

Šolski center Celje
Srednja šola za strojništvo in mehatroniko

UPORABA ODPRTOKODNIH SISTEMOV PRI POSODOBITVI OBDELOVALNIH STROJEV

Avtorji:

Rok MOTORE, S-4.b

Alen OPERČKAL, S-4.b

Aljaž OPERČKAL, S-4.b

Mentor:

Matej VEBER, univ. dipl. inž. el.

Mestna občina Celje, Mladi za Celje

Celje, 2009

KAZALO:

1. Povzetek.....	5
2. Uvod.....	6
3. Teze/hipoteze.....	6
4. Linux proti Windowsu.....	7
4.1 Zakaj smo izbrali Linux.....	7
4.2 Opis EMC ²	9
5. Opis obstoječega delovnega stroja EMCO F1.....	13
6. Opis koračnih motorjev.....	16
6.1 Krmiljenje koračnih motorjev.....	18
7. LPT port.....	22
8. Izdelava ploščice.....	24
8.1 Izris sheme v EAGLU.....	25
8.2 Priključitev in testiranje.....	27
9. Končna/referenčna stikala.....	28
10. Varnost.....	29
11. G koda.....	30
12. Potrditev oziroma zavrnitev tez/hipotez.....	32
13. Zaključek.....	33
14. Zahvala.....	34
15. Viri in literatura.....	35

KAZALO SLIK:

1. Slika 4.1: Simbol za Linux.....	7
2. Slika 4.2: Simbol za Windows.....	7
3. Slika 4.3: Simbol za EMC ²	9
4. Slika 4.4: Izgled programa EMC ²	9
5. Slika 4.5: Obrazložitev izhodov.....	11
6. Slika 4.6: Nastavitev osi.....	12
7. Slika 5.1: EMCO F1 rezkalni stroj.....	13
8. Slika 5.2: Pogled na konzolo.....	14
9. Slika 5.3: Kroglično vreteno.....	15
10. Slika 5.4: Sprednji del rezkalnega stroja.....	15
11. Slika 5.5: Zadnji del rezkalnega stroja.....	15
12. Slika 6.1: Koračni motor.....	16
13. Slika 6.2: Shema in prerez koračnega motorja.....	17
14. Slika 6.3: Koračni motor na EMCO F1.....	17
15. Slika 6.4: Krmilni signali za polnokoračno in polkoračno krmiljenje koračnih motorjev.....	20
16. Slika 6.5: Položaj 1.....	21
17. Slika 6.6: Položaj 2.....	21
18. Slika 6.7: Položaj 3.....	21
19. Slika 6.8: Položaj 4.....	21
20. Slika 7.1: Shematski prikaz LPT porta.....	22
21. Slika 7.2: Signali iz LPT izhodov.....	23
22. Slika 7.3: Preizkusna ploščica z LED diodami.....	23
23. Slika 7.4: Preizkušanje LPT izhodov(deluje).....	23
24. Slika 8.1: Napajalniki.....	24
25. Slika 8.2: Shema vezja za en motor.....	25

<i>26. Slika 8.3: Ploščica za krmiljenje enega motorja</i>	26
<i>27. Slika 8.4: Priključitev in testiranje testne ploščice</i>	27
<i>28. Slika 9.1: Izdelava nosilca za stikala</i>	28
<i>29. Slika 9.2: Namestitev stikal</i>	28
<i>30. Slika 10.1: Gumb za zasilno ustavitev</i>	29
<i>31. Slika 10.2: Senzor za vrata na našem rezkarju</i>	29
<i>32. Slika 11.1: G koda v programu EMC²</i>	31
<i>33. Slika 13.1: Skupinska</i>	33

1. Povzetek

V tej raziskovalni nalogi vam bomo predstavili posodobitev delnega CNC stroja, popolnoma računalniško voden CNC frezalni stroj.

Zaradi vse večje potrebe po inovacijah in napredku, smo se tudi mi v šoli odločili, da posodobimo sedaj že rahlo zastarel frezalni stroj EMCO F1. Delo je potekalo tako doma, kot v šoli, podatke pa smo pridobili iz raznih priročnikov in interneta.

Summary:

In this ressearch we will present to you the updating partial CNC machine into completly computer guided CNC sided molding machine.

The machine that we will update is EMCO F1 sided molding machine.

2. Uvod

Tehnologija se vedno razvija. Za seboj pa pušča stroje, ki so nekdam bili najboljše. Takšen je tudi naš frezalni stroj EMCO F1. Ker takrat še tehnologija ni bila tako razvita kot danes, je ta stroj že malo zastarel, zato smo ga posodobili. S to posodobitvijo se je možnost uporabe spet malo povečala.

3. Teze/hipoteze

Postavili smo tudi nekaj hipotez, ki jih bo z raziskavo potrdili ali zavrgli:

- 1. Hipoteza: S posodabljanjem stroja se bo povečala možnost uporabe tega.
- 2. Hipoteza: Spoznali bomo delovno okolje odprto kodnega operacijskega sistema Linux Ubuntu.
- 3. Hipoteza: Osebni računalnik(PC) in primerno programsko opremo lahko uporabimo za namen vodenja CNC rezkalnega stroja.
- 4. Hipoteza: Z omenjenim pristopom lahko posodobimo starejše obdelovalne stroje.
- 5. Hipoteza: Spoznali bomo tudi problematiko pri avtomatizaciji ali posodobitvi obdelovalnih strojev.

4. Linux proti Windowsu



Slika 4.1: Simbol za Linux



Slika 4.2: Simbol za Windows

Pri izbiri operacijskega sistema za naš projekt smo prišli do točke, kjer smo imeli na izbiro dva sistema. Odprtokodni sistem LINUX, ter malo bolj poznan Windows.

Že na začetku so se začele pojavljati teorije o tem, kateri je boljši, zato smo se odločili, da predstavimo nekaj dejstev, ki so kasneje služila kot kriterij za izbiro.

4.1 Zakaj smo izbrali Linux

Glede na razširjenost po svetu je vsekakor v vodstvu windows, saj že kar nekaj časa kraljuje trgu po vsem svetu. Za informacijo, leta 2008 je Windows predstavljal 91.11% deleža operacijskih sistemov, medtem ko je Linux pristal šele na tretjem mestu z 1.95%. Prehitel ga je še Mac OS z 4.73%, preostali procenti pa pripadajo bolj neznanim sistemom. Zaradi takšnega števila uporabnikov je tudi večina programov prirejenih za Windows okolje, kar pomeni, da nam iskanje le teh ne bi predstavljalo večjih težav.

Ko govorimo o dostopnosti, ima Linux eno dobro lastnost. Vse je namreč brezplačno (freeware), kar pomeni, da si lahko čisto legalno ves programski del projekta (software) enostavno naložimo z interneta ter ga neomejeno kopiramo, uporabljamo in spreminjamo. Za razliko od Windows, kjer je vse zaščiteno z licencami in pogoji uporabe, povrh tega pa še relativno drago.

Po testiranjih obeh sistemov, smo naredili nekaj primerjav, glede na uporabo. Windows je nekoliko bolj prijazen do uporabnika, saj le ta ne potrebuje veliko predznanja za uporabo sistema. Pri delovanju prikaže uporabniku vse potrebne informacije in vse morebitne ukaze predstavi tudi grafično. Poleg tega je marsikaj popolnoma avtomatizirano. Seveda pa vse to ni brez posledic. Ta operacijski sistem je znan po pogostih težavah, ponavadi celo s samim prenehanjem delovanja. Zato ga ne moremo označiti kot najbolj zanesljivega. Ravno obratno

pa velja za Linux. Je izjemno preprosto zgrajen in ne predstavlja velikega bremena za sam računalnik. Po principu »Manj je več«, ta sistem ne vsebuje nepotrebnih programov, ki bi samo trtili potencialne računalnika, zato tudi ne pride do toliko napak, kot pri Windows-u. Poleg tega pa je Linux veliko bolj fleksibilen, saj si ga uporabnik lahko neomejeno spreminja in prilagaja po svojih potrebah, medtem ko je pri Windows omejen znotraj določenih okvirjev.

Glede varnosti pri Windows bi lahko rekli, da je nekje na osnovni ravni. Seveda nekatere verzije vsebujejo avtomatske nadgradnje, požarni zid, protivirusno zaščito...

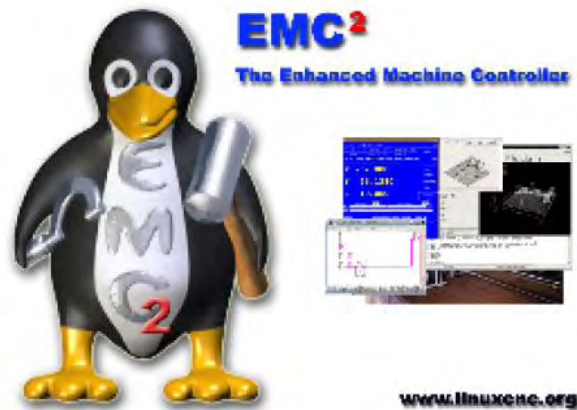
A takšno zaščito je potrebno neprestano posodablјati in nadgrajevati, saj se posodablјajo tudi grožnje in virusi, ki jih lahko dobimo z medmrežja ali prenosnih medijev.

Linux je na tem področju bolj razvit. Na voljo je nešteto nastavitev, ki omogočajo kar najboljšo varnost in optimalno delovanje. Poleg tega pa je količina stvari, ki bi lahko škodovale Linux minimalna v primerjavi s tem kar je nevarno za Windows, ravno zaradi tega ker je Windows najbolj razširjen na trgu.

Glede na dejstva, ki so predstavljena zgoraj, smo prišli do sklepa, da je Linux neprimerljivo boljši kandidat za naš projekt, saj ustreza skoraj vsem kriterijem, ki smo jih postavili ob začetku projektiranja. Cenovno je ugoden, dostopen, prilagodljiv... itd.

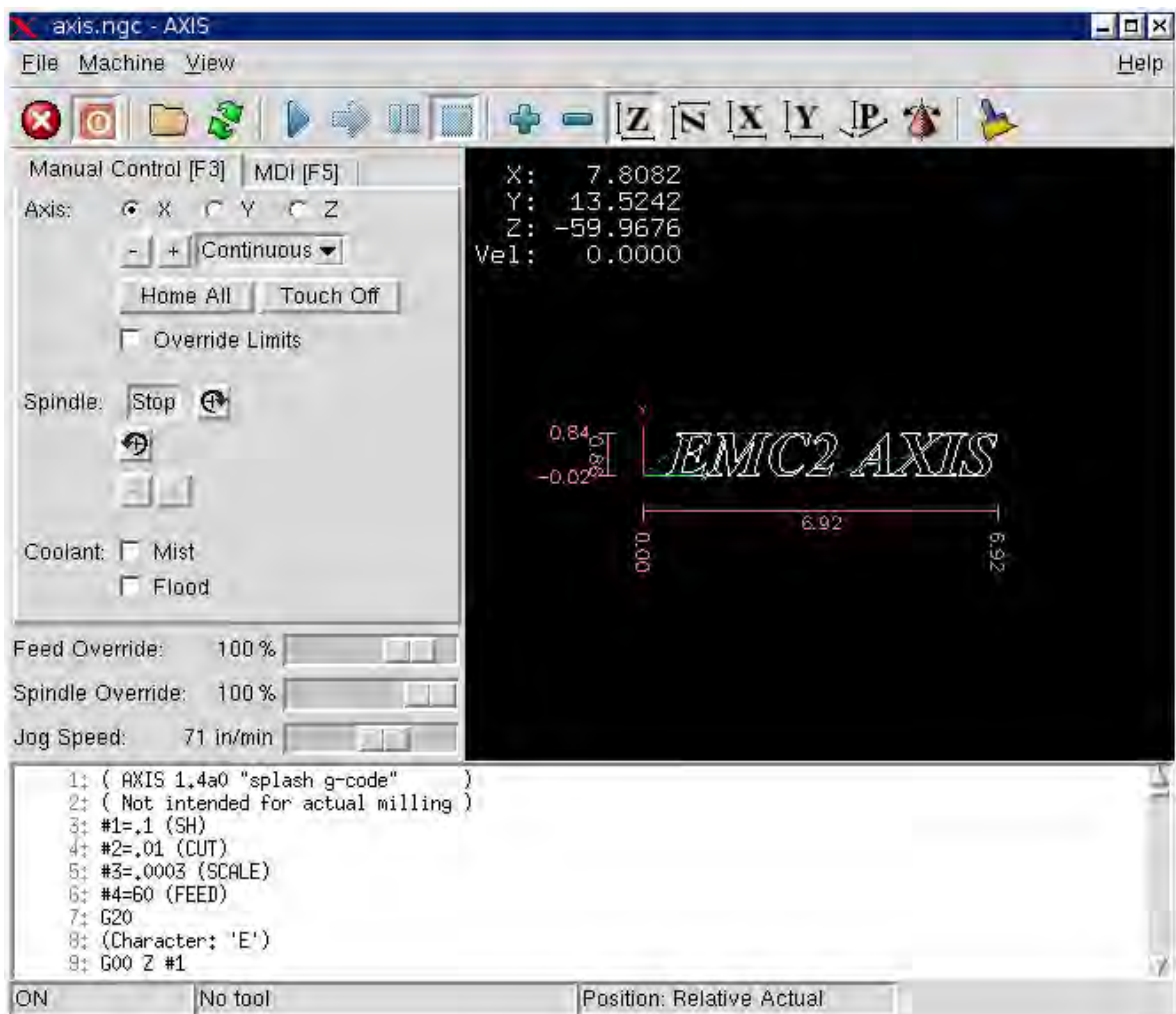
Poleg tega pa je osnova za program EMC2, ki smo ga izbrali, kot programski del projekta (software).

4.2 Opis EMC²



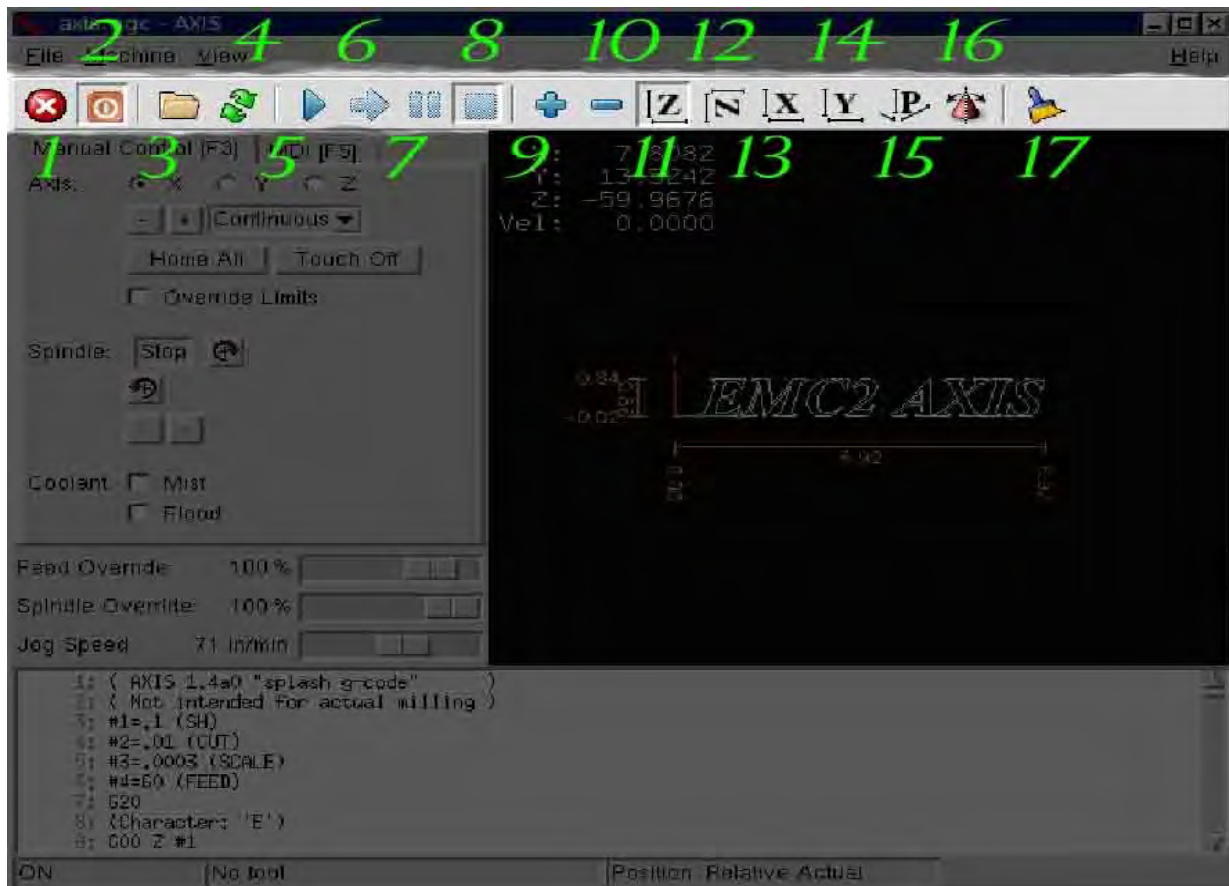
Slika 4.3: Simbol za EMC²

EMC2 ni samo program za uporabo pri CNC obdelovalnih strojih. S tem programom lahko kontroliramo tudi robote in ostale avtomatske naprave. Lahko upravljamo tudi servo motorje, koračne motorje, releje in podobno.



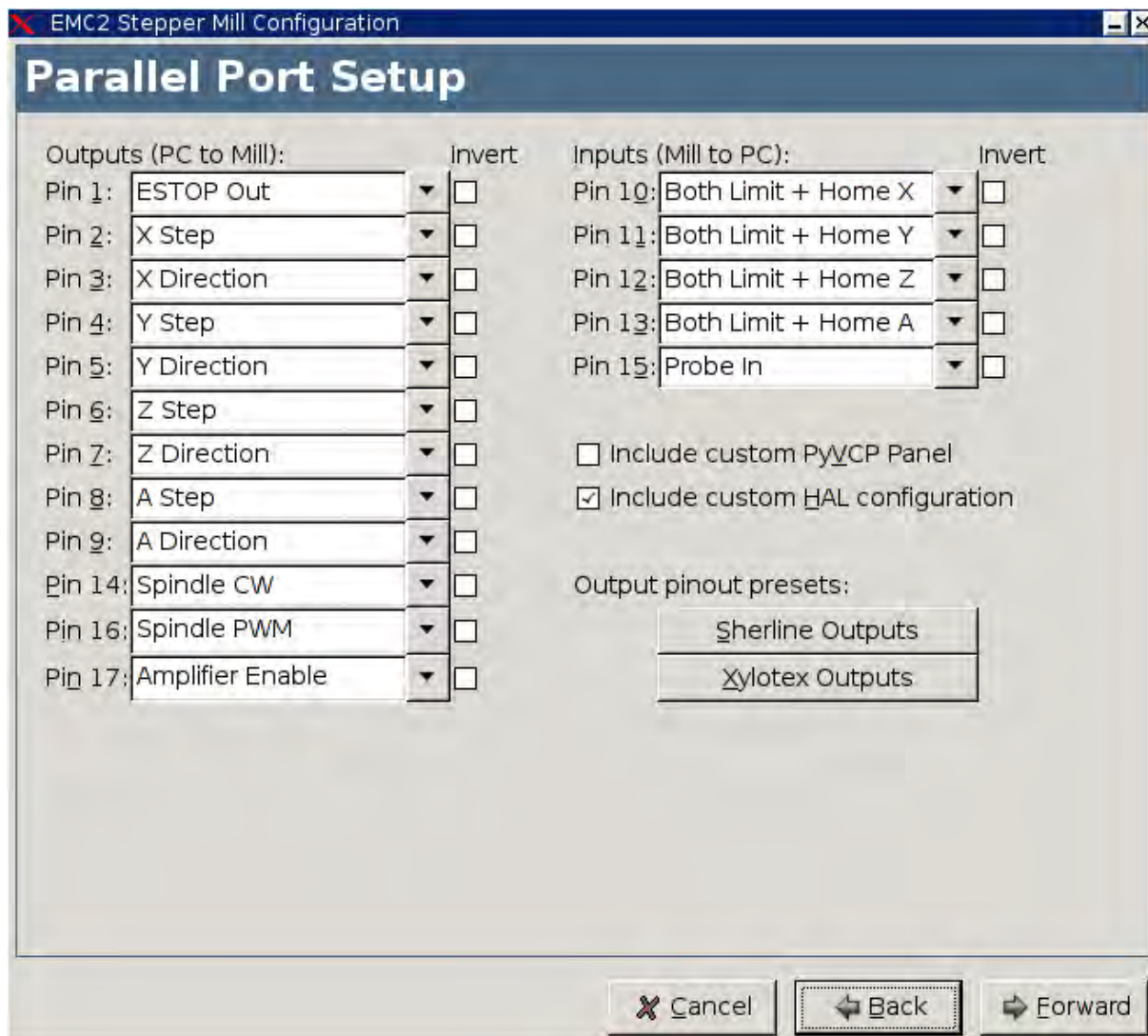
Slika 4.4: Izgled programa EMC²

Obrazložitev gumbov, ki so pomembni v EMC²



1. Zasilna ustavitev (E-Stop).
2. Vključi stroj.
3. Odpri datoteko.
4. Ponovno naloži odprto datoteko.
5. Poženi program.
6. Poženi naslednjo vrstico programa.
7. Pavza.
8. Ustavi program.
9. Povečaj.
10. Oddalji.
11. Pogled od zgoraj „Z.“
12. Pogled od zgoraj „obrnjen Z.“
13. Pogled od spredaj „X.“
14. Pogled iz strani „Y.“
15. Pogled iz perspektive „P.“
16. Obrni pogled, uporabi levo miškino tipko za obračanje.
17. Počisti.

V programu EMC² lahko tudi izberemo, kaj naj pomenijo izhodi na LPT portu. Lahko izberemo tudi ukaz »INVERT«. To pa zato, če je slučajno signal, ki ga dobimo iz računalnika ravno obraten (0V za aktiviranje, 5V neaktivirano).



Slika 4.5: Nastavitev izhodov

Vsako os lahko preizkusimo in konfiguriramo po potrebi. Nastavimo lahko želeno hitrost osi, pospešek osi, število korakov za 360 stopinj...

The image shows a software window titled "EMC2 Stepper Mill Configuration" with a sub-tab "X Axis Configuration". The window contains several input fields and a summary section. The "Motor steps per revolution" field is set to 200. The "Driver Microstepping" field is set to 2. The "Pulley teeth (Motor:Leadscrew)" field is set to 1, and the adjacent field is set to 1. The "Leadscrew Pitch" field is set to 20 rev / in. The "Maximum Velocity" field is set to 1 in / s. The "Maximum Acceleration" field is set to 30 in / s². The "Home location" field is set to 0. The "Table travel" field is set to 0 to 8. The "Home Switch location" field is set to 0. The "Home Search velocity" field is set to 0.05. The "Home Latch direction" dropdown menu is set to "Same". The summary section shows: "Time to accelerate to max speed: 0.0333s", "Distance to accelerate to max speed: 0.0167in", "Pulse rate at max speed: 8000.0Hz", and "Axis SCALE: 8000.0". At the bottom of the window are three buttons: "Cancel", "Back", and "Forward".

Motor steps per revolution:	200	
Driver Microstepping	2	
Pulley teeth (Motor:Leadscrew):	1	: 1
Leadscrew Pitch:	20	rev / in
Maximum Velocity:	1	in / s
Maximum Acceleration:	30	in / s ²
Home location:	0	
Table travel:	0	to 8
Home Switch location:	0	
Home Search velocity:	0.05	
Home Latch direction:	Same	
Time to accelerate to max speed:	0.0333s	
Distance to accelerate to max speed:	0.0167in	
Pulse rate at max speed:	8000.0Hz	
Axis SCALE	8000.0	

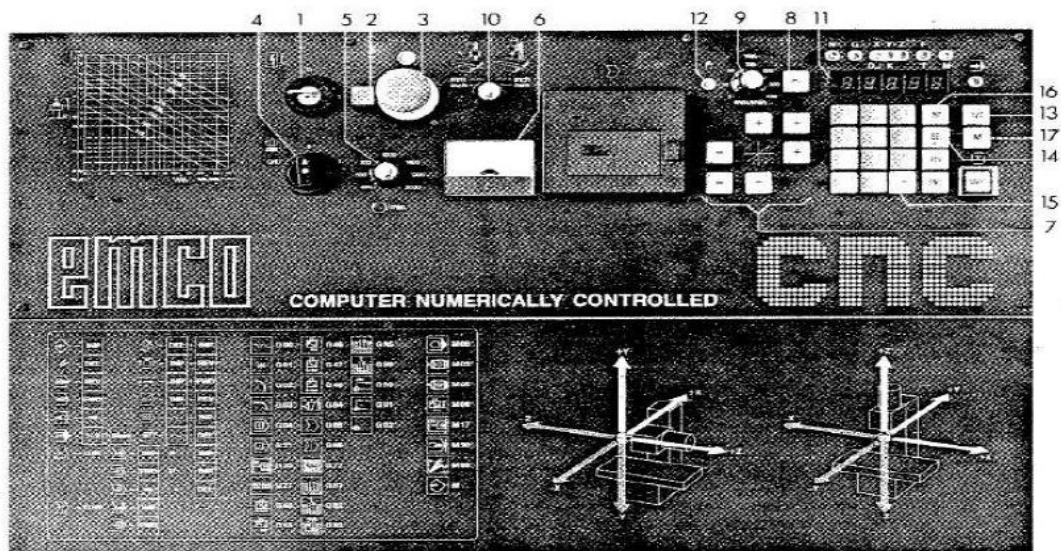
Slika 4.6: Nastavitev osi

5. Opis obstoječega delovnega stroja EMCO F1



Slika 5.1: EMCO F1 rezkalni stroj

Stroj je bil narejen leta 1984. Na šoli so ga uporabljali predvsem pri praksi, zdaj pa že nekaj časa stoji. Stroj v grobem sestavljata dva dela, sam stroj in pa konzola za upravljanje. Stroj je sestavljen iz delovne mize, na kateri je pritrjen primež za lažje delo in pa iz motorja. Miza se pomika v dveh smereh. V smeri X in Y, motor pa se premika v smeri Z. Za premike so uporabljeni koračni motorji. Te motorje lahko vodimo preko konzole s pritiskanjem na tipke (ročno) ali pa lahko vstavimo kaseto s posnetim programom in stroj dela sam (avtomatsko). Na konzoli tudi določimo hitrost vrtenja motorja. Na CNC lahko priključimo tudi televizijo na kateri spremljamo potek rezkanja.



Slika 5.2: Pogled na konzolo

1. Glavno stikalo.
2. Glavna kontrolna luč.
3. Prisilna zaustavitev (»STOP« stikalo).
4. Stikalo za glavno vreteno.
5. Gumb za nastavitev hitrosti glavnega vretena.
6. Ampermeter.
7. Tipke za ročni pomik.
8. Tipka za hitri pomik.
9. Gumb za nastavitev hitrosti pomika.
10. Stikalo za preklap med metričnim in colskim sistemom.
11. LED številčnica.
12. Kontrolna lučka za ročni pomik.
- 13.,14. ,15. ,16. ,17. Tipke za programiranje.

Za prenos iz koračnega motorja na vretena, ki pomikajo delovno mizo in pa za pomik motorja je uporabljen jermen. Vretena, ki služijo za pomik so kroglična. Zaradi tega osi nimajo skoraj nič zračnosti. Navadna vretena se hitreje obrabijo in imajo zračnost, kar pa ni primerno za CNC stroje, ker morajo biti kar se da natančni.



Slika 5.3: Kroglično vreteno.



Slika 5.4: Sprednji del rezkalnega stroja.



Slika 5.5: Zadnji del rezkalnega stroja

6. Opis koračnih motorjev

Koračni motor imenujemo tako zato, ker deluje po korakih. Običajni motorji se vrtijo takoj, ko vključimo stikalo in mu dovedemo napetost. Vrtijo se dokler ne izključimo stikala.

Koračni motor deluje malo drugače. Ko mu dovedemo napetost, se obrne samo za en korak, npr. za 1.8 stopinje. Zato tak motor potrebuje za en cel vrtljaj 200 korakov. 200 krat bi morali vključiti in izključiti stikalo, da bi motor naredil eno revolucijo (360 stopinj - 200×1.8).

Uporabljamo jih zato, ker so precizni. Za primer vzemimo, da želimo os motorja zavrteti za 180 stopinj. Iz podatka, da se motor v enem koraku obrne za 1.8 stopinje lahko izračunamo, da za 180 stopinj potrebujemo 100 korakov. Če mu os podaljšamo z navojno palico, po kateri se premika predmet, in če vemo, da se naprimer ob enem celem vrtljaju palice predmet na njej premakne za 5 milimetrov, sledi povzetek: ko se bo naš motor zavrtel za 360 stopinj (oziroma naredil 200 korakov), se bo naš predmet na palici premaknil za 5 mm.

Torej, če hočemo premakniti predmet za samo 1 mm, potrebujemo 5 krat manj stopinj in 5 krat manj korakov, se pravi $360 \text{ deljeno z } 5 = 72$ stopinj, oziroma $200 \text{ deljeno z } 5 = 40$ korakov našega motorja.

Koračne motorje delimo na dve veji, glede na izvedbo gonilnih tuljav in nato še na število gonilnih faz. To so:

Bipolarni koračni motorji

- Dvofazni.
- Trofazni.
- Petfazni .

Unipolarni koračni motorji

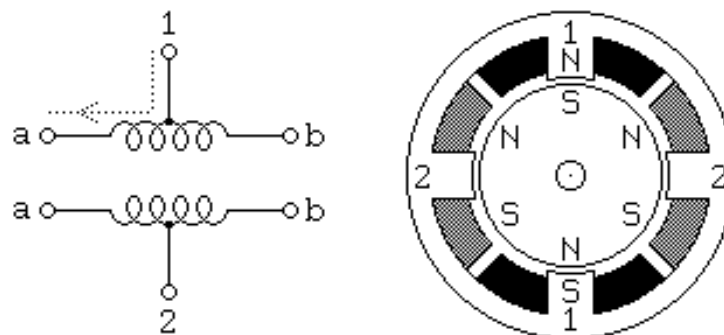
- Dvofazni.
- Trofazni.
- Petfazni.



Slika 6.1: Koračni motor

Unipolarni koračni motorji:

Unipolarni koračni motorji so sestavljeni iz dveh ločenih tuljav za po eno fazo, s čemer je omogočeno enostavnejše krmiljenje tuljav ene faze. Tako v načinu polnega ali polovičnega koraka potrebujemo le dva tranzistorja. S tem smo pocenili krmilno elektroniko, vendar imajo ti motorji v primerjavi z bipolarnimi motorji iste velikosti, manjši vrtilni in držalni moment. Ta posledica izvira iz dejstva, da je naenkrat priključena le polovica žice, ki je na razpolago v motorju za eno fazo, to pomeni, manjše magnetno polje! Seveda lahko navitji 1 in 2 ene faze vezamo zaporedno in motor uporabimo kot bipolarni motor, a ima takšen motor še vedno slabše lastnosti kot isti bipolarni. Slabše lastnosti so posledica večje induktivnosti in upornosti skupnega navitja. Tako bo imel motor pri nizkih vrtljajih podobne lastnosti kot bipolarni, čeprav ga bomo morali krmiliti z večjo napetostjo, ampak bo zaradi povečane induktivnosti navor pri večjih vrtljajih, hitro padel.



Slika 6.2: Shema in prerez koračnega motorja.



Slika 6.3: Koračni motor na EMCO F1

6.1 Krmiljenje koračnih motorjev

KRMILJENJE S KONSTANTNIM TOKOM

Če želimo izgubno moč motorja, zadržati v razumnih okvirjih, moramo imeti nadzor nad tokom skozi navitja. Enostavna in najbolj priljubljena rešitev je zagotoviti motorju tolikšno napetost, kot jo potrebuje za delovanje. Pri tej metodi tok omejuje notranja upornost navitij (RL). Bolj zapletena, a tudi bolj učinkovita in natančna rešitev za omejevanje toka, je uporaba tokovnega generatorja, zaradi katerega dosežemo neodvisnost toka od upornosti navitja. Napajalna napetost takega vezja, pa mora biti višja od napajalne napetosti prej omenjenega vezja, brez tokovnega generatorja.

Pravo primerjavo med vezjema lahko naredimo šele v spremenljivih pogojih delovanja koračnega motorja, oziroma, ko ga obremenimo z dinamičnim bremenom. Omenil sem že, da je navor motorja med drugim odvisen tudi od toka skozi navitja. Med delovanjem motorja se magnetno polje enega polovega para zamenja na vsaka dva koraka. Kako hitro tok narašča po svoji eksponencialni krivulji, je odvisno od induktivnosti navitja, njegove upornosti in priključene napetosti. Tok narašča pri nižji frekvenci in doseže svoje vršno stanje, preden se zamenja njegova smer. Če pa se smer toka zamenja pogosteje, kar pomeni večjo frekvenco korakov, tok več ne doseže svojega vršnega stanja. Moč in navor se zaradi tega bistveno zmanjšata pri višjih frekvencah

VEČJI NAVOR PRI VIŠJI FREKVENCI

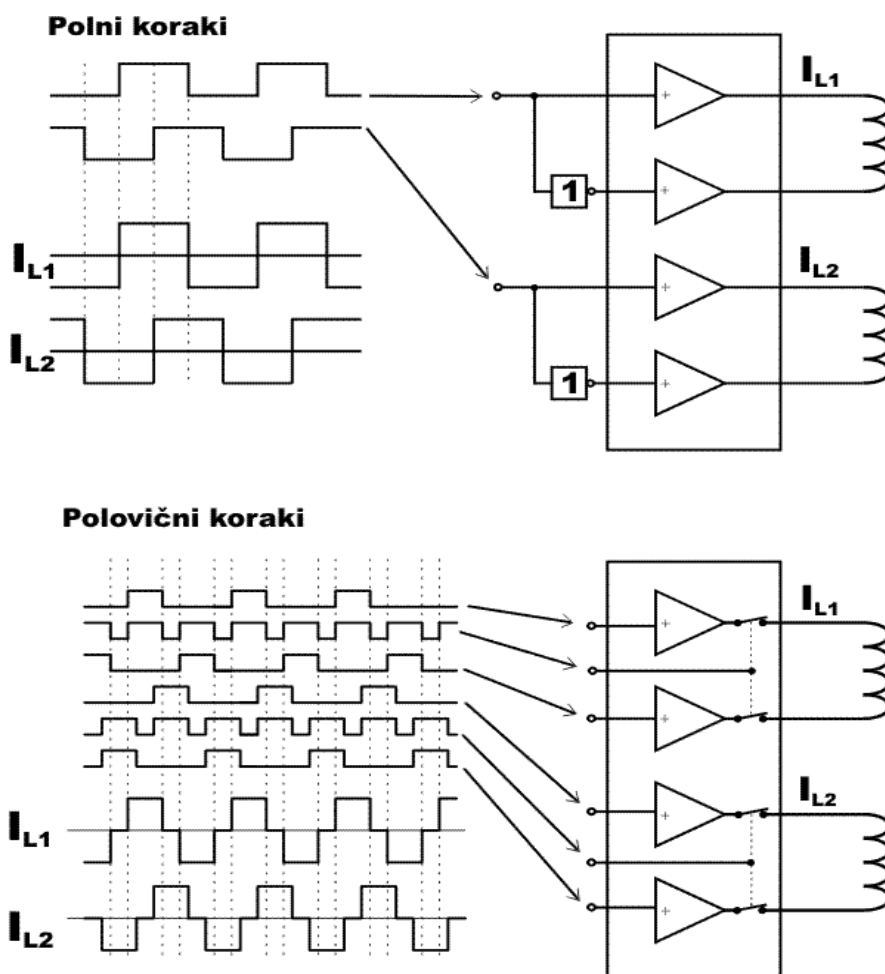
Večji navor pri višjih frekvencah je možen z uporabo tokovnega vira, kot kaže slika 4 desno. Pri tej varianti naj bo napajalna napetost čim višja, kar omogoča hitrejšo naraščanje toka v navitjih. Tokovni generator omejuje tok določene faze in se aktivira šele takrat, ko tok doseže njegovo nazivno vrednost. Do tega trenutka je tokovni generator v zasičenju in napajalna napetost je pripeljana direktno na tuljavo.

PREDNOSTI IN SLABOSTI »HALF STEP« NAČINA KRMILJENJA MOTORJA

Half step, kar po naše pomeni pol koraka, je način delovanja motorja, ko mu spremenimo krmilne signale in se ob vsakem urinem impulzu premakne le za pol koraka.

Poglavitna prednost takega krmiljenja je ta, da se motorju poveča resolucija za faktor 2. Tako dosežemo, da ima motor, ki ima sicer resolucijo $1,8^\circ$, kar pomeni 200 korakov na obrat, sedaj resolucijo $0,9^\circ$, kar pa je 400 korakov na obrat. Vendar to ni vedno glavni razlog, zaradi katerega uporabljamo half step krmiljenje. Velikokrat smo prisiljeni izbrati ta način zaradi motenj v delovanju, ki so posledica resonance motorja. Te so lahko tako močne, da motor v resonanci sploh nima več navora. Zato se začne nepredvidljivo obnašati in lahko izgubi svoj položaj. Razlog temu je dejstvo, da motor sam po sebi tvori mehansko nihalo, saj se magnetno polje statorja obnaša kot vzmet, za maso pa služi sam rotor. Zaradi tega, lahko motor, stimuliran pri tej frekvenci, zaniha. Resonančna frekvenca je odvisna od »napetosti vzmeti«, torej od jakosti magnetnega polja v statorju, ta pa je odvisen od toka. Čim večji je statorski tok, tem višja je resonančna frekvenca. V praksi lahko breme zaduši te oscilacije, vendar samo, če proizvaja dovolj trenja oziroma mehanskega dušenja. V večini primerov polovični koraki pomagajo, da ne pride do tega nezaželenega pojava, saj so koraki pol krajši in rotor ob koraku ne dobi take trenutne hitrosti. Dejstvo, da se polkoračni (half step) način ne posplošuje, pa je nekaj njegovih glavnih slabosti:

- Polkoračno krmiljenje zahteva dvakrat več urinih impulzov od full step načina, zato se frekvenca krmiljenja podvoji.
- V položaju polkoraka je navor motorja približno za polovico manjši od navora v celem koraku.



Slika 6.4: Krmilni signali za polnokoračno in polkoračno krmiljenje koračnega motorja

Zaradi teh dejstev, se polkoračni način uporablja le takrat, ko je urina frekvenca motorja v mejah nevarnega resonančnega območja. Dinamične izgube so tem večje, čim bolj se obremenitev motorja bliža njegovemu maksimalnemu momentu. Pojav se zmanjšuje pri večanju urine frekvence.

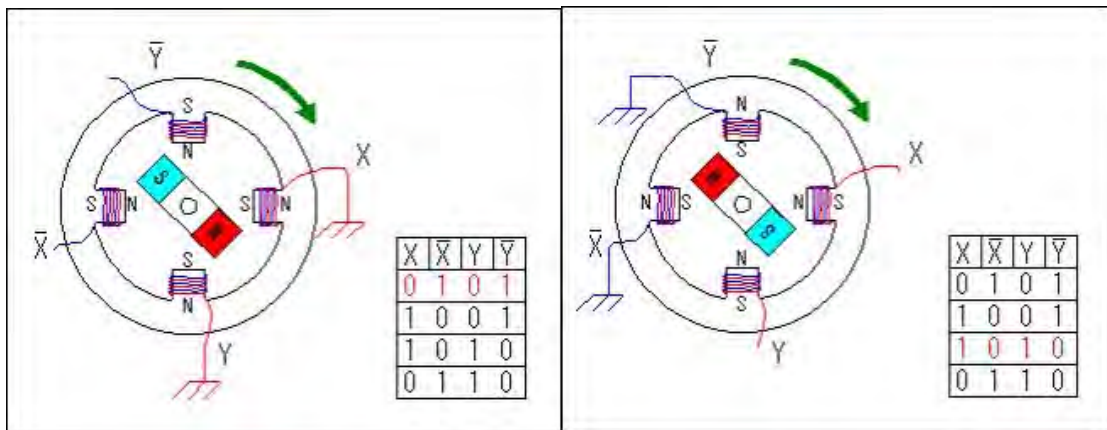
KRMILNI SIGNALI ZA LOGIKO

Običajni enosmerni motor se vrti, ko nanj priključimo enosmerno napetost. Koračni motor pa potrebuje komutiran signal iz večih ločenih, vendar povezljivih ukazov. V 95% današnjih naprav za te signale skrbijo mikroprocesorski sistemi.

Najenostavneje motor krmilimo v polnokoračnem načinu delovanja. Takrat potrebujemo le dva fazno zamaknjena pravokotna signala. Motor se vrti v smeri urinih kazalcev (CW, clockwise) ali v nasprotni smeri urinih kazalcev (CCW, counter clockwise), glede na to, katera faza vodi. Njegovi obrati so premosorazmerni s frekvenco impulzov.

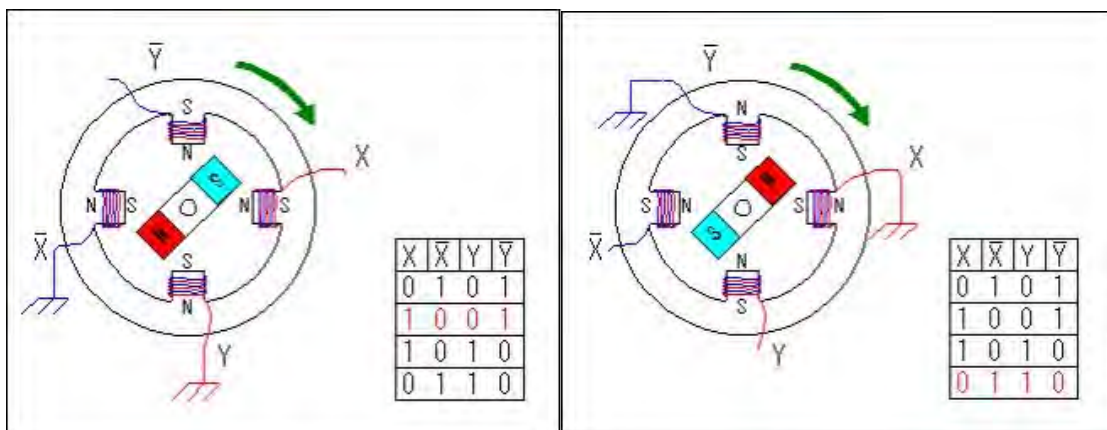
V polkoračnem načinu se stvar zaplete. Osnovna dva krmilna signala se prelevita v najmanj štiri signale. Če potrebujemo na izhodu vezja v določenem trenutku visoko impedanco, kar je nujno potrebno pri višjih obratih, potem potrebujemo kar šest krmilnih signalov. Glede na to, da so vsi signali med sabo sorodni, jih lahko generiramo s standardnimi logičnimi vrati. Vendar pa v primeru, ko potrebujemo polkoračni način delovanja, logika postane dokaj zapletena, zato se tudi podraži, tako vezje pa nam povrh vsega zavzame tudi več prostora. V takem primeru se bolj izplača uporabiti namensko integrirano vezje. Tako vezje nam poleg tega zmanjša zasedenost mikroprocesorja, saj namesto šestih krmilnih signalov potrebujemo le tri, dva statična in enega dinamičnega.

Položaji koračnih motorjev:



Slika 6.5: Položaj 1

Slika 6.7: Položaj 3



Slika 6.6: Položaj 2

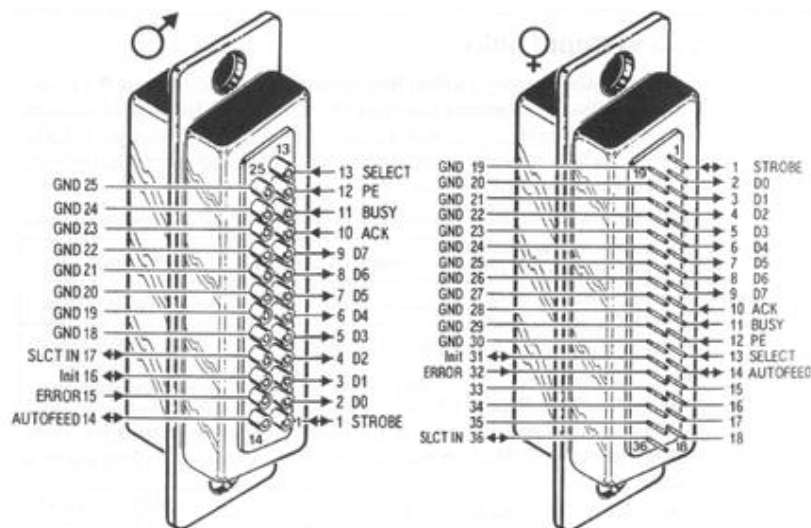
Slika 6.8: Položaj 4

7. LPT port

Lpt (Line Print Terminal) je originalno, ampak še zmeraj najbolj uporabljano ime za paralelni port na računalnikih. Načrtovan in narejen je bil za upravljanje 8 bitnih tiskalnikov. Lpt port se je končno standardiziral v poznih 90-ih letih, kot standard IEEE 1284, vendar pa se v današnjih dneh njegova uporaba manjša zaradi uporabe Universal Serial Bus (USB).

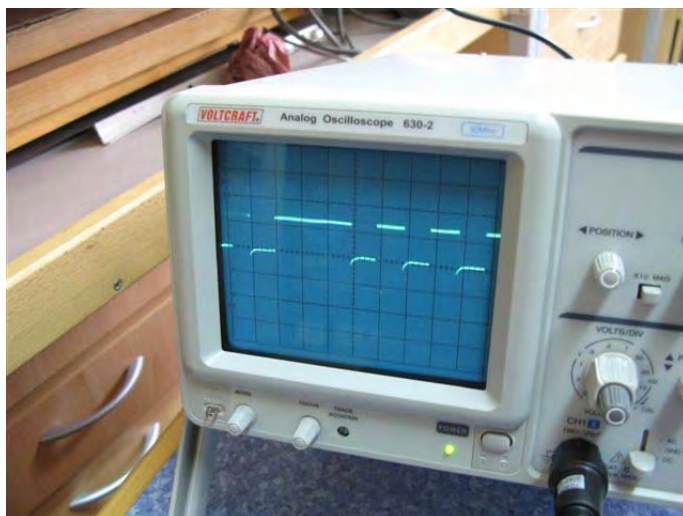
Široko področje naprav je bilo narejenih za povezavo z lpt priključkom. Večina teh naprav je delovalo samo v eni smeri, torej se je samo odzvalo na informacije poslane iz računalnika. Nekaj naprav pa je bilo dvosmernih, torej so tudi povratne informacije prihajale nazaj do računalnika.

V MS-DOS in PC-DOS, se je dalo dostopati do lpt porta kar preko „command line“, operacijski sistem Microsoft Windows pa zaradi varnostnih ne dovoljuje direkten dostop do porta. V Linux-ih je prvi lpt port zmeraj dostopen.

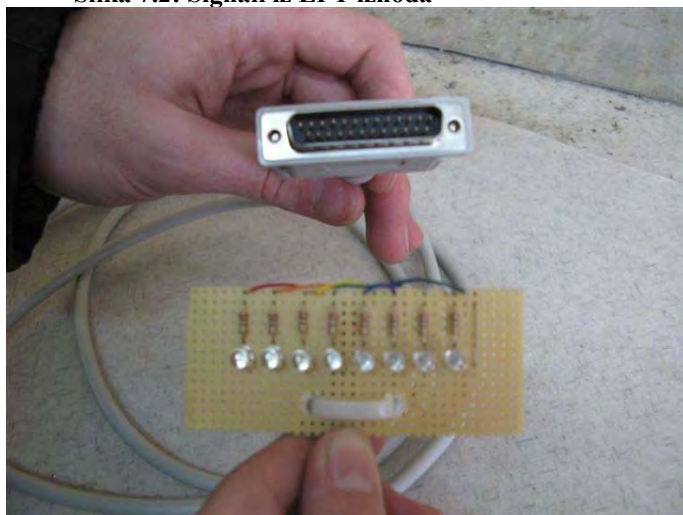


Slika 7.1: Shematski prikaz LPT porta

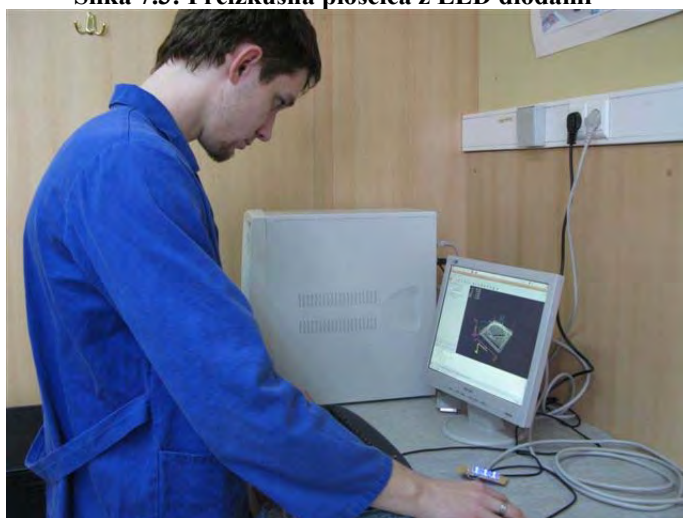
Ker smo hoteli preveriti, kakšne signale dobimo oziroma, če jih sploh dobimo, smo opravili meritve z osciloskopom. Naredili smo tudi ploščico s katero smo preverjali izhode.



Slika 7.2: Signali iz LPT izhoda



Slika 7.3: Preizkusna ploščica z LED diodami



Slika 7.4: Preizkušanje LPT izhodov(deluje)

8. Izdelava ploščice

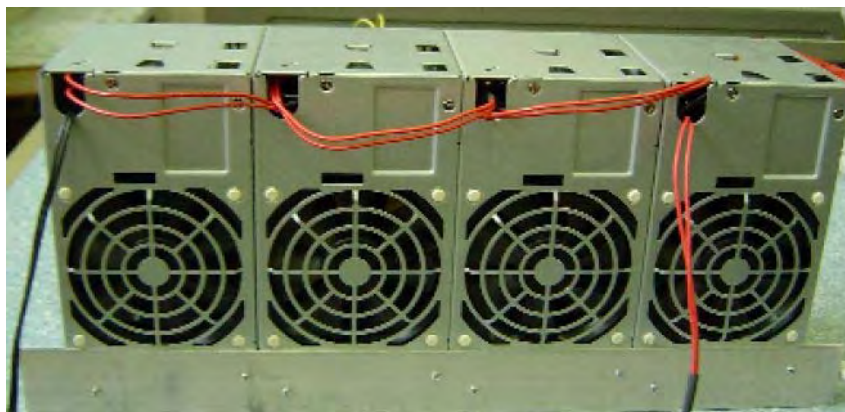
Izbira elementov za krmiljenje koračnega motorja.

Za naš krmilnik koračnih motorjev smo potrebovali:

1. KRMILNIK L297 (3x) - ta krmilnik razdeli signal za korak in smer v štiri signale za vsako navitje v koračnem motorju.
2. MOČNOSTNA STOPNJA uln2803 (3x) - ta stopnja okrepi signale iz krmilnika L297, da so dovolj močni za poganjanje motorja, vendar pa za naše motorje ni bila dovolj močna, zato smo jo zamenjali z tranzistorjem BDX53.
3. TRANZISTOR BDX53c (3x) - tudi ta tranzistor opravlja isto funkcijo kot močnostna stopnja, ampak smo zaradi prevelikega gretja elementa uporabili več tranzistorjev 2n6040(uporabili bi lahko oba).
4. TRANZISTOR 2n6040 (12x) - ta tranzistor nam je najbolj ustrezal, zato smo ga uporabili za naš krmilnik koračnih motorjev.
5. UPOR 3.3k Ω (6x) - ta upor skrbi za konstantno napetost na krmilniku.
6. DIODE 1n4001 (24x) - diode so elementi, ki prepuščajo tok le v eno smer.

Izbira napajalnika

Za naš krmilnik potrebujemo 5V, 46V in konstanten tok 8A. Do teh napetosti lahko pridemo na dva načina. Ali v serijo vežemo več računalniških napajalnikov, ali izdelava lastnega napajalnika, kar je seveda dražje in zahtevnejše. Najboljša izbira je seveda vezava računalniških napajalnikov v serijo. Vsak računalniški napajalnik ima izhod 5V in 12V pri 2A, zato je potrebna vezava štirih napajalnikov.

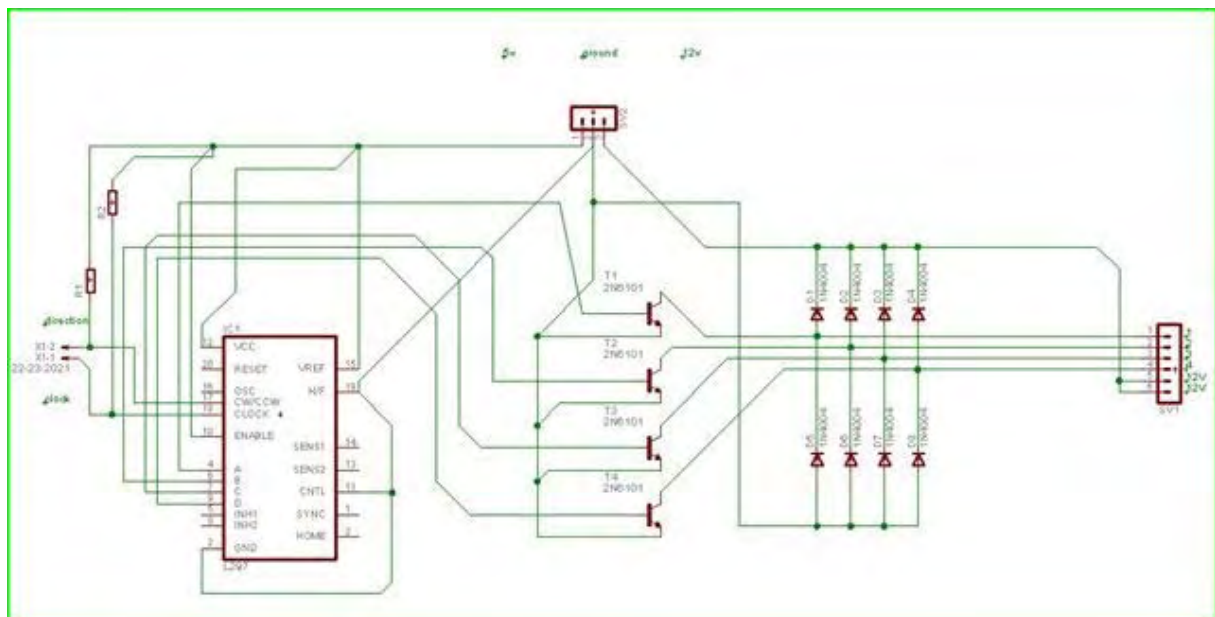


Slika 8.1: Napajalniki

8.1. Izris sheme v EAGLU

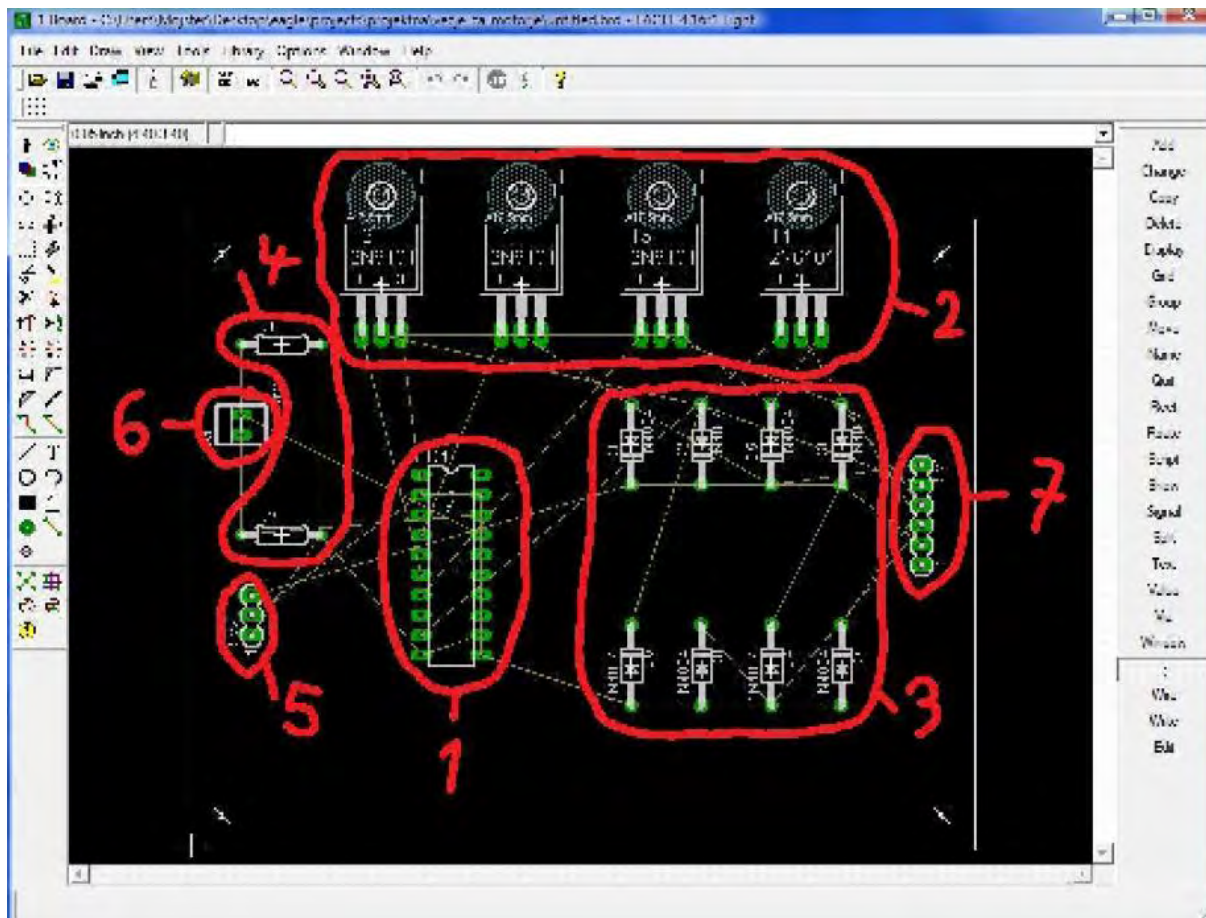
Shema krmilnika koračnih motorjev

Ker je bilo testiranje z testno ploščico uspešno, smo se lotili izdelave prave ploščice. Najprej smo s programom Eagle narisali shemo ploščice:



Slika 8.2: Shema vezja za en motor

Nato pa smo z istim programom še izrisali obliko plošče.



Slika 8.3: Ploščica za krmiljenje enega motorja

Na obeh slikah so prikazani in oštevilčeni elementi:

- 14. Krmilnik L297.
- 14. Tranzistorji 2n6040.
- 14. Diode 1n4001.
- 14. Upora 3.3k Ω .
- 14. Vhod za napetost (+5V, minus, +46V).
- 14. Vhod za signale (korak in smer).
- 14. Izhodi za koračne motorje (1,2,3,4,A,B).

Opis delovanja:

Signala za korak in smer, ki ju pošlje računalnik preko LPT porta, krmilnik L297 razdeli v signale za štiri navitja v koračnem motorju, tako da si sledijo v pravem vrstnem redu, glede na smer in hitrost vrtenja. Ti signali potujejo do tranzistorjev 2n6040, ki jih primerno ojačajo, da lahko poganjajo močne motorje na frezalnem stroju.

8.2. Priklučitev in testiranje

Po začetnih težavah, smo zvezali testno ploščico in jo tudi preizkusili na stroju. Sprva je delovalo, vendar ne dolgo, saj smo imeli premajhno napetost. Težavo smo rešili z vezavo štirih napajalnikov.



Slika 8.4: Priklučitev in testiranje testne ploščice

9. Končna/referenčna stikala

Stroj smo omejili s končnimi oziroma referenčnimi stikali. Ta stikala smo namestili na skrajne lege stroja. Prej so bile samo osi, označene samo kje je skrajna lega, tukaj pa smo tudi namestili stikala.



Slika 9.1: Izdelava nosilca za stikala



Slika 9.2: Namestitev stikal

10. Varnost

Seveda je stroj kot tak, zelo nevarno orodje, zato so za varno uporabo bistvenega pomena varnostni sistemi. Obvezen varnostni sistem na vsakem stroju je gumb za zasilno ustavitev.

Ta popolnoma izključi električni tok potreben za obratovanje stroja.



Slika 10.1: Gumb za zasilno ustavitev

Pri našem stroju je pomemben varnostni element tudi senzor za zaprta vrata delovnega stroja.

Ta senzor pove stroju, da so vrata zaprta in da je pripravljen za delovni proces.



Slika 10.2: Senzor za vrata na našem rezkarju

11. G koda

Začetki G kode segajo nekje v zgodnja šestdeseta leta dvajsetega stoletja, ko so se začele pojavljati zahteve po NC programih za prenovljene stroje. Do takrat so za avtomatske stroje uporabljali preluknjane kartice ipd. Prvič se je G koda standardizirala leta 1980 in tako postavila osnovo kode. Skozi leta se je osnova dopolnjevala in razvijala do takšne, kot jo pozna vsak CNC operater danes.

G koda je postala jezik, ki se uporablja za vodenje CNC strojev po vsem svetu.

Zgradba G kode:

G koda je zgrajena iz blokov (ang. „Block“) oziroma vrstic. En blok lahko vsebuje eno ali več besed sestavljenih iz črke, ki opisuje nastavitve oziroma funkcijo, ki jo mora stroj opraviti ter številko, katera predstavlja vrednost.

Naprimera:

N001 G00 X10.0

N001 predstavlja zaporedno številko bloka.

G00 predstavlja hitri premik.

X10.0 pa predstavlja za koliko enot se premakne v danih oseh (v tem primeru x).

Številke so lahko pozitivne ali imajo negativni predznak (npr. X-10).

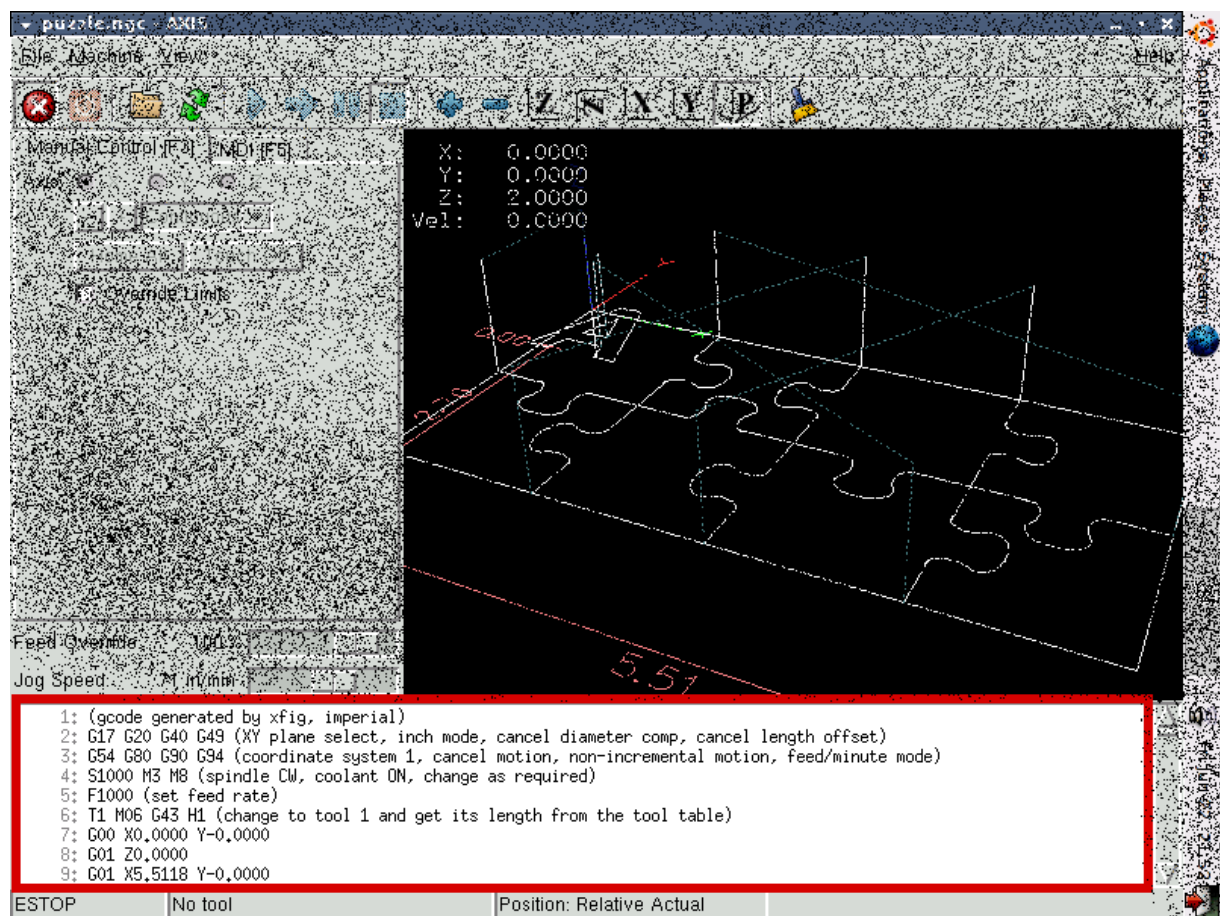
Število decimalk je odvisno od natančnosti (npr. X10.255).

Presledki med besedami niso nujni, vendar so priporočeni zaradi lažjega razbiranja kode.

Prav tako velja za prazne bloke. Če je blok prazen, ga stroj ignorira.

Nekaj osnovnih in najpogosteje uporabljenih ukazov:

Koda	Funkcija	Koda	Funkcija
G0	Hitri premik	M0	Ustavi program
G1	Linearna interpolacija	M1	Ustavi podprogram
G2	Krožna interpolacija smer ure	M2	Končaj program
G3	Krožna interpolacija nasprotna smer	M3	Zaženi vreteno v smeri ure
G4	Ignoriraj podatkovni blok	M4	Zaženi vreteno v nasprotni smeri
G36	Prični poligon	M5	Ustavi vreteno
G37	Končaj poligon	M6	Zamenjava orodja
G54	Koordinatni sistem obdelovanca	M7	Vključi mazanje
G70	Nastavi inče	M8	Vključi hlajenje
G70	Nastavi milimetre	M9	Izključi mazanje oz. hlajenje
Fxx	Hitrost podajanja	M30	Zaključ program in se vrni v izhodišče
G33	Vrezovanje navoja	G90	Absolutno pozicioniranje
G43	Kompensacija dolžine orodja (plus)	G91	Inkrementalno pozicioniranje
G44	Kompensacija dolžine orodja (minus)		



Slika 11.1: G koda v programu EMC²

12. Potrditev oziroma zavrnitev tez/hipotez

1. Hipoteza: S posodabljanjem stroja se bo povečala možnost uporabe le tega.

S posodobitvijo se bo res povečala uporabnost tega CNC stroja. Sicer samo za malo manjše stvari, vendar uporabnost se bo povečala.

2. Hipoteza: Spoznali bomo delovno okolje odprto kodnega operacijskega sistema Linux Ubuntu.

Med posodobitvijo stroja smo dodobra spoznali operacijski sistem Linux Ubuntu in ga tudi s pridom uporabljali.

3. Hipoteza: Osebni računalnik(PC) in primerno programsko opremo lahko uporabimo za namen vodenja CNC rezkalnega stroja.

Osebni računalnik in primerna programska oprema zadostuje za vodenje CNC rezkalnega stroja.

4. Hipoteza: Z omenjenim pristopom lahko posodobimo starejše obdelovalne stroje.

Starejše obdelovalne stroje res lahko posodobimo z primerno programsko opremo in računalnikom, vendar moramo paziti kakšne motorje uporabljamo za pogon. Ker pri servo motorjih potrebujemo še povratno informacijo glede položaja, ponavadi uporabimo inkrementalni dajalnik položaja.

5. Hipoteza: Spoznali bomo tudi problematiko pri avtomatizaciji ali posodobitvi obdelovalnih strojev.

Med celotno posodobitvijo smo naleteli na številne tehnične težave. Paziti moramo na pravilno opremo in pa vse preverjati pred priklopom na dejanski stroj.

13. Zaključek

Med izdelovanjem raziskovalne naloge smo se veliko naučili o elektrotehnik in elektroniki. Spoznali smo kakšni problemi se lahko pojavijo in kako jih najlažje odpraviti.

Mislim, da smo stroju povečali življenjsko dobo in uporabnost vsaj za par let. Tehnika se pa vsak dan spreminja in mi ji moramo slediti.



Slika 13.1: Skupinska

14. Zahvala

Zahvaljujemo se vsem, ki so nam pomagali pri delu, saj brez njih naša raziskovalna naloga ne bi bila končana.

Posebna zahvala je namenjena mentorju Mateju Vebru, univerzitetnem diplomiranim inženirju elektrotehnike, ki nas je ves čas vzpodbujal, nam dajal napotke in nas vodil do pravih informacij.

15. Viri

Instruction book Service parts EMCO F1-CNC

CNC-Ausbildung für jeden erschwilglich

EMCO F1 manual

EMC² User manual

Internet:

http://www2.arnes.si/~sspslavr/k_motor/k_motor.html

<http://hercules.uni-mb.si/SP/CNC/Spetna%20stran/koracni%20motorji.htm>

<http://lpvo.fe.uni-lj.si/Predmeti/EV/Godec%20Andrej/Kontroler.pdf>

http://www.polarhome.com:793/~umnik//index2.php?option=com_content&do_pdf=1&id=21

http://images.google.si/imgres?imgurl=http://www.pef.uni-lj.si/eprolab/comlab/overview/photos/elec-machines-stepmot12.jpg&imgrefurl=http://www.pef.uni-lj.si/eprolab/comlab/overview/ov-elec-machines-si.htm&usq=__m9SX2N_wF5z1pFiLqS2ljMH3UgQ=&h=480&w=640&sz=44&hl=sl&start=9&um=1&tbnid=zBPQdUlb6ea92M:&tbnh=103&tbnw=137&prev=/images%3Fq%3Dkora%25C4%258Dni%2Bmotorji%26hl%3Dsl%26sa%3DN%26um%3D1