



Gregor Steblovnik

**MERILNI SISTEM ZA MERJENJE SILE
PRITISKA IN POLOŽAJA TIPK PRI STROJIH**

Diplomska delo

Maribor, april 2009



UNIVERZA V MARIBORU



FAKULTETA ZA ELEKTROTEHNIKO,
RAČUNALNIŠTVO IN INFORMATIKO
2000 Maribor, Smetanova ul. 17

Diplomska naloga univerzitetnega študijskega programa

MERILNI SISTEM ZA MERJENJE SILE PRITISKA IN POLOŽAJA TIPK PRI STROJIH

Študent: Gregor Steblovnik
Študijski program: Univerzitetni, Elektrotehnika
Smer: Elektronika

Mentor: doc. dr. Iztok KRAMBERGER
Somentor: doc. dr. Mitja SOLAR

Maribor, april 2009



UNIVERZA V MARIBORU



FAKULTETA ZA ELEKTROTEHNIKO,
RAČUNALNIŠTVO IN INFORMATIKO
2000 Maribor, Smetanova ul. 17

Številka: E-2524

Datum: 06. 10. 2008

SKLEP O DIPLOMSKEM DELU

1. **Gregor Steblovnik**, študent univerzitetnega študijskega programa Elektrotehnika, smer Elektronika, izpolnjuje pogoje, zato se mu dovoljuje izdelati diplomsko delo.
2. Tema diplomskega dela je s področja Inštituta za elektroniko in telekomunikacije pri predmetu **VMESNIKI IN PRETVORNIKI**.

MENTOR: doc. dr. Iztok Kramberger

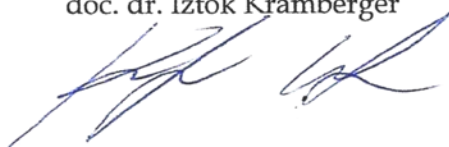
3. Naslov diplomskega dela:
MERILNI SISTEM ZA MERJENJE SILE PRITISKA IN POLOŽAJA TIPK PRI STROJIH
4. Vsebina diplomskega dela:
Razvijte in izdelajte merilni sistem za merjenje sile pritiska in položaja tipk pri strojih, ki bo omogočal simuliran vklop in izklop strojev s pomočjo pritiskanja tipke in pri tem meril silo pritiska ter položaj, v katerem je tipka, ko se stroj dejansko vključi oziroma izključi. Merilni sistem na osnovi 8-bitnega mikrokrmilnika naj preko galvansko ločene zaporedne komunikacije omogoča povezavo na osebni računalnik, pri tem pa izdelajte nadzorni program za prikazovanje in shranjevanje rezultatov na osebni računalniku s pomočjo programskega paketa Labview. Mehanski del merilnega sistema naj bo krmiljen s pomočjo koračnega motorja.
5. Diplomsko delo je potrebno izdelati skladno z "Navodili za izdelavo diplomskega dela" in ga oddati v treh izvodih do 06. 10. 2009 v referatu za študentske zadeve.

PREDSTOJNIK INŠTITUTA



red. prof. dr. Zmago Brezočnik



doc. dr. Iztok Kramberger



DEKAN


red. prof. dr. Igor Tičar

ZAHVALA

Zahvaljujem se mentorju doc. dr. Iztoku Krambergerju za strokovno pomoč in vodenje pri opravljanju diplomskega dela ter doc. dr. Mitji Solarju za somentorstvo pri nastajanju diplomskega dela.

Zahvaljujem se tudi podjetju Elrad International d.o.o. za pomoč, ki so mi jo nudili pri izdelavi diplomske naloge.

Posebna zahvala velja staršem, vsem domačim in Valentini, za vso podporo v času študija.

Merilni sistem za merjenje sile pritiska in položaja tipke pri strojih

Ključne besede: elektronika, zaporedna komunikacija, linearni optični dajalniki, mikrokrmilniki, merilniki sile, vgrajeni sistemi

UDK: 621.38:531.7(043.2)

Povzetek

V diplomskem delu je opisan razvoj in izdelava merilnega sistema za merjenje sile pritiska in položaja tipke pri strojih. Merilni sistem je v svoji osnovni funkciji namenjen merjenju mehansko-električnih karakteristik različnih upravljalnih tipk pri strojih z elektronskim upravljanjem. Sestavljen je iz merilno vpenjalne enote, krmilne enote ter nadzornega sistema. Na krmilno enoto je preko zaporednega RS232 vmesnika priključen merilnik sile ter osebni računalnik, na katerem je nameščen nadzorni sistem. Nadzorni sistem omogoča avtomatsko meritev dveh različnih tipk, in sicer z enojnim ter dvojnimi stikalom. Poleg tega je možno še ročno upravljanje koračnega motorja. To je mogoče tudi s pomočjo tipk, ki so priključene na krmilno enoto. Nadzorni sistem izpisuje merilne rezultate v realnem času na LCD zaslon in na zaslon računalnika, poleg tega pa hkrati shranjuje izmerjene rezultate v datoteko na osebni računalniku.

Measuring system for force and displacement measurement of machine buttons

Key words: electronics, serial communication, linear optical encoder, microcontrollers, force gauge, embedded systems

UDK: 621.38:531.7(043.2)

Abstract

The development of the measuring system is described in the thesis. The basic function of the measuring system is measurement of the electro - mechanical characteristics of different machine buttons. The measuring system is composed of mechanical clamping unit, control unit and surveillance system. Force gauge and personal computer with the surveillance system are over serial RS232 interface connected to the control unit. The surveillance system allows automatic measurement of two styles of buttons, one switch button and two switch button. The surveillance system also enables manual use of stepper motor which can also be done with 3 buttons that are connected to the control unit. Besides real time readout of the measuring results on LCD display and display of the computer the surveillance system saves these results to a file on the personal computer.

KAZALO

1.	UVOD.....	1
2.	PREGLED STANJA.....	3
3.	OPIS MERILNEGA SISTEMA	7
3.1	Merilno vpenjalna enota.....	8
3.1.1.	Koračni motor	9
3.1.2.	Merilnik razdalje.....	10
3.1.3.	Merilnik sile	11
3.2	Delovanje merilno vpenjalne enote	12
3.3	Krmilnik merilnega sistema.....	14
3.3.1.	Napajalno vezje	16
3.3.2.	Krmilnik koračnega motorja	17
3.3.3.	RS232 vmesnik	20
3.3.4.	Vmesnik za LCD zaslon.....	21
3.3.5.	Vmesnik za zaznavanje vklopa in izklopa merjenja.....	22
3.4	Programska oprema mikrokrmilnika	23
3.4.1.	Glavna zanka programa	25
3.4.2.	Prekinitveni servisni programi ter prekinitveni vektorji	25
3.4.3.	Modul za krmiljenje koračnega motorja	27
3.4.4.	Modul za obravnavanje merilne letve	28
3.4.5.	Modul za obravnavanje merilca sile in komunikacije z osebnim računalnikom.....	31
3.4.6.	LCD modul	33
3.5	Sistem za nadzor in prikaz merilnih rezultatov	36
3.6	Razdelitev niza v posamezne dele	38
3.7	Določanje karakterističnih točk.....	40
3.8	Funkcija za nadzor delovanja sistema	42

4. MERITVE IN REZULTATI.....	44
4.1 Meritev električnih karakteristik krmilnika	44
4.1.1. Meritev porabe merilnega sistema.....	44
4.2 Meritev signalov za krmiljenje koračnega motorja	44
4.3 Umerjanje sistema	48
4.4 Izračun vzmetne konstante vzmeti	49
4.5 Meritev tipke z dvojnimi stikalom.....	51
4.6 Primerjava različnih merilnih sistemov.....	54
5. SKLEP	56
6. VIRI, LITERATURA	58
7. PRILOGE.....	60

KAZALO SLIK

Slika 2.1: Wheastonov mostič z notranjo kompenzacijo	4
Slika 2.2: Obremenilna celica z obliko S.....	5
Slika 2.3: Mecmesin MultiTest 1-d, Tinius Olsen H1k, Mark-10 ESM301	6
Slika 2.4: MECMESIN Emperor Lite Software z izmerjeno karakteristiko tipke.....	6
Slika 3.1: Blokovna shema merilnega sistema	7
Slika 3.2: Model merilno vpenjalne enote	8
Slika 3.3: Koračni motor QSH4218	9
Slika 3.4: Prikaz sestave merilne letve	10
Slika 3.5: Merilna letev	11
Slika 3.6: Merilnik sile FG-5000A.....	11
Slika 3.7: Prikaz načina simulacije pritiska prsta	12
Slika 3.8: Merilno vpenjalna naprava z merjencem	13
Slika 3.9: Krmilni modul	14
Slika 3.10: Blokovna shema napajanja vezja	16
Slika 3.11: Primer vezja z L297 in L298 in kontrolni signali.....	17
Slika 3.12: Blokovna shema integriranega vezja L297	18
Slika 3.13: Krmilni signali za polnokoračno in polkoračno krmiljenje koračnega motorja	19
Slika 3.14: Izhodni signali v odvisnosti od krmilnega urinega signala pri polnokoračnem načinu.....	20
Slika 3.15: Izhodni signali v odvisnosti od krmilnega urinega signala pri polkoračnem načinu.....	20
Slika 3.16: Blokovna shema povezave LCD zaslona z mikrokrmilnikom	22
Slika 3.17: LCD zaslon WH1602A, Winstar Display Co.....	22
Slika 3.18: Shema dela vezja za galvansko ločitev vhodov na mikrokrmilnik.....	23
Slika 3.19: Diagram poteka programa	24
Slika 3.20: Del programske kode - Krmiljenje koračnega motorja	28
Slika 3.21: Izhodni signali merilne letve	29
Slika 3.22: Del programske kode: Preverjanje smeri merilne letve in merjenje razdalje premika merilne letve	30
Slika 3.23: Komunikacijski protokol merilca sile	31

Slika 3.24: Oblika niza, ki se prenaša na osebni računalnik.....	31
Slika 3.25: Del programske kode: Pridobivanje znakov v mikrokrmilnik preko prekinjenih vektorjev USART0_RX_vect in USART1_RX_vect.....	32
Slika 3.26: Diagram poteka inicializacije LCD zaslona za 4-bitno povezavo z mikrokrmilnikom.....	33
Slika 3.27: Funkciji za vpis ukaza in za izpis znaka na LCD zaslon.....	34
Slika 3.28: Časovni potek signalov za krmiljenje LCD zaslona.....	35
Slika 3.29: Uporabniški vmesnik za prikaz merilnih rezultatov in nadzor aplikacije.....	37
Slika 3.30: Glavni blokovni diagram uporabniškega vmesnika	38
Slika 3.31: »Koda« funkcije za pretvorbo niza v posamezne dele	39
Slika 3.32: Funkcija za določanje karakterističnih točk	41
Slika 3.33: Funkcija za nadzor delovanja sistema	42
Slika 4.1: Izhodni signal na L297 v odvisnosti od vhodnega urinega signala pri polnokoračnem načinu	45
Slika 4.2: Vsi izhodni signali na L297 pri polnokoračnem načinu.....	46
Slika 4.3: Izhodni signal na L297 v odvisnosti od vhodnega urinega signala pri polkoračnem načinu	47
Slika 4.4: Vsi izhodni signali na L297 pri polkoračnem načinu.....	47
Slika 4.5: Cilindrična tlačna vzmet [17].....	48
Slika 4.6: Karakteristika stiska vzmeti	50
Slika 4.7: Potek meritve karakteristike stiska vzmeti	51
Slika 4.8: Potek signala in karakteristične točke tipke z dvojnim stikalom.....	52
Slika 4.9: Gaussova krivulja pri merjeni vrednosti vklopnih točk A 20 merjencev	53
Slika 4.10: Mecmesin MultiTest 2.5-d, Tinius Olsen H1k, Mark-10 ESM301, naš sistem.....	55

1. UVOD

Diplomsko nalogo smo izdelali v sodelovanju s podjetjem Elrad International d. o. o. iz Gornje Radgone.

Elrad International d. o. o. je eden izmed vodilnih slovenskih proizvajalcev elektronskih sklopov za prosti trg (OEM trg). Za kupce razvijajo in proizvajajo elektronske sklope za končne izdelke na področju malih gospodinjskih aparatov, električnih orodij, bele tehnike, avtomobilske industrije in drugih področjih.

V okviru diplomske naloge smo si zadali cilj zasnovati in izdelati merilni sistem, ki bo omogočal simulacijo vklopa vrtalnega stroja s pomočjo pritiska na tipko za vklop. Hkrati bo sistem meril silo pritiska na tipko za vklop ter položaj tipke, v katerem se stroj dejansko vklopi oziroma izklopi. Območje gibanja tipke, ko nanjo pritiskamo, leži v območju od 0-1cm oziroma od 0-3 cm. Zanimajo nas značilnosti gibanja tipke v fazi vklopa in izklopa stroja, to je sila pritiska in mesto vklopa ter izklopa. Te značilnosti gibanja oziroma gibanja tipke želimo tudi posneti. Odločili smo se torej, da razvijemo sistem, ki bo omogočal natančno merjenje odziva tipke v procesu vklopa in izklopa naprave. V ta namen smo za merjenje premika uporabili merilno letev, ki omogoča merjenje tudi z natančnostjo 20 μm in predstavlja jedro tega sistema. Poleg omenjenih značilnosti tipke, lahko s tem sistemom izvajamo tudi druge meritve, kot na primer merjenje sile in raztezka vzmeti, meritve sile, ki je potrebna za upogib elektronske plošče na določeni razdalji in druge. Mehanski del merilnega sistema za krmiljen vklop stroja pomikamo s pomočjo koračnega motorja. Merilni sistem je preko galvansko ločene zaporedne komunikacije povezan na osebni računalnik, na katerem se v nadzornem programu shranjujejo vse opravljene meritve. Nadzorni program za prikazovanje in shranjevanje rezultatov na osebni računalniku smo razvili s pomočjo programskega paketa Labview.

V drugem poglavju opisujemo razloge za izdelavo takega sistema ter posameznih delov, ki smo jih v sistemu uporabili ter pregled stanja takšnih sistemov na trgu in opis njihovih prednosti oziroma slabosti.

V tretjem poglavju predstavljamo sestavo merilnega sistema. Opisujemo mehansko zgradbo celega sistema in zgradbo elektronske krmilne enote. Podrobneje opisujemo

posamezne enote, ki jih uporabljamo in zgradbo programske opreme krmilnega sistema. Opisujemo tudi nadzorni sistem in posamezne sestavne bloke sistema.

V četrtem poglavju predstavljamo umerjanje sistema ter vse meritve opisanega sistema. Izmerili smo karakteristike posameznih delov krmilnega sistema. Predstavljamo tudi različne meritve, ki smo jih izvedli s pomočjo tega merilnega sistema.

V zadnjem, petem poglavju, povzemamo rezultate celotne raziskave in razvoja, ki smo ga opravili. Navajamo, kje smo naleteli na težave, ter predlagamo možne izboljšave sistema.

2. PREGLED STANJA

V podjetju Elrad International d.o.o. so potrebovali sistem, ki bi omogočal meritev elektromehanskih karakteristik vklopa tipk, vgrajenih v elektronske sklope, proizvedene v podjetju Elrad, za krmiljenje raznih strojev oziroma naprav. Ker so v podjetju Elrad že imeli merilnik sile s stojalom zanj, so želeli, da na njegovi osnovi, razvijemo sistem, ki bo omogočal takšne vrste meritev. Tako smo morali poiskati še primeren merilnik razdalje in razviti krmilni ter nadzorni sistem. Ker so nam dali proste roke glede izbire merilnika razdalje smo se odločili, da bomo uporabili linearni inkrementalni dajalnik, ki omogoča veliko natančnost meritve položaja vklopa in izklopa tipk. V podjetju so želeli, da sistem omogoča meritev, ki bo čimmanj odvisna od človeške napake, zato smo poleg merilnika razdalje dodali sistemu še koračni motor. Krmilni sistem tega koračnega motorja bo izvajal mehanski pomik posebne ročice sistema in s tem avtomatsko simulacijo pritiska na tipko oziroma njenega vklopa. Takšen sistem moramo še nadzorovati preko dodatnega računalniškega sistema. Odločili smo se, da bomo uporabili programski paket LabVIEW, ki je primeren za takšne vrste aplikacij kot je naša. Hkrati s tem, da omogoča razvoj in izdelavo aplikacije za nadzor sistema, lahko v ta programski paket vključimo še sistem za zajemanje in shranjevanje merilnih rezultatov.

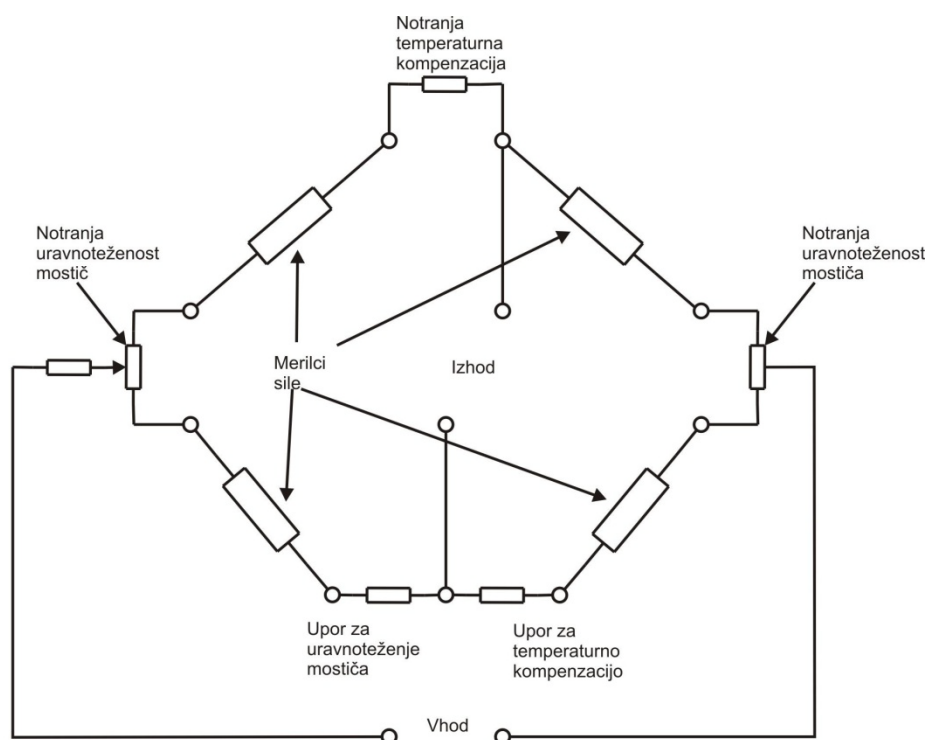
S pregledom stanja smo ugotovili, da na trgu že obstajajo sistemi, ki omogočajo meritve karakteristik stikal. To so profesionalni sistemi in omogočajo več različnih meritev in so seveda tudi relativno dragi. V tabeli 1 prikazujemo primerjavo treh različnih sistemov, ki jih lahko kupimo na trgu:

- Mecmesin Multitest 2.5-d [18]
- Tinius Olsen H1k [20]
- *Mark-10 EMS301* [19]

Slika 2.3 prikazuje te tri sisteme s trga.

Vsi trije sistemi temeljijo na merjenju sile in premika. Na kakšen način oziroma s kakšnim senzorjem merijo premik nam ni uspelo ugotoviti. Predvidevamo pa, da glede na obliko samih sistemov, uporabljajo neke vrste linearnih dajalnikov. Vsem trem je skupna tudi

vrsta merilca sile – digitalni merilniki sile. Čeprav se še vedno uporabljajo mehanski merilniki sile, pa jih v svetu vse bolj izpodrivajo digitalni merilniki. Digitalni merilnik sile je v osnovi ročni instrument, ki vsebuje integrirano obremenilno celico (ang. load cell), elektroniko ter zaslon. Obremenilna celica oziroma senzor sile je v osrčju digitalnega merilca sile. Obremenilna celica je vrsta pretvornika, ki pretvori silo, s katero pritiskamo nanjo, v izhodno električno veličino. Najpogosteje so zgrajene iz uporovnih merilnih lističev (ang. strain gauge) ali s piezo kristalom. Večja kot je sila na senzor, večja je izhodna napetost. Ponavadi so uporabljene 4 obremenilne celice, ki so povezane v obliki Wheatstonovega mostiča, kot je prikazano na sliki 2.1 [21]. Dve se uporabljata za merjenje stiska in dve za merjenje potega. Preko elektronike in mikrokrmilnika je ta sila potem pretvorjena v vrednost sile ter izpisana na zaslon.



Slika 2.1: Wheatstonov mostič z notranjo kompenzacijo



Slika 2.2: Obremenilna celica z obliko S

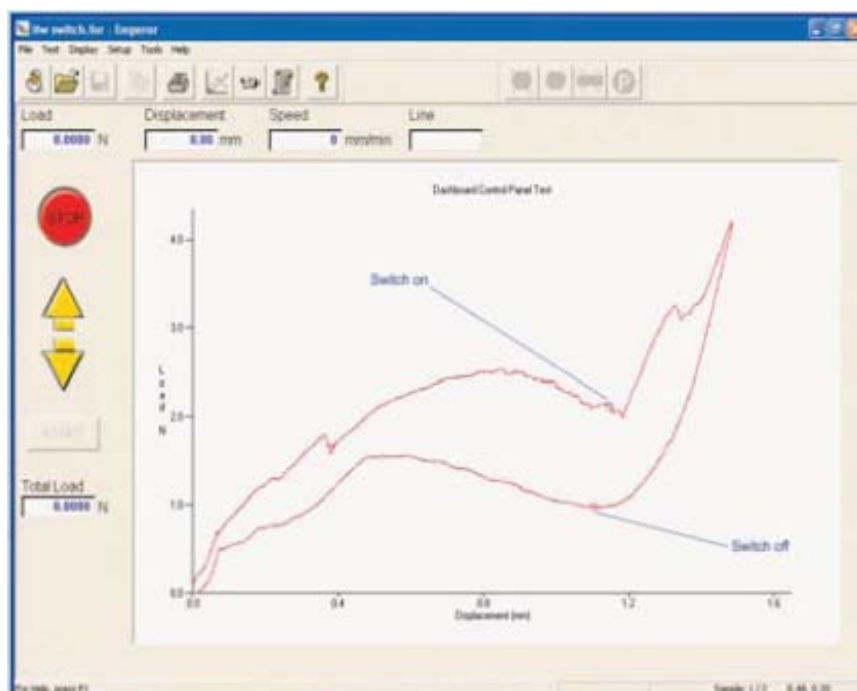
V tabeli 1 lahko razberemo podatke vseh treh navedenih sistemov. Ena od njihovih najslabših lastnosti je ta, da porabljajo za delovanje veliko napajalne energije. V našem sistemu bomo to pomankljivost poskusili izboljšati. Glede na sliko 2.4, ki prikazuje izmerjeno karakteristiko tipke z Mecmesin MultiTest 2.5-d sistemom in njihovo programski opremo, lahko predvidevamo, da ti sistemi omogočajo zgolj merjenje mehanskih lastnosti tipk in drugih naprav, kar za podjetje Elrad predstavlja največjo pomanjkljivost, saj vgrajujejo na elektronske sklope tudi drugačne vrste tipk. To so na primer tipke z magnetno-mehanskim (reed rele) ali elektro-magnetnim (Hallova sonda) preklpom. V teh primer ni dovolj zgolj izmeriti mehansko karakteristiko tipke ampak tudi ugotoviti, v katerem delu te karakteristike se dejanski preklop zgodi. Takšne vrste preklopa se iz mehanske karakteristike tipk ne da ugotoviti.

Tabela 1: Primerjava različnih merilnih sistemov

	Mecmesin MultiTest 2.5-d[18]	Tinius Olsen H1k [20]	Mark-10 EMS301 [19]
Maksimalna sila [N]	2500	1000	1500
Maksimalna razdalja [mm]	350	445	317
Resolucija sile [N]	0,01	Ni podatka	0,1
Resolucija razdalje [mm]	0,01	0,0001	0,01
Hitrost [mm/min]	1-750	0.001 - 1000	0.5-1100
Digitalni izpis	LCD	LCD	LCD
Poraba moči [W]	160	2000	Ni podatka
Komunikacija s PC	RS-232	RS-232	RS-232
Motor	Ni podatka	Ni podatka	Koračni



Slika 2.3: Mecmesin MultiTest 1-d, Tinius Olsen H1k, Mark-10 ESM301



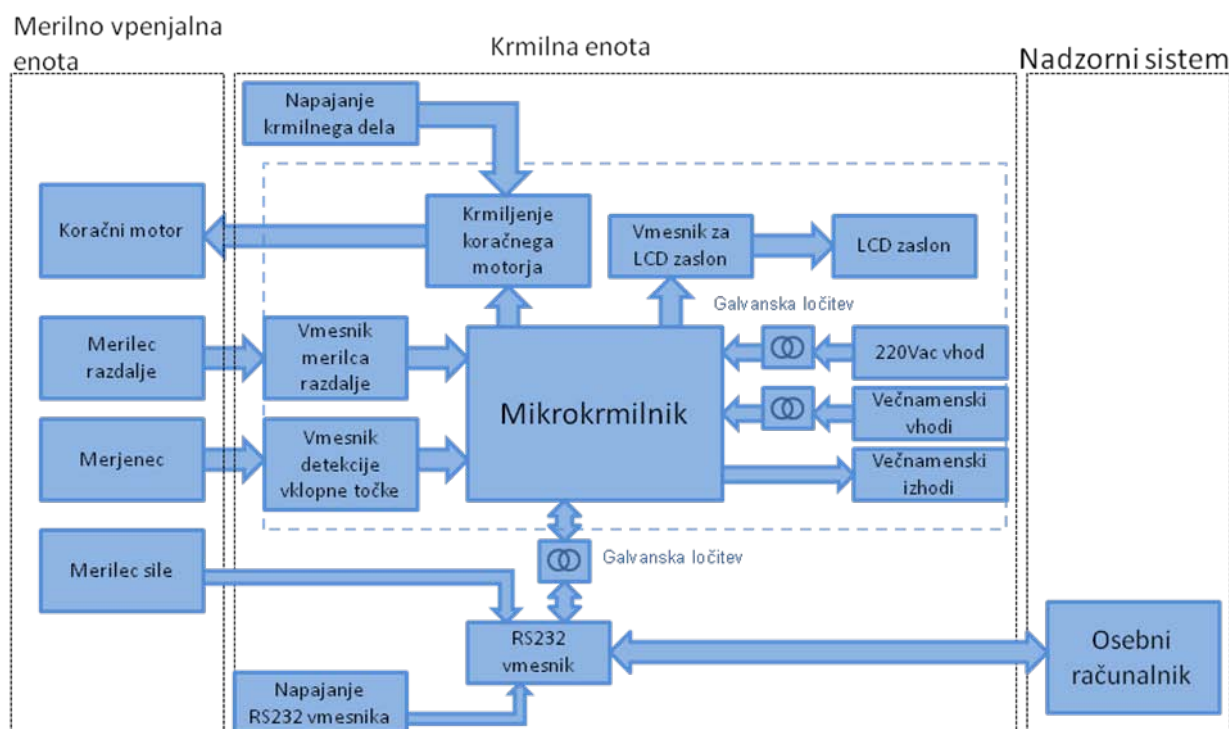
Slika 2.4: MECMESIN Emperor Lite Software z izmerjeno karakteristiko tipke

Tako želimo razviti sistem, ki bo imel podobne značilnosti kot opisani trije sistemi in lastnosti ter zgradbo, kot jo opisujemo v prvem odstavku tega poglavja oziroma, da bo zadostil potrebam podjetja Elrad.

3. OPIS MERILNEGA SISTEMA

Merilni sistem, ki smo ga razvili in izdelali, uporabljamo za merjenje karakteristike pomika tipke v procesu vklopa in izklopa merjenca, ki je v našem primeru vrtalni stroj. Sestavljajo ga trije deli (slika 3.1):

- merilno vpenjalna enota,
- krmilna enota in
- nadzorni sistem.



Slika 3.1: Blokovna shema merilnega sistema

Glavni del merilnega sistema je merilno vpenjalna enota, v katero vstavimo merjenec. Krmilna enota je namenjena za krmiljenje merilno vpenjalne enote in zajemanje podatkov o merjenih značilnostih vklopnega stikala merjenca. Krmilna enota pošilja izmerjene podatke v nadzorni sistem, to je osebni računalnik, kjer se obdelajo in shranijo. Na osebni računalnik teče tudi uporabniški vmesnik za upravljanje sistema. Posamezne enote sistema podrobneje opisujemo v nadaljnjih poglavjih.

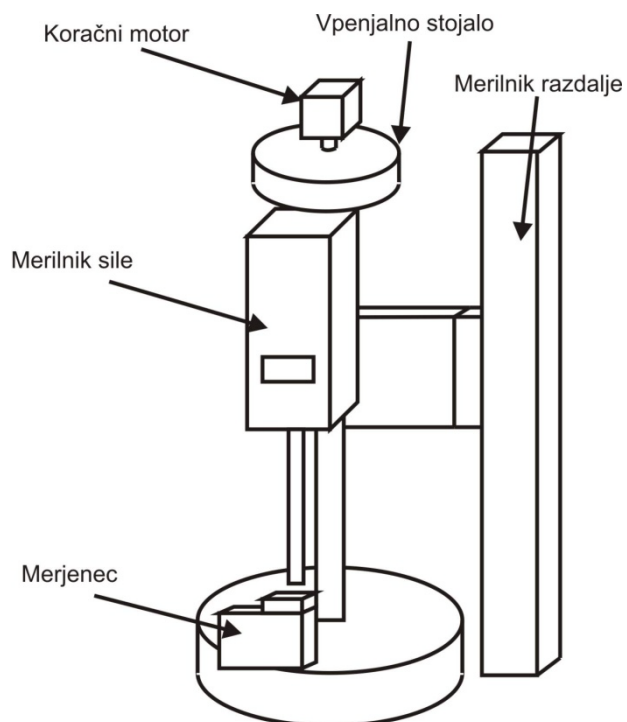
3.1 Merilno vpenjalna enota

Merilno vpenjalno enoto smo sestavili po naših načrtih v podjetju Elrad International d.o.o., ki je tudi naročnik celega sistema. Na

Slika 3.2 je shematski prikaz te enote.

Sestavljajo jo naslednji gradniki:

- Koračni motor,
- merilnik sile,
- merilnik razdalje in
- vpenjalno stojalo.
- Merjenec vpremo v stojalo merilne enote.



Slika 3.2: Model merilno vpenjalne enote

3.1.1. Koračni motor

Koračni motor se ne vrti zvezno kot drugi motorji, ampak po korakih, ki so določeni z njegovo konstrukcijsko izvedbo in načinom krmiljenja. Motor lahko torej zavrtimo najmanj za en korak naenkrat in to v katerikoli smeri. Ker lahko pri takšnem motorju bolje nadziramo vrtenje njegove osi, smo se odločili, da bomo za našo aplikacijo uporabili to vrsto motorja in ne kakšnega izmed konvencionalnih, kot je na primer enosmerni motor ali AC motor. Pri zvezno vrtečih se strojih težje nadzorujemo položaj osi pri njihovi zaustavitvi. Ker pa je v naši aplikaciji pomembno še, da lahko simuliramo pritisk prsta na gumb merjenca čimbolj natančno in nadzorovano, zvezno vrteči motorji niso primerni za našo aplikacijo. Obstaja več vrst koračnih motorjev [6]. Omenimo samo dve glavni skupini:

- bipolarni in
- unipolarni koračni motorji.

Za našo raziskavo smo uporabili bipolarni koračni motor QSH4218 [14] podjetja TRINAMIC MOTION CONTROL. Za ta tip motorja smo se odločili predvsem zaradi primerne velikosti za namestitev na vpenjalno stojalo, dovolj velikega momenta in natančnosti pomika ($1,8^\circ$ na korak oziroma 200 korakov za celi obrat 360°).

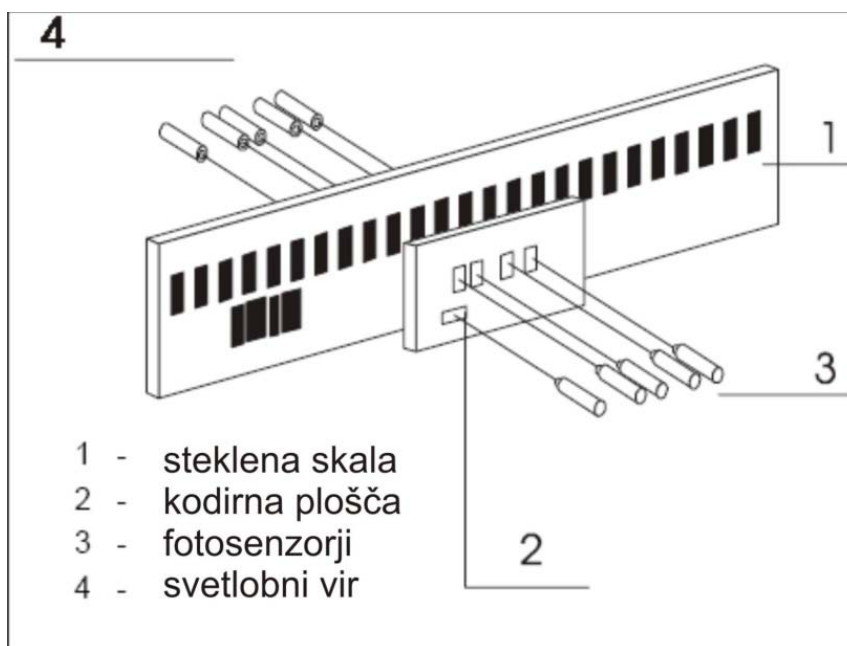


Slika 3.3: Koračni motor QSH4218

3.1.2. Merilnik razdalje

Inkrementalni optični dajalnik je naprava, ki smo jo uporabili kot merilnik razdalje oziroma kot merilnik pomika ročice, ki sproži vklop merjenca. Inkrementalni optični dajalnik pretvori spremembo položaja oziroma gibanje v niz digitalnih signalov [15]. Poznamo rotacijske in linearne optične dajalnike. Uporabili smo linearni inkrementalni optični dajalnik – merilno letev ML300 podjetja MIKROPLUS. Za uporabo takega merilnika razdalje smo se odločili zaradi dovolj velike točnosti in relativno enostavne priključitve na mikrokrmilnik.

Optični dajalnik je sestavljen iz steklene skale, kodirne plošče, fotosenzorjev ter svetlobnega vira (slika 3.4).



Slika 3.4: Prikaz sestave merilne letve

Inkrementalni optični dajalnik deluje po principu optoelektričnega otipavanja. Položaj pridobimo s štetjem rastrskih razdelkov (merilnih korakov)[15].

Inkrementalni optični dajalniki položaja so brezkontaktni in temeljijo na polprevodniški tehnologiji LED diod in fototranzistorjev[15]. LED diode, ki so v dajalniku osvetljujejo fototranzistorje skozi kodirno ploščo. Na kodirno ploščo je nanescena črna koda, ki

prekinja žarke med LED diodo in fototranzistorjem. S tem dobimo signal, ki ga spremenimo v informacijo o položaju. Merilna letev potrebuje napajalno napetost 5V za delovanje in tudi izhodni signali imajo amplitudo TTL vezji, zato za povezavo na naš sistem ni bilo potrebnih nobenih vmesnikov.



Slika 3.5: Merilna letev

3.1.3. Merilnik sile

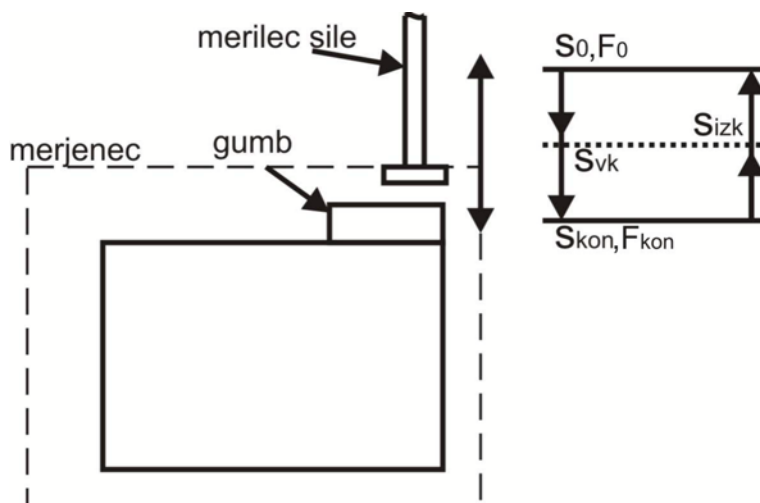
Uporabili smo merilnik sile FG-5000A proizvajalca Lutron[16]. Zanj smo se odločili predvsem zato, ker je že bil na voljo v podjetju Elrad in smo ga lahko enostavno pritrdili na stojalo skupaj z merilnikom razdalje ter koračnim motorjem. Bolj pomemben razlog zakaj je ta merilnik sile primeren za naše potrebe je ta, da ga je možno povezati na katerikoli sistem, ki omogoča zaporedno komunikacijo med moduli. Tako smo ga lahko preko RS232 vmesnika povezali na mikrokrmilnik.



Slika 3.6: Merilnik sile FG-5000A

3.2 Delovanje merilno vpenjalne enote

Merilno vpenjalna enota ima v osnovi dve nalogi. Prva je ta, da z njo simuliramo pritisk prsta na merjenec in s tem vklopa oziroma izklopa merjenca, kar izvajamo z merilcem sile in koračnim motorjem. Druga naloga pa je merjenje »poti«, ki jo napravi gumb merjenca ob delovanju. To nalogo opravlja merilnik razdalje. S pomočjo te enote bomo torej kontrolirano izvajali celoten proces vklopa merjenca in statistično pridobili 6 vrst informacij: silo za aktiviranje stikala preko tipke (F_0), točko dotika gumba, položaj vklopa stikala s pomočjo tipke (S_0), končni položaj tipke (S_{kon}), silo v trenutku, ko je gumb do konca pritisnjen ter položaj izklopa stikala (F_{kon}) – slika 3.7.



Slika 3.7: Prikaz načina simulacije pritiska prsta

S koračnim motorjem pomikamo stojalo iz začetne točke skupno z merilnikom sile in roko merilnika razdalje, s hitrostjo primerno za simulacijo tipke proti merjencu. Ko se merilnik sile dotakne gumba merjenca, dosežemo točko dotika in izmerimo neko začetno silo (F_0). Ta in vse druge podatke pošljemo po zaporedni povezavi v mikrokrmilnik. Mikrokrmilnik nastavi merilnik razdalje na 0 mm. S tem dobimo referenčno točko dotika S_0 merjenca razdalje in začetno silo F_0 , kot je prikazano na sliki 3.7. Glede na to točko potem izmerimo vklopno točko S_{vkl} , končno točko S_{kon} in F_{kon} ter izklopno točko S_{izk} merjenca. Ko se merjenec vklopi, zaznamo ta položaj preko vmesnika za detekcijo vklopa oziroma izklopa (opisali ga bomo kasneje v poglavju 3.3.5). S tem smo dobili vklopno točko merjenca. S pritiskanjem na gumb nadaljujemo dokler ni gumb popolnoma pritisnjen – to je končna točka. Ta položaj zaznamo s tem, da se razdalja, ki jo merimo z merilno letvijo, ne

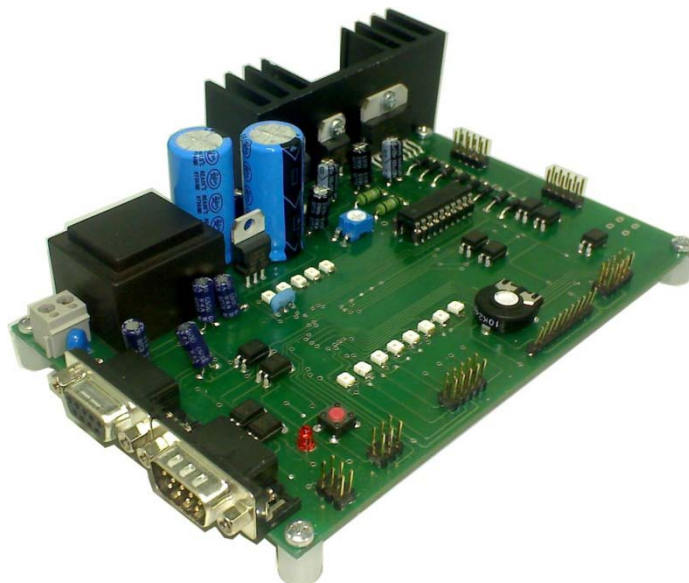
spreminja več. Koračni motor ustavimo in mu spremenimo smer vrtenja, da bomo začeli pritisk gumba na podlago zmanjševati. Gumb počasi popušča in tako poiščemo še izklopno točko merjenca. Koračni motor se vrti še tako dolgo, dokler merilnik sile in merilnik razdalje ne kažeta 0 N oziroma 0 mm. S tem smo opravili meritev in zbrali naslednje podatke:

- S_0, F_0
- S_{vkl}, F_{vkl}
- S_{kon}, F_{kon} ter
- S_{izk}, F_{ikz} .



Slika 3.8: Merilno vpenjalna naprava z merjencem

3.3 Krmilnik merilnega sistema



Slika 3.9: Krmilni modul

Krmilnik merilnega sistema je mikroročunalniški sistem, zgrajen na osnovi Atmelovega mikrokrmilnika AVR Atmega128. Krmilnik izvaja naslednja opravila:

- krmiljenje merilnega sistema,
- zajemanje merilnih rezultatov (S_0 , S_{vkl} , S_{izk} , S_{kon} , F_0 , F_{kon}),
- posredovanje podatkov v nadzorni sistem in
- izpis podatkov na LCD zaslon.

Sestavljajo ga naslednji gradniki:

- napajalno vezje,
- mikrokrmilnik AVR Atmega128,
- vmesnik za krmiljenje koračnega motorja,
- RS-232 vmesnik,
- vmesnik za LCD zaslon,
- vmesnik za detekcijo vklopa oziroma izklopa merjenca ter
- vmesnik merilca razdalje.

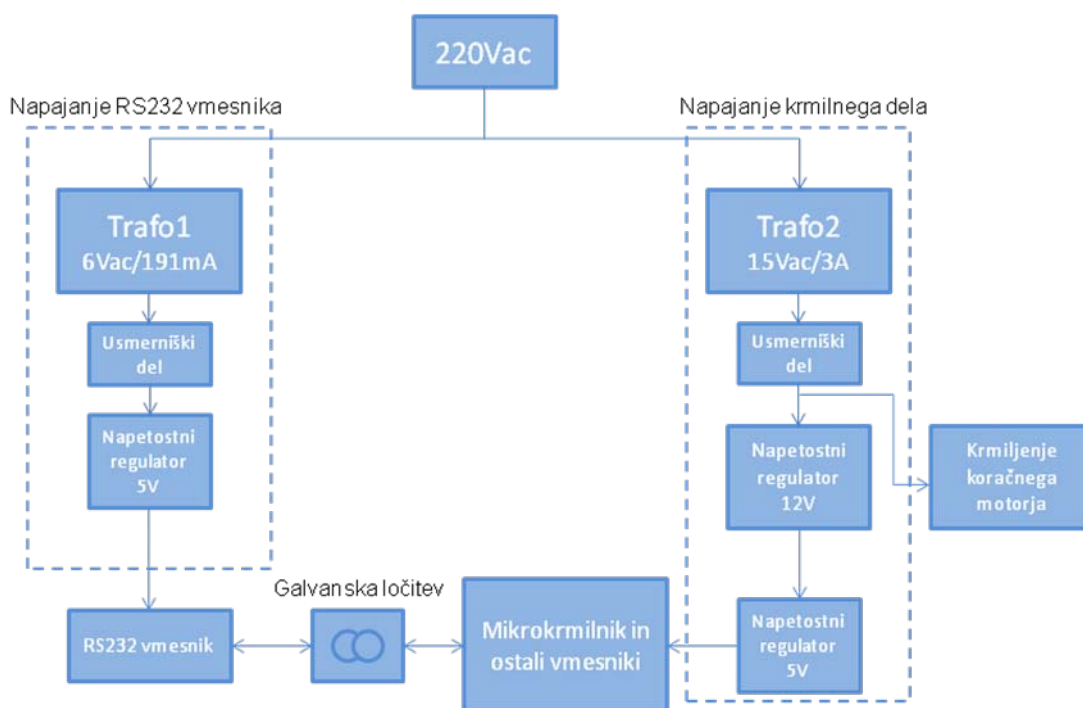
Poleg naštetih vmesnikov in vhodno/izhodnih signalov, ki so pomembni za aplikacijo merilnika sile, smo predvideli še priključke za dodatne vhode in izhode, ki jih AVR mikrokrmilnik podpira. S tem razširimo možnosti uporabe mikroračunalniškega sklopa še za druge aplikacije. Uporabimo ga lahko tudi kot razvojno okolje za Atmega128. Dodali smo še galvansko ločeni vhod, kamor lahko povežemo elektronske sklope, ki delujejo na elektroniki 230Vac. Vsi vhodi, razen vhodov za merilno letev, ki se napaja iz krmilnega sistema, so galvansko ločeni.

Krmilnik je na osebni računalnik povezan preko asinhronne zaporedne povezave (USART - *Universal Synchronous Asynchronous Receiver Transmitter*), kjer poteka komunikacija v obeh smereh. Z merilcem sile pa je povezan preko drugega USART-a, vendar le v smeri proti krmilnemu sistemu.

Procesni del krmilnika merilnega sistema je Atmel-ov mikrokrmilnik Atmega128. Mikrokrmilnik skrbi, preko zaporednega vmesnika, za komunikacijo med osebnim računalnikom in merilnikom sile. Zajema podatke z merilca razdalje in merilca sile ter jih pretvarja v obliko, primerno za obdelavo. Mikrokrmilnik izvaja s pomočjo ustreznih vmesnikov krmiljenje koračnega motorja ter izpis podatkov na LCD zaslonu. V nadaljevanju opisujemo posamezne gradnike sistema.

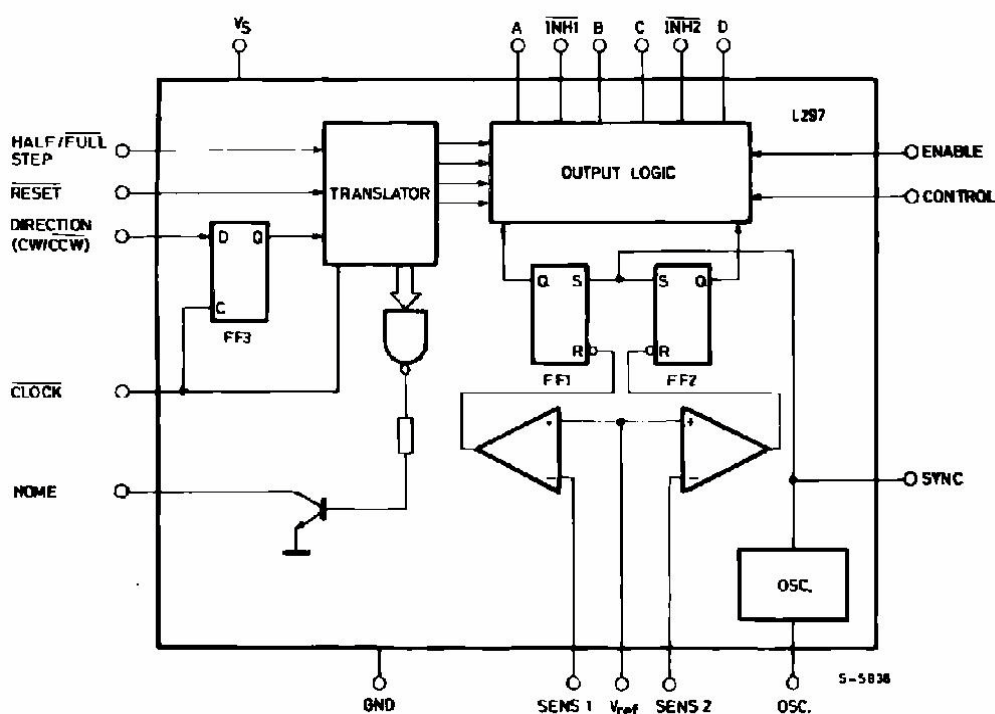
3.3.1. Napajalno vezje

Zaradi zahtev po galvansko ločenem merilnem sistemu smo morali uporabiti dve ločeni napajalni vezji: eno za krmilni del ter eno za napajalno vezje RS232 vmesnika. V napajalniku za krmilni del smo predvideli še dodatno napajanje za koračni motor (slika 3.10). Za ločena napajalnika smo uporabili dva transformatorja. Ker ima vezje MAX232 proizvajalca MAXIM, ki pretvori TTL signale v signale za RS232 komunikacijo, majgno porabo, smo se odločili, da za ta del uporabimo manjši 1,2 watni transformator, ki smo ga vgradili na tiskano ploščico in ima izhodno napteost 6Vac ter izhodni tok 191mA.



Slika 3.10: Blokovna shema napajanja vezja

Za integrirano vezje MAX232 potrebujemo stabilizirano napetost 5V. Izmenično napetost na izhodu transformatorja najprej usmerimo z Graetz-ovim mostičem S250 ter nato z napetostnim regulatorjem 7805 stabiliziramo napetost na 5V. Za napajanje drugega dela krmilnega sistema smo uporabili zunanji transformator z izhodno napetostjo 15Vac/3A. Ta del vezja ja nekoliko drugačen kot za napajanje RS232 vmesnika. Razlikuje se v tem, da smo Graetzov mostič zgradili iz štirih diskretnih diod 1N5401. Te diode zdržijo večji tok kot mostič S250. Za usmerniškim mostičem smo zaporedno vezali regulatorja 7812 in 7805. S tem smo zmanjšali izgubno moč na regulatorju 7805. 12V regulator smo nameravali uporabiti še za napajanje krmilnika koračnega motorja, vendar se je kasneje



Slika 3.12: Blokovna shema integriranega vezja L297

Integrirano vezje L297 (slika 3.12) vsebuje več vhodov za krmiljenje koračnih motorjev. Mi smo uporabili naslednje vhode:

- urin signal (ang. Clock),
- smer vrtenja (ang. Direction - CW/CCW) in
- polovični ali celi korak (ang. Half/Full step).

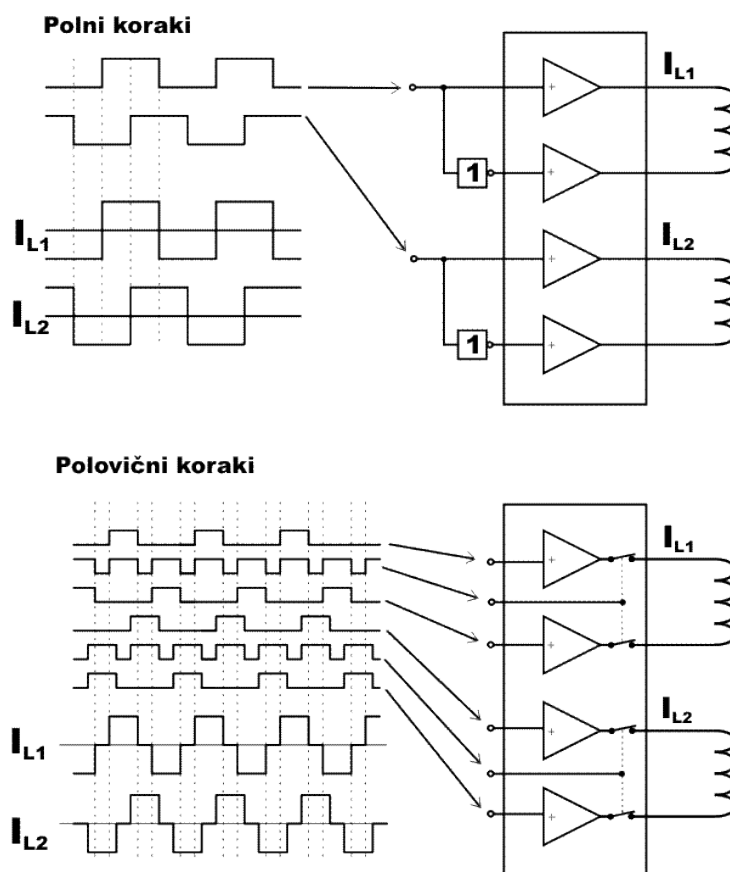
Impulz signala ure krmili vsak korak, ki ga napravi koračni motor pri vrtenju. Aktivni nizek nivo impulza pomakne koračni motor za en korak, ki se izvrši ob rastočem delu enega impulza signala.

Signal za izbiro smeri vrtenja določa, v katero smer se bo koračni motor vrtel.

Vhodni signal za polovični oziroma celi korak krmili način delovanja integriranega vezja L297. Ko je signal na visokem nivoju, izvaja vezje vrtenje motorja za polovični korak. Ko pa je signal na nizkem nivoju, izvaja vezje vrtenje motorja v celem koraku.

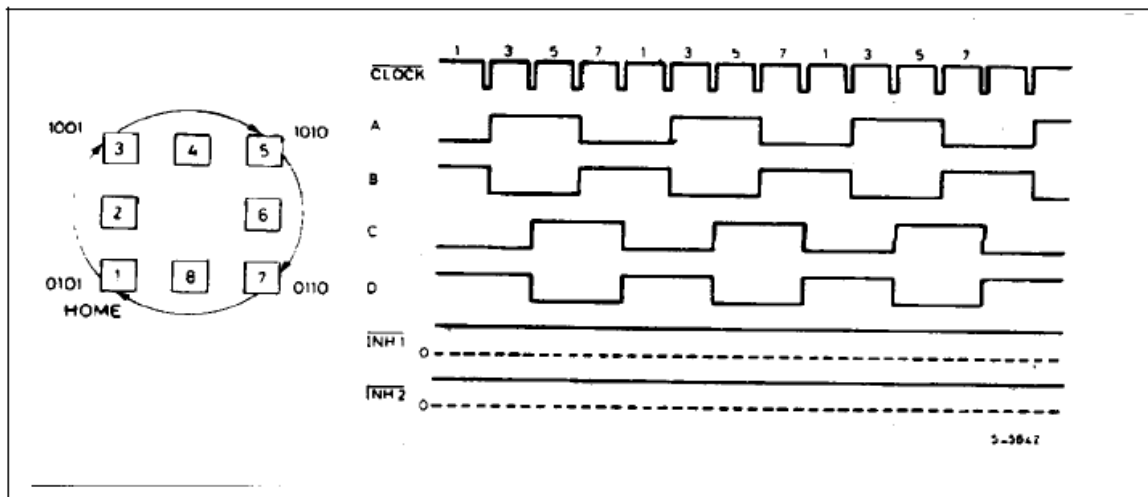
Vrtenje s polovičnim korakom je način delovanja, ko se motor ob spremembi krmilnih signalov in ob vsakem urinem impulzu premakne le za pol koraka (slika 3.13). Poglavitna prednost takega krmiljenja je, da se motorju poveča resolucija za faktor 2. Tako dosežemo,

da dobi motor z resolucijo $1,8^\circ$, kar pomeni 200 korakov na obrat, kolikor bi jih bilo ob polnem koraku, sedaj resolucijo $0,9^\circ$, kar pa pomeni 400 korakov na obrat. S tem povečamo natančnost merilnega sistema. To je tudi pglavitni razlog, zakaj smo se odločili za takšen način krmiljenja koračnega motorja.

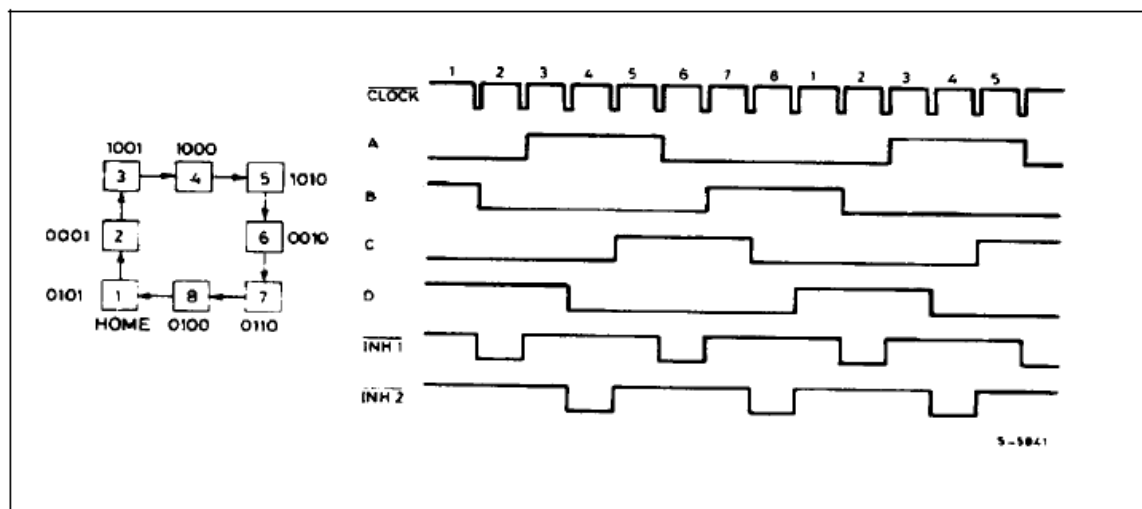


Slika 3.13: Krmilni signali za polnokoračno in polkoračno krmiljenje koračnega motorja

Slika 3.14 in slika 3.15 prikazujeta, kakšni so izhodni signali na integriranem vezju L297 v odvisnosti od vhodnega krmilnega urinega signala in načina, v katerem želimo, da koračni motor deluje.



Slika 3.14: Izhodni signali v odvisnosti od krmilnega urinega signala pri polnokoračnem načinu



Slika 3.15: Izhodni signali v odvisnosti od krmilnega urinega signala pri polkoračnem načinu

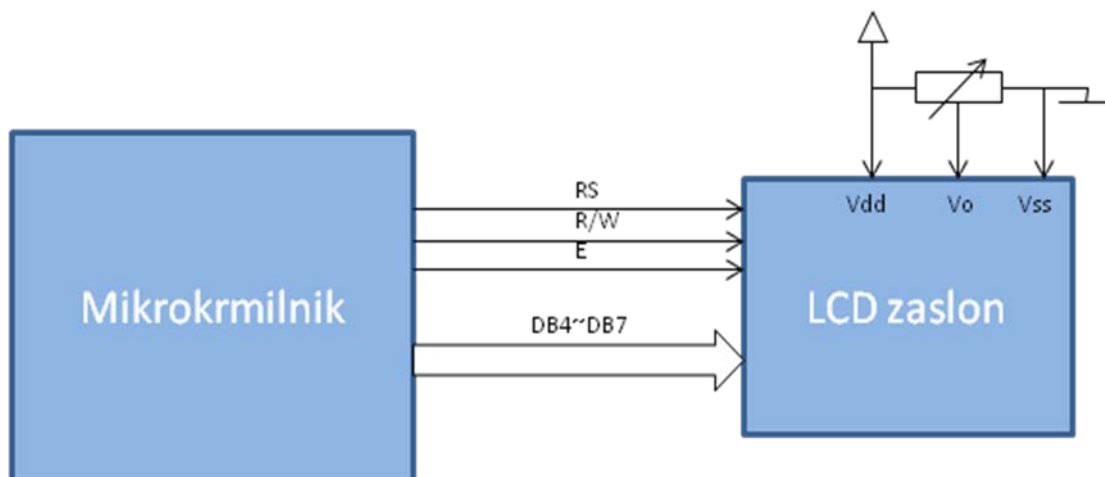
3.3.3. RS232 vmesnik

RS232 standard določa strojne značilnosti zaporednega komunikacijskega vmesnika. Signali tega standarda ne uporabljajo konvencionalnih 0 in 5V nivojev, kot pri TTL in CMOS vezjih. Pri RS232 standardu predstavljajo ti nivoji +5V do +15V za logično »0« ter -5V do -15V za logično »1«. Zaradi tega potrebujemo vmesnik, ki pretvori TTL nivoje v RS232 nivoje in obratno. Za pretvorbo TTL nivojev v RS232 nivoje smo uporabili integrirano vezje MAX232 [12] podjetja MAXIM. Merilni sistem smo preko galvansko ločene zaporedne komunikacije povezali z osebnim računalnikom ter merilnikom sile.

Čeprav omogoča zaporedna komunikacija dokaj velike hitrosti prenosa podatkov, do 115200 Baud-ov (znakov/sekundo) in več, pa smo v našem sistemu uporabili hitrost 9600 Baud-ov. Poglavitni razlog za takšno izbiro hitrosti je bila galvanska ločitev z optičnimi spojniki. Uporabili smo optične spojnike (PC123), ki ne omogočajo hitrih sprememb signalov in s tem hitrejša komunikacije. Za našo aplikacijo je hitrost 9600 baudov dovolj velika za obseg podatkov, ki jih prenašamo. Če bi želeli doseči večje prenosne hitrosti, bi morali uporabiti druge vrste optičnih spojnikov (na primer tip PS9113).

3.3.4. Vmesnik za LCD zaslon

Za lokalni prikaz informacij na strani krmilnika merilnega sistema smo uporabili LCD zaslon podjetja Winstar Display Co [7] s standardnim formatom 2X16 to sta dve vrstici s po 16 znaki (slika 3.17). Za našo aplikacijo zadostuje znakovni zaslon v takšnem obsegu znakov. Priključni konektor ima 16 priključkov. Za krmiljenje zaslona potrebujemo le 10 priključkov (slika 3.16). Vdd in Vss sta napajalna priključka, Vo pa je priključek za nastavitve kontrasta LCD zaslona, ki ga nastavljamo s potenciometrom. Signal RS krmili vpisovanje ukazov oziroma podatkov v krmilnik LCD zaslona. Če je signal RS na visokem nivoju, vpisujemo v signal ukaze, če pa je na nizkem nivoju, vpisujemo podatke oziroma znake, ki se prikažejo na zaslonu. Z linijo R/W izbiramo funkcijo vpisovanja oziroma funkcijo branja registrov krmilnika LCD zaslona. Če postavimo R/W signal na nizek nivo, vpisujemo podatke v krmilnik, sicer pa jih beremo iz njega. Signal E omogoča komunikacijo glavnega mikrokrmilnika z zaslonom. Signali, s pomočjo katerih vpisujemo podatke, so označeni z DB0~DB7. Uporabili smo štiri-bitni način tako, da vpisujemo znake in ukaze preko signalov DB4~DB7.



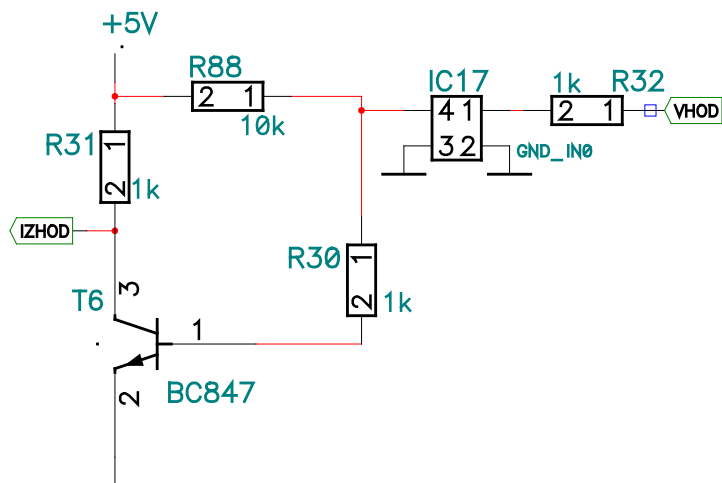
Slika 3.16: Blokovna shema povezave LCD zaslona z mikrokontrolerom



Slika 3.17: LCD zaslon WH1602A, Winstar Display Co.

3.3.5. Vmesnik za zaznavanje vklopa in izklopa merjenca

Vklop in izklop merjenca zaznavamo preko večnamenskih galvansko ločenih vhodov. Galvansko ločene vhode uporabljamo zato, da lahko nanje priključimo tudi merjenca, ki potrebujejo višje napajanje, kot ga lahko daje naš sistem. Enak način galvanske ločitve uporabljamo tudi pri zaporednem vmesniku. Vezje IC17 je optični spojnik tipa PC123, ki zagotavlja galvansko ločitev. Tranzistor smo dodali za prilagoditev nivoja signala.

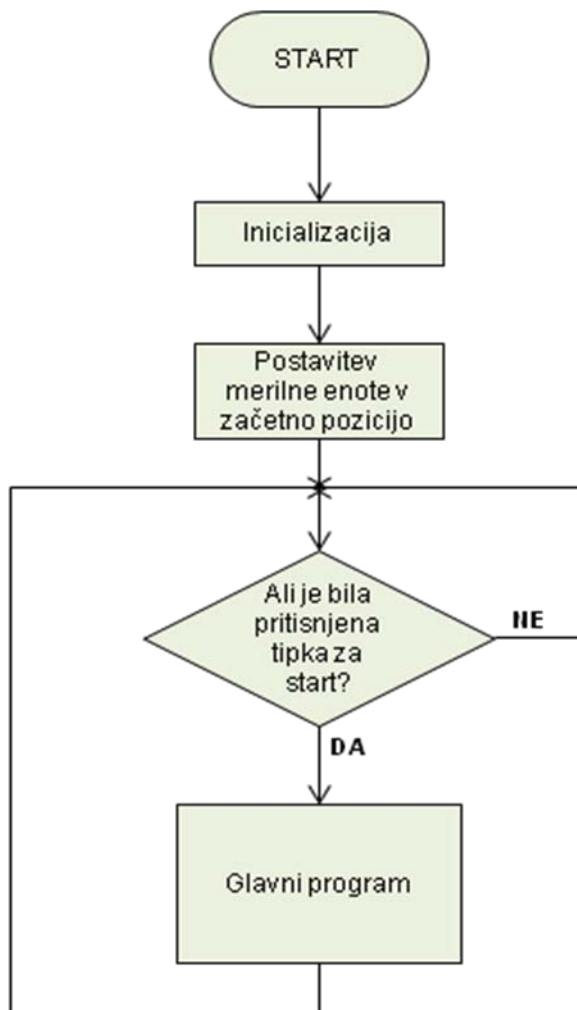


Slika 3.18: Shema dela vezja za galvanško ločitev vhodov na mikrokrmilnik

3.4 Programska oprema mikrokrmilnika

Program mikrokrmilnika smo razvili v programskem jeziku C. Program smo vnašali s pomočjo programskega orodja AVR Studio 4 [4], kot prevajalnik pa smo uporabili orodje WinAVR [3].

Slika 3.19 prikazuje osnovni potek izvajalnega programa. Ob vsaki priključitvi merilnega sistema na napajanje se izvede inicializacija mikrokrmilnika in perifernih enot. Ko se inicializacija konča, mora koračni motor postaviti merilno vpenjalno enoto v začetno pozicijo. Program nato čaka na ukaz za začetek izvajanja glavnega programa oziroma izvajanje postopka meritve. Ukaz za začetek glavnega programa lahko dobi iz nadzornega sistema na osebem računalniku ali pa s pritiskom na tipko za start, ki je ustrezno priključena na enega od vhodov mikrokrmilnika.



Slika 3.19: Diagram poteka programa

Glavni program lahko razdelimo na več modulov:

- glavna zanka programa,
- prekinitveni vektorji in prekinitveni servisni programi,
- modul za krmiljenje koračnega motorja,
- LCD modul,
- modul za servisiranje merilca sile in nadzornega sistema na osebnem računalniku ter
- modul za merilno letev.

Jedro programa predstavlja glavna izvajalna zanka, ki sekvenčno testira ali je bila pritisnjena tipka za začetek in po pritisku te tipke zaporedno izvaja programske module: modul za krmiljenje koračnega motorja, LCD modul, modul za servisiranje merilca sile in

osebnega računalnika kot nadzornega sistema in modul za merilno letev. Prekinitveni vektorji asinhrono in v realnem času prožijo prekinitvene servisne programe za servisiranje posameznih naprav: merilca sile, merilne letve, koračnega motorja in nadzornega sistema. Pri tem koristimo tudi notranje periferne naprave mikrokrmilnika: serijski komunikacijski krmilnik, zunanje prekinitvene vmesnike in časovnike.

3.4.1. Glavna zanka programa

V programskem modulu za krmiljenje koračnega motorja tvorimo signale za krmiljenje koračnega motorja oziroma pripadajočih krmilnih vezij. Motor preko stojala pomika glavo merilca sile iz začetne točke do tipke merjenca. Položaj, ko je merilnik sile dosegel gumb merjenca, zaznamo tako, da se je vrednost sile, ki jo merilnik meri in jo pošilja preko zaporedne povezave na mikrokrmilnik, spremenila oziroma ni več enaka 0. Servisiramo jo v programskem modulu za servisiranje merilca sile in osebni računalnik. V tej točki, točki dotika, se merilna letev, katerega merilna glava se je ob tem času tudi pomikala, postavi na vrednost 0. Ob tem času se krmiljenje motorja ni spremenilo. Ko merilnik sile pomakne tipko do točke vklopa merjenca, se spremeni stanje vhoda na katerega je vezan vmesnik za zaznavanje vklopa oziroma izklopa merjenca. Tako se v tej točki shrani pomik, ki ga je tipka napravila do tega trenutka. Končno točko, to je točka kjer je tipka pritisnjena do konca, smo določili eksperimentalno, in sicer tako, da smo zmerili silo več merjencev in tako določili končno silo 1500 N, preko katere gumba ne moremo oziroma ne smemo več pritiskati. Spremenimo mu smer vrtenja. Tako se ponovi postopek iskanje nasprotne točke 'vklopa', ki je sedaj točka izklopa. Točki izklopa sledi še točka, ko tipka ni več pritisnjena, to je točka dotika. Tako pomaknemo glavo merilca sile in merilno letev le še v začetno pozicijo.

3.4.2. Prekinitveni servisni programi ter prekinitveni vektorji

Pod pojmom "prekinitvev" (angl. Interrupt) razumemo zunanjo zahtevo za prekinitvev izvajanja tekočega zaporedja ukazov in izvedbo zahtevane prekinitvene operacije (na primer prevzem podatka iz zaporednega vmesnika) v realnem času. Zahteva za prekinitvev se lahko pojavi kadarkoli kot posledica nekega dogodka v sistemu ali zunaj njega (na

primer pritiska tipke). Prekinitve se torej pojavljajo *asinhrono* glede na izvajanje programa (tipko lahko pritisnemo kadarkoli). Zato so prekinitve (sistem prekinitvev) na splošno bistvenega pomena za delovanje sistemov v realnem času. Sistem mora na takšne zahteve takoj odgovoriti z njej primernim dejanjem.

Vektor je pomnilniška beseda, ki vsebuje naslov prvega ukaza pripadajočega prekinitvenega programa. Prekinitvenih vektorjev je več, prikazujemo jih v tabeli 2.

Tabela 2: Vsi možni prekinitveni vektorji mikrokrmilnika ATMEGA128 s prikazom uporabe

Vektor.	Naslov programa	Izvor	Vrsta prekinitve	Uporaba/Vstopna točka v prekinitveni servisni program
1	\$0000(1)	RESET	External Pin, Power-on Reset, Brown-out Reset Watchdog Reset, and JTAG AVR Reset	DA
2	\$0002	INT0	External Interrupt Request 0	DA-končna stikala
3	\$0004	INT1	External Interrupt Request 1	DA-končna stikala
4	\$0006	INT2	External Interrupt Request 2	NE
5	\$0008	INT3	External Interrupt Request 3	NE
6	\$000A	INT4	External Interrupt Request 4	NE
7	\$000C	INT5	External Interrupt Request 5	ISR(INT5_vect)
8	\$000E	INT6	External Interrupt Request 6	NE
9	\$0010	INT7	External Interrupt Request 7	ISR(INT7_vect)
10	\$0012	TIMER2 COMP	Timer/Counter2 Compare Match	NE
11	\$0014	TIMER2 OVF	Timer/Counter2 Overflow	NE
12	\$0016	TIMER1 CAPT	Timer/Counter1 Capture Event	NE
13	\$0018	TIMER1 COMPA	Timer/Counter1 Compare Match A	ISR(TIMER0_COMP_vect)
14	\$001A	TIMER1 COMPB	Timer/Counter1 Compare Match B	NE
15	\$001C	TIMER1 OVF	Timer/Counter1 Overflow	NE
16	\$001E	TIMER0 COMP	Timer/Counter0 Compare Match	ISR(TIMER1_COMPA_vect)
17	\$0020	TIMER0 OVF	Timer/Counter0 Overflow	NE
18	\$0022	SPI, STC	SPI Serial Transfer Complete	NE
19	\$0024	USART0, RX	USART0, Rx Complete	ISR(USART0_RX_vect)
20	\$0026	USART0, UDRE	USART0 Data Register Empty	ISR(USART0_RX_vect)

21	\$0028	USART0, TX	USART0, Tx Complete	NE
22	\$002A	ADC	ADC Conversion Complete	NE
23	\$002C	EE READY	EEPROM Ready	NE
24	\$002E	ANALOG COMP	Analog Comparator	NE
25	\$0030(3)	TIMER1 COMPC	Timer/Counter1 Compare Match C	NE
26	\$0032(3)	TIMER3 CAPT	Timer/Counter3 Capture Event	NE
27	\$0034(3)	TIMER3 COMPA	Timer/Counter3 Compare Match A	NE
28	\$0036(3)	TIMER3 COMPB	Timer/Counter3 Compare Match B	NE
29	\$0038(3)	TIMER3 COMPC	Timer/Counter3 Compare Match C	NE
30	\$003A(3)	TIMER3 OVF	Timer/Counter3 Overflow	NE
31	\$003C(3)	USART1, RX	USART1, Rx Complete	ISR(USART1_RX_vect)
32	\$003E(3)	USART1, UDRE	USART1 Data Register Empty	ISR(USART1_RX_vect)
33	\$0040(3)	USART1, TX	USART1, Tx Complete	NE
34	\$0042(3)	TWI	Two-wire Serial Interface	NE
35	\$0044(3)	SPM READY	Store Program Memory Ready	NE

Iz tabele 2 je razvidno kateri so vsi možni prekinitveni vektorji mikrokrmilnika ATMEGA128[2]. Označili pa smo vektorje in vstopne točke v prekinitvene servisne programe, ki smo jih uporabili v naši aplikaciji. Podrobnejše opisujemo te programe, ki servisirajo zunanje prekinitve, v naslednjih nekaj poglavjih.

3.4.3. Modul za krmiljenje koračnega motorja

V poglavju 3.3.2 opisujemo časovne poteke električnih signalov za pravilno krmiljenje motorja. To so trije vhodni signali integriranega vezja L297, ki jih tvorimo s pomočjo mikrokrmilnika. Sedaj bomo opisali, kako te tri signale tvorimo programsko. Najprej jih inicializiramo. Urin signal generiramo s pomočjo programske kode časovnika – TIMER1 (slika 3.20). Druge tri signale pa prav tako postavljamo s pomočjo časovnika, glede na to v katerem delu meritve se nahajamo oziroma, kako želimo, da koračni motor deluje. Časovnike smo uporabili zato, ker delujejo neodvisno od glavnega programa in ga s tem ne

upočasnjujejo. Za krmiljenje koračnega motorja uporabljamo tri ISR – je (Interrupt Service Routine). ISR je prekinitveni servisni podprogram mikrokrmilnika, ki se izvede, ko pride do prekinitve. V našem primeru to prekinitve sproži časovnik. V prvi rutini (ISR(TIMER0_COMP_vect) preverjamo, če želimo spremeniti smer vrtenja motorja ali pa način delovanja. Tako temu primerno nastavimo tudi izhodni signal mikrokrmilnika (KM_H_F), s katerim spremenimo način delovanja motorja. Z ISR(TIMER1_COMPA_vect) ter ISR(TIMER1_COMPB_vect) pa generiramo urin signal za krmiljenje koračnega motorja. Časovnik inicializiramo v inicializacijskem programu in sicer tako, da delujejo v prekinitvenem načinu na različnih frekvencah, ki so odvisne od željene hitrosti motorja.

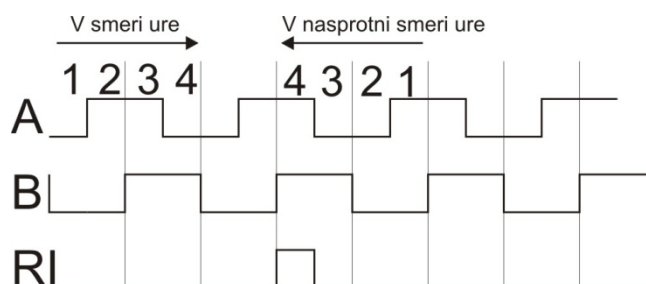
```
ISR(TIMER0_COMP_vect)
{
    if(KM_CW_CCW == 1)           //KM_CW_CCW v nasprotni smeri urne kazalca
        bit_set(PORTD,BIT(4));
    if(KM_CW_CCW == 0)           //KM_CW_CCW v smeri urnega kazalca
        bit_clear(PORTD,BIT(4));
    if(KM_H_F == 1)              //KM_H_F polkoračen način
        bit_set(PORTD,BIT(6));
    if(KM_H_F == 0)              //KM_H_F polnokoračen način
        bit_clear(PORTD,BIT(6));
}
ISR(TIMER1_COMPA_vect)
{
    bit_set(PORTD,BIT(5));       //urin signal na high
}
ISR(TIMER1_COMPB_vect)
{
    bit_clear(PORTD,BIT(5));     //urin signal na low
}
```

Slika 3.20: Del programske kode - Krmiljenje koračnega motorja

3.4.4. Modul za obravnavanje merilne letve

Slika 3.21 prikazuje električne signale, ki jih tvori merilna letve. Signala A in B sta zamaknjena za 90°. Z ustreznim dekodiranjem teh signalov lahko ugotovimo, v katero smer se pomika merilna letve (tabela 3). S pomočjo signala RI ugotovimo položaj merilne letve. V naši aplikaciji ta signal ni potreben, saj položaj določamo s pomočjo končnih stikal. Čeprav je v tabeli 3 prikazano kodiranje rotacijskega dajalnika za obe smeri vrtenja,

pa lahko enako kodiranje uporabimo tudi za merilno letev, saj sta principa kodiranja popolnoma enaka.



Slika 3.21: Izhodni signali merilne letve

Tabela 3: Kodiranje signalov, s katerimi določamo smer merilne letve

V smeri ure			V nasprotni smeri ure		
Stopnja	A	B	Stopnja	A	B
1	0	0	1	1	0
2	1	0	2	0	0
3	1	1	3	0	1
4	0	1	4	1	1

Programsko merimo razdaljo na tak način, da štejemo spremembe nivoja signala. Ker ima mikrokontroler možnost zaznavanja zunanjih sprememb nivojev signalov s prekinitvenimi linijami, smo uporabili prekinitvena vektorja za zaznavanje sprememb signalov, ki sta povezana na zunanji prekinitveni liniji 5 in 7, v programski kodi sta signala A in B označena s P5 za signal A in P7 za signal B. Na ta način smo poenostavili branje signalov z merilne letve. Merilna letev lahko meri dolžino z natančnostjo 20 μm (mikrometrov). Za naš sistem je dovolj velika natančnost 0,1 mm. Kako smo izvedli pretvorbo iz μm v mm je razvidno iz dela programske kode prikazane na sliki 3.22.

```
ISR(INT5_vect)
{
    if(P7==0) //preverja v katero smer se merilna letev premika
    {
        ML_A++; //štejemo prekinitve

        if(ML_A>4) //ko je 5 prekinitvev, se je merilna letev premaknila za 0,1mm
        {
```

```
        ML_A=0;
        ML_Ammd++;           //štejemo desetinke mm
    }
}
flag_INT5 = 1;           //zastavica da se je prekinitev zgodila
}

ISR(INT7_vect)
{
    if(P5==0)           //preverja v katero smer se merilna letev premika
    {
        ML_B++;         //štejemo prekinitve

        if(ML_B>4)     //ko je 5 prekinitvev, se je merilna letev premaknila za 0,1mm
        {
            ML_B=0;    //štejemo desetinke mm
            ML_Bmmd++;
        }
    }
}

void Izracun_Razdalje(void)
{
    ML_skupajd=ML_Ammd-ML_Bmmd; //razdalja premika merilne letve od začetne točke
}
```

Slika 3.22: Del programske kode: Preverjanje smeri merilne letve in merjenje razdalje premika merilne letve

Ko merilna letev postavi linijo A na visoki nivo, to pomeni, da se je merilna letev premaknila za 20 μm , zgodi se prekinitev ISR(INT5_vect). Najprej preverimo, v kakšnem stanju je linija B. Če je signal B na nizkem nivoju, povečamo spremenljivko ML_A za 1. Ko se zgodi peta prekinitev ISR(INT5_vect), povečamo spremenljivko ML_Ammd za ena, kar pomeni, da se je glava merilne letve premaknila za 0,1 mm navzdol. Če bi uporabljali rotacijski dajalnik, bi pomenilo, da se je zavrtil za 0,1 mm v smeri ure. V tem trenutku moramo tudi postaviti spremenljivko ML_A nazaj na 0 zato, da začnemo ponovno šteti prekinitve od začetka. Funkcija Izracun_Razdalja izračunava, za koliko se je glava merilne letve premaknila v katerokoli smer.

3.4.5. Modul za obravnavanje merilca sile in komunikacije z osebnim računalnikom

Eden izmed razlogov, zakaj smo se odločili za mikrokrmilnik ATmega128, je bil tudi ta, da ima možnost priključitve dveh naprav, ki komunicirata preko zaporedne povezave – ima vgrajena dva serijska komunikacijska vmesnika. V našem sistemu sta na ta dva vmesnika priključeni dve zunanji napravi: merilnik sile ter osebni računalnik. Medtem ko smo za povezavo z osebnim računalnikom uporabili dvosmerno komunikacijo, pa je mikrokrmilnik povezan z merilcem sile le enosmerno, saj lahko merilnik sile podatke samo pošilja, ne more jih pa sprejemati.

Za prikaz in pretvorbo podatkov, ki jih merilnik sile pošilja, smo morali spoznati njegov protokol pošiljanja podatkov.

D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
-----	-----	-----	-----	-----	-----	----	----	----	----	----	----	----	----	----	----

Slika 3.23: Komunikacijski protokol merilca sile

Protokol merilnika sile je zasnovan na sporočilu iz 16 zlogov (slika 3.23). Razdelimo jih lahko v več delov. D15 in D0 sta začetni oziroma končni zlog. D14, D13 prikazujeta verzijo merilnika sile. Informacijo o enoti, v kateri je zabeležene izmerjena sila, nosita zloga D12 in D11. Smer sile kaže zlog D10. Zlog D9 pa predstavlja decimalno vejico. Vrednost sile je zapisana z zlogi D8 do D1. Ker smo se odločili, da bomo silo vedno prikazovali le kot pozitivno vrednost ter vedno le v Newton-ih, nas zanima le podatkovni del, tako smo iz niza izločili in obravnavali le zloge D4 do D1, ki predstavljajo rezultat dejanske meritve. Te štiri zloge smo dodali nizu, ki ga pošilja mikrokrmilnik na osebni računalnik. Silo smo na LCD zaslonu prikazovali na enak način, zato smo prav tako upoštevali le zloge D4 do D1.

Za pošiljanje podatkov o meritvi v nadzorni sistem smo razvili lastni protokol (slika 3.24).

;#S	Sila	R	Razdalja	T	Točke	E#;
-----	------	---	----------	---	-------	-----

Slika 3.24: Oblika niza, ki se prenaša na osebni računalnik

Paket, ki ga pošiljamo v nadzorni sistem ni vedno enako dolg, ampak se spreminja glede na izmerjeno razdaljo. Tako je lahko dolg največ 17 zlogov ter najmanj 14 zlogov. Del, kjer se pošilja vrednost izmerjene sile, je dolg 4 zloge. Del za razdaljo se, kot že rečeno,

spreminja in je dolg od 1 do 4 zloge. Del z nazivom 'Točke' je vedno dolg le en zlog in pove točko, katere vrednosti se v danem trenutku pošilja na osebni računalnik.

```
ISR(USART0_RX_vect)
{
  ReceivedByteRdy0 = 1; //postavimo zastavico, da se je zgodila prekinitve(sprejel se je znak)
  ReceivedByte0 = UDR0; // Premaknemo dobljeni byte v spremenljivko ReceivedByte0
  USART0_buffer_r[RX_count0] = ReceivedByte0; //prejeti znak vpišemo v buffer na mest
                                             //kamor kaže RX_count0
  RX_count0++; //štejemo prekinitve da ob naslednji prekinitvi znak ne bo prepisal prej
               //dobljenega znaka
}

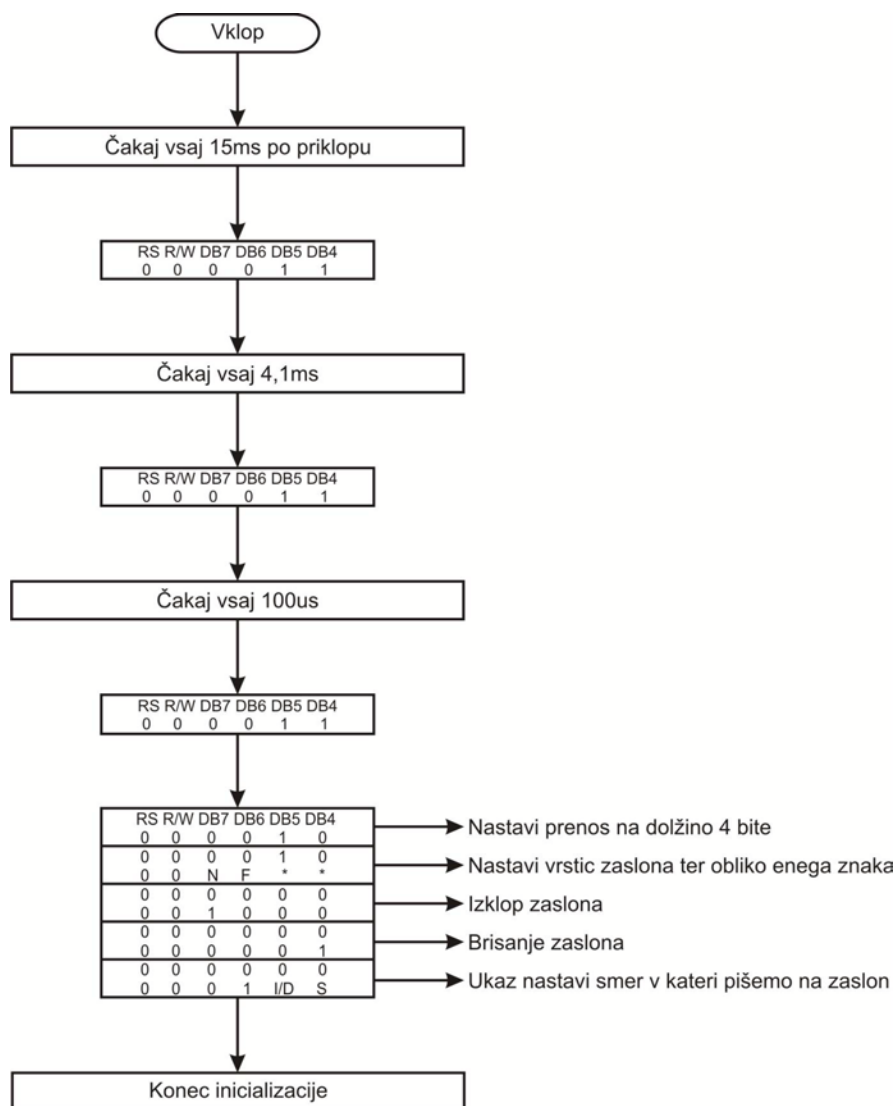
ISR(USART1_RX_vect)
{
  ReceivedByteRdy1 = 1; //postavimo zastavico, da se je zgodila prekinitve(sprejel se je znak)
  ReceivedByte1 = UDR1; // Premaknemo dobljeni byte v spremenljivko ReceivedByte0
  USART1_buffer_r[RX_count1] = ReceivedByte1; //prejeti znak vpišemo v buffer na mesto
                                             //kamor kaže RX_count0
  RX_count1++; //štejemo prekinitve da ob naslednji prekinitvi znak ne bo prepisal prej
               //dobljenega znaka
}
```

Slika 3.25: Del programske kode: Pridobivanje znakov v mikrokrmilnik preko prekinitvenih vektorjev USART0_RX_vect in USART1_RX_vect

Na sliki 3.25 je prikazan del programske kode, s pomočjo katere sprejemamo znake iz zaporednega vmesnika. Ker ima mikrokrmilnik dva vhoda za zaporedno komunikacijo USART0, ki ga uporabljamo za komunikacijo z merilnikom sile, ter USART1, ki ga uporabljamo za komunikacijo z osebnim računalnikom, uporabljamo dva prekinitvena vektorja USART0_RX_vect ter USART1_RX_vect. V primeru, da bi na znake iz zaporedne komunikacijske linije čakali med glavnim programom, bi se le ta zelo upočasnili. Zato uporabljamo mehanizem prekinitvenega vektorja, ki glavnega programa ne upočasnjujejo, saj se prekinitveni servisni podprogram izvaja le takrat, ko je zaporedni komunikacijski sprejemnik mikrokrmilnika sprejel znak. V obeh prekinitvenih podprogramih vsak znak tudi vpisujemo v vmesni pomnilnik (buffer), ki omogoča shranjevanje nizov, ki jih sestavlja več znakov.

3.4.6. LCD modul

Ker smo LCD zaslon povezali na mikrokrmilnik s 4-bitnim podatkovnim vodilom, je LCD zaslon potrebno temu primerno tudi inicilizirati. Diagram poteka inicializacije prikazuje slika 3.26. Inicilizacija se zgodi ob vsakem vklopu krmilnega sistema.



Slika 3.26: Diagram poteka inicializacije LCD zaslona za 4-bitno povezavo z mikrokrmilnikom

Zato, da zapišemo znak na zaslon, ter vpišemo ukaz v krmilnik LCD zaslona, potrebujemo dve različni funkciji, ki sta prikazani na sliki 3.27. S pomočjo teh dveh programskih funkcij tvorimo časovni potek signalov za krmiljenje LCD zaslona. Funkcija LCD_instruction vpiše ukaz, funkcija LCD_putchar pa znak v LCD krmilnik.

```
//////////Funkcija za vpis ukaza v LCD //////////////////////////////////////
void LCD_instruction(char data)
{
    char HIGH,LOW;

    HIGH = data & 0xF0; //maskiramo zgornje 4 bite
    LOW = data & 0X0F; //maskiramo spodnje 4 bite
    LOW = LOW << 4; //spodnje 4 bite premaknemo na mesto zgornjih 4 bitov

    LCD = HIGH; //na vodilo postavimo zgornje 4 bite
    rst_RS_LCD;
    set_EN_LCD;
    rst_EN_LCD;
    delay_us(100);

    LCD = LOW; //na vodilo postavimo spodnje 4 bite
    rst_RS_LCD;
    set_EN_LCD;
    rst_EN_LCD;
    delay_us(100);
}
//////////Funkcija za izpis znaka na LCD //////////////////////////////////////
void LCD_putchar(char data)
{
    char HIGH,LOW;

    HIGH = data & 0xF0; //maskiramo zgornje 4 bite
    LOW = data & 0X0F; //maskiramo spodnje 4 bite
    LOW = LOW << 4; //spodnje 4 bite premaknemo na mesto zgornjih 4 bitov

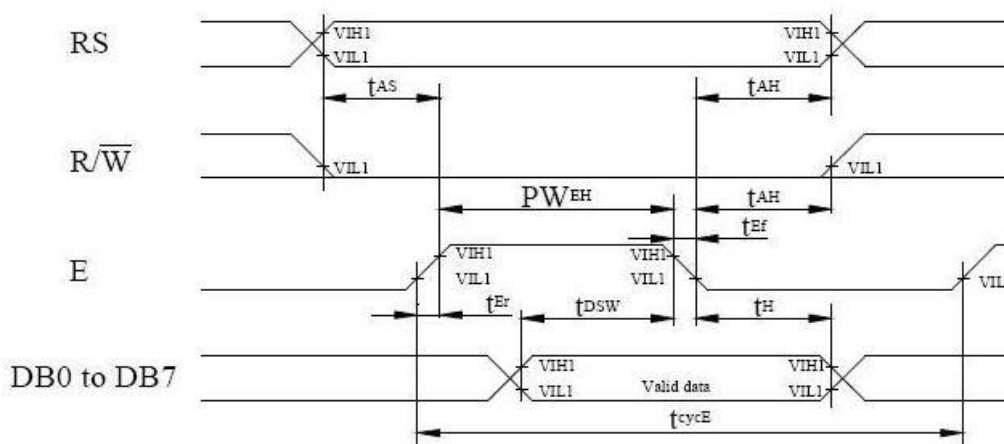
    LCD = HIGH; //na vodilo postavimo zgornje 4 bite
    set_RS_LCD;
    set_EN_LCD; //izpis v RAM
    rst_EN_LCD;
    rst_RS_LCD;
    delay_us(100);

    LCD = LOW; //na vodilo postavimo spodnje 4 bite
    set_RS_LCD;
    set_EN_LCD;
    rst_EN_LCD;
    rst_RS_LCD;
    delay_us(100);
}
```

Slika 3.27: Funkciji za vpis ukaza in za izpis znaka na LCD zaslon

Slika 3.28 prikazuje časovni potek signalov za krmiljenje LCD zaslona, tabela 4 pa vrednosti posameznih časovnih parametrov iz tega časovnega diagrama. Če želimo poslati

ukaz krmilniku LCD zaslona, je potrebno najprej postaviti signal RS na nizek nivo in za tem signal EN držati na visokem nivoju v trajanju vsaj 150 ns. Čeprav bi morali pred tem, ko postavimo EN na visok nivo, počakati vsaj 30 ns, za tem ni potrebno dodajati zakasnitve, saj lahko mikrokrmilnik, ker deluje pri frekvenci 8MHz, naslednji ukaz izvede šele 125 ns kasneje. Je pa potrebno počakati vsaj 100 us za vsak del znaka, ki ga pošljemo krmilniku LCD zaslona, kar je razvidno s slike 3.26, saj bi v nasprotnem primeru krmilnik LCD zaslona dobil ukaze prehitro in jih ne bi bil sposoben dovolj hitro procesirati. V tem načinu krmiljenja LCD zaslona predstavlja vsak znak, ki ga postavimo na vodilo, ukaz za krmilnik LCD zaslona.



Slika 3.28: Časovni potek signalov za krmiljenje LCD zaslona

Tabela 4: Časi posameznih časovnih parametrov za krmiljenje LCD zaslona

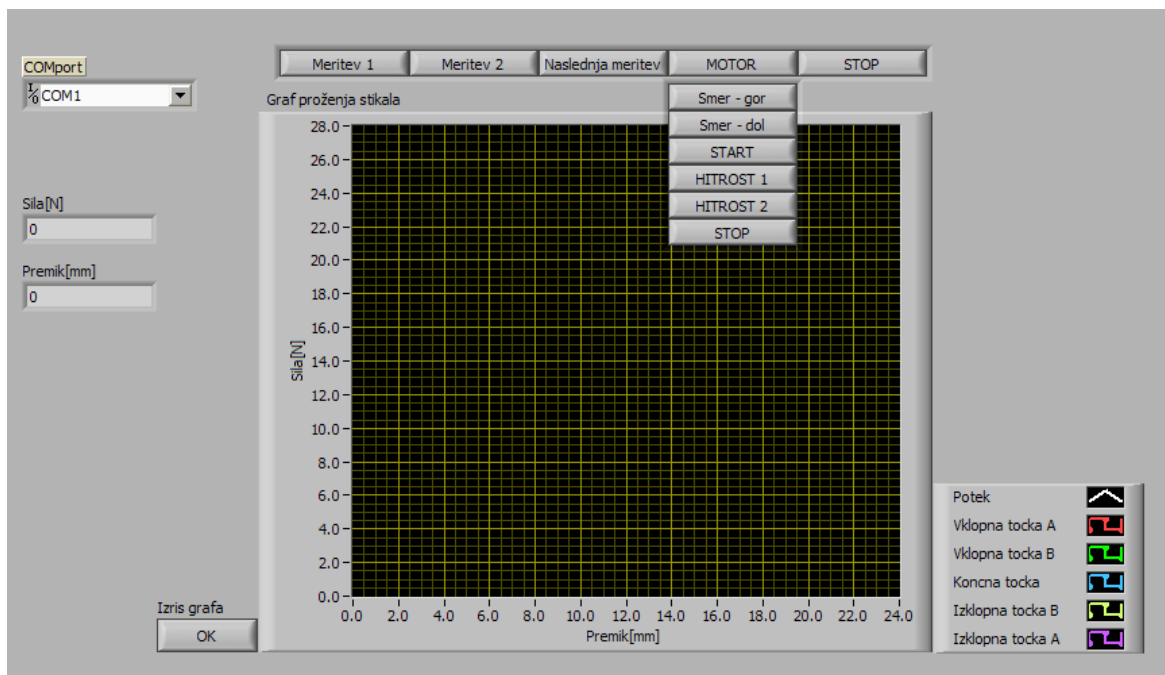
Parameter	Simbol	Min.	Tip.	Max.	Enota
Omogočitveni čas cikla	t_{cycE}	400	□	□	ns
Dolžina omogočitvenega impulza(visoki nivo)	P_{WEH}	150	□	□	ns
Čas upada in dviga omogočitvenega signala	t_{Er}, t_{Ef}	□	□	25	ns
Čas nastavitve naslova (RS,R/W do E)	t_{AS}	30	□	□	ns
Čas zadržanja naslova	t_{AH}	10	□	□	ns
Čas nastavitve podatka	t_{DSW}	40	□	□	ns
Čas zadržanja podatka	t_H	10	□	□	ns

3.5 Sistem za nadzor in prikaz merilnih rezultatov

Za izdelavo sistema za nadzor in prikaz merilnih rezultatov smo uporabili programsko orodje LabVIEW [8] ameriškega podjetja National Instruments. LabVIEW (kratko za **L**aboratory **V**irtual **I**nstrumentation **E**ngineering **W**orkbench) je program in razvojno okolje za grafično programiranje. Programski jezik je imenovan G. LabVIEW in je splošno uporabljen sistem za pridobivanje podatkov, nadzor sistemov in industrijsko avtomatizacijo. Zadnja verzija programskega orodja LabVIEW je 8.6, katero smo uporabili tudi za našo aplikacijo.

S pomočjo programa LabVIEW smo torej razvili uporabniški vmesnik za PC računalnik, s pomočjo katerega upravljamo celotni sistem, hkrati pa shranjujemo izmerjene podatke na disk računalnika.

Slika 3.29 prikazuje uporabniški vmesnik, ki omogoča proženje avtomatske meritve za dva tipa merjenja. To storimo z gumbom *Meritev 1* ali *Meritev 2*. Ko meritev na določenem merjencu opravimo, lahko novo meritev sprožimo z gumbom *Naslednja meritev*. Vmesnik omogoča tudi ročno krmiljenje motorja. V prikazovalnih oknih *Sila[N]* in *Premik[mm]* se med samo meritvijo izpisujeta vrednosti za trenutno silo in premik glave merilnika sile. Ob končani meritvi izrišemo s pomočjo gumba *Izris grafa* za potek premika tipke v odvisnosti od sile. Z gumbom *MOTOR* lahko vklopimo oziroma izklopimo meni za prostoročno krmiljenje motorja.

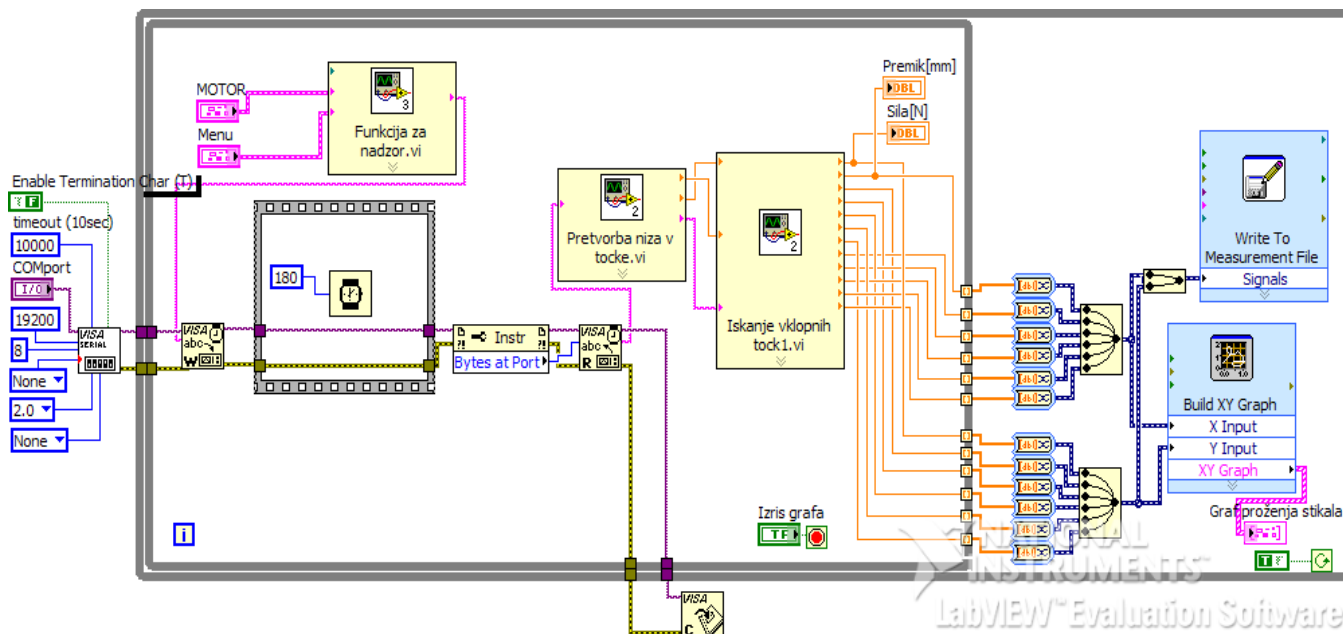


Slika 3.29: Uporabniški vmesnik za prikaz merilnih rezultatov in nadzor aplikacije

Slika 3.30 prikazuje osnovni blokovni diagram uporabniškega vmesnika. V okviru tega diagrama smo razvili naslednje funkcije:

- Razdelitev niza v posamezne dele
- Iskanje karakterističnih točk merjenja
- Pošiljanje ukazov v mikrokrmilnik

Poleg teh funkcij, ki smo jih razvili s pomočjo LabVIEW-a, smo uporabili tudi funkcije, ki so že integrirane v knjižnico orodja LabVIEW. V dodatku D podajamo kratko razlago blokov paketa Labview, ki jih uporabljamo v nadaljevanju. Za inicializacijo in uporabo zaporedne povezave smo uporabili že vgrajene funkcije. Funkcijo *VISA serial* uporabimo za inicializacijo zaporedne povezave. Funkcijo *VISA W* uporabimo za pošiljanje niza iz računalnika v zunanji mikrokrmilnik, *VISA R* pa za prejemanje niza iz mikrokrmilnik v računalnik. Med *VISA W* in *VISA R* moramo vgraditi funkcijo *Zakasnitev* (Delay). Zunaj zanke moramo uporabiti še *VISA C* funkcijo zato, da se ob končanem testu zapre linija (COM port), na katero je priključen sistem. Za risanje poteka meritve smo uporabili funkcijo za izris XY grafa – *Build XY graph*. Za shranjevanje podatkov v datoteko pa funkcijo za zapis signalov v datoteko – *Write To Measurement File*.

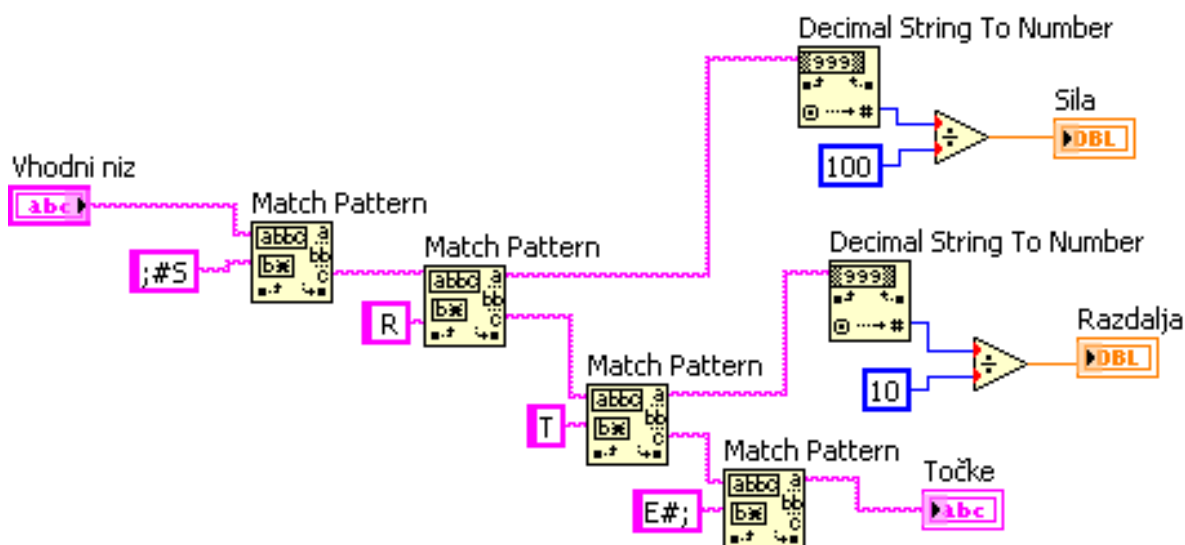


Slika 3.30: Glavni blokovni diagram uporabniškega vmesnika

3.6 Razdelitev niza v posamezne dele

V funkciji za razdelitev niza v posamezne dele razdelimo in pretvorimo vhodni niz, ki ga dobimo preko zaporedne komunikacije. S slike 3.31 je razvidno, da vhodni niz razstavimo na tri sestavne dele – karakteristične veličine merjenja:

- trenutna sila ('Sila')
- trenutna razdalja ('Razdalja')
- karakteristične točke merjenja ('Točke')



Slika 3.31: »Koda« funkcije za pretvorbo niza v posamezne dele

Vhodni niz je sestavljen takole: ';#SxxxxRxxxxTXE#;'. LabVIEW ima v svoji široki knjižnici že funkcije, ki omogočajo razstavljanje in pretvarjanje nizov na zelo enostaven način. V prvi funkciji za razstavljanje niza (*Match Pattern*) poiščemo prvi del vhodnega niza. Glede na protokol, ki ga opisujemo v poglavju 3.4.5, tvori začetni del vhodnega niza zaporedje znakov ';#S'. Funkcija za razstavljanje niza torej prepozna začetno zaporedje in izloči preostali del vhodnega niza. V drugi funkciji za razstavljanje niza poiščemo v vhodnem nizu znak 'R', izločimo silo in ponovno posredujemo v obdelavo preostali del vhodnega niza. Zapis za silo pretvorimo iz ASCII oblike v številčni zapis s pomočjo funkcije *Decimal String To Number*. Izmerjeno silo sprejmemo zapisano v gramih, želimo pa jo prikazati v Newton-ih. Zaradi tega delimo pretvorjeno število z delilnikom 100. Tako dobimo dejansko izmerjeno silo. Kot je razvidno iz slike 3.31 je preostali del izveden na podoben način, le da del, ki nam pomeni posamezne vklopne oziroma izklopne točke, ne pretvarjamo iz ASCII zapisa v številčni zapis. Znaku 'R' v vhodnem zaporedju sledi ASCII vrednost za razdaljo, znaku 'T' pa sledi še vrednost karakteristične točke.

3.7 Določanje karakterističnih točk

To funkcijo sestavlja 5 'case' stavkov, ki določajo oziroma izbirajo karakteristične točke merjenja:

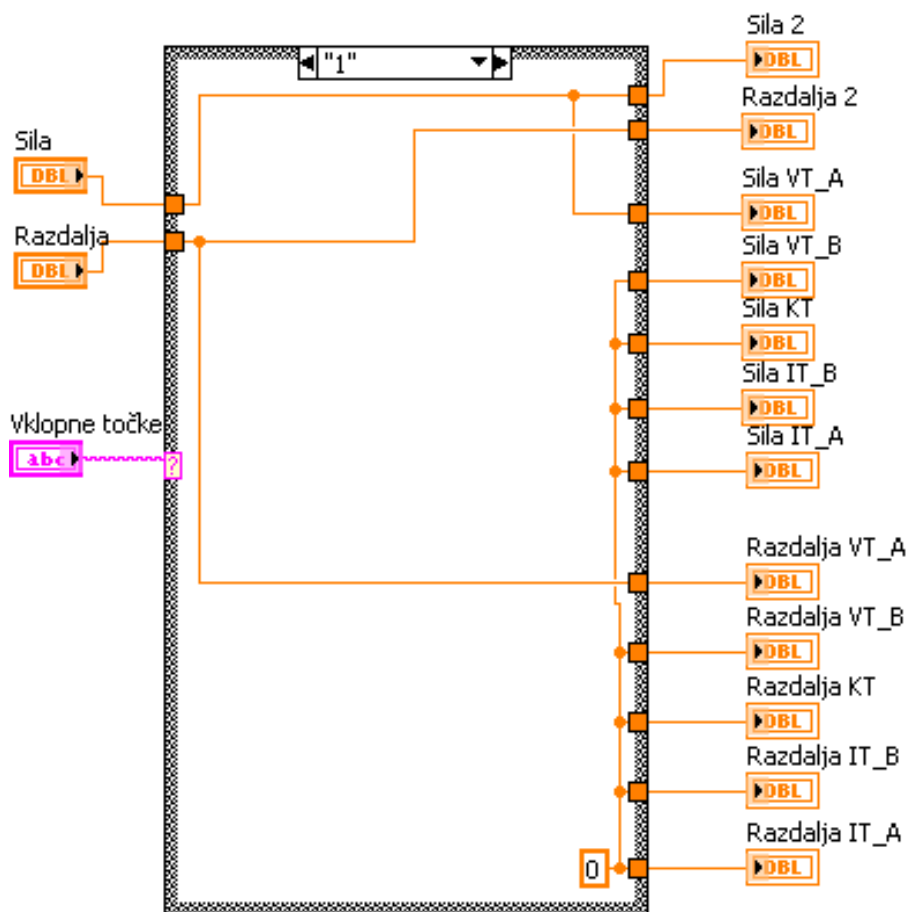
- sila in razdalja ob vklopu merjenja (Sila VT_A, Razdalja VT_A)
- sila in razdalja ob izklopu merjenja (Sila IT_A, Razdalja IT_A)
- sila in razdalja ob končni točki merjenja (Sila KT, Razdalja KT)
- v primeru merjenja, ki ima vgrajeni dve stikali, dobimo še dodatno silo in razdaljo ob vklopu in izklopu merjenja (Sila VT_B, Razdalja VT_B, Sila IT_A, Razdalja IT_A)

Niz, ki ga pošiljamo preko zaporedne komunikacije iz mikrokrmilnika, bi bil relativno dolg, če bi za vsako karakteristično točko pošiljali točno vrednost sile in razdalje, pošiljamo le znak za indikacijo karakteristične točke, ki se je zgodila pri tisti sili in razdalji, ki jo takrat pošiljamo. Tako za vsako od karakterističnih točk dobimo svojo indikacijo iz niza (Slika 3.24) dobljenega preko zaporedne komunikacije iz krmilnega dela (3.4.5):

- 'T1' – sila in razdalja ob vklopu merjenja
- 'T2' - sila in razdalja ob izklopu merjenja
- 'T3' - sila in razdalja ob končni točki merjenja
- 'T4' – sila in razdalja za drugo vklopno točko merjenja z dvema stikaloma
- 'T5' – sila in razdalja za drugo izklopno točko merjenja z dvema stikaloma

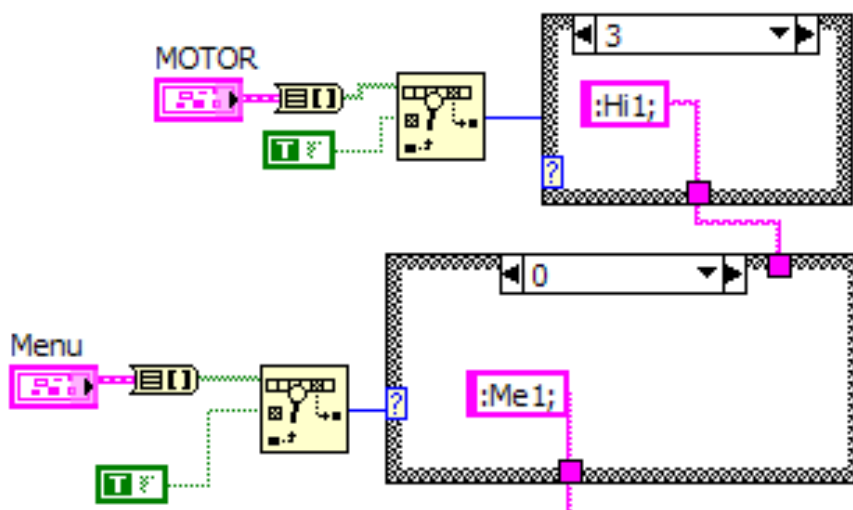
Na sliki 3.32 je prikazan le del funkcije za določanje sile in razdalje ob vklopu merjenja. Opisali bomo le ta del, saj so drugi enaki in se razlikujejo le v tem, da določamo drugo karakteristično točko merjenja. Vse tri veličine silo, razdaljo in točke smo iz niza izluščili že v predhodni funkciji. Tako v tem delu dobimo na vhod funkcije trenutne vrednosti veličin, ki so bile poslane preko zaporedne komunikacije. V primeru, ki je prikazan na sliki 3.32, dobimo vrednost 1 za karakteristične točke, kar nam pomeni Vklonno točko A. Tako priredimo vrednost sile in razdalje, ki je bila poslana poleg vrednosti 1 za karakteristično

točko, spremenljivkama Sila VT_A ter Razdalja VT_A. Tako smo dobili vrednosti za prvo karakteristično točko merjenja.



Slika 3.32: Funkcija za določanje karakterističnih točk

3.8 Funkcija za nadzor delovanja sistema



Slika 3.33: Funkcija za nadzor delovanja sistema

Podobno kot v prejšnjem poglavju je tudi na sliki 3.33 prikazan le del funkcije, ki opravlja nadzor delovanja sistema. Razlika od prej je le v tem, da se v prejšnjem primeru »case« stavki izvajajo glede na prejeti niz, v tem primeru pa se »case« stavki izvajajo glede na to, kateri gumb smo pritisnili v meniju na zaslonu računalnika. V primeru, ki je na sliki 3.33, se potem, ko pritisnemo *Meritev 1*, v sistem posreduje zaporedje "MI;". To zaporedje sproži izvedbo avtomatske meritve za merjenec 1. Ostali primeri, ki jih lahko izberemo v meniju na zaslonu, so:

- ":Me2;" se posreduje v primeru, ko želimo, da se izvede avtomatska meritev za merjenec 2 in pritisnemo tipko *Meritev 2*.
- ":NMe;" se posreduje za naslednjo avtomatsko meritev prej izbranega merjenca, ko pritisnemo tipko *Naslednja meritev*.
- ":STp;" se posreduje, če hočemo ustaviti meritev, tipka STOP.
- V primeru, ko pritisnemo tipko *Motor* in želimo meritev opraviti prostoročno, se preko zaporedne povezave posredujejo v mikrokrmilnik naslednja zaporedja:
 - ":SmG;" če želimo da motor premika ročico gor, tipka Smer - gor
 - ":SmD;" če želimo da motor premika ročico dol, tipka Smer - dol

- ":*St*;" za začetek meritve, tipka START
- ":*Hi1*;" za počasno vrtenje motorja, tipka HITROST 1
- ":*Hi2*;" za hitrejše vrtenje motorja, tipka HITROST 2.

4. MERITVE IN REZULTATI

4.1 Meritev električnih karakteristik krmilnika

4.1.1. Meritev porabe merilnega sistema

Merilni sistem lahko deluje v različnih načinih, zato se poraba sistema temu primerno tudi spreminja. Čeprav lahko sistem napajamo z napetostjo od 15V do maksimalno 42V, smo se odločili, da ga bomo uporabljali z minimalno zahtevano napetostjo. Tako je ob napetosti 15V poraba toka različna. Odvisna je predvsem od uporabe koračnega motorja.

Ko koračnega motorja ne uporabljamo je poraba toka okoli 0,250 A. Moč, ki jo sistem v tem primeru porabi je:

$$P = U \cdot I = 15V \cdot 0,250A = 3,75 W. \quad (4.1)$$

Ob uporabi motorja pa se poraba spreminja. Razlog temu je, da lahko z delom vezja, ki je namenjen krmiljenju koračnega motorja, porabo na motorju spreminjamo. Uporabili smo koračni motor, ki lahko prejme maksimalen tok 1A. Ker pa nam ni bilo potrebno uporabiti celotne moči motorja in bi si v primeru, da bi jo, lahko motor zelo segreval, smo porabo toka na motorju nastavili približno na 0,75 A. Tako se poraba celotnega sistema zelo poveča in je v tem primeru:

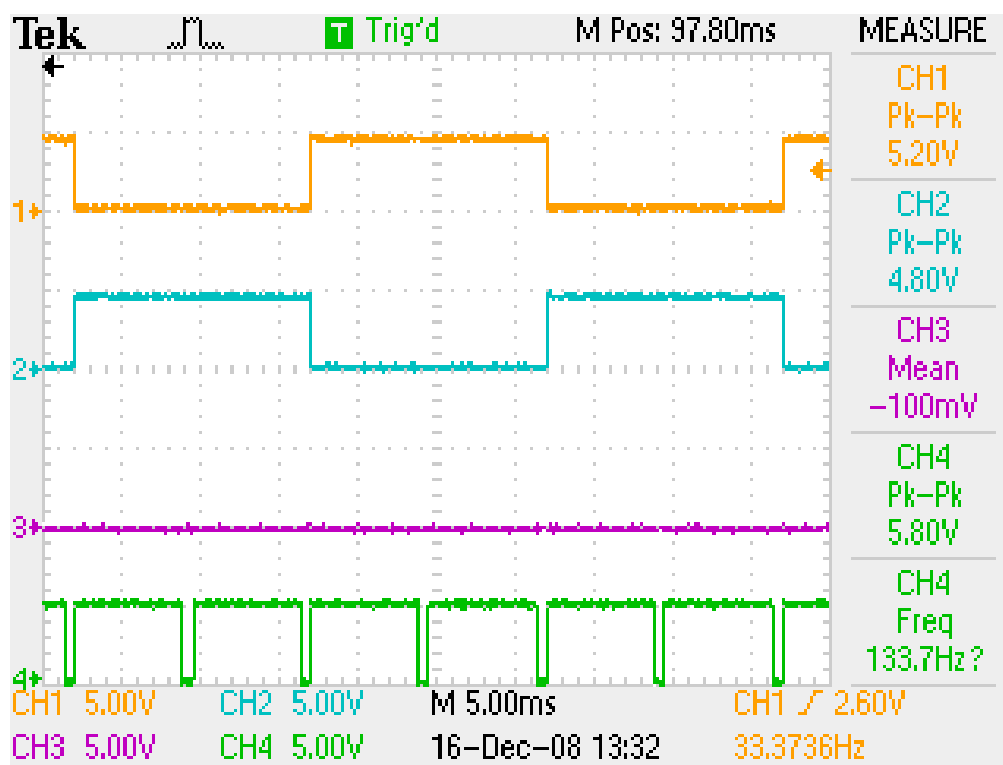
$$P = U \cdot I = 15V \cdot 1A = 15 W. \quad (4.2)$$

Te meritve smo morali opraviti pred načrtovanjem napajalnika, da smo ga lahko pravilno dimenzionirali.

4.2 Meritev signalov za krmiljenje koračnega motorja

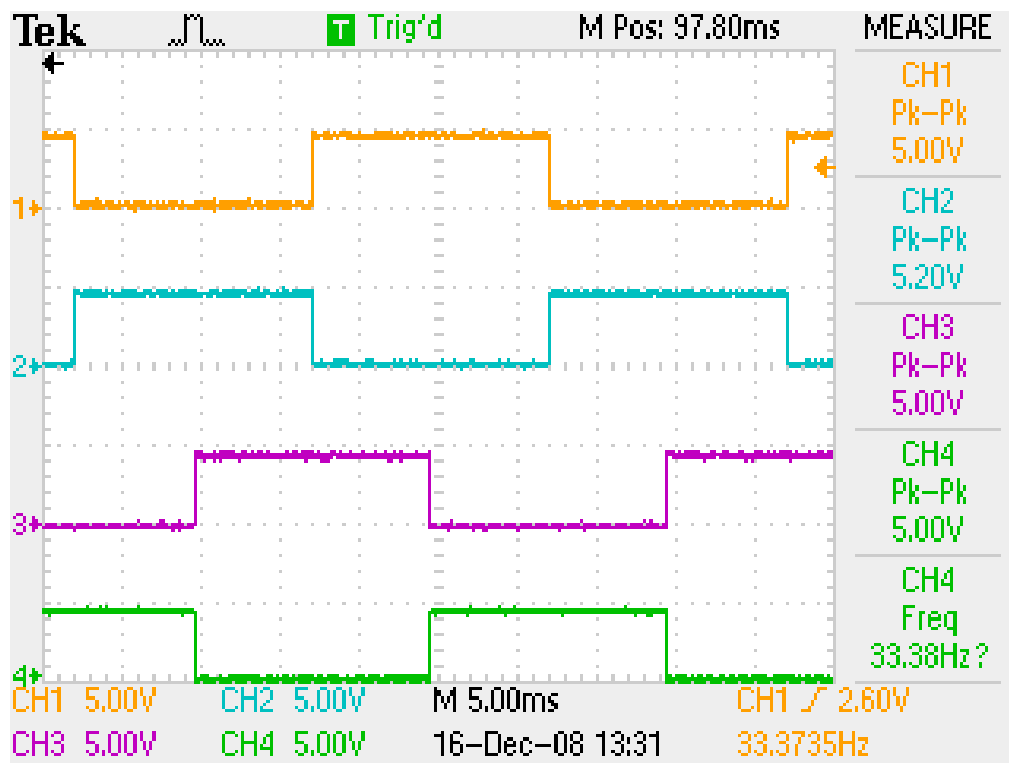
Za pravilno delovanje merilnega sistema je pomembno, da tudi koračni motor krmilimo pravilno in s pravnimi signali ter, da se temu primerno tudi koračni motor obnaša. Tako smo izmerili signal, glede na katerega integrirano vezje L297 generira izhodne signale. Merili smo izhodne signale L297 v dveh načinih delovanja, to je pri polnokoračnem in

polkoračnem načinu. Urin impulz, ki se uporablja za »vrtenje« koračnega motorja, kjer en urin impulz pomeni premik koračnega motorja za en korak, smo nastavili na frekvenco pri kateri deluje koračni motor med samo meritvijo, to je 133Hz. Na sliki 4.1 vidimo, kako je odvisen izhodni signal na L297 od vhodnega urinega signala pri polnokoračnem načinu. Izhodna signala na L297 smo z osciloskopom merili s kanaloma 1 in 2, s kanalom 3 smo merili signal za način delovanja, kanal 4 pa smo uporabili za merjenje urinega signala. S slike 4.1 je razvidno, da je kanal 3 na nizkem nivoju, kar za L297 pomeni, da bo koračni motor deloval v polnokoračnem načinu. Tako je perioda izhodnega signala dolga 4 vhodne urine impulze. Ker koračni motor krmilimo s 4 izhodnimi signali, iz tega sledi, da se koračni motor prestavi za en korak za vsak urin impulz. Pri polnokoračnem načinu je tako frekvenca izhodnega signala 33Hz (Kanal 1 in 2 na sliki 4.2).



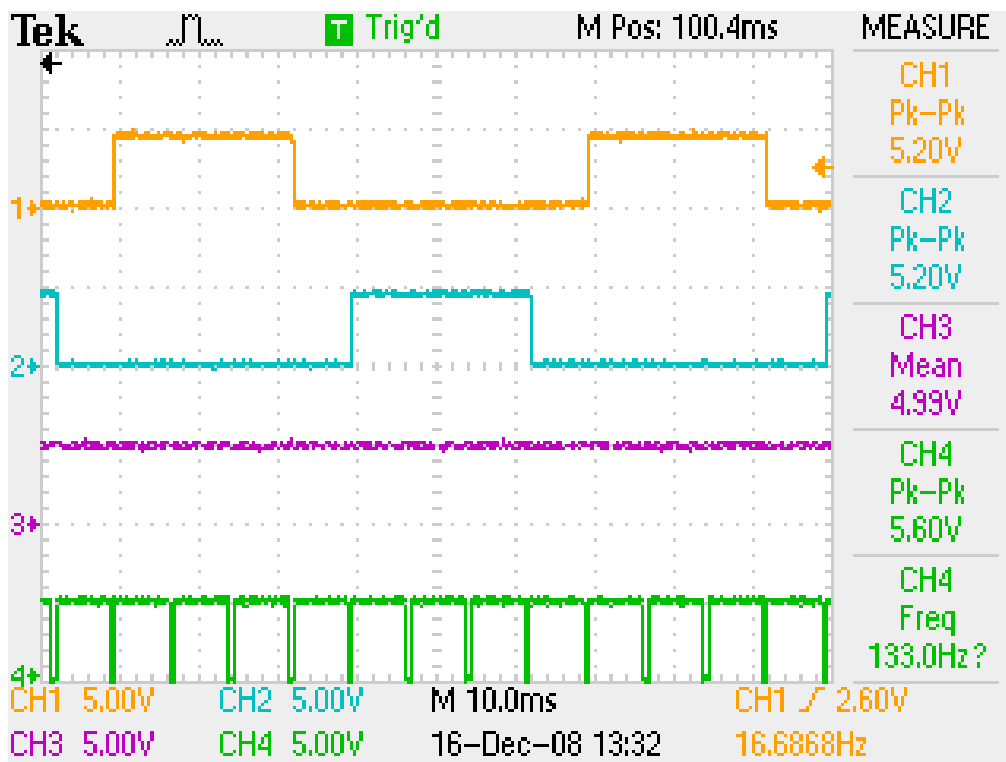
Slika 4.1: Izhodni signal na L297 v odvisnosti od vhodnega urinega signala pri polnokoračnem načinu

Na sliki 4.2 je prikazana meritev vse 4 izhodnih signalov. Kanal 1 merimo na izhodu A, kanal 2 na izhodu B, kanal 3 in 4 pa na izhodu C in D. Razvidno je, da sta si izhoda A in B ter C in D invertirana. V parih sta izhodna signala tudi povezana na navitji motorja. Tuljavi se izmenično vzbujata. Zaradi tega se spreminja magnetno polje motorja in s tem se motor vrti.

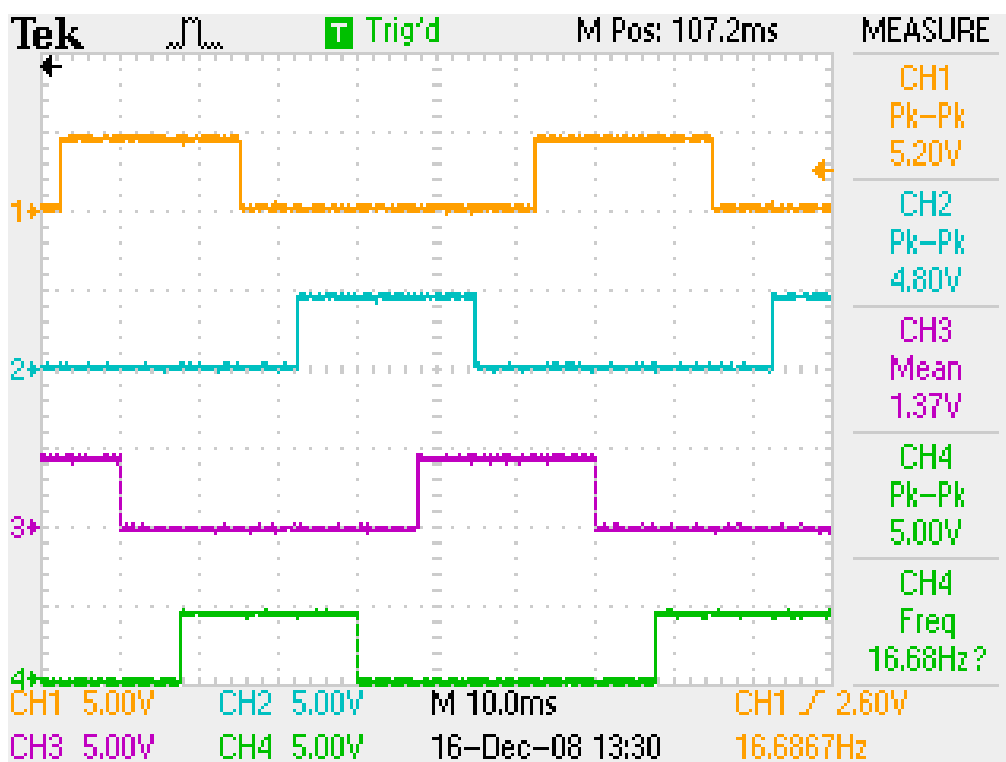


Slika 4.2: Vsi izhodni signali na L297 pri polnokoračnem načinu

Na sliki 4.3 je prikazana meritev signalov na L297 pri polkoračnem načinu. Od polnokoračnega načina na sliki 4.1 se razlikuje po tem, da je signal merjen na kanalu 3, to je signal za določanje načina delovanja, zdaj na visokem nivoju in pomeni, da L297 krmili motor v polkoračnem načinu. Temu primerno se spremenijo tudi izhodni signali, merjena na kanalu 1 in 2. Urin signal je enak kot pri polnokoračnem načinu. Deluje na frekvenci 133Hz. Pri polkoračnem načinu pa je perioda izhodnega signala dolga 8 urinih impulzov. To pomeni, da se koračni motor v tem načinu delovanja vrti počasneje. Meritev frekvence izhodnega signala na sliki 4.4 nam to dokaže. Frekvenca je 16 Hz, kar je ravno pol manj kot pri polnokoračnem načinu. Vidimo, da se bo koračni motor pri enaki frekvenci delovanja, pri polnokoračnem načinu vrtel enkrat hitreje. Če je koračni motor pri polnokoračnem načinu za to, da se je zavrtel za en krog potreboval 200 urinih impulzov oziroma 200 korakov, lahko sklepamo, da bo pri polkoračnem načinu potreboval 400 urinih impulzov oziroma 400 korakov.



Slika 4.3: Izhodni signal na L297 v odvisnosti od vhodnega urinega signala pri polkoračnem načinu



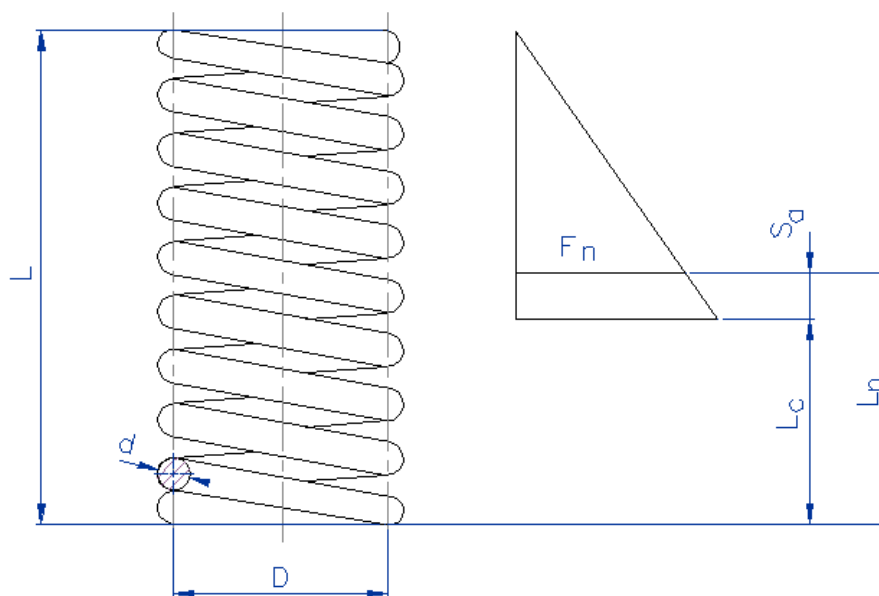
Slika 4.4: Vsi izhodni signali na L297 pri polkoračnem načinu

4.3 Umerjanje sistema

Ker smo kot merilnik razdalje uporabili linearni dajalnik in preko mikrokrmilnika pretvorili signale dajalnika v mm, smo morali naš sistem tudi umeriti. To smo storili z izračunom poteka stiska vzmeti v odvisnosti od sile.

Uporabili smo vzmet z naslednjimi podatki:

- premer žice: $d = 1,0\text{mm}$,
- srednji premer vzmeti: $D = 15,0\text{mm}$,
- dolžina neobremenjene vzmeti: $L = 32,0\text{mm}$,
- popolnoma stisnjena vzmet: $L_c = 9,0\text{mm}$
- strižni modul po DIN17223B: $G = 81500\text{ N/mm}^2$ in
- število aktivnih ovoje vzmeti: $i_f = 7$.



Slika 4.5: Cilindrična tlačna vzmet [17]

4.4 Izračun vzmetne konstante vzmeti

Vzmetna konstanta (c) je razmerje med obremenitvijo vzmeti in povosom vzmeti pri tej obremenitvi.

$$c = F/f = (G \cdot d^4) / (8 \cdot D^3 \cdot i_f) \quad [17] \quad (4.3)$$

Izračun vzmetne konstante glede na podatke proizvajalca:

$$c = F/f = (G \cdot d^4) / (8 \cdot D^3 \cdot i_f) = (81500 \cdot 1^4) / (8 \cdot 15^3 \cdot 7) = 0,43 \text{ N/mm.} \quad (4.4)$$

Izračun vzmetne konstante s podatki vzmeti izmerjenimi s kljunastim merilom:

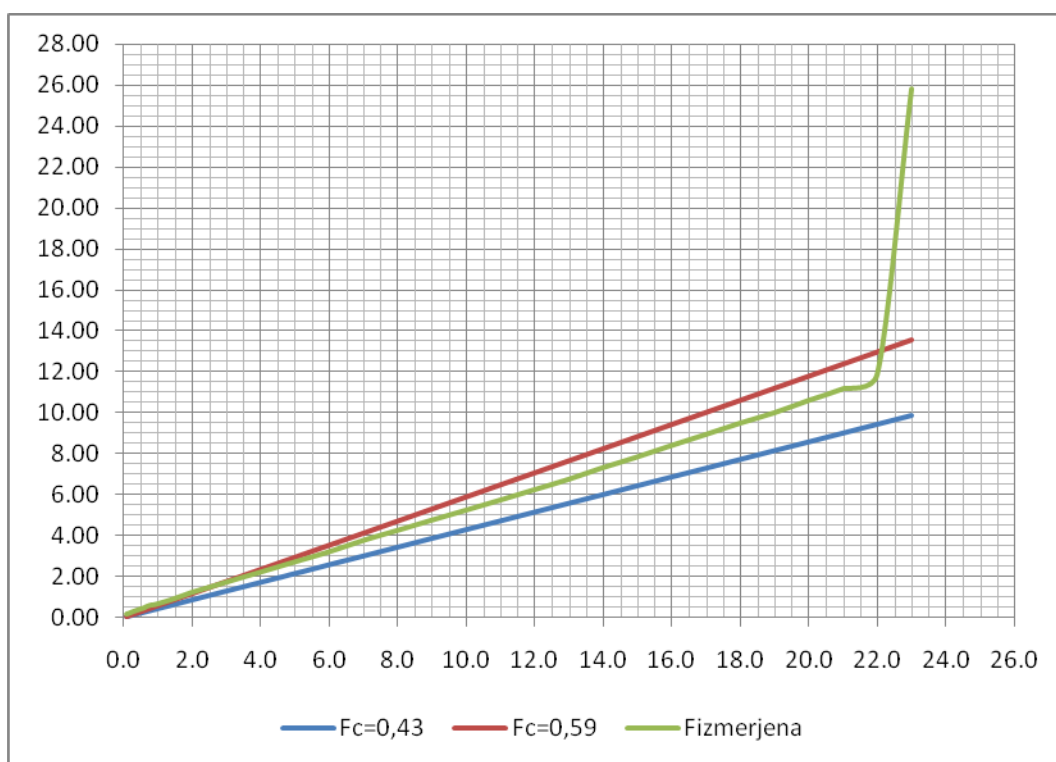
$$c = F/f = (G \cdot d^4) / (8 \cdot D^3 \cdot i_f) = (81500 \cdot 1,02^4) / (8 \cdot 13,85^3 \cdot 7) = 0,59 \text{ N/mm} \quad (4.5)$$

Razlog za tako veliko razliko med izračunanima konstantama je ta, da ima proizvajalec verjetno veliko toleranco v katero še dopušča karakteristike vzmeti. Drugi razlog je tudi ta, da s kljunastim merilom ne moremo izmeriti vzmeti tako natančno.

Tabela 5: Preračun in meritev karakteristike stiska vzmeti

		$c = 0,43 \text{ N/mm}$	$c = 0,59 \text{ N/mm}$	
$L_{\text{trenutna}}[\text{mm}]$	$f[\text{mm}]$	Fizračunana[N]	Fizračunana[N]	Fizmerjena[N]
0.1	31.9	0.04	0.06	0.15
0.2	31.8	0.09	0.12	0.21
0.3	31.7	0.13	0.18	0.28
0.4	31.6	0.17	0.24	0.34
0.5	31.5	0.22	0.30	0.40
0.6	31.4	0.26	0.35	0.45
0.7	31.3	0.30	0.41	0.53
0.8	31.2	0.34	0.47	0.59
0.9	31.1	0.39	0.53	0.61
1.0	31.0	0.43	0.59	0.65
1.5	30.5	0.65	0.89	0.91
2.0	30.0	0.86	1.18	1.21
2.5	29.5	1.08	1.48	1.46
3.0	29.0	1.29	1.77	1.71
3.5	28.5	1.51	2.07	1.96
4.0	28.0	1.72	2.36	2.20
4.5	27.5	1.94	2.66	2.45
5.0	27.0	2.15	2.95	2.70
6.0	26.0	2.58	3.54	3.19

7.0	25.0	3.01	4.13	3.75
8.0	24.0	3.44	4.72	4.24
9.0	23.0	3.87	5.31	4.74
10.0	22.0	4.30	5.90	5.23
11.0	21.0	4.73	6.49	5.72
12.0	20.0	5.16	7.08	6.23
13.0	19.0	5.59	7.67	6.74
14.0	18.0	6.02	8.26	7.31
15.0	17.0	6.45	8.85	7.83
16.0	16.0	6.88	9.44	8.39
17.0	15.0	7.31	10.03	8.93
18.0	14.0	7.74	10.62	9.48
19.0	13.0	8.17	11.21	9.99
20.0	12.0	8.60	11.80	10.59
21.0	11.0	9.03	12.39	11.16
22.0	10.0	9.46	12.98	11.88
23.0	9.0	9.89	13.57	25.81

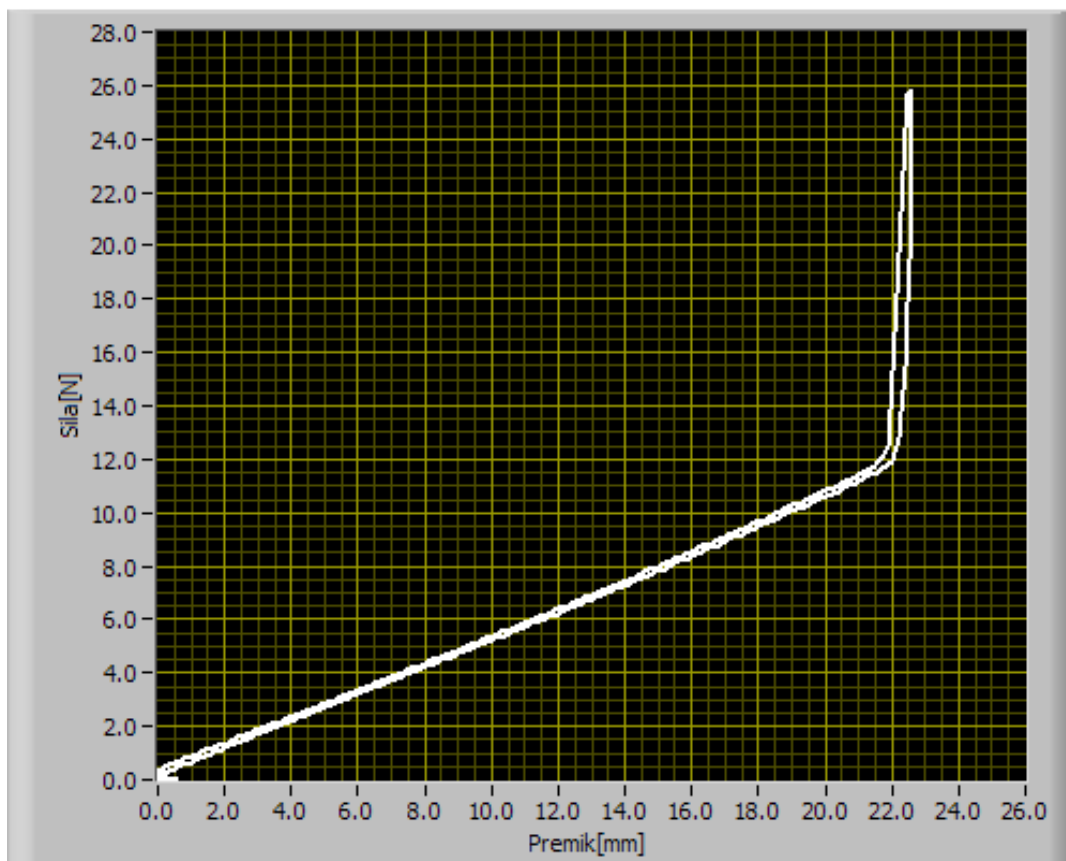


Slika 4.6: Karakteristika stiska vzmeti

Po tem, ko smo izračunali karakteristiko stiska uporabljene vzmeti, smo le to še izmerili z našim sistemom. Vzmet smo pritrdili na ravno ploščo in jo postavili pod »roko« merilca

sile. Nanjo smo pritiskali s hitrostjo 0,1 mm na 0,3 sekunde. Razlog za takšno hitrost je bil ta, da lahko merilnik sile pošilja podatke najhitreje vsake 0,3 sekunde. Čeprav v tabeli 5 nismo uporabili vseh podatkov, pa smo meritev opravili s takšno hitrostjo zato, da smo dobili najbolj natančne podatke kot smo jih lahko. Na sliki 4.7 je tako prikazan natančen graf karakteristike stiska vzmeti.

Graf stiska vzmeti



Slika 4.7: Potek meritve karakteristike stiska vzmeti

Iz tabele 5 ter grafa na sliki 4.6 je razvidno, da je oblika izmerjene karakteristike vzmeti zelo podobna izračunanim. Iz tega lahko sklepamo, da merilni sistem deluje dovolj natančno in daje verodostojne rezultate.

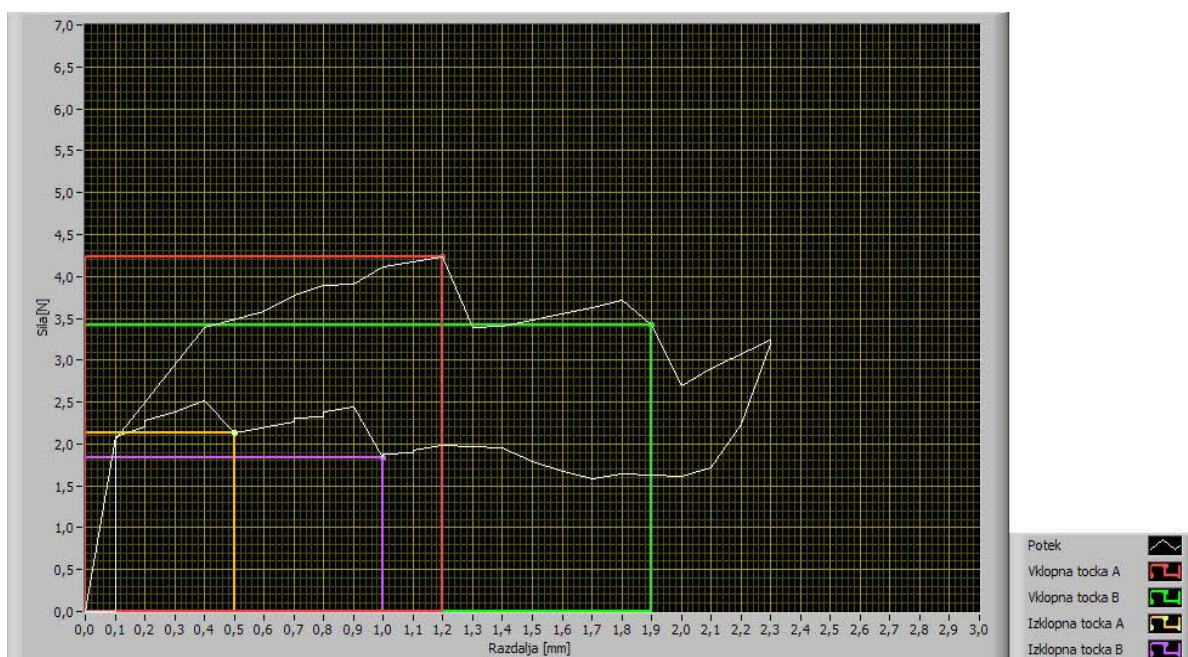
4.5 Meritev tipke z dvojnimi stikalom

V tem poglavju bomo opisali meritev, za katero je v osnovi ta sistem, ki smo ga v okviru tega diplomskega dela razvili, namenjen. To meritev smo opravili zato, da smo preverili, če se tipka merjenca, ki je v bistvu sestavljena iz dveh stikal, obnaša specifikacijam

primerno. Predvsem je bilo pomembno to, da se stikali ne preklopita istočasno, ampak morata imeti nek zamik. Kot je razvidno s slike 4.8, smo merili 5 karakterističnih točk tipke merjenca. To so:

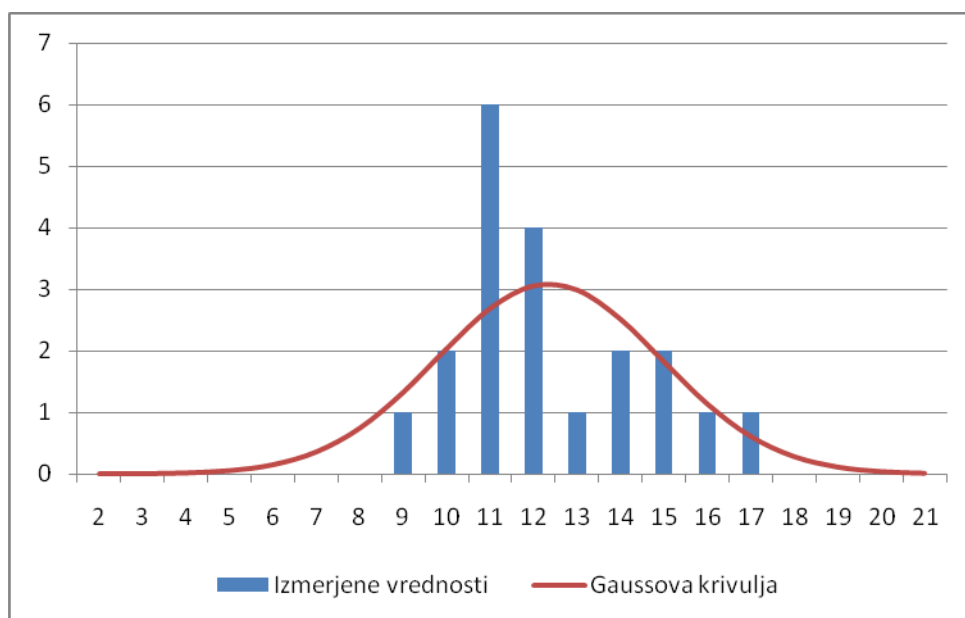
- vklopna točka A, ki predstavlja trenutek ob vklopu prvega stikala,
- vklopna točka B predstavlja trenutek ob vklopu drugega stikala,
- končna točka predstavlja trenutek, ko je stikalo do konca pritisjeno,
- izklopna točka A predstavlja trenutek ob izklopu prvega stikala in
- izklopna točka B, ki predstavlja trenutek ob izklopu drugega stikala.

Potek signala in karakteristične točke merjenca prikazuje slika 4.8. Prikazuje pravilno delovanje tipke merjenca. S slike 4.8 vidimo, kako se spremeni sila po vsakem vklopu oziroma izklopu stikala. Razvidno je, da med potekom vklapljanja stikala sila, s katero pritiskamo na tipko, narašča in v trenutku po vsakem vklopu pade. Do tega pride zato, ker se je stikalo preklopilo in se pritisk na tipko zaradi tega rahlo sprosti. Ob popuščanju tipke pa pride do obratnega pojava, saj se takrat stikalo preklopi na prejšnjo pozicijo, zaradi katere je potrebna večja sila na tipko in zato v trenutku po izklopu stikal sila rahlo naraste.



Slika 4.8: Potek signala in karakteristične točke tipke z dvojnim stikalom

Ker smo to meritev opravili na večjem številu elektronskih sklopov, in sicer 20, nas je zanimalo, koliko od teh sklopov je statistično najbolj ustrezalo specifikacijam. Na sliki 4.9 je prikazana Gaussova krivulja za vklop prvega stikala, to je vklopna točka A. Vidimo, da bi se vklopna točka A morala pojaviti največkrat nekje med 1,2 do 1,3 mm od začetne pozicije tipke. S slike 4.9 pa je razvidno, da se je pri večini merjencev vklopna točka pojavila že malo prej, med 1,1 in 1,2 mm. Razlog, da pa bi se naj, glede na Gaussovo krivuljo, ta točka zgodila drugje, pa je v tem, da smo merili elektronske sklope, ki so bili reklamirani in niso ustrezali zahtevam kupca. V dodatku C. dodajamo Gaussove krivulje še za preostale karakteristične točke merjenja.



Slika 4.9: Gaussova krivulja pri merjeni vrednosti vklopnih točk A 20 merjencev

4.6 Primerjava različnih merilnih sistemov

V tabeli 6 podajamo primerjavo treh različnih sistemov s trga s sistemom, ki smo ga razvili v okviru tega diplomskega dela. Slika 4.10 prikazuje vse tri sisteme s trga in naš sistem.

Tabela 6: Primerjava različnih merilnih sistemov z našim

	Mecmesin MultiTest 2.5-d[18]	Tinius Olsen H1k [20]	Mark-10 EMS301[19]	Naš sistem
Maksimalna sila [N]	2500	1000	1500	50
Maksimalna razdalja [mm]	350	445	317	330
Resolucija sile [N]	0,01	Ni podatka	0,1	0,01
Resolucija razdalje [mm]	0,01	0,0001	0,01	0,1
Hitrost [mm/min]	1-750	0.001 - 1000	0,5-1100	0,005-600
Digitalni izpis	LCD	LCD	LCD	LCD
Poraba moči [W]	160	2000	Ni podatka	15
Komunikacija s PC	RS-232	RS-232	RS-232	RS-232
Motor	Ni podatka	Ni podatka	Koračni	Koračni

Že na prvi pogled lahko vidimo, da je največja prednost obstoječih sistemov v maksimalni sili, ki jo sistemi omogočajo, saj od našega omogočajo tudi do 50 – krat večjo merjeno silo. Razlog za takšno razliko v maksimalni merjeni sili je v tem, da smo mi uporabili že obstoječi merilnik sile, ki omogoča samo tako veliko maksimalno merjeno silo, kot določa njegova specifikacija. Omogoča pa zato kar veliko resolucijo merjenja sile. Ker smo merilnik razdalje lahko izbrali sami in smo se odločili za linearni optični dajalnik, maksimalna merjena razdalja ne odstopa toliko od ostalih merilnih sistemov. Razlog, da je resolucija meritve razdalje v našem sistemu slabša kot v ostalih pa je v tem, da je predstavljala takšna resolucija dovolj veliko natančnost merjenja tipk za naše potrebe. Maksimalna hitrost pomikanja »glave« merilnega sistema je sicer v našem sistemu nižja kot v ostalih, je pa minimalna hitrost manjša kot v dveh in nam s tem omogoča boljši nadzor in natančnost meritve. Velika prednost našega sistema pred ostalimi pa je veliko manjša poraba moči. Poleg veliko manjše porabe moči je prednost našega sistema pred ostalimi tudi ta, da lahko nanj električno priključimo izhod merjenca. Tako ne zmerimo le mehansko karakteristiko tipk ampak tudi dejanski vklop oziroma izklop. Pri strojih, ki uporabljajo elektromehanske tipke, je že sama izmerjena karakteristika tipke dovolj, da se

da ugotoviti, kdaj je tipka vklopila oziroma izklopila (slika 2.4). Pri strojih, ki pa uporabljajo drugačne vrste tipke, kot je na primer tipka z magnetno-mehanskim (reed rele) ali elektro-magnetnim (Hallova sonda) preklopom, kjer se vklop oziroma izklop ne zgodi zgolj mehansko ampak elektro-magnetno-mehansko pa zgolj izmerjena karakteristika tipke ni dovolj, ampak je potrebno tudi izmeriti, kdaj se je dejansko preklon električno zgodil. V takšnih primerih predstavlja naš sistem bistveno prednost pred ostalimi. Čeprav so sistemi s trga lahko uporabljeni za veliko različnih meritev, naš pa je bil izdelan zelo namensko, lahko strojno in programsko opremo našega sistema enostavno priredimo še drugim meritvam te vrste.



Slika 4.10: Mecmesin MultiTest 2.5-d, Tinius Olsen H1k, Mark-10 ESM301, naš sistem

5. SKLEP

Rezultat našega razvojnega dela je bila izdelava merilnega sistema za merjenje sile pritiska in položaja tipk pri strojih.

Ker smo za merilnik sile vedeli kakšnega bomo uporabili, je bilo potrebno ugotoviti kakšen merilnik razdalje bi bil najbolj ustrezen za naše potrebe. Prvotno smo želeli uporabiti kljunasto merilo, ki omogoča priključitev preko zaporedne povezave. A smo ugotovili, da bi bila mehanska namestitev preveč okorna in bi zelo težko zagotovili natančno meritev premika tipk. Tako smo se odločili za linearni optični dajalnik, ki ga lahko relativno enostavno priključimo na mikrokrmilnik. Poleg tega ima dajalnik podobne mere, kot jih ima vpenjalno stojalo. Zaradi tega smo ga lahko na enostaven način namestili na celotni sistem. Kot se je kasneje pokazalo, je največ težav povzročala mehanska namestitev koračnega motorja, saj smo ga morali zelo natančno namestiti na os vpenjalnega stojala. Če je bila ta namestitev nenatančna se je motor pogosto ustavljal. To težavo bi lahko rešili s postavitvijo koračnega motorja preko zobnikov na zunanji del vpenjalnega stojala, ki je namenjen za ročno upravljanje. S tem bi razbremenili koračni motor in zelo povečali ločljivost simulacije pritiska na merjeno tipko.

Načrtovanje in izdelava krmilnega dela ni predstavljala večjih težav, saj nismo bili omejeni niti z velikostjo tiskane ploščice niti z izbiro vrste elementov. Poleg tega smo uporabili še predhodne izkušnje za programiranje zaporedne komunikacijske povezave. Glede na to, da je to vrsto komunikacije z osebnim računalnikom že skoraj izrinila USB komunikacija pa bi bilo v nadaljnjem razvoju smotrno merilni sistem nadgraditi s to vrsto zaporedne komunikacije. V ta namen bi lahko uporabili vezje za pretvorbo RS232 komunikacije v USB ali pa celo mikrokrmilnik z že vgrajeno USB komunikacijo.

Z nadzornim sistemom smo imeli težave le v začetku, in sicer zaradi nepoznavanja programskega orodja LabVIEW.

S pomočjo sistema, ki smo ga razvili v okviru tega diplomskega dela, lahko torej izvajamo naslednje vrste meritev:

- meritev tipke z enojnim preklopom

- meritev tipke z dvojnimi preklopom
- meritve tipk z ne-mehanskimi preklopom
- meritev karakteristik tlačnih vzmeti
- meritev ovitja tiskanih vezij in LCD zaslonov

Merilni sistem za merjene sile pritiska in položaja tipk pri strojih, smo zasnovali tako, da ga lahko v prihodnje prilagodimo za kakršne koli vrste meritev, kjer merimo sile pritiska na določene sestavne dele naprav in njihovega položaja do natančnosti 0,01N in 100 μ m.

6. VIRI, LITERATURA

- [1] M. Solar, Meritve v elektroniki, 1. izdaja, Založništvo FERI, Maribor, 2001.
- [2] Atmel Corporation, Atmel 8-bit AVR ATmega128, 2007.
- [3] WinAVR, <http://winavr.sourceforge.net/>, 2006.
- [4] AVR Studio, http://www.atmel.com/dyn/products/tools_card.asp?tool_id=2725, 2008.
- [5] *Atmega128*, http://www.atmel.com/dyn/resources/prod_documents/doc2467.pdf, 2007.
- [6] Krmiljenje koračnih motorjev, http://www2.arnes.si/~sspslavr/k_motor/k_motor.html, 2004.
- [7] Winstar LCD WH1602, http://www.winstar.com.tw/products_detail_ov.php?ProID=21, 2008.
- [8] National Instrument LabVIEW, <http://www.ni.com/labview/>, 2008.
- [9] Stepper Motor Basics, <http://www.solarbotics.net/library/pdflib/pdf/motorbas.pdf>, 2001.
- [10] ST Microelectronics L297, <http://www.st.com/stonline/products/literature/ds/1334.pdf>, 2001.
- [11] ST Microelectronics L298, <http://www.st.com/stonline/products/literature/ds/1773.pdf>, 2001.
- [12] MAXIM Fundamentals of RS232 Serial Communications, <http://pdfserv.maxim-ic.com/en/an/AN83.pdf>, 2001.
- [13] Iskra TELA merilne letve, http://www.iskra-tela.si/img/pc/merilne_letve.pdf, 2008.
- [14] TRINAMIC QMOT Motor QSH4218, http://www.trinamic.com/tmc/media/Downloads/QMot_motors/QSH4218/QSH4218_manual.pdf, 2007.
- [15] INKREMENTALNI DAJALNIK
http://lrt2.fe.uni-lj.si/lrtme/meri_pret/Andrej_Velikonja_seminar_Absolutni_dajalniki_zasuka.pdf, 2008
- [16] FG-5000A
<http://www.pedak.nl/Lutron/pdf/FG-5000A.pdf>
- [17] PRERAČUN TLAČNIH VZMETI
<http://www2.lecad.si/documents/seminarji/resitve/3.6/index.html>, 1999

- [18] MECMESIN Multitest-d
<http://www.mecmesin.com/ucm/admin/upload/documents/2MultiTest-dBrochure01EQ.pdf>
- [19] MARK-10 ESM301
http://www.mark-10.com/new_manuals/manualESM301.pdf
- [20] Tinius Olsen H1K
<http://www.tiniusolsen.com/pdf/BenchTop.pdf>
- [21] Load Cell
<http://www.omega.com/literature/transactions/volume3/load.html>

7. PRILOGE

Seznam prilog:

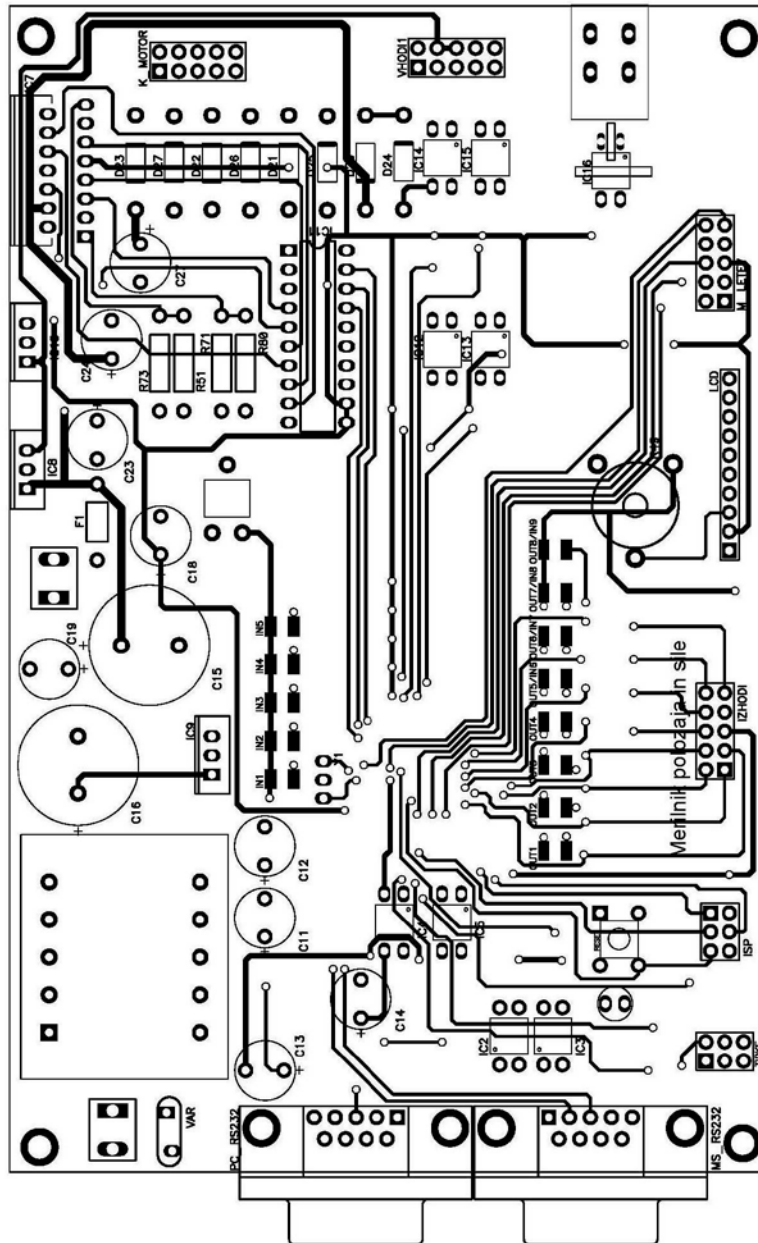
- A. Vsebina priložene zgoščenke
- B. Shema vezja in predloge tiskanega vezja
- C. Grafi gaussovih krivulj za vse karakteristične točke merjenih tipk z dvojnimi stikalom
- D. Legenda blokov uporabljenih v LabVIEW - u
- E. Naslov študenta
- F. Kratek življenjepis

A. Vsebina priložene zgoščenke

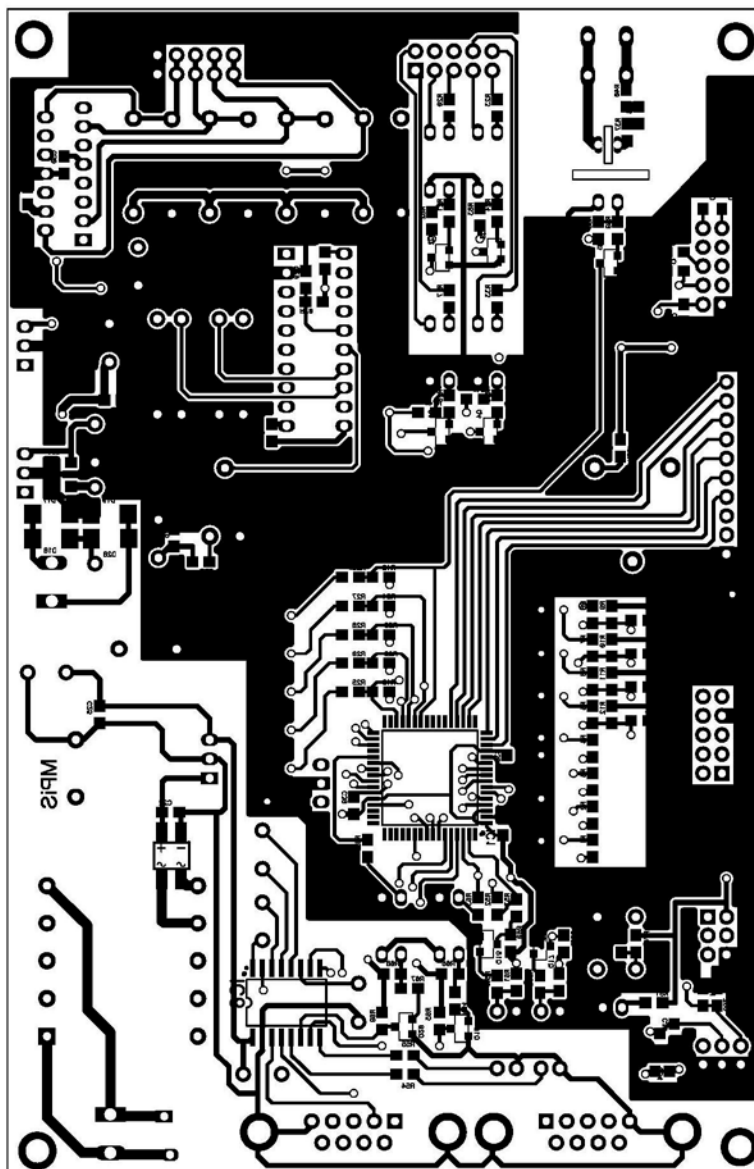
- Diplomaska naloga v .pdf formatu (»diploma.pdf«)
- Diplomaska naloga v .doc formatu (»diploma.doc«)
- Shema vezja v .pdf formatu (mapa »doc«)
- TIV v .pdf formatu (mapa »PCB«)
- Tehnični podatki, pridobljeni iz spleta v .pdf formatu (»tehnični podatki«)

B. Shema vezja in predloga tiskanega vezja

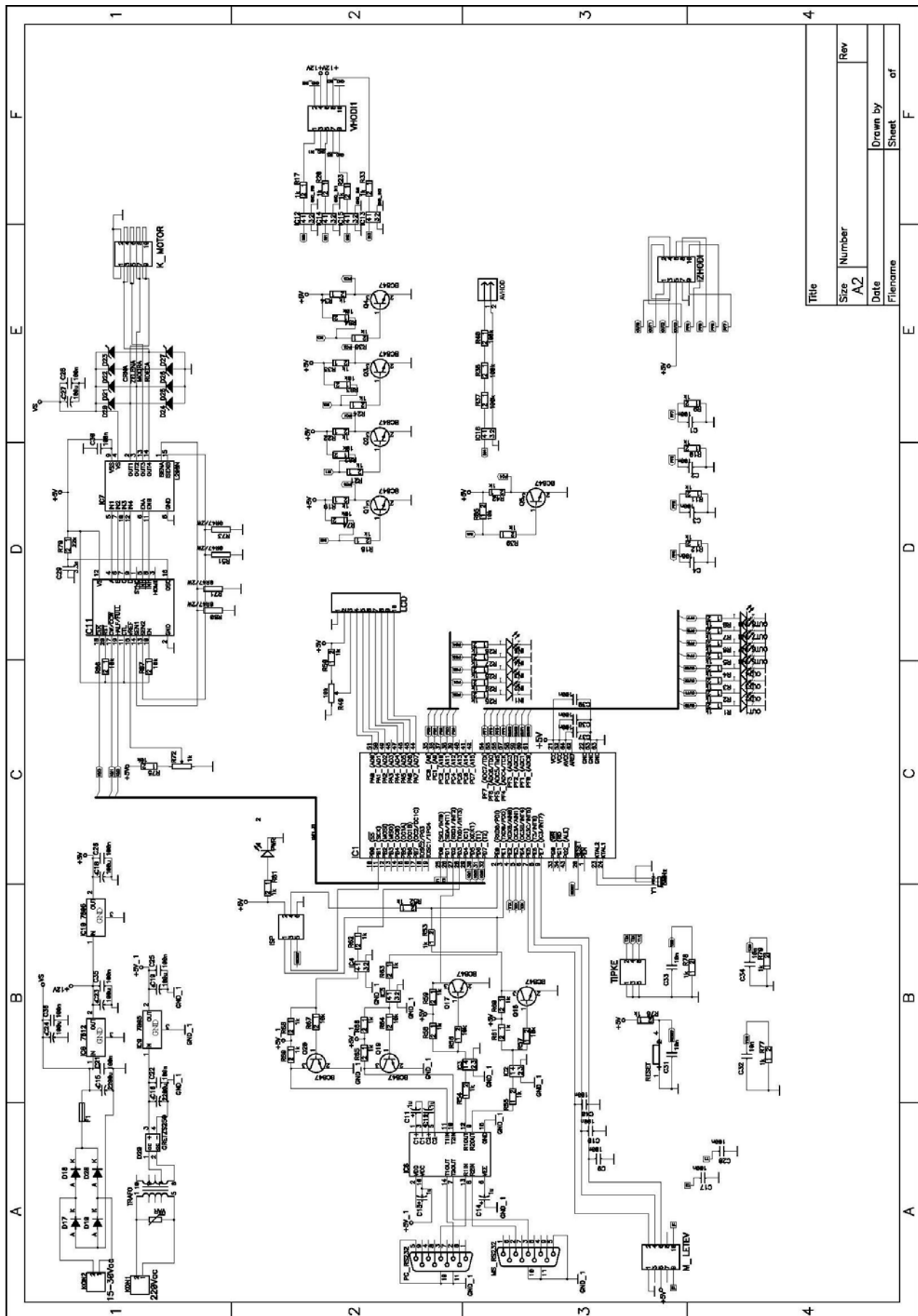
Predloga TIV- zgoraj



Predloga TIV- spodaj



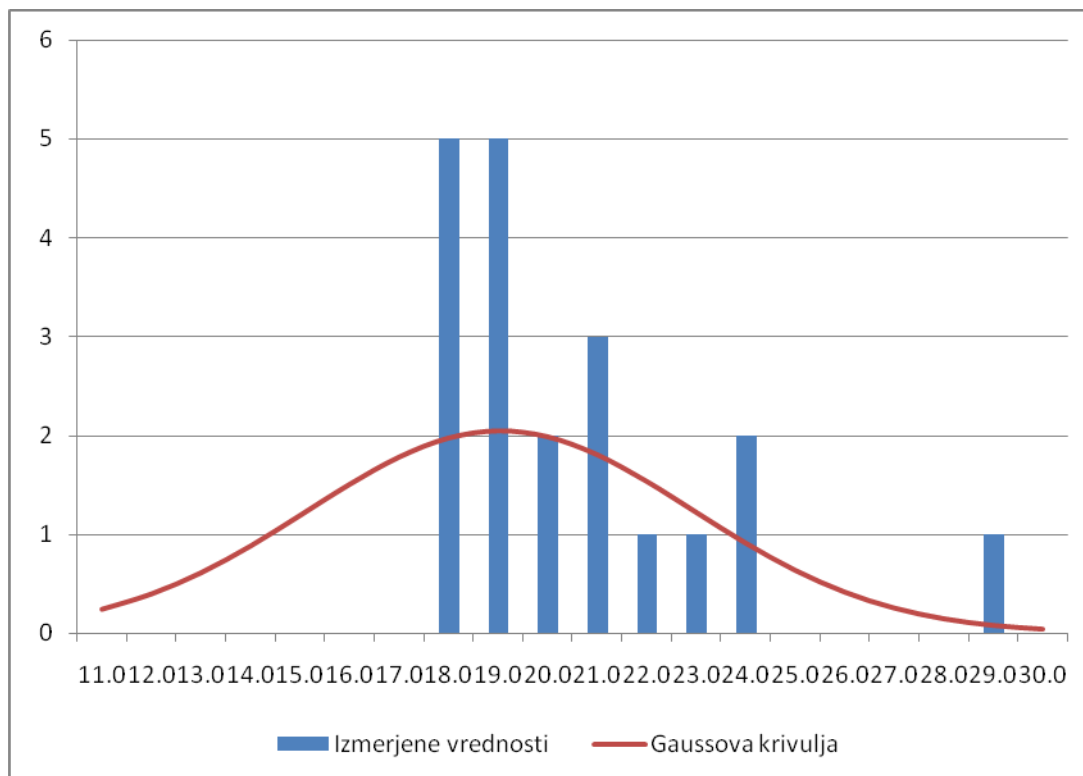
Schema krmilnika



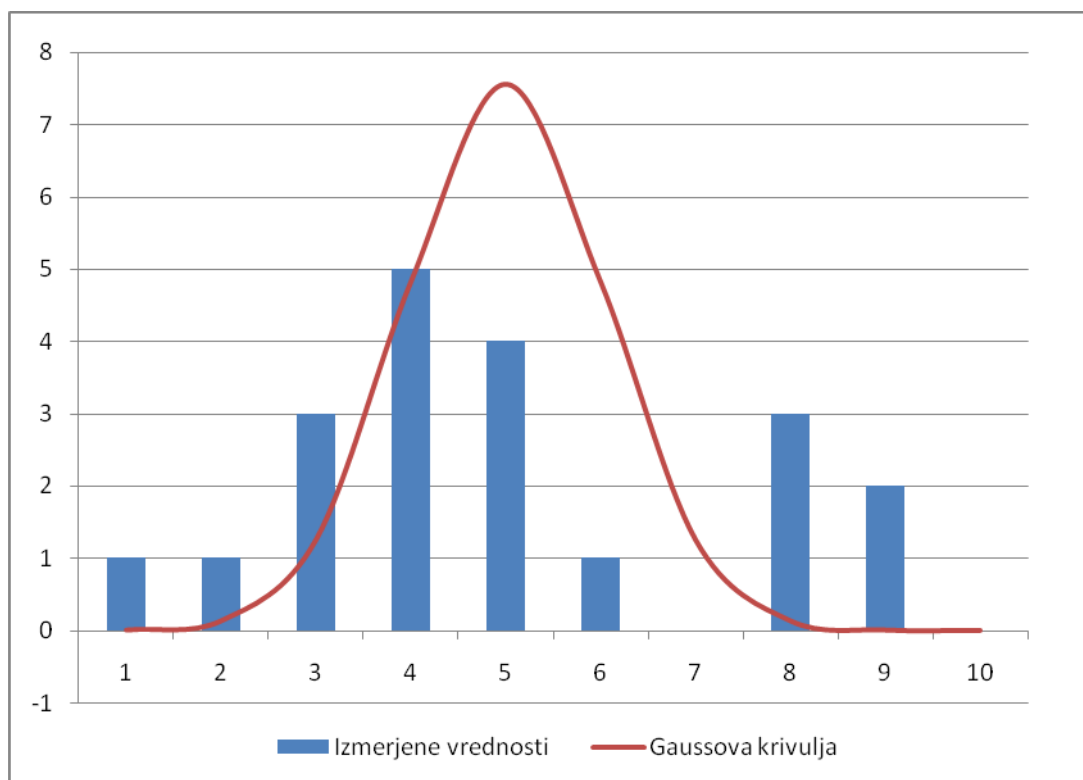
Title	
Size	Number
A2	
Date	Rev
Filename	Drawn by
	Sheet
	of

C. Grafi gaussovih krivulj za vse karakteristične točke merjenih tipk z dvojnimi stikalom

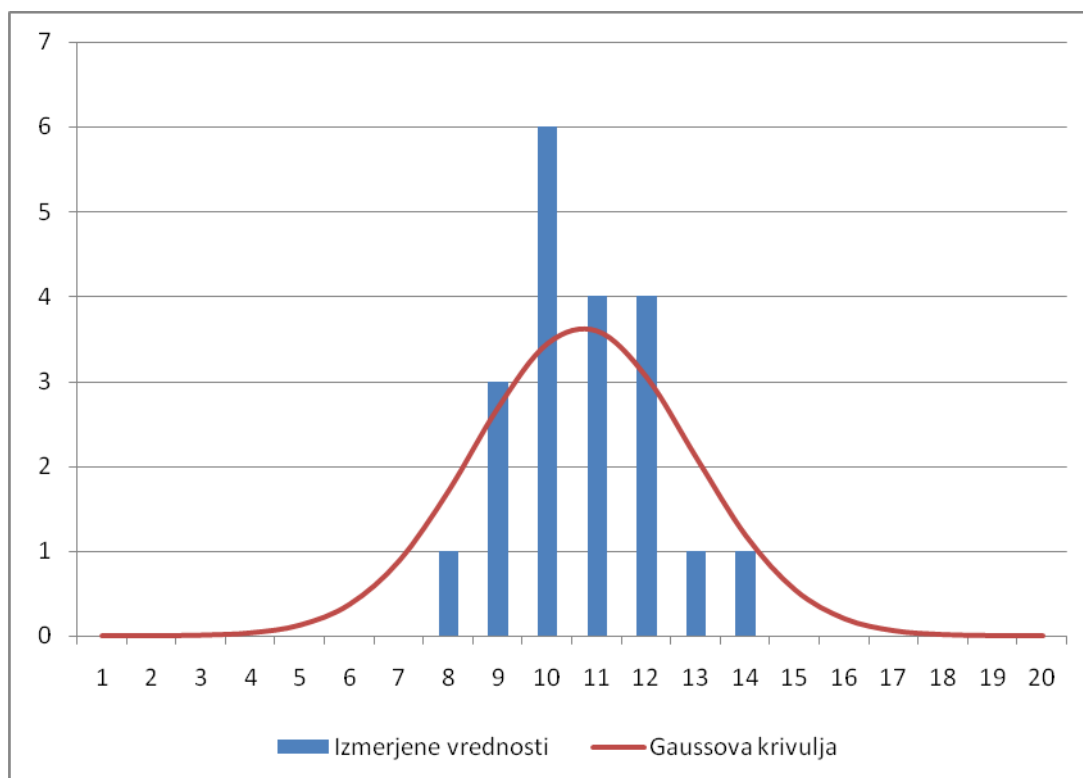
Vklopna točka B.



Izkopna točka A.




Izkopna točka B.




D. Legenda blokov uporabljenih v LabVIEW - u


- MOTOR Menu



→ skupina več elementov




→ pretvorba skupine v polje




→ preverjanje enodimenzionalnega polja


COMport




→ tip izvora vhodnega in izhodnega niza




→ nastavitve VISA serijske povezave




→ pošiljanje signala preko VISA serijske povezave




→ branje signala iz VISA serijske povezave




→ zapiranje VISA serijske povezave




→ 32 bitno decimalno število




→ 64 bitno realno število




→ string



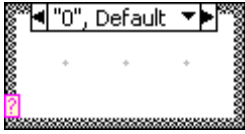
→ iskanje znaka v nizu



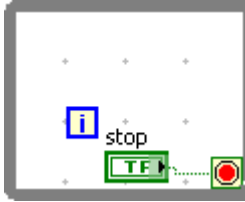
→ pretvorba decimalnega niza v število



→ deljenje



→ »case« pogojni stavek



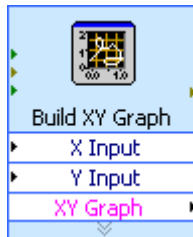
→ »while« zanka



→ pretvorba v dinamični podatkovni tip



→ združevanje signalov



→ prirejanje podatkov izrisanih na XY graf

E. Naslov študenta

Gregor Steblovnik

Rečica ob Paki 37

3327 Šmartno ob Paki

Tel.: +386 31 344 226

E-pošta: gregor.steblovnik@triera.net

F. Kratek življenjepis

Rojen: 12. 11 1981 Celje

Šolanje: 1988–1996 Osnovna šola Bratov Letonja Šmartno ob Paki

1996–2000 I. Gimnazija v Celju