u>Slika 2: Prvotno električno vezje</u>	9
<u>Slika 3: Načrt električnega vezja</u>	10
<u>Slika 4: Sprednja doza</u>	13
<u>Slika 5: Doza s postavljenimi elementi</u>	13
<u>Slika 6: Ozemljitev plošče</u>	14
<u>Slika 7: Vezava v sprednji dozi</u>	14
<u>Slika 8: Postavitev elementov v zadnji dozi</u>	15
<u>Slika 9: Rezkanje odprtin na pokrovu zadnje doze</u>	15
<u>Slika 10: Izdelava žic za krmiljenje</u>	16
<u>Slika 11: Električna vezava na zadnji strani plošče</u>	16
<u>Slika 12: Plošča arduino MEGA</u>	17
<u>Slika 13: L298N modul</u>	18
<u>Slika 14: Koračni motor NEMA 17</u>	18
<u>Slika 15: LCD ekran</u>	19
<u>Slika 16: Tipkovnica</u>	19
<u>Slika 17: Napajalnik</u>	20
<u>Slika 18: LM 2596 DC-DC pretvornik</u>	20
<u>Slika 19: XL4005 DC-DC pretvornik</u>	21
<u>Slika 20: Program 1. del</u>	23
<u>Slika 21: Program 2. del</u>	24
<u>Slika 22: Program 3. del</u>	25
<u>Slika 23: Program 4. del</u>	26
<u>Slika 24: Program 5. del</u>	26
<u>Slika 25: Program 6. del</u>	27
<u>Slika 26: Program 7. del</u>	27
<u>Slika 27: Program 8. del</u>	28
<u>Slika 28: Naprava za rezanje žice 3D model</u>	29
<u>Slika 29: Kovinska plošča-nebarvana</u>	31
<u>Slika 30: Kovinska plošča-temeljna barva</u>	31
<u>Slika 31: Notranji del drsnika za zgornje kolo s privarjeno osjo</u>	32
<u>Slika 32: Varjenje zunanjega dela drsnika za zgornje kolo</u>	32
<u>Slika 33: Drsnik z zgornjim kolesom 3D model</u>	33
<u>Slika 34: Nosilec motorja za rezanje z motorjem</u>	33
<u>Slika 35: Rezkanje izreza za vodilo nosilca za zgornje kolo</u>	34
<u>Slika 36: Priprava na varjenje nosilca za gonilni motor</u>	34
<u>Slika 37: Nosilec za gonilni motor z motorjem</u>	36
<u>Slika 38: Pogonsko kolo z osovinama-3D model</u>	36
<u>Slika 39: Struženje zunanje osi pogonskega kolesa</u>	36
<u>Slika 40: Načrt zunanje osovine pogonskega kolesa</u>	37
<u>Slika 41: Struženje notranje osi pogonskega kolesa</u>	37
<u>Slika 42: Načrt notranje osi pogonskega kolesa</u>	38
<u>Slika 43: Struženje gonilnega plastičnega kolesa</u>	38
<u>Slika 44: Načrt osi kolesa za zgornje kolo</u>	39
<u>Slika 45: Struženje plastičnega kolesa za drsnik</u>	39
<u>Slika 46: Škarje</u>	40
<u>Slika 47: Škripci in osovini</u>	41
<u>Slika 48: Končni vzvod</u>	42
<u>Slika 49: Varnostna zaščita</u>	43

1 POVZETEK

Za raziskovalno nalogo sem se odločil izdelati napravo, ki bo na mero rezala žice ali pa kakšen drug element. V raziskovalni nalogi sem najprej opisal princip delovanja naprave, ki sem jo izdelal. Potem sem postavil cilje naloge in pa hipoteze, kako naj bi naprava delovala kot končni izdelek. Naprej je potem opis načrtovanja, oziroma postopek načrtovanja, ki sem ga ubral, to pa se deli na električno in strojno načrtovanje. Opisana je izdelava električnega dela in vezanje električnih komponent, ki pa imajo naprej tudi vsaka svoj opis. Naprej je načrtovanje programa in pa program, ki je uporabljen pri napravi. Opisana je tudi izdelava strojnega dela izdelka, to je izdelava individualnih elementov in potem združitev vseh v eno konstrukcijo. Potem je opisana še združitev in uskladitev strojnega dela z električnim. Na koncu pa so potrjene oziroma ovržene hipoteze, postavljene na začetku in pa ugotovitve ob koncu raziskave.

1.1 Ključne besede

- Naprava za avtomatsko rezanje žice
- Arduino Mega 2560
- Vodila
- Škarje

2 UVOD

2.1 Predstavitev raziskovalnega problema

Izdelal bom izdelek, katerega osnovna funkcija bo rezanje žice na željeno dolžino. Ideja za to se mi je porodila pri vezanju večje elektro omare. Opazil sem namreč, da včasih potrebujemo veliko žic iste dolžine, ki si jih po navadi za lažje delo vnaprej narežemo. Ob tem sem razmišljal, da bi lahko delo olajšalo oziroma pospešilo, če bi mi te žice lahko pripravil nekdo drug. Dobil sem idejo, da bi izdelal napravo s funkcijo priprave žic. Tako sem se potem lotil izdelave izdelka. Glavni namen naprave bi bil rezanje žic za potrebe vezanja elektro omar, lahko pa jo uporabimo tudi za druge namene, na primer za rezanje vrvi na mero, rezanje termo skrčk, rezanje manjših cevi in tako dalje.



Slika 1: Naprava za rezanje žice

2.2 Delovanje naprave

Naprava deluje tako, da žico oziroma element, ki ga želimo imeti narezanega, najprej nastavimo na držalo, potem pa napeljemo element skozi vodila in skozi gonilna kolesa tako, da je tik pred začetkom škarij. Ko začne naprava obratovati, se nastavljena dolžina odmeri od položaja, ki smo ga nastavili, se pravi od začetka škarij. Naprava ima dve funkciji. Prva in osnovna funkcija je to, da žico odmeri na takšno dolžino, ki jo nastavimo, na koncu pa jo odreže. Lahko nastavimo, da se to ponovi tolikokrat, kolikor žic oziroma kosov želimo imeti. Druga funkcija pa je namenjena izdelovanju žic za vezanje elektro omare. Deluje enako kot prva, le da žici na vsaki strani zareže izolacijo, tako da jo lahko potem brez pripomočkov odstranimo in na žico pritrdimo votlici.

2.4 Hipoteze

- Naprava odmeri in odreže žico
- Odstopanje dolžine odrezane žice ni večje od 10 % nastavljene dolžine
- Odreže željeno oziroma nastavljeno količino žic
- Cena vseh potrebnih elementov je manj kot 200 €

2.5 Opis raziskovalnih metod

Moja raziskovalna naloga se je začela z idejo. Na začetku sem si načine izvedbe in zasnova naloge poskušal zamisliti, nato sem si nekatere ideje začel zapisovati, na koncu pa sem naredil načrt. Kasneje sem določene načrte spremenil, da je bilo vse lažje izvedljivo. Pri izdelavi sem potreboval veliko orodij, saj je bilo za izdelavo potrebnega kar nekaj strojnega dela. Uporabljal sem stružnico, CO₂ varilnik, rezkar in nekaj drugih ročnih orodij. Potrebnega je bilo tudi nekaj programiranja in vezanja.

3 NAČRTOVANJE

Celotno načrtovanje se deli na dva dela, in sicer na strojno načrtovanje in na izdelavo načrta za elektro povezave.

Na začetku, pred načrtovanjem, je bilo potrebno pridobiti nekaj idej, zato sem pred načrtovanjem iskal ideje v svoji okolici. Ko sem dobil kako idejo, sem si le to zapisal, narisal in izdelal 3D model v programu za lažjo predstavo.

Na tak način sem si pomagal pri oblikovanju realne slike. Določene ideje, so se namreč šele po premisleku in opazovanju skice izkazale za neizvedljive, bodisi zaradi prevelikih stroškov, ki bi pri tem nastali, ali zaradi prevelike kompleksnosti, ali pa preprosto zato, ker na koncu ne bi bile tako učinkovite. Zaradi tega sem potem iskal druge rešitve, ki so se mi zdele primernejše, učinkovitejše in enostavnejše.

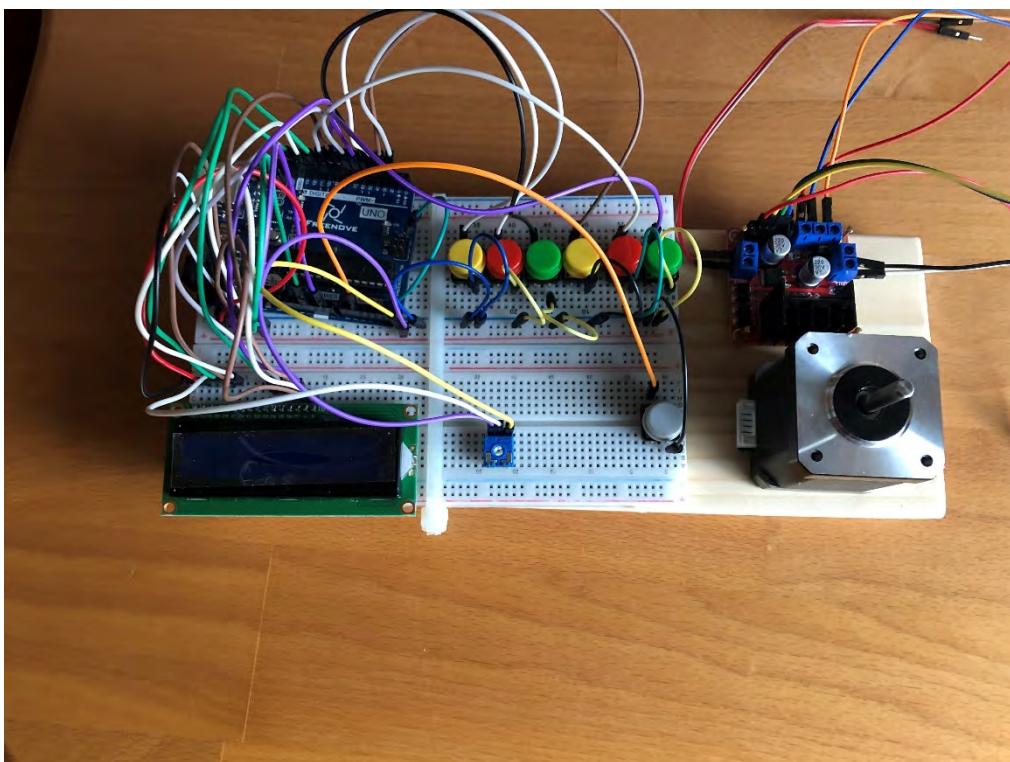
Ko sem na koncu prišel do kakšne ideje, ki se je zdela smiselna, sem jo zapisal in začel ustvarjati skice, na katerih sem elemente združeval. Kasneje sem te skice spremenjal in določene stvari dodajal. Ko sem dobil neko končno idejo, kako naj bi bil izdelek videti, sem narisal obliko in razporeditev elementov in tako sem dobil končno obliko, iz katere sem potem naprej izhajal.

Na podlagi nekega zanesljivejšega načrta sem potem začel z iskanjem ustreznih elementov in elektronskih komponent. Ko sem vedel, katere elektronske komponente bom uporabil sem lahko začel z izdelavo električnega načrta. Tega sem najprej narisal, kasneje pa sem načrt naredil v programu.

4 IZDELAVA ELEKTRIČNEGA DELA

4.1 Načrt za električno vezje

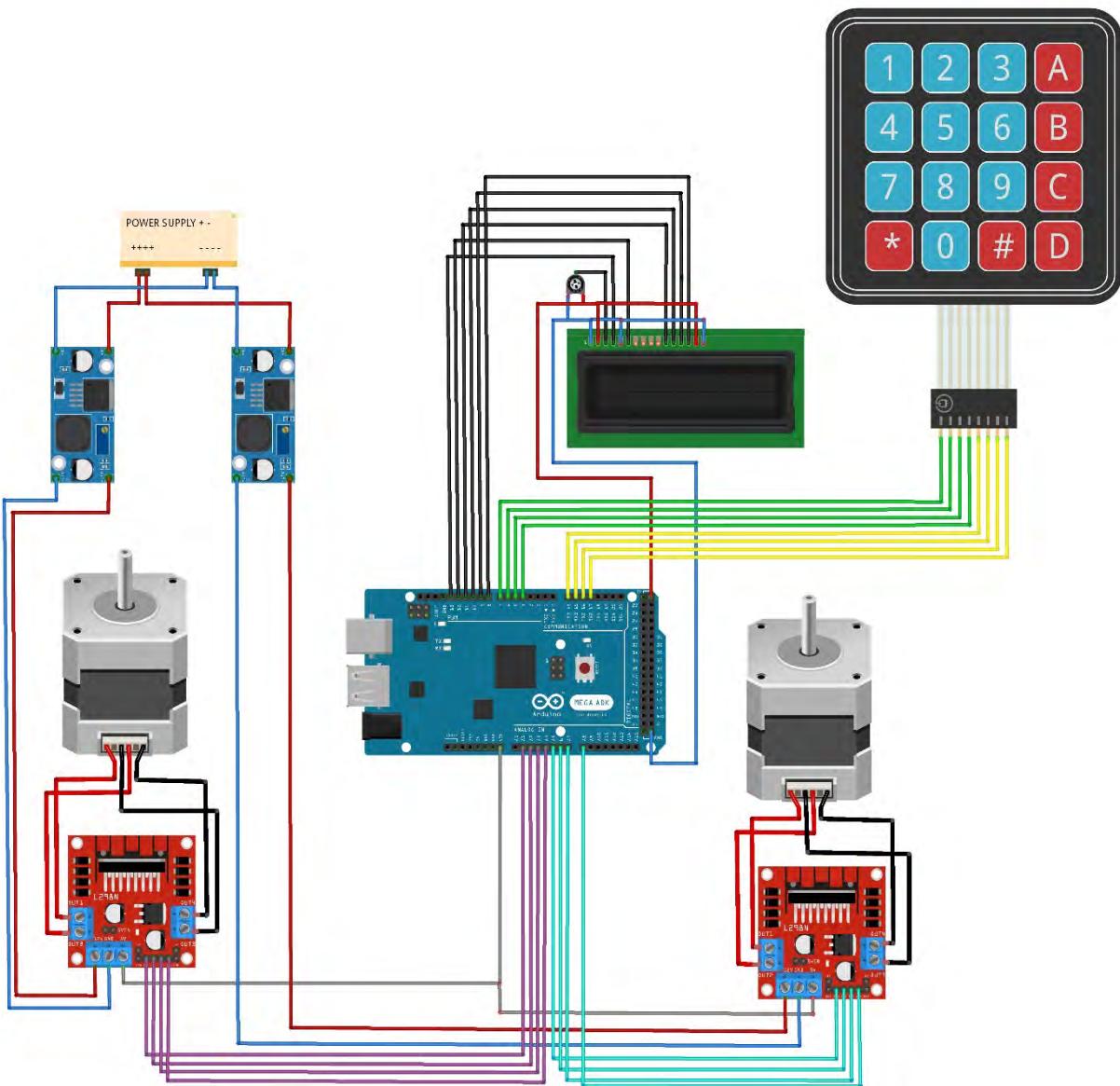
Še preden sem dobil konkretno idejo o tem, na kakšen način bom naredil električno izvedbo svoje naprave, sem z enostavnimi vezavami in programiranjem testiral različne načine. Prvotno sem imel idejo, da bom za vnašanje vrednosti uporabljal tipke (in ne tipkovnice). Tako sem sestavil vezje in napisal program za ta način izvedbe. Kmalu pa se je pokazalo, da to ni ravno najboljši način, saj je bilo nastavljanje vrednosti na tak način nekoliko bolj zamudno in zato ne ravno najbolj optimalno (vrednosti so se višale s postopnim klikanjem na tipke).



Slika 2: Prvotno električno vezje

Zato sem se odločil, da bom namesto tipk uporabil tipkovnico. Ta način se mi zdi najbolj optimalen, saj je vnašanje najhitrejše in tudi najbolj enostavno (vrednosti vnesemo le z pritiskom na številke). Tako sem prišel do neke ideje, na podlagi katere sem lahko začel izdelovati načrt.

Predpogoj za izdelavo elektro načrta je bila izbira elektro komponent, zato sem pred načrtovanjem dobro razmislil, katere komponente bom potreboval. Ko sem izbral ustrezne komponente, za katere sem menil, da bodo najboljše za mojo nalogo, sem najprej narisal preprosti načrt na papirju, potem pa sem v programu Fritzing narisal vse povezave med elementi. V program sem moral določene komponente dodati oziroma prenesti s spleta, tako da sem lahko naredil identično vezje, kot sem ga imel v resnici.



Slika 3: Načrt električnega vezja

4.2 Izbira elementov in testiranje

Za svoj projekt sem poskušal najti takšne elemente, ki so lahko dostopni, tako da bi ob kakšni okvari enostavno prišel do njih. Poskušal pa sem se tudi izogniti previsokim stroškom zato sem za svojo napravo uporabil cenovno ugodne komponente.

4.2.1 Arduino in dodatki

Zaradi vseh navedenih razlogov sem se odločil, da bom za osnovo uporabil mikrokrmlnik arduino, potem pa sem iskal module, ki so z njo kompatibilne.

Potem sem uporabil dva modula L298n motor driver, saj je ta modul enostaven za uporabo in pa deluje brez težav. Ob dolgi uporabi se je sicer pokazalo, da se čip na vezju pregreva, zato sem na oba modula namestil večji hladilnik, tako da je zdaj segrevanje minimalno tudi ob daljši uporabi.

Izbral sem tudi LCD ekran, ki je 4-vrstični in 20-stolpčni za bolj pregleden zapis. Za vnos podatkov sem izbral tipkovnico s šestnajstimi tipkami, saj se zdi za mojo napravo najbolj primerna.

Potem pa sem se še odločil za nekaj dodatnih komponent:

- potenciometer, s katerim lahko nastavljam kontrast LCD ekrana;
- reset tipko, ki sem jo zvezal na mikrokrmlnik tako, da se le ta resetira ob njenem pritisku;
- dvopolno stikalo, s katerim vklapljam ozioroma izklapljam dovodno izmenično napetost.

4.2.2 Motorja

Uporabil sem dva koračna motorja, in sicer oba tipa NEMA 17. Ta tip motorja sem izbral, ker je lahko dostopen, poceni in ima dobre karakteristike.

Pri izbiri gonilnega motorja nisem imel večjih težav. Potreben je bil le koračni motor z visoko natančnostjo, tako da sem lahko nastavljal število korakov glede na potrebni kot zasuka. Navor pri gonilnem motorju ni tako pomemben, saj mu ni potrebno premagovati velike sile.

Drug motor, pa služi temu, da z vrtenjem spreminja pozicijo škarij, s katerimi se reže žica. Ker morajo škarje žico odrezati, je tu potrebna večja sila. Če bi želet motor neposredno povezati na škarje, bi potreboval zelo močen motor, ki pa je tudi precej dražji. Zato sem se odločil, da bom uporabil navaden koračni motor, katerega moč bom povečal z vzvodom.

Tako sem potem izdelal vzvod iz štirih škipcev in tako potrebno silo zmanjšal za štirikrat in podaljšal pot, ozioroma povečal število obratov motorja za štirikrat.

4.2.3 Napajanje

Za napajanje sem izbral 60-vatni napajalnik, katerega izhodna napetost je nastavljiva od 11 do 13 volтов in je enosmerna, njegova napajalna napetost pa je 230 ali pa 110 volтов izmenične napetosti.

Vsak motor je potreboval drugačno napetost. Gonilni motor ne potrebuje velike moči, saj tam ni velike obremenitve, zato sem tam potreboval manjšo napetost, da sem se izognil nepotrebnem toku. Motor za vzvod pa je kljub vzvodu potreboval nekoliko večjo moč, saj mora premagovati večjo silo. Zato je le ta potreboval višjo napetost.

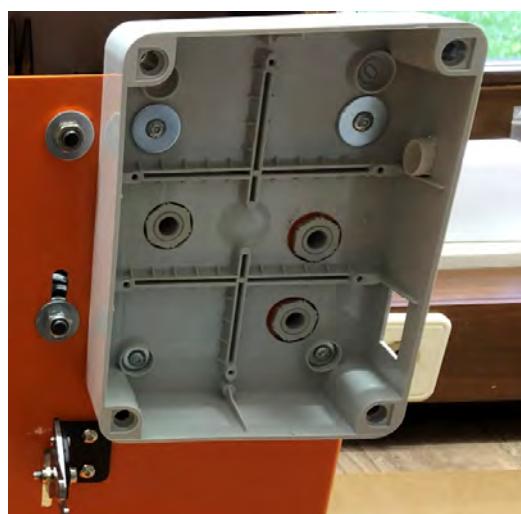
Ker nisem želel napajanja posebej za vsak motor, sem uporabil pretvornika napetosti, s katerima sem izhodno napetost zmanjšal. Uporabil sem dva modula in sicer LM2596, katerega najvišji dovoljeni tok ne presega treh amperov in pa XL4015, katerega najvišji dovoljeni tok znaša pet amperov. Oba modula delujeta na isti princip in sicer tako, da vhodno napetost zmanjšata na željeno.

Pri nastavitevah napetosti sem poskušal dobiti čim manjši tok, saj sem s tem zmanjšal porabo in gretje, zato sem v fazi izdelave opravil kar nekaj meritev in prilagajanj, da sem prišel do neke ustrezne vrednosti. Za gonilni motor sem tako nastavil napetost na 6 voltov in tako dobil porabo okoli 1.2 ampera, za drug motor pa sem napetost nastavil na 9 voltov in tako dobil porabo okoli 2 ampera.

4.3 Vezanje elementov

Na samem začetku elektro izvedbe sem najprej izdelal idejno vezje na testni plošči in začel s pisanjem programa. Kasneje sem potem določene elemente in vezave dodajal ali spreminal. To me je potem vodilo do končne ideje, ki se je zdela najbolj primerna. Vezje sem potem spremenil in ga ponovno zvezal na testni plošči. To vezje sem vzporedno s programom potem testiral in spreminal. Ko sem vse skupaj dokončal in sem bil zadovoljen z vezavo in programom, sem vezje na testni plošči razdrl in začel z pripravo in vezanjem na plošči.

Na začetku sem moral pripraviti doze. Za sprednjo stran kovinske plošče sem izbral dozo velikosti 200x150x70. Vanjo sem dal napajalnik, mikrokrmilnik in glavno stikalo, potem pa sem na vrata doze namestil LCD ekran, tipkovnico, reset tipko in potenciometer za nastavitev kontrasta na ekrantu. Na dozi je bilo potrebno izdelati izrez za glavno stikalo ob strani, izrez za zračenje doze in pa izrez za LCD ekran ter žice od tipkovnice na vratih. Skozi hrbtno stran doze in kovinsko ploščo je bilo tudi potrebno izvrtati luknje za uvodnice, tako da sem lahko povezal elemente z zadnje strani plošče na elemente v dozi. Izvrtal sem tri luknje, in sicer eno za dovodni kabel, drugo za vodnike iz napajjalnika in tretjo za krmilne žice.



Slika 4: Sprednja doza

Napajalnik in mikrokrmlnik sem pritrdil na ploščo iz pleksi stekla, ki sem jo pred tem odrezal na mero in jo potem privijačil na dozo. Nato sem skozi uvodnico napeljal dovodni kabel. Fazo in nulo sem priklopil na stikalo, z zaščitnim vodnikom pa sem šel direktno na usmernik. Iz stikala sem nato nulo in fazo priklopil na usmernik. Na plus in minus usmernika sem priklopil po dva vodnika in sicer za napajanje mikrokrmlnika ter za napajanje modulov za motorja.



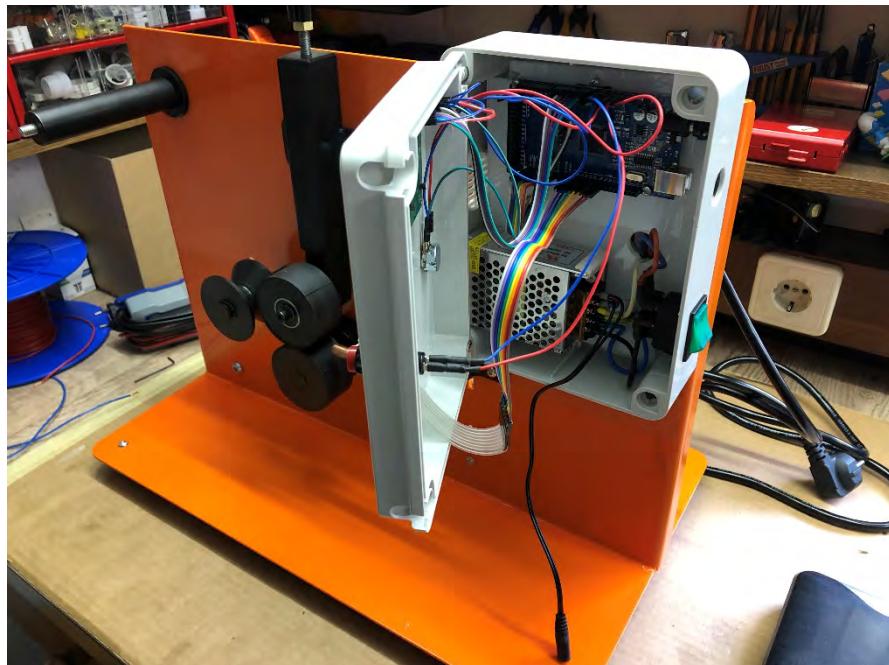
Slika 5: Doza s postavljenimi elementi

Z vodnikoma za napajanje modulov in z ozemljitvijo sem potem šel skozi uvodnico na drugo stran plošče. Zaščitni vodnik sem zadaj privijačil na ploščo in jo s tem ozemljil.



Slika 6: Ozemljitev plošče

Ko sem imel osnovne elemente v sprednji dozi postavljene, sem na njena vrata pritrdil LCD ekran, tipkovnico, tipko in potenciometer. Znotraj doze sem potem te elemente povezal z mikrokrmlnikom. Pine mikrokrmlnika, na katere sem povezal elemente, sem si zapisal na papir tako, da sem jih lahko potem v programu nastavil.



Slika 7: Vezava v sprednji dozi

Po izdelavi sprednje doze pa sem moral narediti še dozo, v katero sem dal, L298N modula in pa napetostna pretvornika za koračna motorja. Za to sem potreboval manjšo dozo, ki sem jo postavil na zadnjo stran kovinske plošče.

Na tej dozi je bilo potrebno na vsaki strani narediti izrez za hladilnika L298N modulov, ker sem se odločil, da bosta ta zunaj doze zaradi boljšega odvajanja toplote. Moral sem izvrnati tudi tri luknje za uvodnice. Ko sem imel to narejeno, sem znotraj doze privijačil vse elemente in namestil uvodnice. Potem sem na pokrovu doze z rezkarjem izrezal nekaj rež za dovod zraka, zato da se ob primeru gretja elementov topli zrak odvaja.

NAPRAVA ZA AVTOMATSKO REZANJE ŽICE



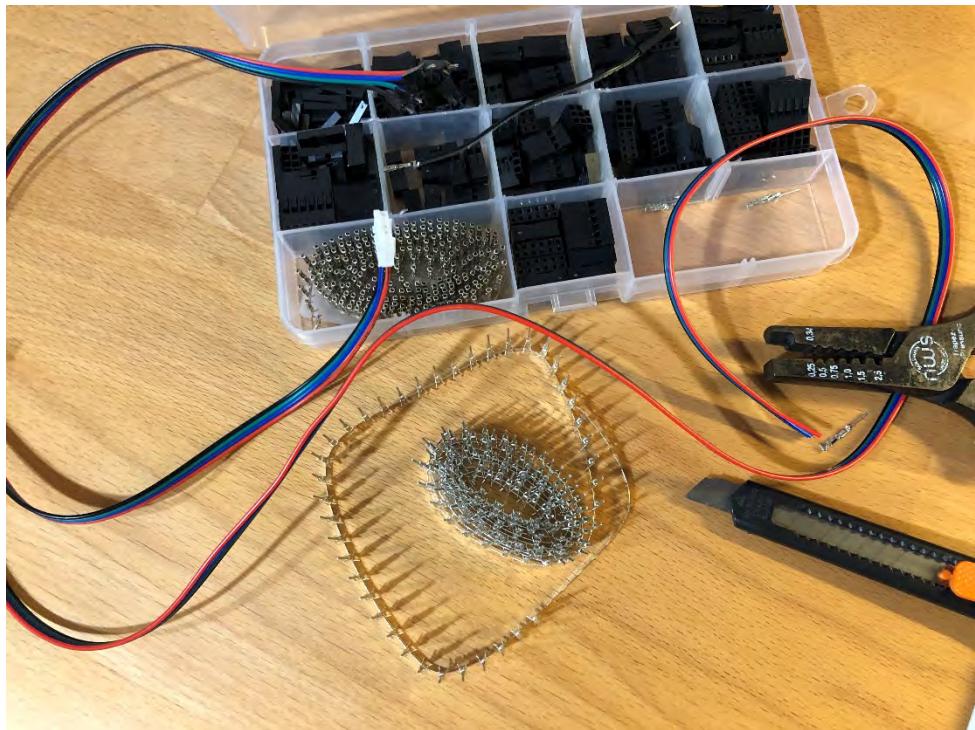
Slika 8: Postavitev elementov v zadnji dozi



Slika 9: Rezkanje odprtin na pokrovu zadnje doze

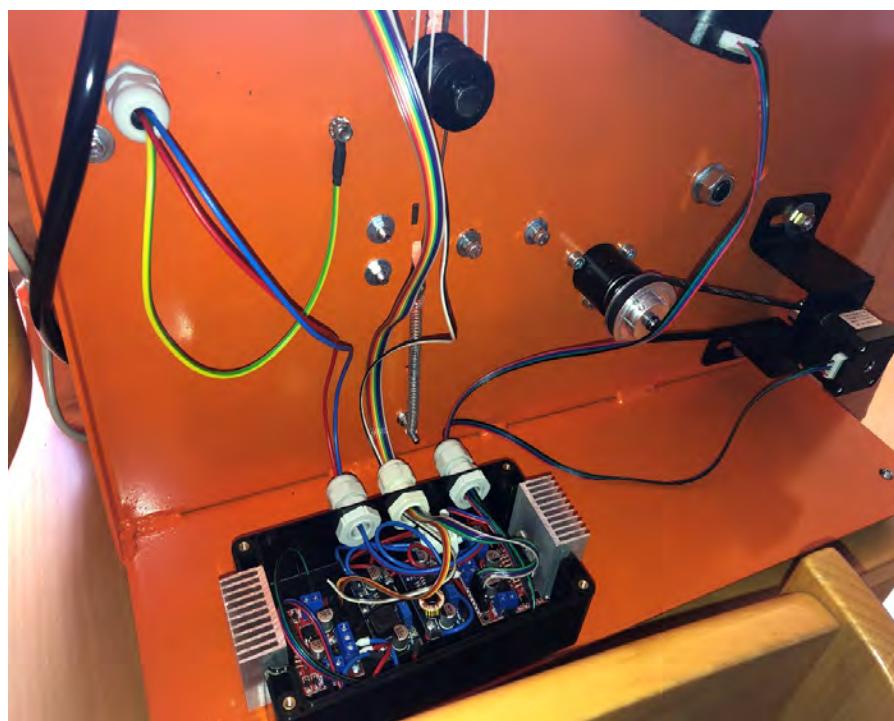
Ko sem imel pripravljeno še to dozo, sem najprej povezal pozitivni in negativni vodnik iz napajjalnika na oba napetostna pretvornika in potem iz napetostnih pretvornikov na L298N modula. Potem pa sem se lotil priprave žic za krmilje. Povezave med zadnjo dozo in mikrokrmilnikom so namreč precej dolge, zato sem moral krmilne žice izdelati sam. Prav tako sem moral izdelati povezovalne žice med L298N modulom in koračnim motorjem.

NAPRAVA ZA AVTOMATSKO REZANJE ŽICE



Slika 10: Izdelava žic za krmiljenje

Ko sem imel žice pripravljene, sem jih le še napeljal skozi uvodnice in jih priklopil na oba modula in mikrokrmilnik (glede na to, na katere pine mikrokrmilnika sem priklopil pine od L298N modulov, sem potem v programu nastavil smer vrtenja posameznega motorja). Čisto na koncu pa sem na sponke vsakega izmed L298N modulov privijačil le še žice in jih potem povezal na posamezen motor.



Slika 11: Električna vezava na zadnji strani plošče

5 ELEKTRO ELEMENTI

5.1 Arduino MEGA 2560

Arduino je ploščica tiskanega vezja, ki je namenjena za hob elektroniko. Glavni element oziroma jedro ploščice je mikrokrmilnik AT mega 2560. Poleg tega pa je na ploščici še usb vtičnica tipa B, vtičnica za DC napajanje, tipka za reset, varovalka, nekaj led diod, kondenzatorjev, uporov...Ima tudi 54 digitalnih pinov, ki jih lahko porabimo za vhode ali izhode in pa 16 analognih vhodov.

Arduino se lahko napaja na več načinov, in sicer preko usb vtičnice, preko katere nalagamo tudi program nanj, lahko pa se napaja z napetostjo med 7 in 12 voltov. Tok, ki teče skozi ploščo, ne sme presegati 200 mA.

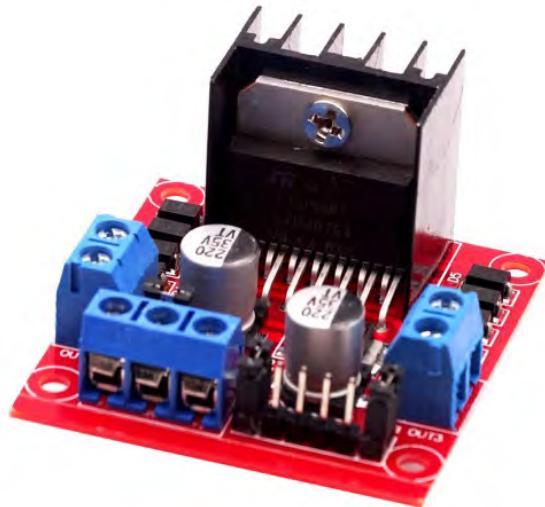
Za programiranje ima arduino svoje okolje, ki je narejeno tako, da dobro pomaga programerju. Za lažje programiranje lahko uporabljamo številčne knjižnice, ki jih imamo na voljo v okolju.



Slika 12: Plošča arduino MEGA

5.2 L298N modul

Je modul za krmiljenje elektro motorjev. Na njem lahko krmilimo do dva DC motorja na enkrat ali pa en bipolarni koračni motor. Je manjše elektronsko vezje, na katerem je več elementov. Glavni je čip, na katerega je že v osnovi pritrjen hladilnik. Na vezju pa imamo tudi nekaj sponk in pa pinov za lažjo povezovanje z krmilnikom in pa motorjem. Imo visoko delovno napetost, ki lahko znaša vse do 40 voltov, ne sme pa biti manjša od 5 voltov, skozenj pa lahko teče tok do 3 ampere.



Slika 13: L298N modul

5.3 Koračni motor NEMA 17

Nema 17 koračni motor je bipolarni koračni motor. Je precizen koračni motor z 1.8° koračnega kota. To pomeni, da naredi 200 korakov za en celoten zasuk. Vsaka faza vzame 1.2 ampera toka pri napetosti štirih voltov in dovoljuje navor 3.2 kilograma na centimeter.

Delovna temperatura takega motorja znaša od -10 pa do 40°C . Poznamo tudi različne type tovrstnih motorjev, ki pa se v osnovi razlikujejo le po velikosti in pa navoru (poznamo na primer model 49Nm ali pa 59Nm).



Slika 14: Koračni motor NEMA 17

5.4 LCD ekran 2004

LCD ekran 2004 je ekran, ki ima 20 stolpcev in pa štiri vrstice. Vsako polje posebej ima 5×8 pik, preko katerih se dobi zapis. Na zadnji strani ima 16 pinov, ki jih po večini porabimo za povezavo z mikrokrmlnikom. Z mikrokrmlnikom se ga krmili tako, da se nastavi ustrezeno polje in se vanj vnese nek določen znak, ta znak se potem v polju osvetli in dobimo izpis na ekranu.

Napaja se ga z napetostjo 5 voltov. Z spremjanjem upornosti na enem izmed pinov pa lahko nastavljamo tudi kontrast.



Slika 15: LCD ekran

5.5 Tipkovnica 4 x 4

Tipkovnica 4 x 4, je tipkovnica, sestavljena iz šestnajstih tipk, ki so razporejene 4 v vrsto in po 4 v stolpec. Ima 8 izhodnih vodnikov, ki se jih priklopi na krnilnik in se bere katera izmed tipk je pritisnjena. Vrednosti se preberejo na tak način, da en izmed pinov pove stolpec drugi pa vrstico v kateri je bila tipka pritisnjena.



Slika 16: Tipkovnica

5.6 Napajalnik DC 12V 5A

Napajalnik DC 12V 5A, je napajalnik, ki se napaja z izmenično napetostjo. Vrednost vhodne napetosti lahko nastavimo z stikalom znotraj usmernika, izbiramo lahko med 110 VAC in pa 230 VAC. Izhodna napetost je enosmerna in jo lahko z potenciometrom nastavljamo med 11 in pa 13 voltov. Napajalnik je 60-vatni, kar pomeni, da da pri 12 voltih 5 amperov toka. Je tudi zelo majhne velikosti (dimenziije: 85x60x34)



Slika 17: Napajalnik

5.7 LM 2596 DC-DC pretvornik

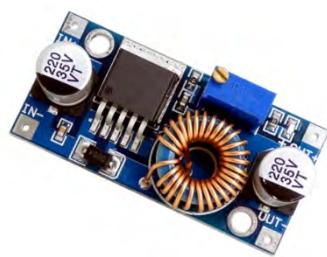
Je napetostni regulator, ki omogoča zmanjševanje vhodne napetosti. Vrednost napetosti na vhodu je lahko od 1.23 do 37 voltov. Deluje tako, da po priklopu vhodne napetosti, zmanjšujemo vrednost izhodne napetosti na željeno vrednost. Ta regulator dopušča najvišji tok treh amperov.



Slika 18: LM 2596 DC-DC pretvornik

5.8 XL 4005 DC-DC pretvornik

Deluje enako kot LM 2596 regulator, le da ta dovoljuje tok do 5 amperov.



Slika 19: XL4005 DC-DC pretvornik

6 IZDELAVA PROGRAMA

Izdelava programa za mojo napravo je potekala v več delih. Delila se je na izdelavo programa za menije na LCD ekranu, na branje znakov pritisnjениh na tipkovnici in potem shranjevanje teh v medpomnilnik in na izdelavo programa za vrtenje koračnih motorjev.

Na začetku sem za vsak tip programa naredil nek osnovni tip kode, ki sem jo potem nadgrajeval in prilagal, da je ustrezala mojemu namenu. Ko sem imel vsak tip programa narejen in sem se naučil logiko tega, sem se lotil združevanja programov. Najprej sem se lotil združitve LCD ekrana in pa tipkovnice, tako da mi je LCD izpisoval vrednosti, ki sem jih vtipkal na tipkovnici. Ker pa sem moral vtipkane vrednosti nekam shraniti, tako da si jih je program zapomnil, sem moral vse shraniti v medpomnilnik.

Na tak način sem potem vrednosti, ki sem jih z tipkovnico vnesel, lahko tudi uporabil v funkcijah za vrtenje motorjev. Na tak način sem določil število korakov motorjev in pa število ponovitev.

Med samim pisanjem programa je bilo seveda potrebno reševati mnogokatere probleme, ki so se pojavili. Pomagal sem si s pisanjem komentarjev, tako da se na koncu nisem zgubil v dolgem združenem programu.

NAPRAVA ZA AVTOMATSKO REZANJE ŽICE

```
1 #include <Keypad.h> //Knjižnica za tipkovnico
2 #include <Stepper.h> //Knjižnica za stepper motorje
3 #include <LiquidCrystal.h> //Knjižnica za LCD display
4 LiquidCrystal lcd(A5, A4, A3, A2, A1, A0); //Določimo pine za LCD
5
6 byte customChar_z[8] = { //Izdelava znaka "ž": Vsako polje LCD ekrana ima 5 x 8 manjših polj (5 vodoravno in 8 navpično).
7     0b1010,           //Mi lahko poljubno ustvarimo znak, ki ga želimo, in sicer tako, da označimo manjša polja znotraj večjega polja.
8     0b00100,          //Na tisto mesto kjer želimo, da je polje osvetljeno vpišemo vrednost 1 (0 pomeni ni osvetljeno).
9     0b11111,          //Na takšen način lahko potem poljubno ustvarimo znak, ki ga želimo.
10    0b00011,
11    0b01110,
12    0b11000,
13    0b11111,
14    0b00000
15 };
16 // Stepper motorja
17 const int stepsPerRevolution = 200; //Določimo število korakov za en obrat
18 Stepper Gonilni_motor(stepsPerRevolution, 14, 15, 16, 17); //Določimo pine za prvi motor
19 Stepper myStepper2(stepsPerRevolution, 21, 20, 19, 18); //Določimo pine za drugi motor
20
21 //LCD meniji
22 int currentScreen = 0; //Spremenljivka za trenutni ekran
23 const int numScreens = 4; //Konstanta za število ekranov, ki jih imamo
24 String screens[numScreens][3] = {"Pozdravljeni!", "ZA NADALJEVANJE", "PRITISNITE #"}, //Nastavimo nize za posamezen ekran
25 {"VNESITE DOLZINO:", "cm"}, {"VNESITE KOLICINO:", "kom."}, {"", ""}; //Nastavimo nize za posamezen ekran
26
27 //konstante in enako za vse funkcije
28 const byte NewBuffer_COLS = 16; //število mest v medpomnilniku
29 const byte maxInputSize = 16;
30 char znak; //določimo znak "znak"
31 int currentLine = 1; //Določimo spremenljivko za trenutno vrstico in jo damo na 1
32 char number; //določimo znak "number"
33
34 //dolzina function
35 int currentCol_dolzina = 0; //Trenutni stolpec za dolžino nastavimo na 0
36 char inputBuffer_dolzina[maxInputSize + 1];
37 byte currentBufferPosition_dolzina = 0; //Nastavimo trenutno pozicijo medpomnilnika dolzina na 0
38
39 //kolicina function
40 int currentCol_kolicina = 0; //Trenutni stolpec za kolicino nastavimo na 0
41 char inputBuffer_kolicina[maxInputSize + 1];
42 byte currentBufferPosition_kolicina = 0; //Nastavimo trenutno pozicijo medpomnilnika kolicina na 0
43
44 //keyboard
45 char keys[4][4] = { //Nastavimo znake za tipkovnico
46     {'1', '2', '3', 'A'},
47     {'4', '5', '6', 'B'},
48     {'7', '8', '9', 'C'},
49     {'*', '0', '#', 'D'}
50 };
```

Slika 20: Program 1. del

NAPRAVA ZA AVTOMATSKO REZANJE ŽICE

```
51 };
52 byte rowPins[4] = {2, 3, 4, 5}//Nastavimo pine za vrstice
53 byte colPins[4] = {6, 7, 8, 9}//Nastavimo pine za stolpce
54 Keypad myKeypad = Keypad(makeKeymap(keys), rowPins, colPins, 4, 4);
55 //SETUP-----
56 void setup() {
57   lcd.begin(20, 4);
58   lcd.createChar(0, customChar_z);
59   Gonilni_motor.setSpeed(100);//Nastavimo hitrost gonilnega motorja na 100
60   myStepper1.setSpeed(200);//Nastavimo hitrost drugega motorja na 200
61   lcd.setCursor(0, 2);//nastavimo začetni zapis na LCD ekranu
62   lcd.print/screens[currentScreen][1]; //to naredi v setupu zato ker se vrednosti v 2. vrstici skozi program pomikajo...zato to vpisemo v setup in je to vpisano samo do trenutka ko pritisnemo "#" ali ".."
63   lcd.setCursor(0, 3);
64   lcd.print/screens[currentScreen][2];
65 }
66 //LOOP-----
67 void loop() {
68   lcd.setCursor(3, 0);
69   lcd.print/screens[currentScreen][0];
70   char keyPressed = myKeypad.getKey(); //Ustavimo znak "keyPressed" in ga enačimo z pritiskom na tipko, s tipkovnice
71   if ((keyPressed == '#') || (keyPressed == '..')) //Če je pritisnjen znak enak # ali ..
72   {
73     lcd.clear(); //počisti LCD display
74   }
75   if (keyPressed) { //Ob pritisku na tipko
76     znak = keyPressed; //keyPressed enačimo z "znak" in
77     firstFunction(); //začne se 1. funkcija
78   }
79 }
80 //FIRST FUNCTION-----
81 void firstFunction()
82 {
83   if (znak == '#') //če pritisnemo na #
84   {
85     if (currentScreen == 1)//in če je trenutni ekran na poziciji 1,
86     {
87       currentScreen = 2; //gre ekran na pozicijo 2
88       lcd.setCursor(0, currentLine);
89       lcd.print(inputBuffer_kolicina); //in napiše vrednost, ki je halčena v medpomnilniku "inputBuffer_kolicina"
90       lcd.print(" "); //naredi presledek
91       lcd.print/screens[currentScreen][1]; //in napiše vrednost, ki jo imamo za trenutni ekran definirano v nizu("kom.")
92     }
93     else if (currentScreen == 2) //če je ekran na poziciji 2,
94     {
95       currentScreen++; //gre ekran na pozicijo 3
96       lcd.setCursor(0, currentLine);
97       lcd.print/screens[currentScreen][1]; //in napiše vrednost, ki jo imamo za trenutni ekran definirano v nizu
98     }
99   else if (currentScreen == 3)//če je ekran na poziciji 3,
100   {
101 }
```

Slika 21: Program 2. del

NAPRAVA ZA AVTOMATSKO REZANJE ŽICE

```
101     currentScreen = 1;//gre ekran na pozicijo 1
102     lcd.setCursor(0, currentLine);
103     lcd.print(inputBuffer_dolzina);//in napiše vrednost, ki je naložena v medpomnilniku "inputBuffer_dolzina"
104     lcd.print(" ");//naredi presledek
105     lcd.print/screens[currentScreen][1];//in napiše vrednost, ki jo imamo za trenuten ekran definikano v nizu("cm")
106   }
107 else //če nič od zgoraj napisanega ne velja(ko prvič pritisnemo na tipko)
108 {
109   currentScreen = 1;//gre ekran na pozicijo 1
110   lcd.setCursor(0, currentLine);
111   lcd.print(inputBuffer_dolzina);//in napiše vrednost, ki je naložena v medpomnilniku "inputBuffer_dolzina"
112   lcd.print(" ");//naredi presledek
113   lcd.print/screens[currentScreen][1];//in napiše vrednost, ki jo imamo za trenuten ekran definikano v nizu("cm")
114 }
115 }
116 if (znak == '*')//če pritisnemo na *
117 {
118   if (currentScreen == 0)//in če je trenutni ekran na poziciji 0,
119   {
120     currentScreen = 0;//se ne zgodi nobena sprememba
121   }
122 else if (currentScreen == 1)//če je trenutni ekran na poziciji 1,
123   {
124   currentScreen = 3;//gre ekran na pozicijo 3
125   lcd.setCursor(0, currentLine);
126   lcd.print/screens[currentScreen][1];//in napiše vrednost, ki jo imamo za trenuten ekran definikano v nizu
127 }
128 else if (currentScreen == 2)//če je ekran na poziciji 2,
129   {
130   currentScreen = 1;//gre ekran na pozicijo 1
131   lcd.setCursor(0, currentLine);
132   lcd.print(inputBuffer_dolzina);//in napiše vrednost, ki je naložena v medpomnilniku "inputBuffer_dolzina"
133   lcd.print(" ");//naredi presledek
134   lcd.print/screens[currentScreen][1];//in napiše vrednost, ki jo imamo za trenuten ekran definikano v nizu("cm")
135 }
136 else//če nič od zgoraj napisanega ne velja(če je trenutni ekran na poziciji 3)
137   {
138   currentScreen--;//gre ekran na pozicijo 2
139   lcd.setCursor(0, currentLine);
140   lcd.print(inputBuffer_kolicina);// in napiše vrednost, ki je naložena v medpomnilniku "inputBuffer_kolicina"
141   lcd.print(" ");//naredi presledek
142   lcd.print/screens[currentScreen][1];//in napiše vrednost, ki jo imamo za trenuten ekran definikano v nizu("kom")
143   }
144 }
145
146 if (currentScreen == 1) //Na prvi poziciji ekrana se začne funkcija v kateri vnašamo dolžino
147 {
148   dolzinaFunction();
149 }
150 if (currentScreen == 2) //Na drugi poziciji ekrana se začne funkcija v kateri vnašamo količino
```

Slika 22: Program 3. del

NAPRAVA ZA AVTOMATSKO REZANJE ŽICE

```

152     kolicinaFunction();
153 }
154 if (currentScreen == 3) //Na tretji poziciji ekranpa se začne funkcija, ki izpisuje različne izpise
155 { //glede na podatke, ki smo jih, ali jih nismo vnesli prej)
156     izbira_programaFunction();
157 }
158 }
159 }
160 // SECOND FUNCTION-----
161 void dolzinaFunction() //Deluje tako, da lahko izpisujemo samo številke. Številke izpisuje v liniji(spodnji liniji LCDja)
162 { //tukaj nudi shranjuje v medpomnilnik "inputBuffer_dolzina". Na LCD poleg števila napiše tudi "cm"(centimetrov)
163     Potem lahko z tipko 'A' zapišemo zadnjo številko v zaporedju(npr. 123, ob pritisku 'A' je 12).*/
164 {
165     number = znak; //V tej funkciji znak enačimo z number, ker bomo vnašali številske podatke
166     if ((number != '') && (number != '#') && (number != '=') && //če je pritisnjena tipka različna od tukaj vnešenih znakov
167         (number != 'B') && (number != 'C') && (number != 'D')) {
168         if (currentCol_dolzina < NewBuffer_COLS) { //če je trenutni stolpec na LCDju manjši kot vrednost v "NewBuffer_COLS"(16)
169             lcd.setCursor(currentCol_dolzina, currentLine); //nastavi kurzor na "currentCol"(0) in "currentLine"(1-druga vrsta)
170             lcd.print(number); //napiše vrednost iz "znak"(tj. tipka, ki je pritisnjena)
171             currentCol_dolzina++; //stolpec potem pomakne, da dobimo linijski izpis
172             led.print(" "); //za številom naredi presledek
173             led.print(screens[currentScreen][1]); //in napiše "cm"
174             if (currentBufferPosition_dolzina == maxInputSize) //če je vrednost v "currentBufferPosition"(trenutno je 0) manjša od "maxInputSize"(16)
175             {
176                 inputBuffer_dolzina[currentBufferPosition_dolzina] = number; //v char inputBuffer[currentBufferPosition](trenutna vrednost je 0) naloži vrednost od char number(ima vrednost vtipkanega števila)
177                 currentBufferPosition_dolzina++; //mesto v liniji pomakne za eno naprej, tako da dobimo linijski izpis tako kot na LCD
178                 inputBuffer_dolzina[currentBufferPosition_dolzina] = '\0'; // s tem dolžimo konec linije "end of the string"
179             }
180         }
181     }
182 } else if (number == 'A') { //če pritišnemo znak 'A' na tipkovnici
183
184     if (currentCol_dolzina != 0) { //s tem programu prepričujemo, da bi šel v minus(da ne izpisuje negativnih števil)
185         currentCol_dolzina--; //trenutni stolpec premakne za eno mesto nazaj
186         lcd.setCursor(currentCol_dolzina, currentLine); //kursor LCDja postavi na novo postavljeni stolpec in vrsto
187         lcd.write(" "); //in na tem mestu izpiše prazno mesto izbrise število, ki je bilo napisano v tem stolpcu
188         lcd.print(screens[currentScreen][1]);
189         lcd.setCursor(currentCol_dolzina + 3, currentLine); //kursor se pomakne na zadnje mesto zapisa pred izbrisom z "A" znakom(kjer ostaja črka m od zapisa cm)
190         lcd.print(" "); //in jo izbriše
191     }
192 } //če je pozicija v inputBuffer različna od nič, se npr.ova ponica pomakne za eno v - in tam konča niz(imbriše zadnjo število)
193 if (currentBufferPosition_dolzina != 0) inputBuffer_dolzina[--currentBufferPosition_dolzina] = '\0'; //medpomnilniku preprečimo, da bi šel v minus
194
195 }
196 }
197 //THIRD FUNCTION-----
198 void kolicinaFunction() // Te funkcije je precej podobna prejšnji. Razlikuje je v tem, da vrednosti te funkcije vnašamo v drug medpomnilnik in da imamo drugo stran na LCD displayu.
199 { //Tu namreč vnašamo kolikočino in nismo zato imeli vrednosti vpisane v tej funkciji, shranjene posebej.
200     Števila se izpisujejo v liniji in jih shranjuje v medpomnilnik "inputBuffer_kolicina". Na LCD poleg števila napiše tudi "kom."(komadov)
201     Potem lahko z tipko 'A' zapišemo zadnjo številko v zaporedju/
202 }

```

Slika 23: Program 4. del

```

203 {
204     number = znak; //znak enačimo z number, ker bomo vnašali številske podatke
205     if ((number != '') && (number != '#') && (number != '=') && //če je pritisnjena tipka različna od tukaj vnešenih znakov
206         (number != 'B') && (number != 'C') && (number != 'D')) {
207         if (currentCol_kolicina < NewBuffer_COLS) { //če je trenutni stolpec na LCDju manjši kot vrednost v "NewBuffer_COLS"(16)
208             lcd.setCursor(currentCol_kolicina, currentLine); //nastavi kurzor na "currentCol"(0) in "currentLine"(1-druga vrsta)
209             lcd.print(number); //napiše vrednost iz "znak"(tj. tipka, ki je pritisnjena)
210             currentCol_kolicina++; //stolpec potem pomakne, da dobimo linijski izpis
211             led.print(" "); //za številom naredi presledek
212             lcd.print(screens[currentScreen][1]); //in napiše "kom"
213             if (currentBufferPosition_kolicina < maxInputSize) //če je vrednost v "currentBufferPosition"(trenutno je 0) manjša od "maxInputSize"(16)
214                 inputBuffer_kolicina[currentBufferPosition_kolicina] = number; //v char inputBuffer[currentBufferPosition](trenutna vrednost je 0) naloži vrednost od char number(ima vrednost vtipkanega števila)
215                 currentBufferPosition_kolicina++; //mesto v liniji pomakne za eno naprej, tako da dobimo linijski izpis tako kot na LCD
216                 inputBuffer_kolicina[currentBufferPosition_kolicina] = '\0'; // s tem dolžimo konec linije "end of the string"
217             }
218         }
219     }
220 } else if (number == 'A') { //če pritišnemo znak 'A' na tipkovnici
221
222     if (currentCol_kolicina != 0) { //s tem programu prepričujemo, da bi šel v minus(da ne izpisuje negativnih števil)
223         currentCol_kolicina--; //trenutni stolpec premakne za eno mesto nazaj
224         lcd.setCursor(currentCol_kolicina, currentLine); //kursor LCDja postavi na novo postavljeni stolpec in vrsto
225         lcd.write(" "); //in na tem mestu izpiše prazno mesto izbrise število, ki je bilo napisano v tem stolpcu
226         lcd.print(screens[currentScreen][1]);
227         lcd.setCursor(currentCol_kolicina + 3, currentLine); //kursor se pomakne na zadnje mesto zapisa pred izbrisom z "A" znakom(kjer ostaja črka m od zapisa kom)
228         lcd.print(" "); //in jo izbriše
229     }
230     if (currentBufferPosition_kolicina != 0) inputBuffer_kolicina[--currentBufferPosition_kolicina] = '\0'; //medpomnilniku preprečimo, da bi šel v minus
231 }
232
233 //FOURTH FUNCTION-----
234 void izbira_programaFunction() //V tej funkciji imamo v primeru, da smo vnesli vse vrednosti dolžino in kolikočino za želenega programa. Izberamo lahto med navadnim programom, kjer nam npravila le odmeri
235 //in odreže željeno dolžino in kolikočino žice in pa program za ţice, ki pa deluje tako da nam odmeri nastavljeno dolžino ţice, poleg tega pa nudi na vsaki strani
236 //srečne lom izčlascje, kar nam pomaga pri odstrelitvi izčlascje. V primeru, da želimo izbrati navaden program pritisnemo tipko A na tipkovnici,
237 //za izbirko programa za ţice pa pritišnemo B. V primeru, da posabljemo vpisati kakšno vrednost(bodisi kolikočino ali pa dolžino), nas pa so program v tej funkciji spomne
238 //to naredi tako, da na stranu izpiše "VNESITE VREDNOSTI!".*/
239
240 {
241     lcd.setCursor(6, 0);
242     lcd.print("VPISTE!"); //Na LCD displayu se avtomatsko izpiše napis "VNESITE VREDNOSTI!"
243     lcd.setCursor(5, 1);
244     lcd.print("VREDNOSTI!");
245     if ((atol(inputBuffer_dolzina) != 0) && (atol(inputBuffer_kolicina) != 0)) //če vrednosti v medpomnilnikih za dolžino in kolikočino niso enake "0"(se pravi, da imamo vpisane vrednosti)
246     {
247         lcd.clear(); //počisti zapis na LCDju
248         lcd.setCursor(0, 0); //v zgornjem desnem kotu ekранa
249         lcd.print(inputBuffer_dolzina); //izpiše vrednost iz medpomnilnika kolikočina(koliko komadov nam bo izdelal)
250         lcd.print(" "); //naredi presledek
251         lcd.print("x"); //napiše x, ki ve mišljenu kot krt(operationa množenja)

```

Slika 24: Program 5. del

NAPRAVA ZA AVTOMATSKO REZANJE ŽICE

```

152 led.print(" "); //spet naredi presledek
153 led.print(inputBuffer_dolzina); //potem pa izpiše vrednost naloženo v medpomnilniku dolžina(kakšne dolžine bodo narezani komadi)
154 led.print("cm"); //na koncu izpiše cm (centimetrov).
155 led.setCursor(0, 1); //Na začetku druge vrstice
156 led.print("IZBERITE PROGRAM ! "); //napisi "IZBERITE PROGRAM !".
157 led.setCursor(0, 2); //Na začetku tretje vrstice
158 led.setCursor(0, 3); //Na začetku šette vrstice
159 led.print("Program za A"); //napisi: "Program za A"
160 led.print("Program za B"); //napisi: "Program za B"
161 led.setCursor(11, 3); //te dve vrstice so za zapis črke "Ž"
162 led.write(Byte(0));
163 led.setCursor(12, 3); //dokonča zapis v zadnji vrstici
164 led.print("Iz: B"); //tako, da na koncu dobimo zapis: "Program za žice: B"(uporabniku pove, da mora za izbiro navadnega programa pritisniti tipko A)
165
166 if (znak == 'A') //Če je pritisnjena tipka na tipkovnici enaka "A"
167 {
168     motorFunction(); //se sačne motorFunction.
169 }
170 if (znak == 'B') //Če je pritisnjena tipka na tipkovnici enaka "B"
171 {
172     motorFunction2(); //se sačne motorFunction2.
173 }
174 }
175
176 void motorFunction() //Ta funkcija je za delovanje motorja. Deluje tako, da glede na vneseno dolžino potri naredi dolžino kolikih korakov, kateri povzroči vretenje in s tem premik žice.
177 //motor pa naredi število korakov potrebnih da žica odide potem pa isto kolikih korakov naredi v obratno smer, tako da se vrne v prvotno pozicijo. Teme skupaj ponavljajo kolikor imamo vneseno kolikino komadov.)
178 {
179     for (int i = 0; i < #obolinputBuffer_kolicina; i++) //Dokler je ponovitev manj kot pa je vnesena vrednost v medpomnilniku za koliko
201     {
202         delay(500); //Med vrtenjem motorjev se na LCD izpiše napis "NAPRAVA JE POD OBRAZOVANJEM".
203         lcd.clear();
204         lcd.setCursor(6, 0);
205         lcd.print("NAPRAVA");
206         lcd.setCursor(6, 1);
207         lcd.print("JE POD");
208         lcd.setCursor(4, 2);
209         lcd.print("OBRAZOVANJEM");
210
211         Gonolini_Motor.step(atol(inputBuffer_dolsina) * 33.368); //Nastavimo število korakov. Dobimo jih iz vrednosti v medpomnilniku dolžina potem pa to vrednost pomnožimo z konstanto, s katero dobimo najbolj natančen rezultat
212         analogWrite(14, LOW); //Na navitja motorja damo vrednost "0", tako da nimajo porabe že jih ne potrebujemo
213         analogWrite(15, LOW);
214         analogWrite(16, LOW);
215         analogWrite(17, LOW);
216         delay(100);
217         myStepper2.step(200); //Drugi korάni motor naredi en obrat v eno smer
218         delay(100);
219         myStepper2.step(-200); // in en obrat v drugo(tako da se vrne v začetno položaj)-naredi toliko korakov, da se preko škarji žica odreže
220         analogWrite(18, LOW); //Na navitja motorja damo vrednost "0"
221         analogWrite(19, LOW);
222     }

```

Slika 25: Program 6. del

```

301     analogWrite(19, LOW);
302     analogWrite(20, LOW);
303     analogWrite(21, LOW);
304 } //Ko se postopek vrtenja motorjev konča ->
305 lcd.clear(); //se LCD ekran počisti
306 lcd.setCursor(2, 0);
307 led.print("PRITISNITE TIPLI"); //in izpiše "PRITISNITE TIPLI"
308 led.setCursor(2, 1);
309 led.print(" " + ali + " "); // " " ali # =
310 led.setCursor(2, 2);
311 led.print("Za ponovni vnos"); // "Za ponovni vnos"
312 // vrednosti v medpomnilniku se posabljajo ->
313 currentCol_dolzina = 0; //pozicijo kurSORja ponovno nastavimo na 0
314 currentBufferPosition_dolzina = 0; //pozicijo za zapis v medpomnilniku nastavimo na 0
315 inputBuffer_dolzina[currentBufferPosition_dolzina] = '\0'; //določimo konec linije
316
317 currentCol_kolicina = 0; //pozicijo kurSORja ponovno nastavimo na 0
318 currentBufferPosition_kolicina = 0; //pozicijo za zapis v medpomnilniku nastavimo na 0
319 inputBuffer_kolicina[currentBufferPosition_kolicina] = '\0'; //določimo konec linije
320
321 void motorFunction2() //ta funkcija deluje motorja na kakšničini, da ima žico, ki ji naprava kurSorje še na sveti atnost narezano po 1 cm izolacijo.
322 Izvedba poteka tako: Najprej potri motor naredi število korakov, ki sodobujejo temu da se žica pomakne za 1 cm. Na tem mestu drugi motor naredi
323 število korakov, da se izolacija naloži na te. Potem potri motor naredi število korakov potrebnih, da se tica pomakne na nastavljeni dolžino.
324 Du potem spet drugi motor naredi izolacijo. Prvi motor spet pomakne žico za 1 cm, na koncu pa drugi motor naredi toliko korakov, da se žica odreže.
325
326
327 for (int i = 0; i < #obolinputBuffer_kolicina; i++)//Dokler je ponovitev manj kot pa je vnesena vrednost v medpomnilniku za koliko
328 {
329     delay(300); //Med vrtenjem motorjev se na LCD izpiše napis "NAPRAVA JE POD OBRAZOVANJEM".
330     lcd.clear();
331     lcd.setCursor(6, 0);
332     lcd.print("NAPRAVA");
333     lcd.setCursor(6, 1);
334     lcd.print("JE POD");
335     lcd.setCursor(4, 2);
336     lcd.print("OBRAZOVANJEM");
337     Gonolini_Motor.step(33.34); //gonolini motor naredi toliko korakov, da se žica pomakne za lcm
338     delay(100);
339     myStepper2.step(172); //drugi motor naredi toliko korakov, da zareže izolacijo
340     delay(100);
341     myStepper2.step(-172); //in se vrne v prvotno položaj
342     Gonolini_Motor.step(atol(inputBuffer_dolzina) * 33.368); //Nastavimo število korakov. Dobimo jih iz vrednosti v medpomnilniku dolžina potem pa to vrednost pomnožimo z konstanto, s katero dobimo najbolj natančen rezultat
343     delay(100);
344     myStepper2.step(172); //drugi motor spet naredi toliko korakov, da zareže izolacijo
345     delay(100);
346     myStepper2.step(-172); //in se vrne v prvotno položaj
347     Gonolini_Motor.step(33.4); //gonolini motor spet naredi toliko korakov, da se žica pomakne za lcm
348     analogWrite(14, LOW); //Na navitja motorja damo vrednost "0"
349     analogWrite(15, LOW);
350     analogWrite(16, LOW);

```

Slika 26: Program 7. del

NAPRAVA ZA AVTOMATSKO REZANJE ŽICE

```

327 for (int i = 0; i < stol[inputBuffer_kolicina]; i++)//Dokler je ponovitev manj kot pa je vnesena vrednost v medpomnilniku za kolicino
328 {
329     delay(300); //Med vrtenjem motorjev se na LCD izpiše napis "NAPRAVA JE POD OBRAZOVANJEM".
330     lcd.clear();
331     lcd.setCursor(6, 0);
332     lcd.print("NAPRAVA");
333     lcd.setCursor(6, 1);
334     lcd.print("JE POD");
335     lcd.setCursor(4, 2);
336     lcd.print("OBRAZOVANJEM");
337     Gonilni_motor.step(33.34);//gonilni motor naredi toliko korakov, da se žica pomakne za lcm
338     delay(100);
339     myStepper2.step(-172);//drug motor naredi toliko korakov, da zareže izolacijo
340     delay(100);
341     myStepper2.step(-172); //in se vrne v prvoten položaj
342     Gonilni_motor.step((aCol[inputBuffer_dolzina] * 33.368)///Nastavimo število korakov. Dobimo jih iz vrednosti v medpomnilniku dolzina potem pa to vrednost pomnožimo z konstanto, s katero dobimo najbolj natančen rezultat
343     delay(100);
344     myStepper2.step(172); //drug motor spet naredi toliko korakov, da zareže izolacijo
345     delay(100);
346     myStepper2.step(-172); //in se vrne v prvoten položaj
347     Gonilni_motor.step(33.4); //gonilni motor spet naredi toliko korakov, da se žica pomakne za lcm
348     analogWrite(14, LOW); //Na navitja motorja damo vrednost "0"
349     analogWrite(15, LOW);
350     analogWrite(16, LOW);
351     analogWrite(17, LOW);
352     delay(100);
353     myStepper2.step(200); //Drugi kracnji motor naredi en obrat v eno smer
354     delay(100);
355     myStepper2.step(-200); // in obrat v drugo(tako da se vrne v začeten položaj)-naredi toliko korakov, da se preko žice odreši
356     analogWrite(18, LOW);
357     analogWrite(19, LOW);
358     analogWrite(20, LOW);
359     analogWrite(21, LOW);
360     //Ne se ponovno vrtenje motorjev konča ->
361     lcd.clear(); //cek LCD ekran počisti
362     lcd.setCursor(0, 0);
363     lcd.print("PRIZISNITE TIPKI"); //in izpiše "PRIZISNITE TIPKI"
364     lcd.setCursor(5, 1);
365     lcd.print(" " ali # ""); // ali #
366     lcd.setCursor(2, 2);
367     lcd.print("Sa ponovni vnos"); //Sa ponovni vnos
368
369 currentCol_dolzina = 0;//ponovno posicijo kurSORja ponovno nastavimo na 0
370 currentBufferPosition_dolzina = 0;/ponicijo za zapis v medpomnilniku nastavimo na 0
371 inputBuffer_dolzina[currentBufferPosition_dolzina] = '\0';//dolocimo konec linije
372
373 currentCol_kolicina = 0;//ponovno posicijo kurSORja ponovno nastavimo na 0
374 currentBufferPosition_kolicina = 0;//ponicijo za zapis v medpomnilniku nastavimo na 0
375 inputBuffer_kolicina[currentBufferPosition_kolicina] = '\0';//dolocimo konec linije
376 }

```

Slika 27: Program 8. del

7 STROJNA IZVEDBA

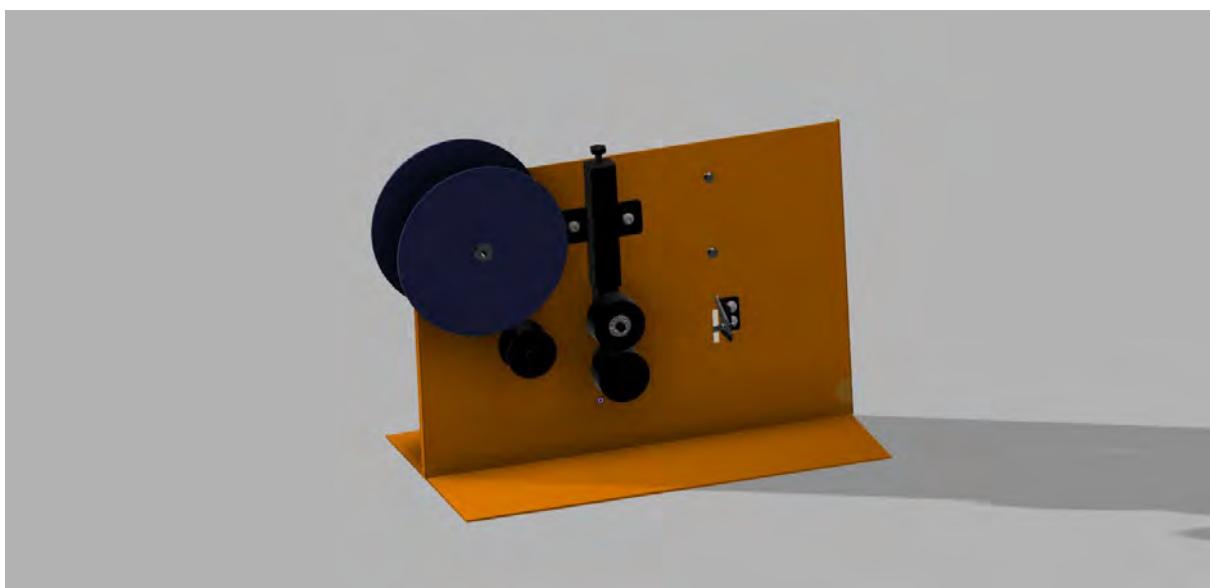
7.1 Potek strojne izvedbe

Strojna izvedba je bila za moj izdelek ključna, saj sem ves električni del prilagajal strojnemu. Vse skupaj sem si najprej zamislil, kar sem kasneje prenesel na papir. Skice, ki sem jih narisal na papir, so mi omogočile lažjo predstavo o tem, kako bo izdelek na koncu videti. Veliko je bilo namreč stvari, ki so se mi pred risanjem zdele enostavne, ko pa sem vse skupaj prenesel na papir, se je za kar nekaj stvari izkazalo, da jih sam ne bom mogel izvesti. Zaradi tega sem bil primoran veliko idej opustiti in poiskati enostavnejše rešitve. Tako sem potem počasi prihajal do primernih rešitev, ki so bile izvedljive. Vse skupaj sem potem povezal in narisal načrt, na katerem sem elemente povezal v celoto.

Ko sem imel dokončno idejo narisano na papirju, sem se lotil bolj konkretnega načrta, v katerem sem poskušal čim bolj natančno narisati izdelek, tako da je bil videti, kot naj bi bil videti na koncu izdelave. Prvi pogoj je bil poiskati oziroma približno določiti mere izdelka.

Izdelek sem si zamislil tako, da sem za osnovo uporabil kovinsko ploščo, na katero sem potem pritrjeval elemente. Zato sem moral na začetku poiskati primerno velikost te plošče. Vse elemente sem približno ponazoril na plošči in tako dobil ustrezno mero izdelka.

Ko sem imel ploščo narejeno, sem lahko začel z iskanjem primernih mer za elemente, ki sem jih moral postaviti na ploščo. Potem pa se je začelo moje načrtovanje 3D modela v Fusionu. Za posamezne elemente sem poiskal mere, jih prenesel v program in narisal natančen 3D model. Na osnovi modela sem, ko sem elemente v programu združeval, lahko določene mere prilagodil tako, da je na koncu res vse sodilo skupaj. Ko sem imel načrt narejen, sem se moral pri nadalnjem delu samo držati mer in pozicij, ki so bile narisane v načrtu in je strojna izvedba potekala brez težav.



Slika 28: Naprava za rezanje žice 3D model

Po izdelavi načrta sem se lahko lotil strojnega dela. Največ je bilo dela na stružnici, se pravi izdelave osovin, saj je bilo teh potrebnih izdelati največ. Prav tako sem moral izvrtati kar nekaj izvrtin in nekaj kosov privariti skupaj.

7.2 Izdelava posameznih elementov

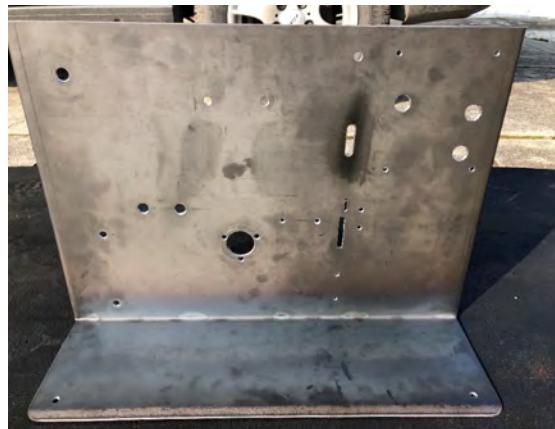
7.2.1 Izdelava plošče

Ploščo sem poskušal narediti čim manjšo, tako da sem se znebil nepotrebne teže, še zmeraj pa je mogla biti dovolj velika, da sem nanjo lahko pritrdil vse željene elemente, in sicer kovinske in elektronske elemente. Zamislil sem si mere vodil, koles in škatel za elektroniko. Vse te mere sem razporedil na neko površino in izmeril mero, ki bi ustrezala za velikost plošče. Tako sem potem dobil mero in sem lahko izdelal ploščo.

Za njeno izdelavo sem uporabil 3-milimetrsko kovinsko ploščo, ki sem jo dal odrezati na mero. Ker je za takšno površino plošče, kot sem jo potreboval, ta debelina nekoliko pretanka, sem se odločil da bom okrivil robove plošče na pravi kot. Na tak način sem celotno ploščo nekoliko ojačal, tako da sem se znebil nezaželenega ukrivljanja plošče. Na koncu, ko sem imel ploščo ukrivljeno sem privaril le še spodnji del, tako da sem konstrukcijo lahko postavil.

Veliko dela je bilo s pripravo plošče na pritrditev elementov. Potrebno je bilo izvrtati kar nekaj lukenj in daljših izrezov. Nekaj lukenj je bilo potrebno narediti za uvodnice, tako da sem lahko električne elemente na sprednji strani povezal z elementi na zadnji strani plošče, večino lukenj sem pa naredil za vijke. Odločil sem se namreč, da posameznih elementov ne bom varil na ploščo temveč jih bom pritrdil na njo z vijaki. Tako se lahko vsaka posamezna komponenta kadarkoli zamenja.

Ko sem imel na plošči izdelane vse izvrtine, sem, moral ploščo prebarvati. Prebarval sem jo zato, da sem kovino zaščitil pred rjo in pa za lepši videz. Pred samim barvanjem sem moral celotno ploščo dobro očistiti in zbrusiti vrhnjo plast železa tako, da se je barva dobro prijela površine. Najprej sem na ploščo nanesel temeljno barvo. Potem pa sem jo dvakrat prebarval z oranžno barvo. Ko se je barva v celoti posušila je bila plošča končana.



Slika 29: Kovinska plošča-nebarvana



Slika 30: Kovinska plošča-temeljna barva

7.2.2 Izdelava drsnika za zgornje kolo

Pri moji napravi se žica pomika med dvema kolesoma, med katerima je spodnje kolo pogonsko, ki je preko prenosa povezano na motor, zgornje pa služi temu, da ustvari pritisk na žico tako, da s pomočjo gumijastega materiala, ki je na kolesih žica teče skozi ob premiku motorja. Da sem ustvaril pritisk, ki drži žico med kolesoma sem mogel za zgornje kolo narediti nekakšen nosilec, ki kolo nateguje na spodnjega. Tako sem ustvaril drsnik znotraj katerega je vzmet, ki pritiska del nosilca na katerem je kolo, proti spodnjemu kolesu.

Za osnovo sem vzel dve kvadratni kovinski pohištveni cevi. Izbral sem takšne dimenzije, da je manjša cev sodila v večjo, tako sem potem obe cevi odrezal na mero in na zgornje strani pri vsaki privaril kos kovine, tako da sem naredil nekakšna pokrova. Na ceveh sem potem na vrhu izvrtil luknji in naredil navoja, tako da sem lahko na zunanjih cevi privijačil vijak, s katerim sem kasneje lahko nastavljal s kakšno silo zgornje kolo pritiska na spodnjega, na notranji cevi pa, da sem lahko pritrnil držalo za vzmet. Na spodnjo stran notranje cevi sem moral tudi privariti os, na katero sem potem namestil kolo. Na zunanjih cevih pa sem moral še ob robovih privariti kosa kovine in narediti skoznju luknje za vijke, tako da sem lahko ta del pritrnil na ploščo. Ko sem imel cevi dokončane

NAPRAVA ZA AVTOMATSKO REZANJE ŽICE

sem le še poiskal primerno vzmet, ki sem jo namestil med zunanjim in notranjim cev. Na koncu sem le še vse skupaj zbrusil in očistil ter potem prebarval.



Slika 31: Notranji del drsnika za zgornje kolo s privarjeno osjo



Slika 32: Varjenje zunanjega dela drsnika za zgornje kolo



Slika 33: Drsnik z zgornjim kolesom 3D model

7.2.3 Izdelava nosilcev za koračna motorja

Ker sem pri svoji napravi uporabil dva koračna motorja, sem mogel narediti dva nosilca, s katerima sem ju lahko postavil na ploščo. Izdelava nosilca motorja za vzzvod ni bila zahtevna. Potreboval sem pravokoten konec kovine. Na širšem delu sem zarisal luknje za motor (za fiksiranje motorja), na ožjem delu pa sem le izvrtil dve luknje tako, da sem nosilec lahko pritrdil na ploščo.



Slika 34: Nosilec motorja za rezanje z motorjem

Nosilec za gonilni motor pa sem naredil tako, da lahko po želji nastavljam nateg jermenja, ki je vpet na motorju. Za nosilec sem izdelal nekakšen omega profil, zato sem potreboval dva kovinska kotnika in pa eno povezovalno ploščico. Najprej sem na obeh kotnikih zrezkal utora za nastavljanje natega jermenja.



Slika 35: Rezkanje izreza za vodilo nosilca za zgornje kolo

Nato sem se lotil izdelave ploščice za pritrditev motorja. Zarisal sem si luknje potrebne za pritrditev motorja in jih potem izvrtal. Ko sem imel vse tri konce dokončane, sem se lotil varjenja. Vsakega od pravokotnih kovin sem privaril na vsak konec ploščice, tako da je nastal omega profil. Na koncu sem oba nosilca zbrusil in prebarval.



Slika 36: Priprava na varjenje nosilca za gonilni motor



Slika 37: Nosilec za gonilni motor z motorjem

7.3 Izdelava elementov na stružnici

Napravo za rezanje žice sem nameraval narediti tako, da bi bilo odstopanje zaradi strojne izvedbe čim manjše. Izogniti sem se moral odstopanjem zaradi nenatančne izdelave in zaradi obrabe po daljšem času uporabe naprave. Zato sem vse vrteče se dele izdelal popolnoma natančno, obrabo pa sem preprečil z uležajenjem.

Izdelati sem moral 3 osovine, dva plastična kolesa in pa kolo za vodilo, za katere pa je bilo potrebno kar nekaj načrtovanja.

Na začetku sem se vsega lotil v programu Fusion, kjer sem si vsako os posebej zamislil in jo narisal v 3D modelu. S tem sem si delo precej olajšal, saj sem lahko v programu vse osovine in kolesa združil skupaj in jih postavil na ploščo. Tako sem lahko dobil povsem natančne mere, ki sem jih potreboval pri izdelavi.

Izdelovanje na stružnici sem razdelil na izdelavo pogonskega kolesa, ki jo sestavlja izdelava dveh kovinskih osovin in pa plastičnega kolesa ter na izdelavo kolesa za drsnik, ki jo sestavlja izdelava ene osi in pa plastičnega kolesa.

7.3.1 Izdelava pogonskega kolesa

Namen pogonskega kolesa je vlečenje žice. Os, ki je na zadnji strani plošče preko prenosa povezana z koračnim motorjem, vrtenje tega motorja prenaša na sprednje kolo, preko katerega teče žica.

Osi sem si zamislil tako, da imam zunanjo in pa notranjo os, ki sta dani ena v drugo. Zunanja os služi pritrditvi na ploščo in pa vstavitvi ležajev. Sama je primeru (se ne vrati) in v resnici služi le držanju zunanjega dela ležaja, tako da se notranja os lahko vrati. Notranja os je os na katero pride pritrjen zobnik za zobati jermen, tako da se nanj lahko preko zobatega jermenega poveže koračni motor. Na drugi strani pa je na njo togo pritrjeno kolo, tako da se z vrtenjem motorja vrati tudi kolo.



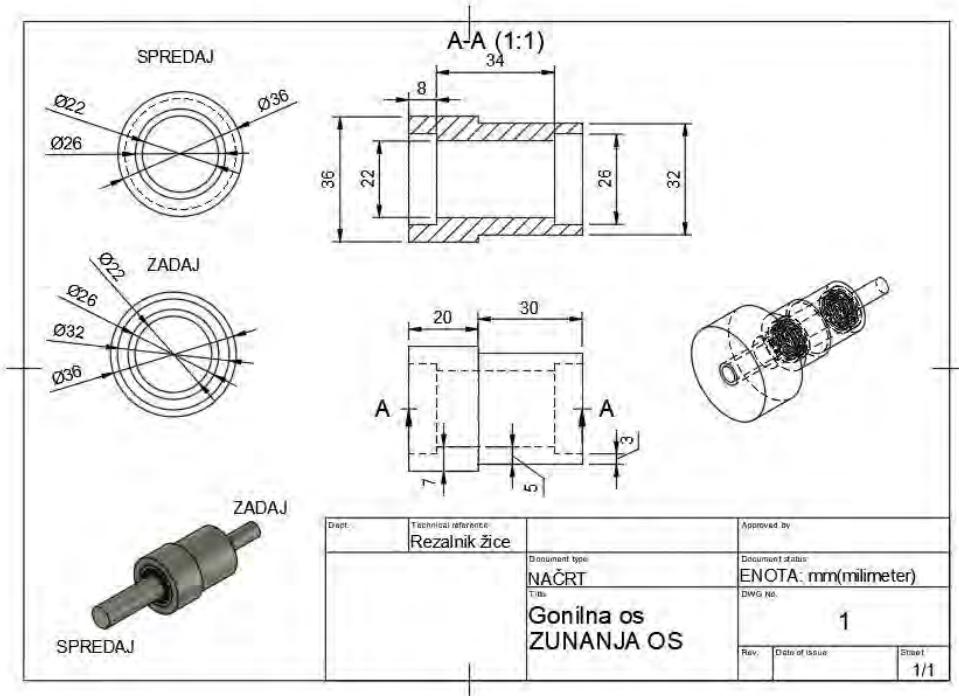
Slika 38: Pogonsko kolo z osovinama-3D model

Najprej sem se lotil izdelave zunanje osi. Izdelal sem jo tako da ima na eni strani debelejši rob, tako da jo ob pritrditvi na ploščo lahko točno pritrdimo na ploščo. Na notranji strani pa sem moral z veliko natančnostjo postružiti luknjo za debelino in premer ležaja na vsaki strani (na koncu se je ležaj moral vtisniti v os).



Slika 39: Struženje zunanje osi pogonskega kolesa

NAPRAVA ZA AVTOMATSKO REZANJE ŽICE



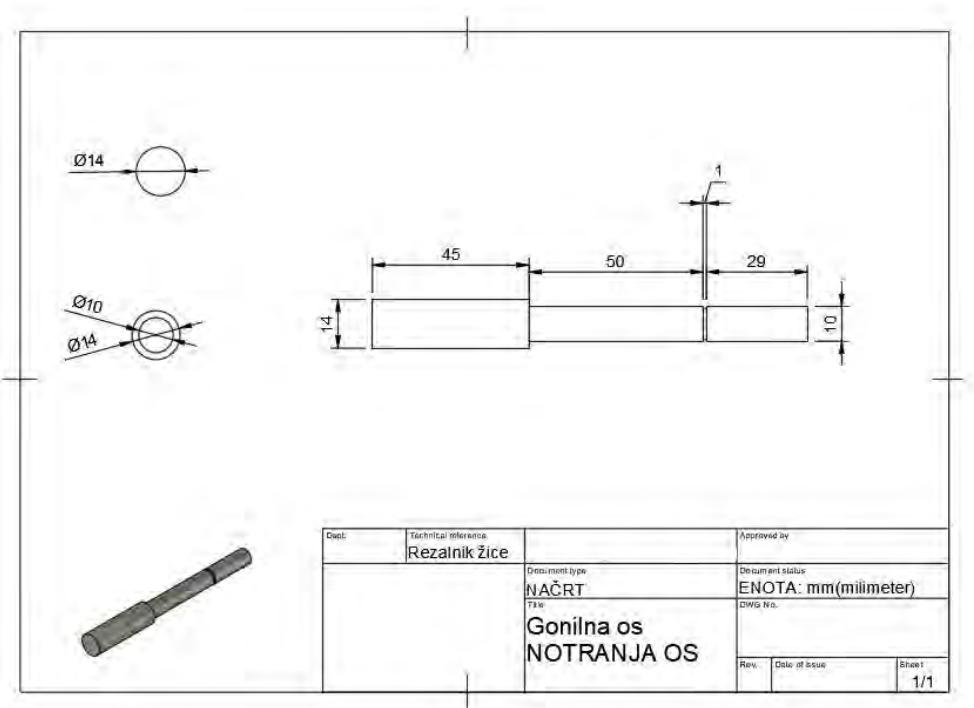
Slika 40: Načrt zunanje osovine pogonskega kolesa

Za notranjo os sem moral postružiti os tako, da se je na eni strani zobnik lahko pritrdil na njo, na drugi pa plastično kolo. Na nekoliko bolj notranjih straneh pa jo je bilo potrebno natančno postružiti za vtisk ležajev.



Slika 41: Struženje notranje osi pogonskega kolesa

NAPRAVA ZA AVTOMATSKO REZANJE ŽICE



Slika 42: Načrt notranje osi pogonskega kolesa

Kolo sem naredil iz plastike za struženje in ima premer okoli 60 mm. Kolesu sem moral le postružiti del za vijačenje vijaka in izvrtati luknjo, tako da sem ga lahko pritrdil na os.



Slika 43: Struženje gonilnega plastičnega kolesa

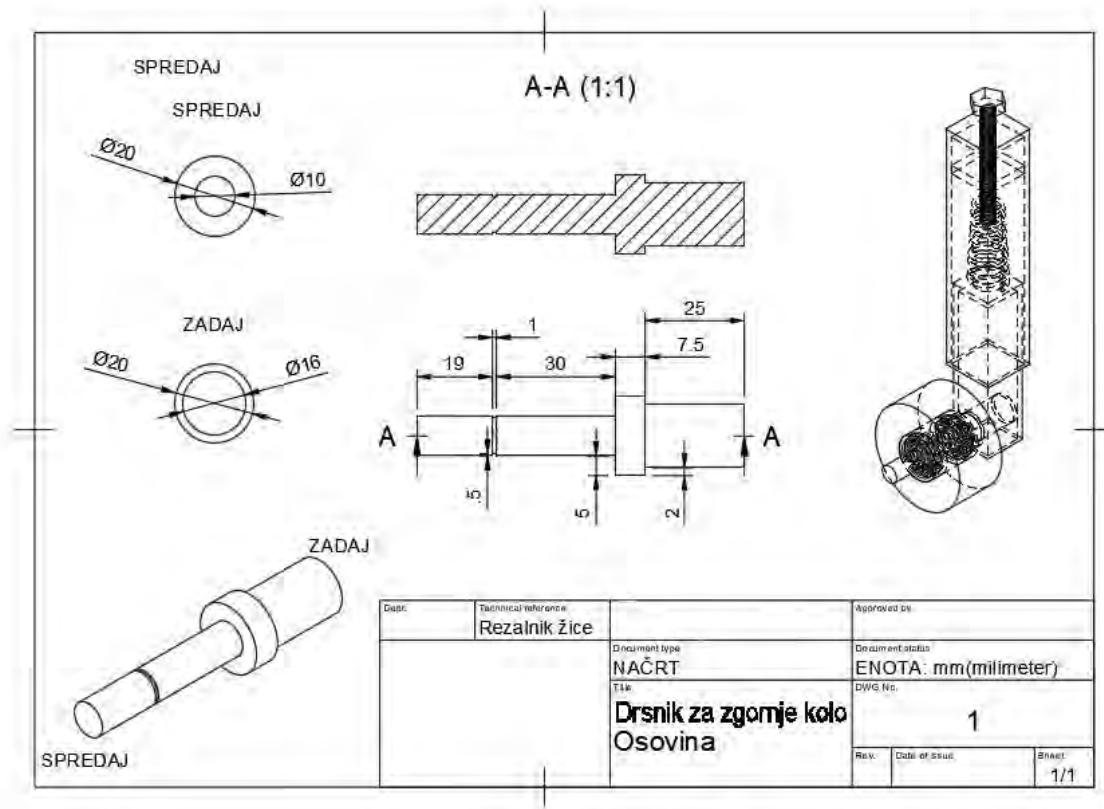
Na koncu sem moral vtisniti ležaje to sem naredil tako, da sem ležaj vtisnil najprej na eni strani notranje osi, potem sem ta isti ležaj vtisnil v zunanjo os, le da sem tu vtisnil zunanji obroč ležaja. Potem pa sem še drug ležaj vtisnil hkrati na notranjo in v zunanjo os.

7.3.2 Izdelava kolesa za drsnik

Osovina tega kolesa je privarjena na drsnik za zgornje kolo tako, da se s premikanjem drsnika premika tudi kolo.

Najprej sem na spodnji strani notranje kvadratne cevi drsnika izvrtal luknjo tako da sem os lahko vtaknil v njo.

Potem sem začel z struženjem osi. Na eni strani sem os postružil za premer luknje na cevi, tako da je pasala v njo. Na drugi strani pa sem jo natančno postružil za vtisk dveh ležajev.



Slika 44: Načrt osi kolesa za zgornje kolo

Kolo sem izdelal iz plastike za struženje in ima premer okoli 60 mm. Na vsaki strani kolesa sem mogel na notranji strani postružiti plastiko za premer ležaja.



Slika 45: Struženje plastičnega kolesa za drsnik

V tem primeru sem ležaje vtiskal po sledečem postopku: prvi ležaj najprej na notranjo stran osi, nato isti ležaj na kolo, potem pa na drugi strani, drugi ležaj hkrati na os in kolo.

7.4 Izdelava škarij in vzdova

Mehanizem za rezanje na moji napravi sestavljajo ščipalne klešče kot osnova, potem pa je na zadnji strani narejen vzvod preko katerega motor z vrtenjem dviguje in spušča klešče.

Na začetku sem želel narediti mehanizem za rezanje na preprost način, in sicer tako da bi ploščo fiksiral klešče, ki bi jih dvigoval in spuščal z vrvico, ki bi se ovijala okoli osi motorja.

Klešče sem res naredil tako, da sem jih na eni strani pritrdil na ploščo. Naredil sem tudi nosilec za njih tako, da sem dobil dobro stabilnost klešč.



Slika 46: Škarje

Za rezanje žice pa se je kmalu izkazalo, da je takšen koračni motor brez vzvoda precej prešibak. Zato sem začel razmišljati o tem, da bi ga zamenjal z večjim motorjem, vendar pa bi potreboval velik motor z veliko močjo, ki pa se ni zdel primeren za mojo napravo. Zaradi tega razloga sem začel razmišljati na kakšen način bi lahko izdelal vzvod, s katerim bi pridobil na moči.

Odločil sem se, da bom uporabil škripce. Približno sem izračunal potrebno moč, ki jo potrebujem za rezanje žice in se odločil, da bom uporabil sistem z štirimi škripci. Na tak način sem štirikrat povečal moč motorja oziroma sem ga za štirikrat razbremenil in pa povečal pot, ki jo mora za isto delo opraviti, za štirikrat.

Za škripce sem se odločil, da jih bom naredil kar sam. Na začetku sem moral v ploščo narediti izrez, po katerem se bo en par škripcev pomikal. Nato pa sem se lotil izdelave osovin za škripce. Izdelal sem dve enaki osovine in sicer obe takšni, da sem na njiju lahko dal dva kolesa.

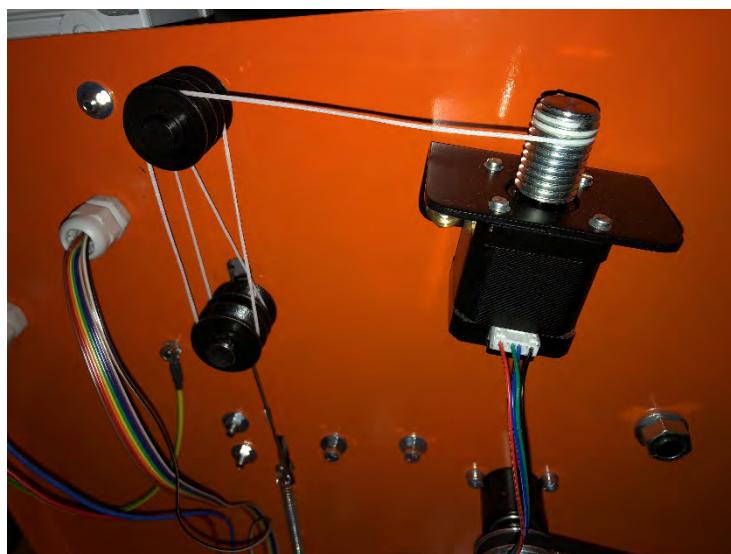
Za tem pa sem se lotil izdelave koles. Za kolesa sem uporabil plastiko za struženje premora 30mm. Naredil sem štiri kolesa. Vsakemu od njih sem na stružnici pod kotom 45 stopinj zarezal kanal, ki služi kot vodilo za vrvico.



Slika 47: Škripci in osovini

Ko sem imel narejene vse kose, sem jih sestavil skupaj in pritrdil na ploščo. Klešče sem povezal s spodnjo osjo oziroma z osjo, ki se premika, tako da se škarje premikajo skupaj z njo. Na spodnjo stran pa sem pritrdil tudi vzemet, tako da se klešče lažje vračajo v prvotni položaj.

Na osi motorja je bilo še potrebno narediti element, po katerem se bo vrvica natančno ovijala in odvijala. Za to sem uporabil vijak večjega premera in ga odrezal. Na eni strani sem mu izvrtil luknjo, tako da sem ga lahko namestil na os motorja. Na tak način sem dosegel, da se vrvica navija po navojih, tako da je z vsakim vrtljajem povsem enak premik škarij.



Slika 48: Končni vzvod

7.5 Sestava vseh elementov

Ko sem imel narejene vse posamezne elemente, sem lahko začel s pripravo na sestavo vsega skupaj. Najprej je bilo potrebno vse zaščititi.

Najprej je bilo vse kovinske dele (vse osovine, zvarjene dele...) potrebno dobro očistiti in zbrusiti. Potem sem vse prebarval z temeljno barvo. Nato pa sem vse manjše dele, ki pridejo pritrjeni na ploščo prebarval z črno barvo. Ploščo sem pa med tem dvakrat prebarval z oranžno barvo.

Ko sem imel vse dele posušene sem se lahko lotil sestavljanja moje naprave. Vsi elementi so na ploščo privijačeni, tako da se lahko bilo kdaj snamejo z nje.

8 VARNOST

Ker sem z vzvodom precej povečal moč motorja in s tem tudi moč škarij, je postala naprava nevarna v območju rezanja. Potrebno je bilo narediti neko zaščito tako, da se je preprečila možnost, da bi oseba po pomoti prišla v stik s škarjami, med obratovanjem. Odločil sem se, da bom zaščito naredil iz akrilnega stekla. Zato sem to najprej odrezal na mero, potem pa sem ga s segrevanjem okrivil tako, da se je prilegal okoli škarij.



Slika 49: Varnostna zaščita

9 NAVODILA ZA UPORABO

9.1 Ročno

Žico oziroma element, ki ga želimo uporabiti na napravi, moramo najprej ročno napeljati, tako da lahko naprava funkcionala. Če imamo možnost, element postavimo na stojalo za odvijanje. Potem ta element napeljemo, najprej po spodnji strani vodilnega kolesa, na to med gonilno kolo in zgornje kolo, naprej ga napeljemo skozi bakreno cev in končamo, ko se element dotakne škarji.

9.2 Elektro

Vtič na napravi je najprej treba vtakniti v vtičnico. Na desni strani sprednje doze je dvočlenno stikalo, s katerim vklopimo napravo.

Na vratih sprednje doze imamo en potenciometer in tipko. S potenciometrom lahko nastavljamo kontrast LCD ekranja. S pritiskom na tipko pa celotni program resetiramo.

9.3 Programsко

Na začetku se nam na LCD ekranu pojavi pozdravni napis. Z tipko »#« na tipkovnici se začnemo pomikati naprej po menijih, z tipko »*« pa se pomikamo nazaj po menijih. Ob prvem pritisku na tipko »#«, se nam pojavi meni za vnos dolžine elementa, ki ga želimo imeti narezanega. Vrednosti vnesemo s pritiskom številk na tipkovnici, v primeru da se zmotimo pa lahko vrednosti pobrišemo z tipko »A«. S ponovnim pritiskom na tipko »#« nadaljujemo na meni 2, kjer vnesemo količino kosov, ki jih želimo imeti narezane.

V primeru, da smo pozabili vnesti določeno vrednost, bodisi količino ali pa dolžino, nas program ne spusti v naslednji meni, temveč nas z napisom na ekranu opomni, da smo vrednosti pozabili vnesti.

V primeru, da smo vnesli vse vrednosti (dolžino in količino), se z naslednjim pritiskom na tipko »#« pomaknemo na meni, kjer izberem tip programa, za katerega želimo, da ga naša naprava izvede. V tem meniju nam v zgornjem desnem kotu ekranja izpiše dolžino in količino, ki smo ju prej nastavili. Spodaj pa imamo na ekranu napisano, katero tipko naj pritisnemo za izbiro prvega programa in katero za izbiro drugega.

S pritiskom na tipko »A« izberemo navaden program. Ta program deluje preprosto tako, da predmet, ki ga napeljemo skozi vodila, nareže na takšno dolžino, kot smo jo nastavili, to pa ponovi tolikokrat, kolikor imamo nastavljenou vrednost za količino.

S pritiskom na tipko »B« izberemo program za rezanje žice. Ta program deluje tako, da žico, ki jo napeljemo skozi vodila, odmeri na dolžino, ki jo imamo nastavljeno in ji doda dva centimetra. Na vsaki strani žice pa škarje po en centimeter zarežejo izolacijo, tako da jo potem lažje odstranimo in lahko nanjo namestimo tulce. Celoten postopek se ponovi tolikokrat kot določa nastavljenou vrednost za količino.

NAPRAVA ZA AVTOMATSKO REZANJE ŽICE

Med delovanjem bilo katerega programa, se na ekranu izpiše napis »Naprava je pod obratovanjem«. Po zaključitvi postopka pa lahko s pritiskom na tipko »#« ponovno vnesemo vrednosti za količino in dolžino.

10 RAZPRAVA

Na začetku sem si postavil cilj, da moja naprava odmeri in pa odreže neko količino žic. To sem dosegel, saj moja naprava odmeri žico na tako dolžino, kot sem jo predhodno nastavil in jo potem tudi odreže. To ponovi tolkokrat, kot določa izbrana vrednost, zato je tudi moj drugi cilj, da naprava odreže nastavljenou količino žic, dosežen.

Pri točnosti se je na koncu izkazalo, da je nekaj odstopanj. Vendar so odstopanja mnogo manjša kot pa sem jih predvidel na začetku, ta zdaj znašajo največ 2% nastavljenou količine. Ta odstopanja bi lahko še odpravili z daljšimi testiranji in prilagajanji znotraj programa.

Na začetku sem si tudi zadal cilj, da stroški naprave ne bi bili višji od 200 €. Zadani cilj sem izpolnil, saj so bile vse uporabljene komponente preproste in cenovno ugodne.

11 ZAKLJUČEK

V svoji raziskovalni nalogi sem prišel do spoznanj, ki temeljijo na vseh hipotezah, ki sem jih postavil na začetku. Prišel sem do spoznanja, da se da vnesti podatke oziroma vrednosti, ki so potem vidne pri strojni izvedbi. Natančnost se da precej dobro zagotoviti, z natančno strojno izvedbo in potem s primernimi nastavtvami v programu. Spoznal sem tudi, da je z lastnim delom, mogoče zmanjšati mnogo stroškov, ki bi sicer nastali.

Na samem začetku me je najbolj skrbela izvedba za odmerjanje žice, saj sem menil da oprijem med kolesoma ne bo zadosten in žice ne bosta dovolj držala. Vendar se je izkazalo, da je takšna izvedba kot sem jo uporabil precej optimalna. Tudi izdelava mehanizma za rezanje je bila nekoliko bolj zapletena, saj je bila potrebna velika moč na motorju, za rezanje žice. Tudi ta problem pa sem brez večjih težav rešil.

12 VIRI IN LITERATURA

- Primer programa z meniji. Pridobljeno 5.12.2020 s <https://github.com/VRomanov89/EEEnthusiast/tree/master/03.%20Arduino%20Tutorials/01.%20Advanced%20Button%20Control/ButtonSketch>
- Arduino forum. Pridobljeno 20.1.2021 s <https://forum.arduino.cc/>
- Nema 17 datasheet. Pridobljeno 17.2.2021 s <https://datasheetspdf.com/pdf/1260602/Schneider/NEMA17/1>
- Arduino mega. Pridobljeno 25.3.2021 s https://www.geeetech.com/wiki/index.php/Arduino_Mega_2560

IZJAVA*

Mentor **Gregor Kramer** v skladu z 20. členom Pravilnika o organizaciji mladinske raziskovalne dejavnosti »Mladi za Celje« Mestne občine Celje, zagotavljam, da je v raziskovalni nalogi z naslovom **Naprava za avtomatsko rezanje žice**, katere avtor je **Alen Jakopič**:

- besedilo v tiskani in elektronski obliki istovetno,
- pri raziskovanju uporabljeno gradivo navedeno v seznamu uporabljeni literature,
- da je za objavo fotografij v nalogi pridobljeno avtorjevo dovoljenje in je hranjeno v šolskem arhivu,
- da sme Osrednja knjižnica Celje objaviti raziskovalno nalošo v polnem besedilu na knjižničnih portalih z navedbo, da je raziskovalna naloga nastala v okviru projekta Mladi za Celje,
- da je raziskovalno nalošo dovoljeno uporabiti za izobraževalne in raziskovalne namene s povzemanjem misli, idej, konceptov oziroma besedil iz naloge ob upoštevanju avtorstva in korektnem citiranju,
- da smo seznanjeni z razpisni pogoji projekta Mladi za Celje.

Celje, 12. maj 2021



Podpis mentorja

Podpis odgovorne osebe

*

POJASNILO

V skladu z 20. členom Pravilnika raziskovalne dejavnosti »Mladi za Celje« Mestne občine Celje je potrebno podpisano izjavo mentorja (-ice) in odgovorne osebe šole vključiti v izvod za knjižnico, dovoljenje za objavo avtorja (-ice) fotografskega gradiva, katerega ni avtor (-ica) raziskovalne naloge, pa hrani šola v svojem arhivu.