Programsko rješenje za razmjenu podataka između frekvencijskog pretvarača i SIMATIC 1500 PLC-a temeljeno na Modbus RTU/RS 485 protokolu

Šarić, Luka

Master's thesis / Diplomski rad

2018

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **Josip Juraj Strossmayer University of Osijek, Faculty of Electrical Engineering, Computer Science and Information Technology Osijek / Sveučilište Josipa Jurja Strossmayera u Osijeku, Fakultet elektrotehnike, računarstva i informacijskih tehnologija Osijek**

Permanent link / Trajna poveznica: https://urn.nsk.hr/urn:nbn:hr:200:905111

Rights / Prava: In copyright / Zaštićeno autorskim pravom.

Download date / Datum preuzimanja: **2024-04-25**

Repository / Repozitorij:

Faculty of Electrical Engineering, Computer Science and Information Technology Osijek

SVEUČILIŠTE JOSIPA JURJA STROSSMAYERA U OSIJEKU

FAKULTET ELEKTROTEHNIKE, RAČUNARSTVA I INFORMACIJSKIH TEHNOLOGIJA

Sveučilišni studij

PROGRAMSKO RJEŠENJE ZA RAZMJENU PODATAKA IZMEĐU FREKVENCIJSKOG PRETVARAČA I SIMATIC 1500 PLC-a TEMELJENO NA MODBUS RTU/RS 485 PROTOKOLU

Diplomski rad

Luka Šarić

Osijek, 2018.

FERIT
FAKULTET ELEKTROTEHNIKE, RAČUNARSTVA I INFORMACIJSKIH TEHNOLOGIJA OSIJEK

Obrazac D1: Obrazac za imenovanje Povjerenstva za obranu diplomskog rada

Osijek, 18.09.2018.

Odboru za završne i diplomske ispite

Imenovanje Povjerenstva za obranu diplomskog rada

Ime i prezime studenta:	Luka Šarić
Studij, smjer:	Diplomski sveučilišni studij Elektrotehnika
Mat. br. studenta, godina upisa:	D 932, 25.09.2017.
OIB studenta:	66521277019
Mentor:	Doc.dr.sc. Tomislav Keser
Sumentor:	
Sumentor iz tvrtke:	Filip Belić
Predsjednik Povjerenstva:	Doc.dr.sc. Tomislav Rudec
Član Povjerenstva:	Izv. prof. dr. sc. Alfonzo Baumgartner
Naslov diplomskog rada:	Programsko rješenje za razmjenu podataka između frekvencijskog pretvarača i SIMATIC 1500 PLC-a temeljeno na Modbus RTU/RS 485 protokolu
Znanstvena grana rada:	**Procesno računarstvo (zn. polje računarstvo)**
Zadatak diplomskog rada:	Korištenjem SIMATIC 1500 PLC-a s dodatnim RS485 komunikacijskim modulom potrebno je izraditi programsko rješenje koje omogućava dvosmjernu komunikaciju s frekvencijskim pretvaračem. Frekvencijski pretvarač pokreće asinkroni motor te je potrebno očitavati trenutne vrijednosti struje, snage i brzine vrtnje motora za vrijeme rada.
Prijedlog ocjene pismenog dijela ispita (diplomskog rada):	Izvrstan (5)
Kratko obrazloženje ocjene prema Kriterijima za ocjenjivanje završnih i diplomskih radova:	Primjena znanja stečenih na fakultetu: 3 bod/boda Postignuti rezultati u odnosu na složenost zadatka: 2 bod/boda Jasnoća pismenog izražavanja: 3 bod/boda Razina samostalnosti: 3 razina
Datum prijedloga ocjene mentora:	18.09.2018.

Potpis mentora za predaju konačne verzije rada u Studentsku službu pri završetku studija:	Potpis:
	Datum:

FERIT
FAKULTET ELEKTROTEHNIKE, RAČUNARSTVA
I INFORMACIJSKIH TEHNOLOGIJA **OSIJEK**

IZJAVA O ORIGINALNOSTI RADA

Osijek, 26.09.2018.

Ime i prezime studenta:	Luka Šarić
Studij:	Diplomski sveučilišni studij Elektrotehnika
Mat. br. studenta, godina upisa:	D 932, 25.09.2017.
Ephorus podudaranje [%]:	1

Ovom izjavom izjavljujem da je rad pod nazivom: **Programsko rješenje za razmjenu podataka između frekvencijskog pretvarača i SIMATIC 1500 PLC-a temeljeno na Modbus RTU/RS 485 protokolu**

izrađen pod vodstvom mentora Doc.dr.sc. Tomislav Keser

i sumentora

moj vlastiti rad i prema mom najboljem znanju ne sadrži prethodno objavljene ili neobjavljene pisane materijale drugih osoba, osim onih koji su izričito priznati navođenjem literature i drugih izvora informacija.
Izjavljujem da je intelektualni sadržaj navedenog rada proizvod mog vlastitog rada, osim u onom dijelu za koji mi je bila potrebna pomoć mentora, sumentora i drugih osoba, a što je izričito navedeno u radu.

Potpis studenta:

ZAHVALA

Prvo bih se zahvalio svome mentoru doc. dr. sc. Tomislavu Keseru na bezuvjetnom pomaganju, strpljenju i sugestijama pri izradi ovog diplomskog rada.

Zahvaljujem se i tvrtki Teo – Belišće d.o.o. i sumentoru Filipu Beliću, te Luki Gostinskom na omogućavanju potrebne opreme, ali i velikom strpljenju na moja mnogobrojna pitanja.

Zahvalio bih se svim svojim prijateljima, koji su mi kroz fakultetske godine učinili studiranje lakšim i zabavnijim.

Posebne zahvale idu cijeloj mojoj obitelji koja je uvijek bila uz mene i bila mi bezuvjetna podrška. Hvala i mojoj baki što mi je bila potpora.

Za kraj, najveće HVALA mojim roditeljima, a posebno majci, na potpori, strpljenju i ljubavi u sretnim trenucima i onim tužnim. Hvala joj što mi je svojim sugestijama uvijek pokušavala dati do znanja da je ona UVIJEK uz mene i da ću pomoć dobiti od nje bez obzira na sve.

Veliko Vam HVALA svima!

SADRŽAJ

1. UVOD 1

1.1. Opis i zadatak rada 3

2. SUSTAVI UPRAVLJANJA FREKVENCIJSKIM PRETVARAČIMA 4

2.1. PLC 4

2.2. SCADA 5

2.3. Komunikacija 9

2.4. Frekvencijski pretvarač 13

2.5. Asinkroni motor 15

3. REALIZACIJA SUSTAVA UPRAVLJANJA 19

3.1. Korišteni programski alati 19

3.2. Korišteni PLC i frekvencijski pretvarač 23

3.3. MODBUS protokol pri realizaciji sustava 32

3.4. Integracija sustava 35

4. TESTIRANJE 39

4.1. Rezultati testiranja u realnom pogonu 39

4.2. Analiza realnog pogona 42

5. ZAKLJUČAK 44

LITERATURA 45

SAŽETAK 46

ABSTRACT 47

ŽIVOTOPIS 48

PRILOG 1. FBD DIO KODA U TIA PORTALU 49

PRILOG 2. SCL DIO KODA U TIA PORTALU 56

PRILOG 3. STATIČKE VARIJABLE KORIŠTENE U PROGRAMU 61

PRILOG 4. PLC TAGOVI 72

PRILOG 5. HMI SUČELJE NA ENGLESKOM JEZIKU 73

PRILOG 6. HMI SUČELJE NA NJEMAČKOM JEZIKU 74

PRILOG 7. HMI TAGOVI 75

1. UVOD

Od početaka čovjekovog vremena nastoje se što više olakšati bilo kakve radnje. U početku je to bila izrada oruđa, gdje je čovjek koristio samo ono što je našao u prirodi. Nakon toga su se počele otvarati prve radionice u kojima je korištena pomoć životinja i raznih jednostavnih alata. Tako je u 3. st. pr. Kr. grčki inženjer Ctesibius izumio vodeni sat, što je bio prvi kakav takav oblik automatizacije.

Uslijedila je postupna mehanizacija uvođenjem alata s ciljem olakšavanja posla čovjeku, a sve se to odvijalo, isprva, u manufakturama te poslije u tvornicama. U 17. st nizozemski znanstvenik C. Drebbel izumio je termostat. Sredinom 18.st E. Lee je patentirao kontrolni mehanizam za pokrivanje lopatica vjetrenjače, dok je J. De Vaucanson patentirao automatizirani stroj za tkanje. Pretkraj tog stoljeća, točnije 1771., R. Arkwright izumio je prvi potpuno automatizirani vodeni mlin, dok je 14 godina poslije O. Evans izumio automatizirani mlin koji je mljeo brašno i tako stvorio prvi automatizirani industrijski proces.

Daljnjim razvojem tehnologije nastoji se čovjekova prisutnost svesti na minimum, a sve njegove poslove bi preuzeli razni strojevi koji bi se automatizirali, te bi se tako stvorio učinkovitiji tehnički proces. Stroj koji je nadogradio proces automatizacije je parni stroj J. Watta koji je preuzeo izum centrifugalnog regulatora od C. Huygensa (17. st.). Regulatoru je veću važnost dao J. C. Maxwell u svom radu o teorijskoj bazi za razumijevanje upravljanja. Tijekom 1920-ih izumom elektroničkog pojačala riješen je problem negativne povratne veze, vrlo bitnog dijela teorije upravljanja. 1940-ih i 1950-ih njemački matematičar Flugge-Lotz osmislio je teoriju diskontinuiranog automatskog upravljanja koje je svoju primjenu pronašlo u sustavima navigacije i upravljanja paljbom tijekom 2. svjetskog rata.

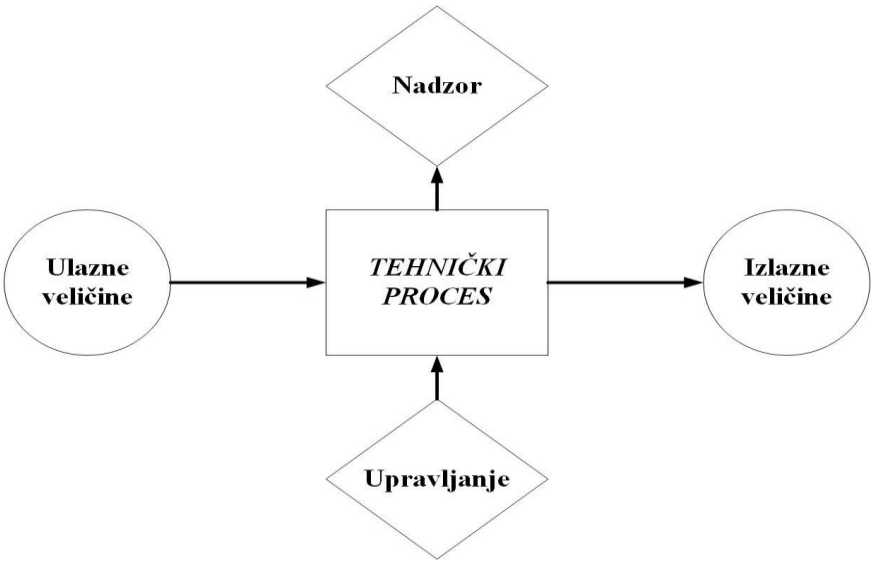
Tijekom elektrifikacije tvornica, između 1900-ih i 1920-ih, relejna logika se uvelike počela koristiti. To je zahtijevalo dosta prostora zbog velikih ormara i širenja postrojenja, te su se 1920-ih centralne upravljačke sobe počele graditi, no sve do ranih 1930-ih većina procesa se koristila tehnikom „upali-ugasi“. Operatori su pratili zbivanja u procesu uz pomoć snimača koji su iscrtavali grafove, no da bi napravili neke ispravke morali su zatvarati ventile ili isklapati prekidače. Centralne upravljačke sobe su imali kodirana svjetla koja bi pomogla pri održavanju

određenih promjena ručno. Godine 1959. Port Arthur rafinerija tvrtke Texaco je postala prvo postrojenje koje je koristilo digitalno upravljanje, a to je bio tek početak, [1, 2].

S obzirom na to da je 1970-ih cijena sklopovske podrške računala bila u padu, u industrijskim se postrojenjima počinju koristiti programabilni logički kontroleri (skraćeno PLC – *eng. Programmable Logical Controller*). Najveća prednost PLC-ova je bila to što su mogli zamijeniti na tisuće releja, što je dovelo do uštede na sklopovlju, ali i na prostoru. Pošto se PLC mogao postaviti u blizini upravljanog uređaja, bilo je potrebno osmisliti najbolji način za upravljanje. To je postignuto centralnim nadzorom i upravljanjem, odnosno SCADA sustavom (*eng. Supervisory Control And Data Acquisition*). [3]

Dakle, automatizacija je tehnologija koja zamjenjuje ljudsku snagu. Postupkom automatizacije nekog postrojenja moguće je povećati proizvodnju, poboljšati kvalitetu proizvodnje (isključiti mogućnost čovjekove pogreške) i stvoriti učinkovitiji nadzor i upravljanje, no to za sobom povlači veliku manu – smanjenje radnih mjesta. Gotovo svako postrojenje u svijetu sadrži neki oblik upravljanja, a sva postrojenja sadrže upravljanje motorima.

Postrojenja, u kojima je barem dio automatiziran, nazivaju se tehnički sustavi. Prema [4] skup događanja unutar tehničkog sustava je tehnički proces, koji je ujedno i najbitnija komponenta tehničkog sustava. U tehničkom se procesu može odvijati proizvodnja, pretvorba energije ili obrada informacija. Slikom 1.1. bit će prikazana blok shema tehničkog sustava.



Slika 1.1. *Blok shema tehničkog sustava.*

Diplomski rad će biti napravljen baziran na prethodnoj blok shemi. Ulazne veličine u ovom diplomskom radu predstavljat će napon. Tehnički proces bit će sadržan u elektromotoru gdje će se električna energija pretvarati u mehaničku. Upravljanje tehničkim procesom izvršit će se pomoću programabilnog logičkog kontrolera, dok će se nadzor izvršiti preko grafičkog sučelja od strane čovjeka. Izlazne veličine ovog procesa bit će brzina vrtnje, snaga i struja. Također, bitna komponenta diplomskog rada bit će i komunikacija kojom će se vršiti upravljanje.

1.1. Opis i zadatak rada

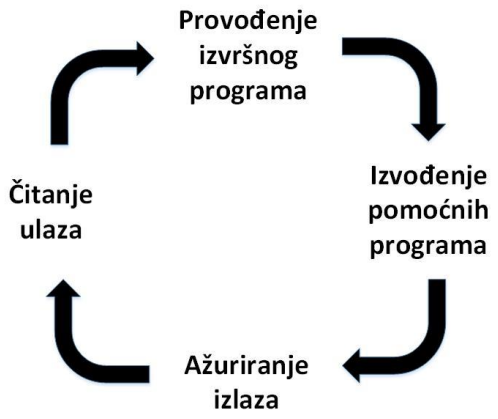
Unutar ovog rada želi se napraviti upravljanje i nadzor jednog dijela postrojenja – frekvencijskog pretvarača i asinkronog motora. To će se postići uz pomoć Siemensovog programabilnog logičkog kontrolera SIMATIC 1500 koji će se spojiti s frekvencijskim pretvaračem EURA E800 uz pomoć komunikacijskog protokola MODBUS RTU/RS485. Upravljanje i nadzor će se odvijati na grafičkom (HMI) sučelju, te će biti moguće očitavati trenutne vrijednosti struje, snage i brzine vrtnje motora za vrijeme rada.

2. SUSTAVI UPRAVLJANJA FREKVENCIJSKIM PRETVARAČIMA

Ovim poglavljem proći će se kroz teoriju svih sastavnica diplomskog rada. U prvom potpoglavlju opisat će se način rada PLC uređaja. Nastavit će se s općom teorijom SCADA sustava za nadzor, dok će treće potpoglavlje obuhvatiti teoriju komunikacije između PLC uređaja i frekvencijskog pretvarača. Upravo će princip rada frekvencijskog pretvarača biti opisan u četvrtom potpoglavlju, dok će sve zaokružiti princip rada i opis asinkronog motora.

2.1. PLC

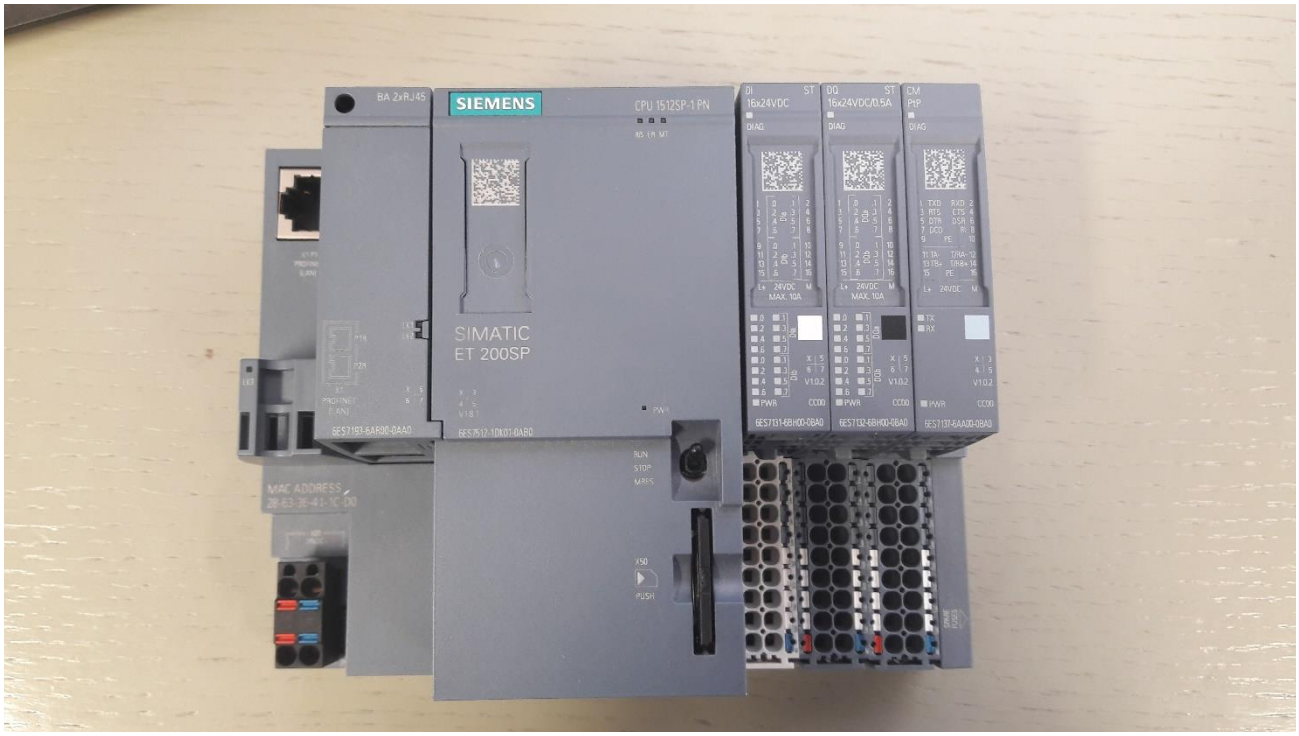
PLC je, zapravo, digitalni regulator koji obavlja elektronički uređaj s ciljem automatizacije i upravljanja. Ono svoj rad zasniva na mikroprocesorima koji nastoje obaviti određeni zadatak u stvarnom vremenu. Cikličko izvođenje programa osnovni je princip rada PLC-ova. Osnovne zadaće u cikličnom načinu rada prikazane su slikom 2.1.



Slika 2.1. *Ciklični način rada PLC-a.*

Gruba podjela dijelova PLC-ova sastoji se od upravljačke jedinice, čiju jezgru čini procesor s aritmetičkom jedinicom i memorijom, te periferijskih ulazno/izlaznih modula. Memoriju čini nepromjenjivi sadržaj, unutar kojeg se nalazi operacijski sustav (ROM), ali i promjenjivi (podatkovni) sadržaj unutar kojega korisnik upisuje željeni program (RAM). Zadnje navedena memorija podijeljena je na više dijelova: mjerači vremena, brojila, zastavice, stanje procesnih ulaza, stanje procesnih izlaza, podaci sustava, te korisnički program. Periferijski moduli mogu biti razni, a najviše je modula: digitalnih ulaza/izlaza, analognih ulaza/izlaza, brojila, mjerača vremena, dijagnostike, za regulaciju, te onih za pozicioniranje, [4].

Izgled PLC uređaja, i to Siemensovog PLC-a SIMATIC S7-1500, dan je sljedećom slikom.



Slika 2.2. *PLC SIMATIC S7-1500 s pripadajućim modulima.*

Kako bi se mogao isprogramirati PLC uređaj, tri jezika/načina su dostupna, uz još dodatni četvrti:

- relejna shema (*eng. Ladder diagram* – skraćeno LAD),
- lisni program (*eng. Statement list* – skraćeno STL),
- funkcijski blok dijagram (*eng. Function block diagram* – skraćeno FBD),
- strukturirano upravljani jezik (*eng. Structured Control Language* – skraćeno SCL).

Osim Siemensovog SIMATIC-a, popularni proizvođači PLC-ova su ABB, Schneider, Rockwell (Allen-Bradley), Mitsubishi, General Electrics i mnogi drugi.

2.2. SCADA

U automatiziranom sustavu bitno je provesti nadzor određenog procesa. To je moguće ostvariti uz pomoć sučelja čovjek-stroj (HMI) tako da se prikažu relevantni podaci i mjerne veličine bitne za taj sustav. Isprva sinoptički paneli, na kojima je bila prikazana shema tehnološkog procesa uz razne instrumente te svjetlosne i zvučne signale, zadovoljavali su samo dio potreba nadzora. Nemogućnost detaljnog prikaza zbivanja u procesu nadoknadio se nadzornim prostorijama s

velikim zrcalnim površinama u blizini procesa. Tu manu nadomjestila su suvremena računala na kojima je moguće prikazati detaljno proces, te osim toga prikazati i obraditi rezultate procesnih veličina i to sve na osobnim računalima u Windows okruženju. Sve naredbe moguće je poslati uz pomoć tipkovnice, miša ili *touch screen* panela.

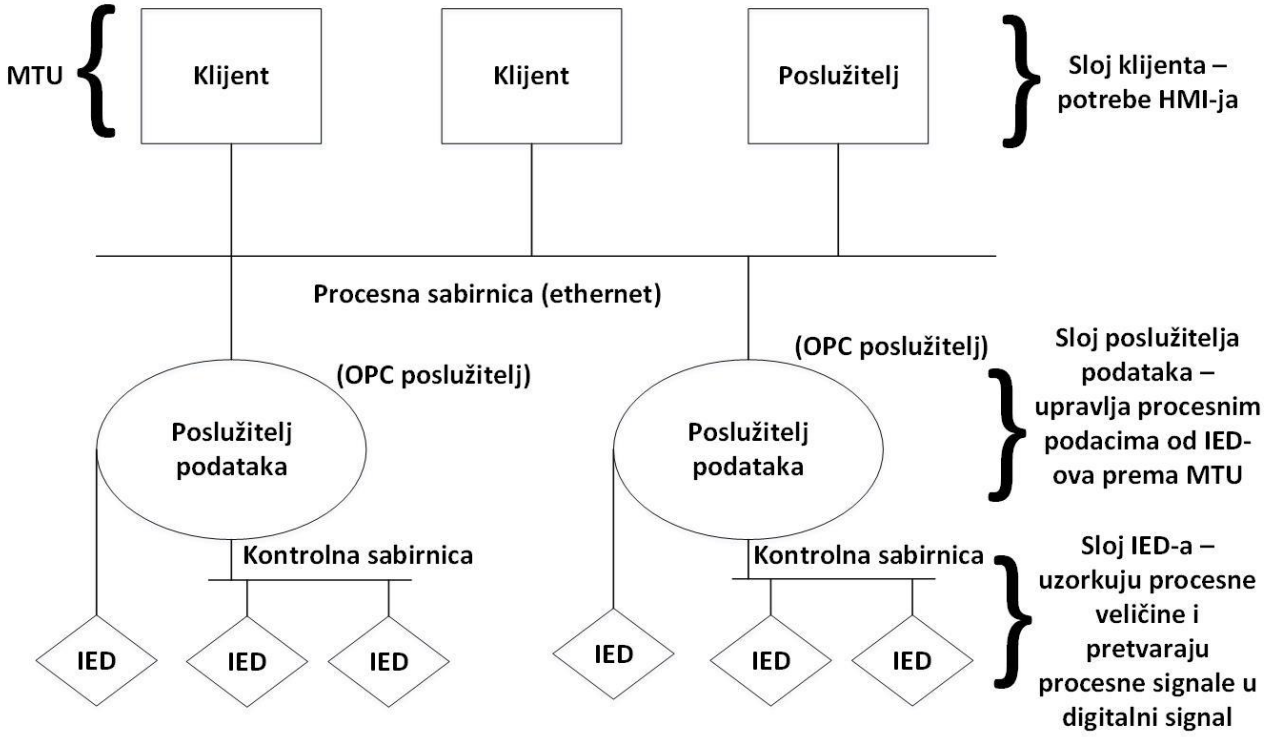
Nadzor procesa je moguće odraditi kroz tri različite strukture:

- centralna struktura – s jednim ili više računala,
- decentralna struktura – s više računala,
- hijerarhijska struktura – s više računala (posjeduje središnju i lokalnu nadzornu jedinicu).

Osim standardne računalne opreme za realizaciju nadzora procesa (računalo s visoko-rezolutnim monitorom, tipkovnicom, mišem) koriste se i *touch screen* paneli, svjetlosna pera, HDD-ovi, vanjska memorija, pisači, čitači magnetskih kartica, čitači otiska prsta, te operaterski paneli.

Problem unificiranja programske podrške nadzora riješen je izradom SCADA-e. Osim što služi za nadzor procesa, s njom je moguće i prikupljati razne podatke, stvaranje baze podataka za pohranu i čuvanje vrijednosti procesnih veličina, obrada podataka, registracija alarma i raznih događaja te izvještaj, kao i grafički prikaz stanja sustava uz njegovo rukovanje, [4].

SCADA-ina arhitektura može biti sklopovska i programska. Sklopovska arhitektura bazira se na radu sklopovlja s kojima komunicira, a tu su uglavnom tzv. pametni elektronički uređaji (*eng. Intelligent Electronic Devices* – skraćeno IED). Osim IED-a SCADA sadrži još dvije komponente: MTU (*eng. Master Terminal Unit*) i RTU (*eng. Remote Terminal Unit*). IED pruža sučelje u podatkovnoj komunikaciji sa SCADA-om. Svrha MTU komponente je preuzimanje, arhiviranje, obrađivanje i prosljeđivanje svih prikupljenih podataka te predstavlja središnje računalo SCADA sustava. RTU su uređaji koji prikupljaju podatke s IED-ova. Sklopovsku strukturu moguće je vidjeti na slici 2.3.



Slika 2.3. *Sklopovska struktura SCADA-e.*

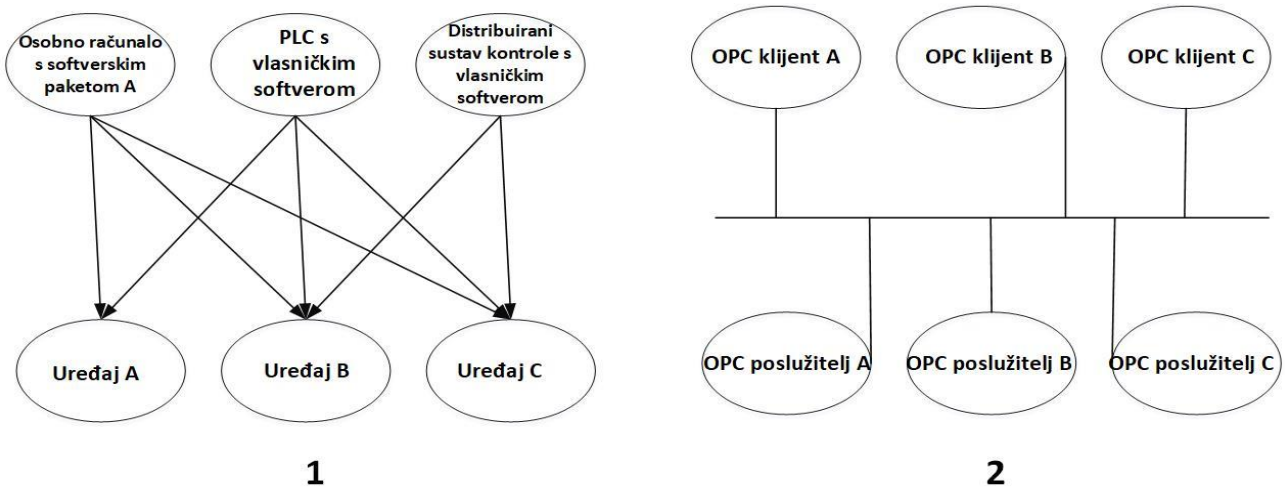
Što se tiče programske arhitekture SCADA sustava, može se reći da je ona dosta složena pošto korisniku omogućuje mnoga sučelja (HMI, planiranje, analizu, razne Windows aplikacije itd.), omogućuje sučelja s puno perifernih jedinica nadzorne jedinice i jedinica za automatizaciju, kao i mnoge komunikacijske protokole za razna povezivanja s IED-ovima.

Također, vrlo bitne karakteristike SCADA sustava jesu skaliranje i zalihost. Prvi pojam označava mogućnost proširenja SCADA-e dodavanjem procesnih veličina ili poslužitelja, dok drugi označava višak obavijesti u komunikaciji kako ne bi došlo do pogreške u sustavu.

SCADA posjeduje korisničko i programsko sučelje. Korisničko sučelje ima tri funkcije prikaza, a to su prikazi: trenutnog stanja procesa, povijesti događanja u procesu i kretanja procesnih veličina, te alarma i događanja u procesu. U velikim i skupim SCADA sustavima korisničko sučelje može imati četiri dijela: operatorsko sučelje, sučelje za održavanje, sučelje za posredovanje sa sustavom

za automatsko vođenje procesa te poslovno sučelje. U praksi se najčešće koriste operatorsko (HMI-P) i inženjersko (HMI-E) sučelje, [4].

Programsko sučelje SCADA-e bazirano je oko OPC-a (*eng. Open Process Control*) koji povezuje programske komponente između poslužitelja koji prikuplja podatke i IED-ova koji daju te podatke. Pomoću OPC-a moguće je vrlo jednostavno povezivanje uređaja. Razlika između povezivanja unutar sustava za automatsko vođenje procesa bez i s OPC-om vidljivo je slikom 2.4.



Slika 2.4. *Povezivanje unutar sustava za automatsko vođenje procesom bez (1) i s OPC-om (2).*

OPC predstavljaju zapravo naredbe koje su prikupljene u programsku biblioteku (*.dll*), te se mogu povezati s aplikacijama klijenata. Sastoji se iz tri glavne komponente:

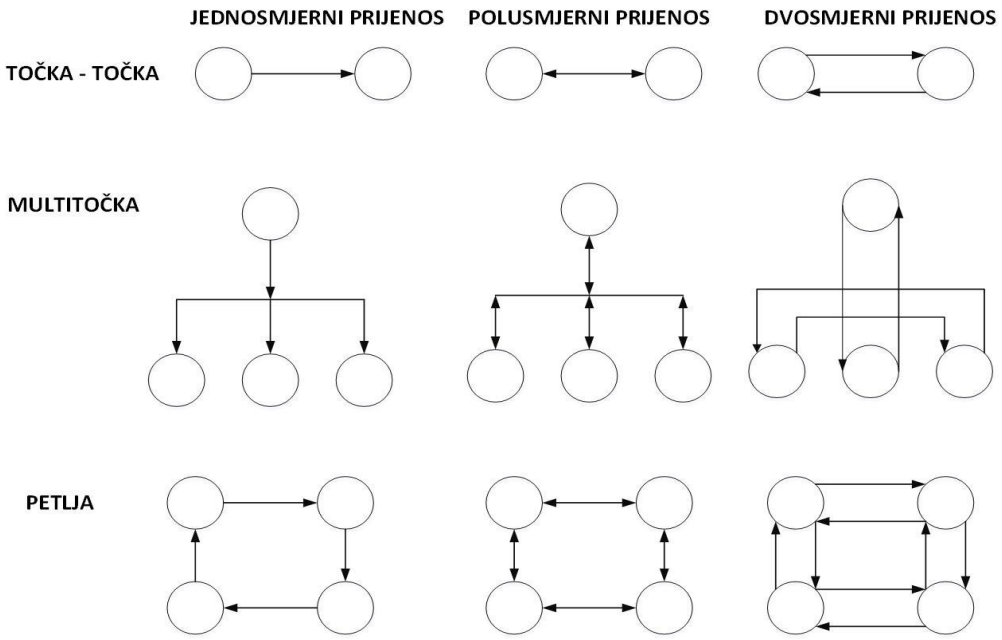
- OPC-DA – adresiranje prikupljenih procesnih varijabli, glavni klijenti su vizualizacija i opsluga, te registracija podataka,
- OPC-AE – informacije o alarmima i događajima, glavni klijenti su zapisivači alarma i događaja,
- OPC-HDA – moguće je povijesne podatke dohvatiti iz zapisa u memoriji, prikupiti i obraditi, glavni klijenti su prikazi trendova i histograma, [4].

Kao i za PLC uređaje, i za SCADA sustave najpoznatiji proizvođači su Siemens, ABB, Rockwell, General Electrics itd.

2.3. Komunikacija

Komunikacija je vrlo bitna za usklađivanje rada više računala pri vođenju procesa. S obzirom na to da je ovdje riječ o digitalnoj vezi, a računala koriste digitalne signale, nije potrebno pretvaranje signala što je slučaj kod izravnog povezivanja računala i periferije. Uz to informacija je očuvana i određena na bilo kojoj udaljenosti prijenosa.

Postoji nekoliko osnovnih struktura komunikacijskog sustava, a jedan od njih će se koristiti i u ovom radu. Slika 2.5. prikazuje osnovne komunikacijske strukture koji povezuju računalnu opremu u sustav za obradu podataka.



Slika 2.5. *Osnovne strukture sustava komunikacije.*

Informacije je moguće prenositi na razne načine, pa tako mediji koji mogu prenositi informacije su najčešće:

- jetkani (bakreni) vodovi unutar uređaja,
- trakasti kabeli unutar ili između uređaja,
- upleteni vodovi i višežilni kabeli,
- koaksijalni kabeli,
- optički kabeli,
- radio-veza.

Najjeftinija izvedba jesu dva upletena voda, koja su još oklopljena u industriji kako bi se informacije zaštitile od smetnji. Ukoliko je potrebno postići velike kapacitete prijenosa, tu je najbolje rješenje koaksijalni kabel. U današnje vrijeme sve je češća upotreba optičkih kabela zbog njihove neosjetljivosti na električka i magnetska polja. Za veće udaljenosti između dva odredišta za komunikaciju se koriste optički kabeli i radio-veze, [4].

Prijenos podataka u digitalnom obliku odvija se na dva načina: serijski i paralelno. Serijska sučelja mogu imati sinkroni i asinkroni način prijenosa podataka. Asinkroni prijenos je raširenji od sinkronog i najznačajniji predstavnik mu je RS 232C sučelje. Predstavnik sinkronog prijenosa je RS 485 sučelje koji će biti korišten u ovom diplomskom radu. Od paralelnih sučelja najznačajniji predstavnik je IEEE 488.

RS 485 je, dakle, standard serijskog sinkronog prijenosa podataka koji je bitan i često korišten u industriji. Svrha mu je na užem prostoru stvoriti komunikacijski put između više računala ili inteligentnih jedinica. Smatra se vanjskom serijskom sabirnicom, te je iz tog razloga u danom trenutku njime dozvoljen prijenos jedne poruke, u jednom smjeru između dva uređaja. Dakle, vodeći se slikom 2.5. to bi bila struktura točka-točka s polusmjernim prijenosom, no potrebno je naglasiti kako se komunikacija odvija između *mastera* i *slave-a*. Tablicom 2.1. bit će dane osnovne karakteristike RS 485 standarda.

Tablica 2.1. *Karakteristike RS 485 sučelja.*

Način prijenosa	serijski, sinkroni
Broj sugovornika	32 (moguće proširiti)
Duljina kabela	do 1400 m
Brzina prijenosa	< 100 kb/s (do 1200 m), < 10 Mb/s (do 20m)
Protokoli	SDLC, HDLC, FeildBus, PROFIBUS, PROFINET, MODBUS

Protokoli iz prethodne tablice koriste se kako bi se omogućila komunikacija preko RS 485 sabirnice. Najrasprostranjeniji jesu PROFINET i PROFIBUS, no ipak će se raditi s MODBUS protokolom i to RTU načinom rada (postoji još ASCII i TCP). Razlog korištenja ovog protokola je taj što ga frekvencijski pretvarač jedinog podržava.

MODBUS protokol služi za serijsku komunikaciju prilikom prijenosa podataka i princip rada mu je zasnovan na *master-slave* arhitekturi. Svaka *slave* jedinica protokola posjeduje svoju adresu, te reagira ona jedinica kojoj je naredba i namijenjena. RTU način (*eng. Remote Terminal Unit*) zasniva svoj rad na izravnom prijenosu heksadecimalnih znamenki, bez pretvorbe (konvertiranja) u ASCII kod, [5]. Kako bi se podaci pakirali u riječ, potrebno je postaviti redoslijed bitova koji je prikazan na slici 2.6.

Start	1	2	3	4	5	6	7	8	Par	Stop

(1)

Start	1	2	3	4	5	6	7	8	Stop	Stop

(2)

Slika 2.6. *Redoslijed bitova pri pakiranju podataka sa (1) i bez (2) paritetne provjere.*

Poruke u MODBUS-u su strukturirane isto za sve načine rada. Posjeduje pet dijelova: okvir, adresni dio, funkcijski kod, podatkovni dio i dio za provjeru ispravnosti poruke. Strukturu poruke, kao i duljinu pojedinog dijela (nisu iste za svaki način rada), moguće je vidjeti idućom slikom.

POČETAK	ADRESNI DIO	FUNKCIJSKI DIO	PODATKOVNI DIO	DIO ZA PROVJERU ISPRAVNOSTI PORUKE	KRAJ
28 bita	8 bita	8 bita	n x 8 bita	16 bita	28 bita

Slika 2.7. *Struktura poruke MODBUS-a.*

Početak i kraj poruke predstavljaju okvir strukture poruke, a za RTU način rada se koriste pauze koje traju barem 3,5 znakovnih intervala (uzima se višekratnik znakovnih intervala). Adresni dio je sastavljen od 8 bitova. Raspon adresa za *slave* jedinicu je od 1 do 247, dok nulta adresa predstavlja *broadcast* naredbu. Što se tiče *master* jedinice, ona postavlja određenu adresu koja

prima poruku, dok *slave* jedinica postavlja svoju adresu u adresni dio. Funkcijski je dio, kao i adresni, sastavljen od 8 bitova. Raspon poruke je od 0 do 255, te posjeduje funkcijski kod poruke koji za zadatak ima prenijeti određenu naredbu *slave* jedinici. Potvrda o primitku odrađuje se tako da *slave* jedinica šalje natrag nepromijenjenu poruku, dok u slučaju neke greške šalje funkcijski kod s bitom ne većim od 1. Podatkovni dio sastavljen je od para heksadecimalnih znamenki, koji za RTU način rada jedan znak. Za podatkovni dio poruke *master* jedinica unosi adresu registara ili vanjskih pinova za pristup, broj traženih podataka, ali i broj bajtova podataka i same podatke, ukoliko je te podatke potrebno negdje upisati. *Slave* jedinice ne sadrže uvijek podatkovni dio, jer to nije potrebno za neke naredbe. Ukoliko sadrži podatkovni dio, tada *slave* jedinica šalje podatke koji se traže, ali i kod greške za slučaj kada se naredba ne može izvršiti. Zadnji dio strukture poruke kod MODBUS-a čini dio za provjeru ispravnosti poruke koji je obavezan i neovisan o odabiru paritetne provjere znakova. Za RTU način rada provjera se provodi pomoću CRC metode (*eng. Cyclic Redundancy Check*). Metodu odrađuje *master* jedinica pri slanju poruke, a prije kraja poruke upisuje rezultat. *Slave* jedinica za svaki primitak poruke ponovno koristi CRC metodu, te ju uspoređuje s onim što *master* jedinica pošalje. U slučaju razlika u rezultatu, izbacuje se greška u vidu ne reagiranja *slave* jedinice. Za takav slučaj *master* jedinica posjeduje unaprijed postavljeno vrijeme (tzv. *timeout*) za čekanje odgovora *slave* jedinice. To vrijeme je dovoljno dugo za mogućnost reakcije *slave* jedinice. Poslije isteka tog vremena *master* jedinica prekida komunikaciju. Isti je ishod ukoliko *master* jedinica adresira *slave* jedinicu, [5,6].

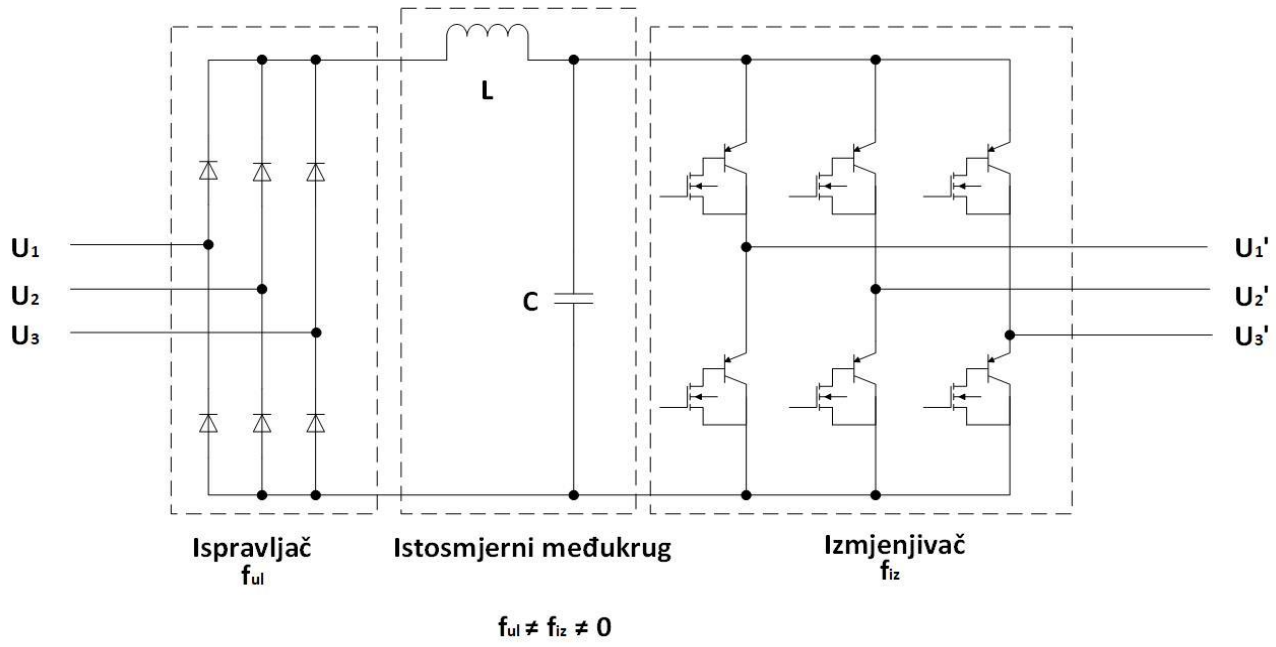
CRC metoda koristi se na cijeloj poruci, no u provjeru ulazi samo osam podatkovnih bitova svake riječi (ne ulaze *start, stop* i *paritetni* bitovi). Metoda se sastoji od dva bajta, od kojih je prvi uvijek niži, a drugi viši bajt. CRC metoda funkcionira na sljedeći način:

1) 16-bitni registar se napuni jedinicama,
2) operacija *XOR* (ekskluzivno ili) se provede nad 8-bitnim znakom i sadržajem registra,
3) pomicanje rezultata za jedan bit prema najnižem bitu (LSB – *eng. Least Significant Bit*),
4) nula se upisuje u najviši bit (MSB – *eng. Most Significant Bit*),
5) ukoliko je LSB = 1, provodi se operacija *XOR* nad sadržajem registra i nekoj prethodno definiranoj vrijednosti.

Korake 2) – 5) treba ponavljati osam puta za svaki bajt podataka, uz naglasak da se operacija *XOR* provodi s trenutnim sadržajem registra za svaki novi bajt. Nakon obrade cijele poruke, ona se upisuje kao CRC, [5].

2.4. Frekvencijski pretvarač

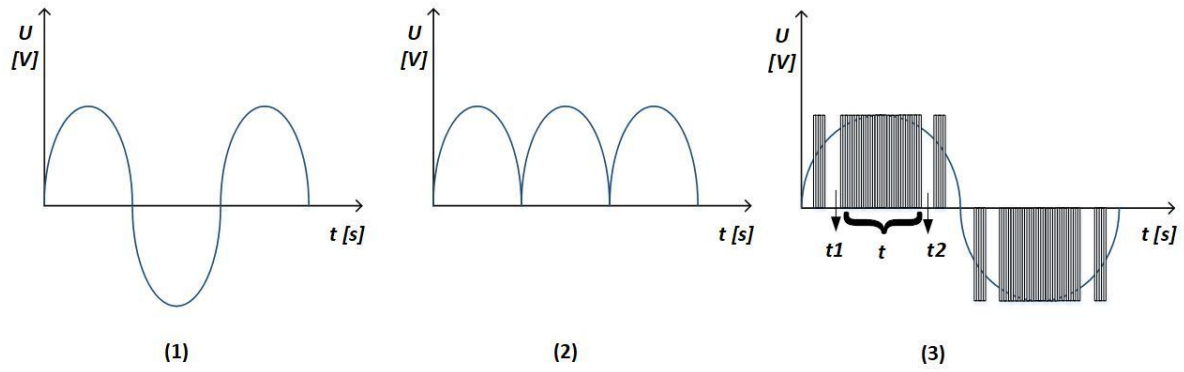
Još jedna bitna sastavnica ovoga diplomskog rada je frekvencijski pretvarač ili pretvarač frekvencije. Može se definirati kao pretvarač koji frekvenciju na ulazu (izvora), uz pomoć pretvaračkih komponenti, pretvara u željenu frekvenciju na izlazu (trošila), različitu od one na ulazu. Frekvencijski pretvarač čine tri osnovna dijela, a to su: ispravljač, istosmjerni međukrug i izmjenjivač. Ukoliko ispravljač i izmjenjivač rade neovisno jedan o drugome, tada je riječ o neizravnim frekvencijskim pretvaračima, no ukoliko se valni oblik napona mreže s ulaznom frekvencijom uz pomoć sklapanja pretvaračkih komponenti sintetizira u valni oblik napona trošila sa željenom izlaznom frekvencijom jednostrukim procesom pretvorbe, tada je riječ o izravnim frekvencijskim pretvaračima. Razne su izvedbe pretvarača, no u većini slučajeva koristi se za trofazne elektromotore. U tom slučaju, ispravljač i izmjenjivač se nalaze u trofaznom mosnom spoju, a ispravljač je sastavljen od dioda, dok je izmjenjivač od IGBT tranzistora (*eng. Insulated-gate bipolar transistor*), [7]. Sljedećom slikom prikazat će se trofazna shema strukture frekvencijskog pretvarača.



Slika 2.8. *Trofazna shema strukture frekvencijskog pretvarača.*

Prethodnom slikom je vidljivo kako su frekvencije ulaza f_{ul} i izlaza f_{iz} različite jedna od druge i različite od nule što ovaj pretvarač čini frekvencijskim pretvaračem.

Kompliciranost pretvaranja frekvencije izmjeničnog sinusnog oblika rješava se uz pomoć ispravljača koji izmjenični trofazni napon pretvara u monofazni istosmjerni ispravljeni valni oblik napona koji ide prema istosmjernom međukrugu. Funkcija istosmjernog međukruga nije da izravno doprinosi promjeni frekvencije, već ona služi, uz pomoć prigušnica i kondenzatora, kao filter izmjenične komponente napona koja može ostati poslije ispravljanja napona, a da ne dođe do izmjenjivača. Uz to, može i spriječiti harmonijsku distorziju od povratka u napajanje frekvencijskog pretvarača. Posljednja komponenta frekvencijskog pretvarača je izmjenjivač koji se sastoji od IGBT tranzistora (bolji od tranzistora zbog veće brzine preklapanja i smanjene proizvodnje topline) stvarajući tako istosmjerne impulse koji zapravo imitiraju sve tri faze izmjeničnog sinusnog valnog oblika. Ti impulsi diktiraju i valni oblik napona i frekvenciju. Izmjenjivači u današnjim frekvencijskim pretvaračima koriste tehniku pulsno – širinske modulacije kako bi mogli regulirati napon i frekvenciju, [8]. Slikom 2.9. prikazat će se valni oblici nakon što prođu pojedine komponente u frekvencijskom pretvaraču.



Slika 2.9. *Valni oblici na ulazu u ispravljač (1), ulazu u istosmjerni međukrug (2) i izlazu iz izmjenjivača (3).*

Kako je vidljivo na prethodnoj slici, valni oblik napona, mrežne frekvencije ulazi u frekvencijski pretvarač (1), nakon toga ga ispravljač ispravi te se dobiju pozitivne (apsolutne) vrijednosti napona (2). Zadnji valni oblik (3) je dobiven pulsno – širinskom modulacijom, odnosno tehnikom

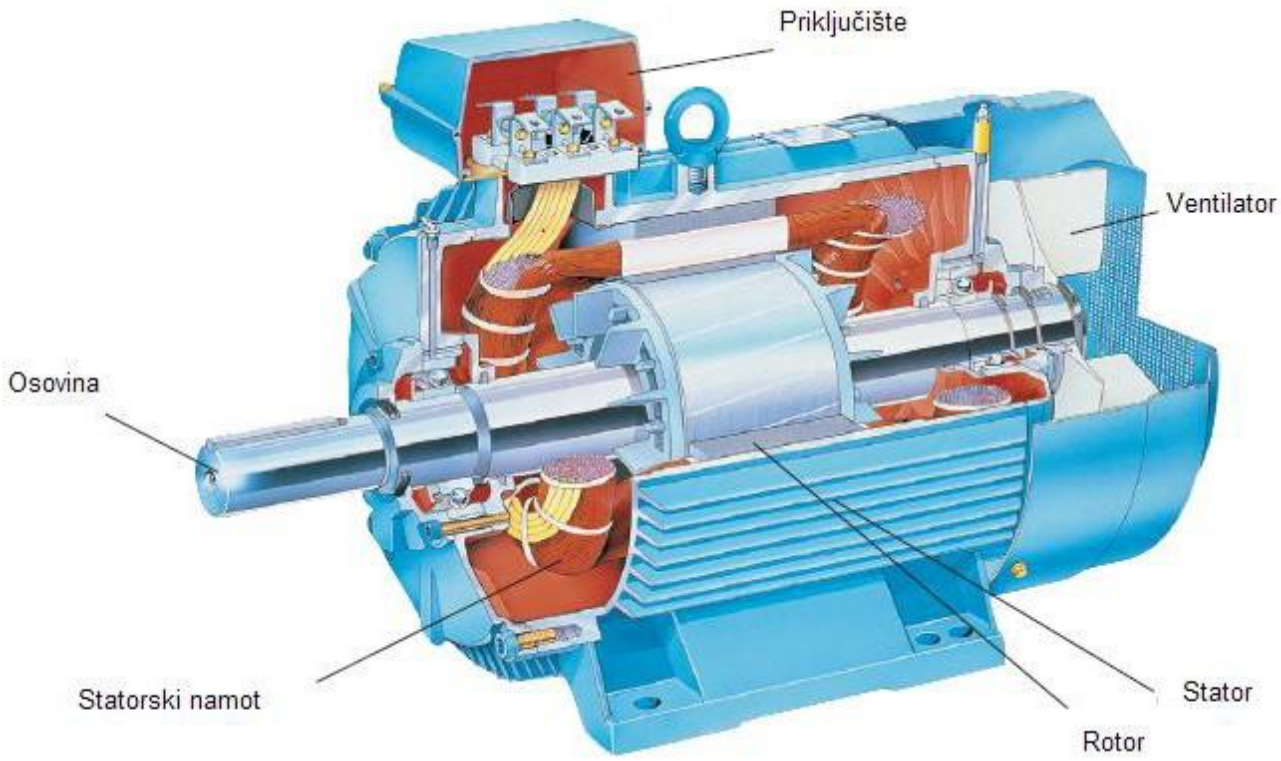
upravljanja kojom se omogućava da se voltsekunde na izlazu mijenjanju prema željenom, i unaprijed utvrđenom, zakonu, [7]. Cilj je stvoriti određeni valni oblik uz pomoć uklopa-isklopa tranzistora. Može se primijetiti niz pravokutnih impulsa, sklopne frekvencije, koji imaju stalnu amplitudu, dok mu se širina može mijenjati. Tranzistor je uklopljen kada postoje ti pravokutni nizovi, dok je inače isklopljen. Vrijeme uklopljenog tranzistora (*t*) daje efektivnu vrijednost izlaznog napona, dok suma vremena isklopljenog tranzistora (*t1* i *t2*) daje izlaznu frekvenciju. Suma pozitivnog dijela daje pozitivni dio sinusnog valnog oblika koji je prikazan na slici, a negativni dio se dobije sumom negativnog niza pravokutnih impulsa. Potrebno je naglasiti kako mora biti ostvareno da je sklopna frekvencija tranzistora puno veća od željene frekvencije na izlazu. S obzirom na to da se na izlazu dobije sinusni valni oblik, iz tog razloga je riječ o sinusnoj pulsno – širinskoj modulaciji.

Upravo zbog te promjene frekvencije frekvencijski pretvarač omogućuje učinkovitije korištenje motora, te tako produljuje životni vijek samog motora. Primjer tomu je velika struja prilikom pokretanja motora, dok zbog svog načina rada frekvencijski pretvarač omogućuje *soft start* te tako uvelike smanjuje struju prilikom pokretanja.

Najpoznatiji primjeri frekvencijski pretvarači su Danfoss, Sinamics (Siemens), Schneider Electric, Eura (LS Control) i mnogi drugi.

2.5. Asinkroni motor

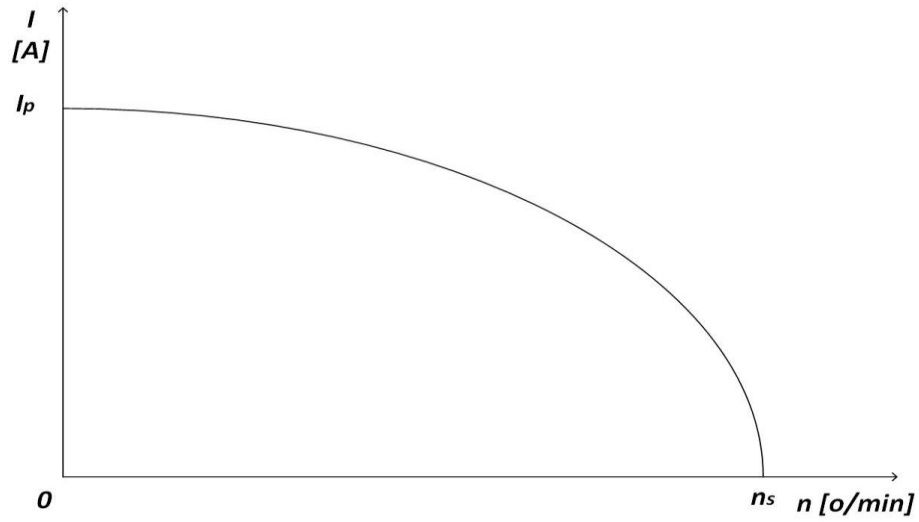
Jedna od komponenata za ovaj diplomski rad je, između ostaloga, jedan od najpoznatijih izuma velikog znanstvenika, Nikole Tesle. Asinkroni ili indukcijski motor je stroj koji se pogoni izmjeničnom strujom, a glavna karakteristika mu je manja brzina okretanja rotora u odnosu na brzinu okretanja magnetskog polja statora. Ta razlika među brzinama naziva se klizanje. Asinkroni motori se dijele na dvije vrste, a to su: klizno-kolutni i kavezni. Kavezni asinkroni motor češće je u upotrebi, a ime je dobio zbog toga što na rotoru ima kratkospojene vodiče koje sliče kavezu. Iz tog razloga, dalje u tekstu, pod pojmom asinkroni motor mislit će se na asinkroni kavezni motor. Motor se može spajati u trokut ili zvijezdu, a u praksi se motori manji nazivnih snaga (do 2,2 kW) spajaju u zvijezdu, dok oni većih (iznad 3 kW) u trokut. Sljedećom slikom prikazat će se izgled asinkronog motora, s pripadajućim osnovnim dijelovima.



Slika 2.10. *Kavezni asinkroni motor, [9]*.

Princip rada asinkronog motora zasniva se na tomu da prilikom priključenja na napon mreže, kroz višefazne simetrične namote statora poteče izmjenična struja koja stvara okretno magnetsko polje. Te struje stvaraju okretno protjecanje, koje u zračnom rasporu motora stvara okretno polje indukcije. Okretno polje indukcije u statoru stvara protuelektromotornu silu koja drži ravnotežu s naponima mreže, te ne dopušta prekomjerno povećanje struje. Nadalje, okretno polje stvoreno u zračnom rasporu inducira struje u rotoru pošto je on kratkospojen. Inducirane struje namota rotora moraju biti višefazne i simetrične kako bi mogle dati okretno protjecanje koje će dovesti do okretanja rotora, [10].

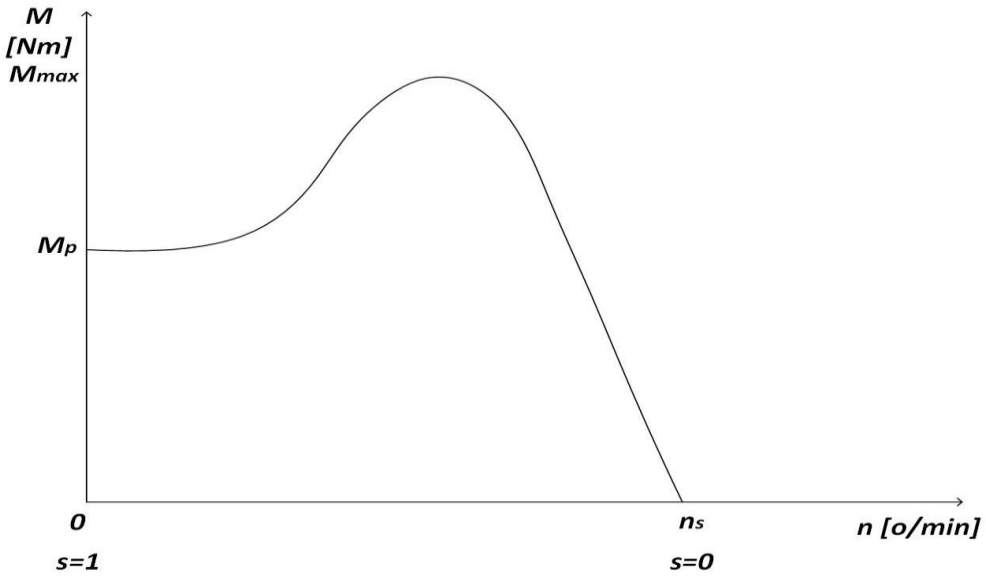
Prikazivanjem raznih ovisnosti može se lakše shvatiti kako se mijenjaju određene veličine u odnosu na neke druge. Slika 2.11. prikazat će ovisnost rotorske struje o brzini vrtnje (klizanju).



Slika 2.11. *Ovisnost rotorske struje o brzini vrtnje (klizanju)*

Vidljivo je kako se najveća struja, tzv. potezna struja I_p, pojavljuje prilikom pokretanja motora, dok je brzina vrtnje jednaka nuli. Ona može biti i sedam do deset puta veća od nazivne struje. Povećanjem brzine vrtnje, struja postaje sve manja, a mijenja se po nelinearnom zakonu. Pri sinkronoj brzini n_s, struja je jednaka nula.

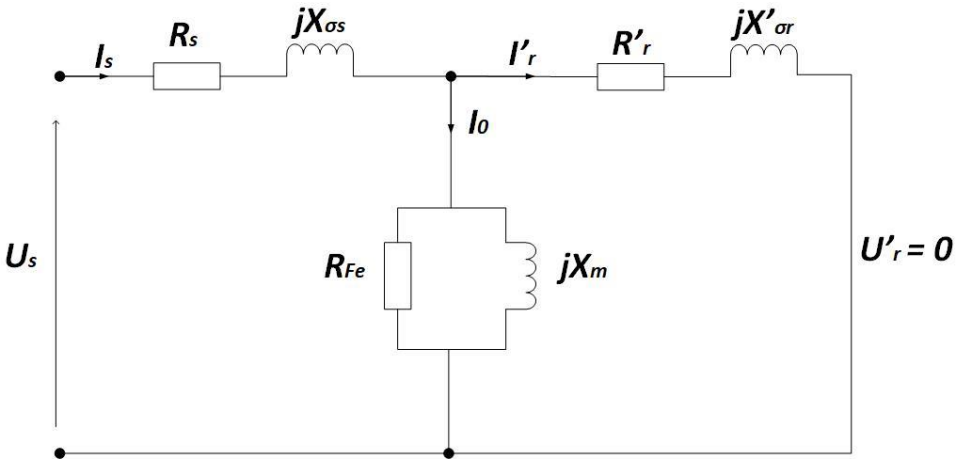
Iduća slika prikazuje momentnu karakteristiku asinkronog motora s pripadajućim značajnim veličinama.



Slika 2.12. *Momentna karakteristika asinkronog motora*

Momentnom karakteristikom vidljivo je da prilikom pokretanja motora, tj. pri poteznoj struji postoji potezni moment M_p koji je prisutan pri klizanju $s=1$. Nadalje, moment postepeno raste dok ne dođe do prekretnog ili najvećeg momenta M_{max}, te nakon toga opada do sinkrone brzine n_s (gdje je $s=0$). S obzirom na to da je riječ o asinkronom motoru, taj moment neće pasti na nulu, već će imati neku vrijednost za određenu radnu točku.

Nadomjesna shema od velike je važnosti ukoliko se želi bolje promotriti matematički model stroja. Slika 2.13. prikazuje pojednostavljenu nadomjesnu shemu asinkronog motora



Slika 2.13. *Pojednostavljena nadomjesna shema asinkronog motora*

Kao što je vidljivo prethodnom slikom asinkroni motor se modelira s otporima statora R_s i rotora (reducirano) R'_r, rasipnim reaktancijama statora $jX_{\sigma s}$ i rotora (reducirano) $jX'_{\sigma r}$, te reaktancijom magnetiziranja X_m i gubicima u željezu zbog histereze i vrtložnih struja R_{Fe}. S obzirom na to da su tijekom normalnog rada stroja gubici u željezu vrlo mali, često se zanemaruju, te u nadomjesnoj shemi ostaje samo reaktancija magnetiziranja.

3. REALIZACIJA SUSTAVA UPRAVLJANJA

Unutar ovog poglavlja bit će dan kratki opis programskih alata koji su korišteni pri sastavljanju ovog diplomskog rada. Nadalje, prikazat će se osnovne karakteristike korištenog PLC-a sa svojim modulima, kao i frekvencijskog pretvarača s pripadajućim asinkronim motorom. Slijedit će opis postavljanja komunikacije između frekvencijskog pretvarača i TIA Portala. Ovo poglavlje će zaokružiti integracija sustava u kojima će se preko blok dijagrama predočiti rad programa u PLC-u, te princip rada vizualizacije programa, odnosno SCADA-e.

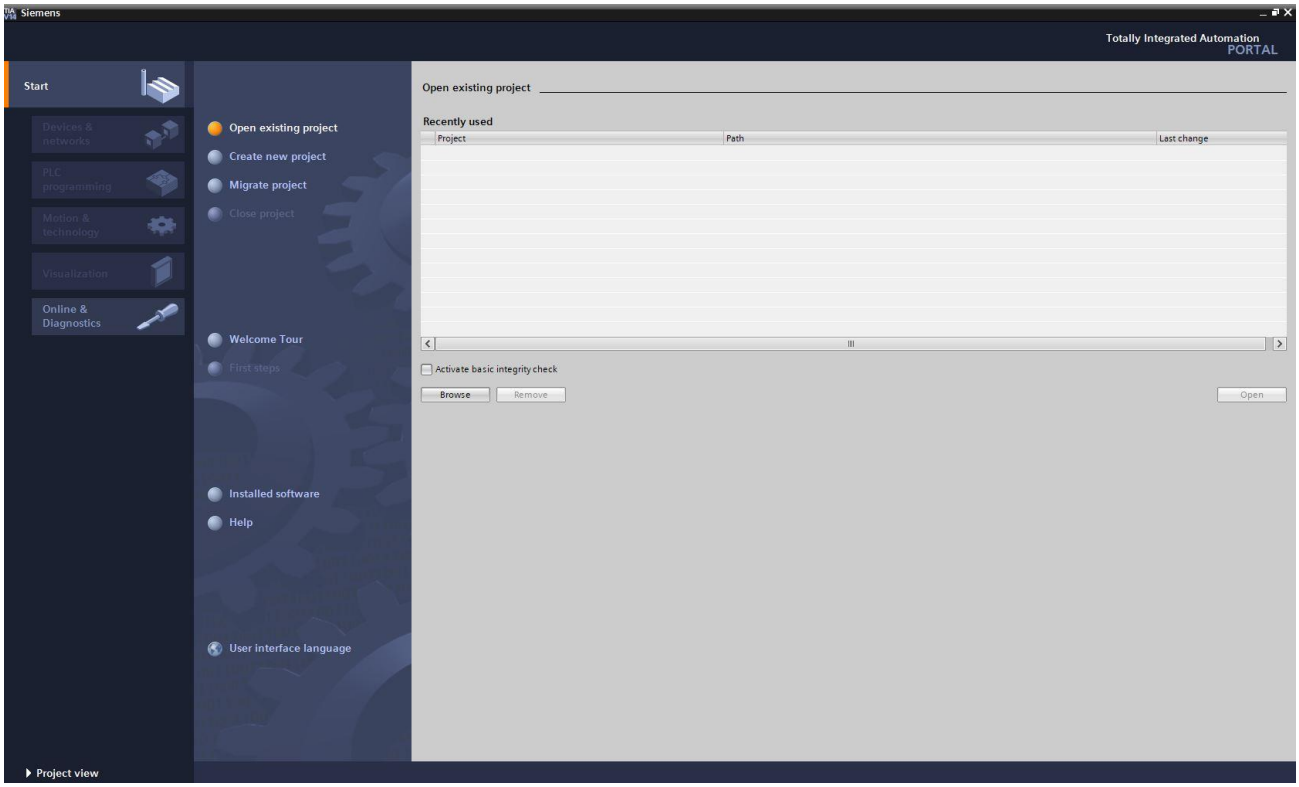
3.1. Korišteni programski alati

Kroz ovo potpoglavlje dat će se kratki opis svih korištenih programskih alata u diplomskom radu. Prvi će biti predstavljen Siemensov program za automatizaciju, nadzor i upravljanje – TIA Portal. Potpoglavlje će završiti s programskim alatom koji je korišten za prikaz i vizualizaciju unutar diplomskog rada, Microsoft Visio.

3.1.1. TIA Portal

TIA Portal (*eng. Totally Integrated Automation Portal*) Siemensov je programski paket koji omogućuje korisniku potpuno digitalnu automatizaciju postrojenja. Sastoji se iz nekoliko dijelova: SIMATIC STEP 7 za PLC programiranje, SIMATIC WinCC za vizualizaciju preko grafičkog sučelja, SINAMICS Startdrive za parametrizaciju pogona, program SIMCODE za upravljanje motorom, te dio za distribuciju energije, [11].

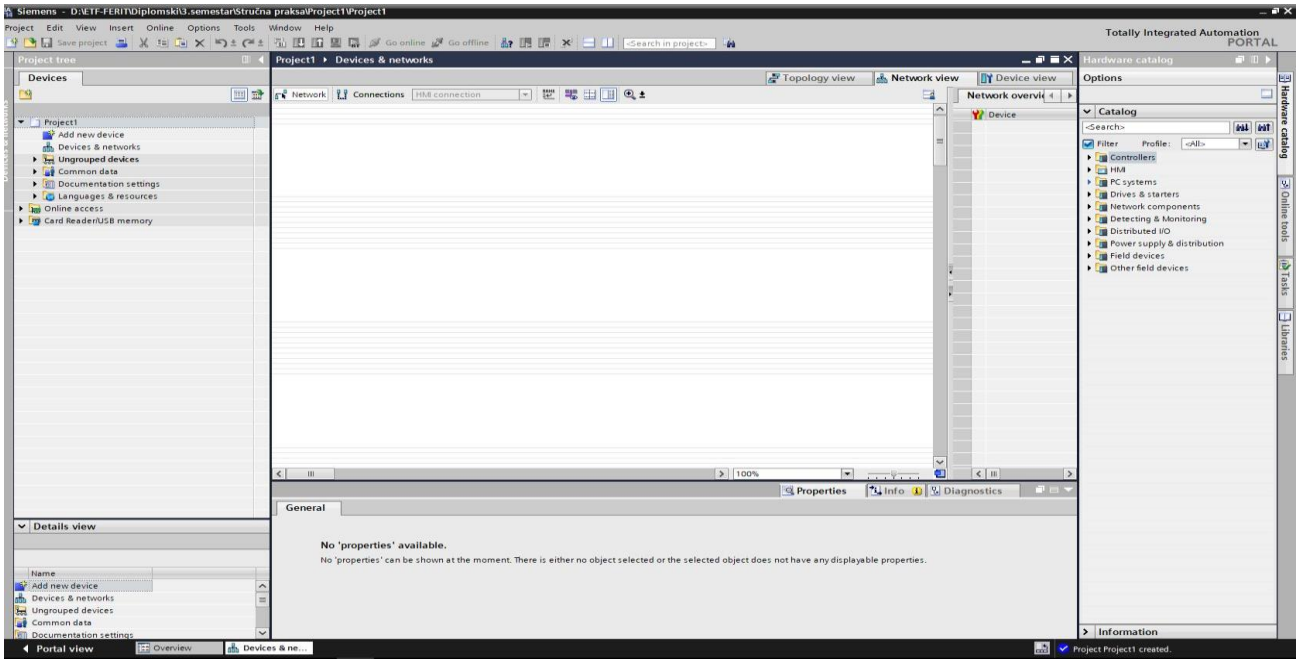
Izgled početnog sučelja TIA Portala prikazat će se sljedećom slikom.



Slika 3.1. *TIA Portal.*

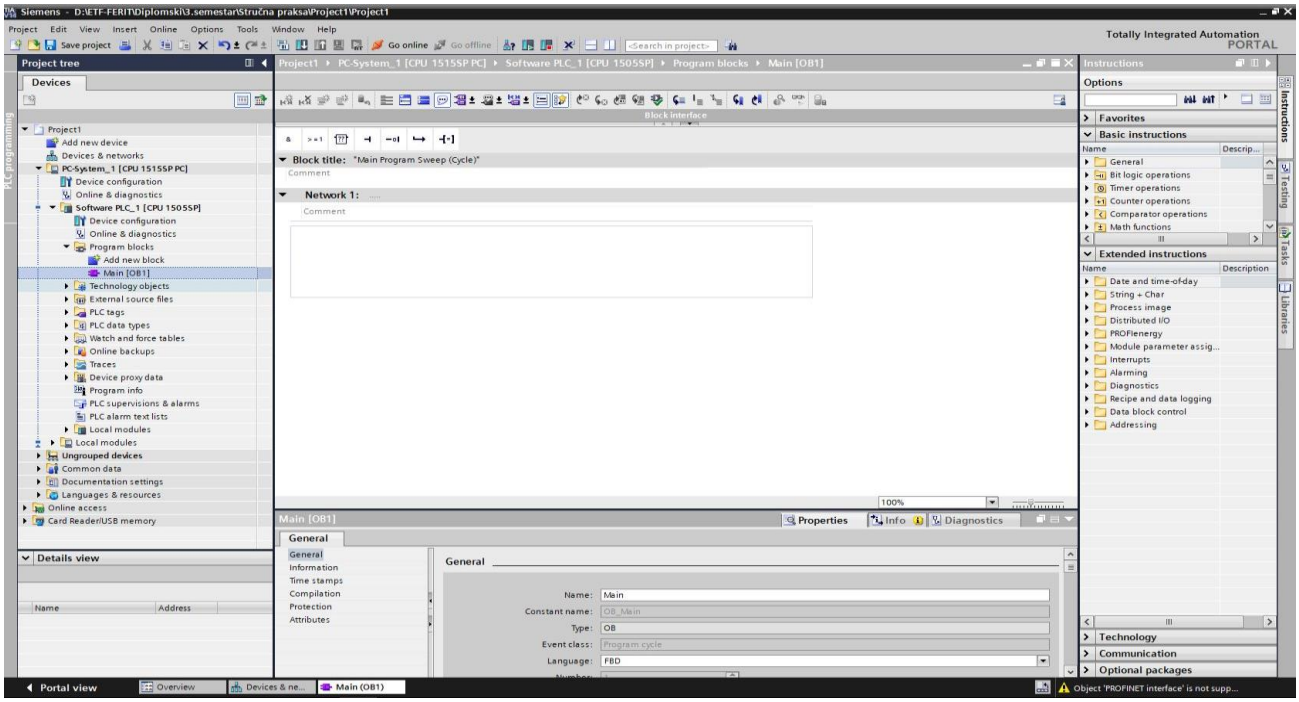
Prethodnom slikom uočava se izgled sučelja TIA Portala. Zasjenjene kartice *Devices & Network*. *PLC programming*, *Motion& technology* i *Visualization*, a koje se nalaze s lijeve strane prozora, nije moguće otvoriti prije nego li se izabere jedna od opcija u središnjem dijelu prozora (*Open existing project*, *Create new project* ili *Migrate project*).

Konfiguracija korištenih uređaja i komunikacija postavljaju se u kartici *Devices & Network* koji je prikazan na slici 3.2.



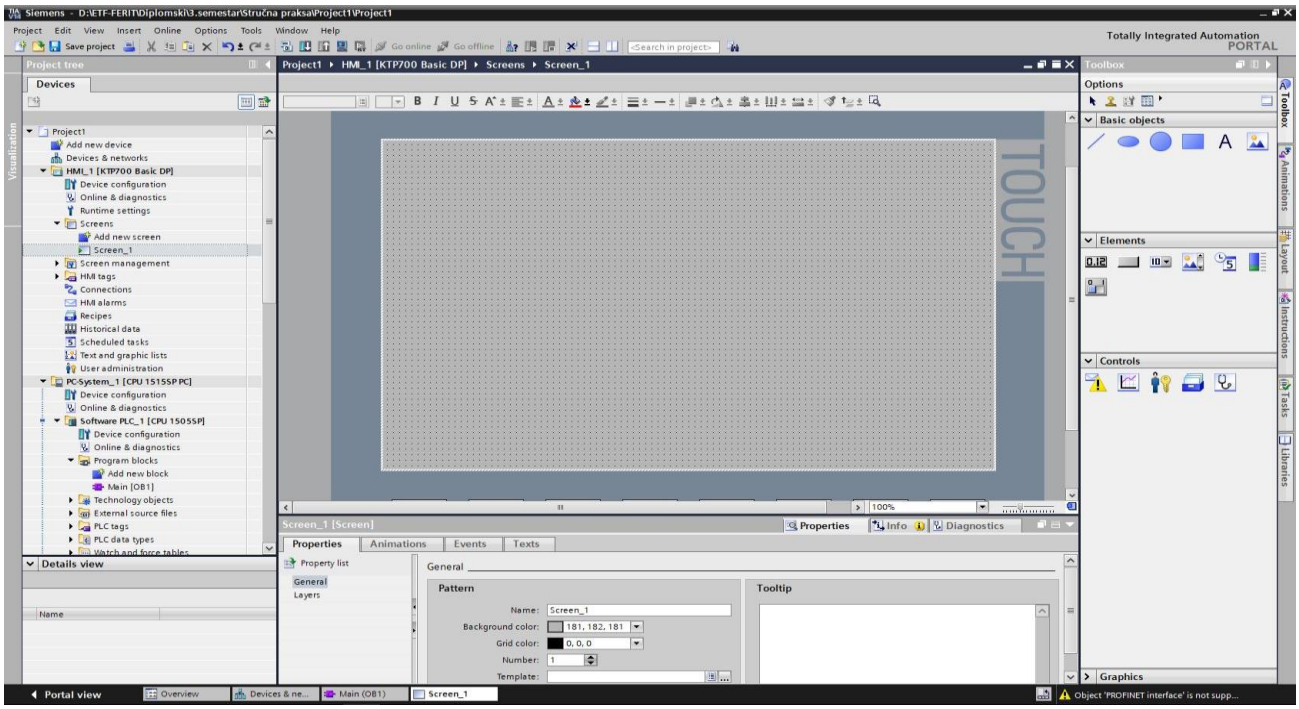
Slika 3.2. *Kartica Devices & Network.*

Kroz SIMATIC STEP 7 može se automatizirati postrojenje ili dio postrojenja uz pomoć četiri načina: STL, FBD, LAD i SCL. Zbog jednostavnosti i najboljeg razumijevanja izabran je FDB. Izgled sučelja SIMATIC STEP 7 unutar TIA Portala prikazan je sljedećom slikom.



Slika 3.3. *SIMATIC STEP 7.*

Vizualizacija, a samim time i nadzor, je zadnji dio kojeg je potrebno napraviti u TIA Portalu. Njome se omogućuje lakše shvaćanje dijela postrojenja kojim se upravlja, te se na interaktivan način povezuje vezu između stroja i čovjeka. Izgled dijela grafičkog sučelja kojim se vizualizira automatizirani dio prikaza će se slikom 3.4.

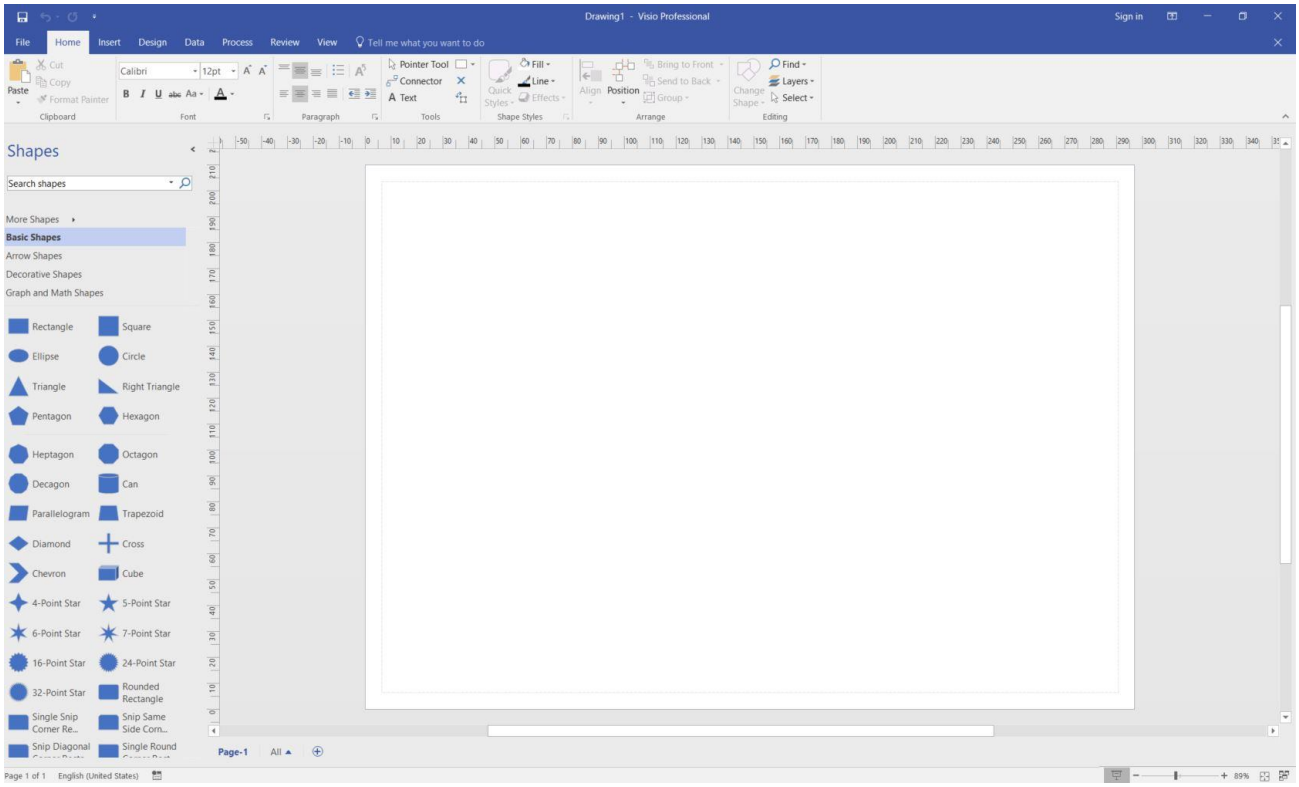


Slika 3.4. *HMI grafičko sučelje*

3.1.2. Alati za vizualizaciju

Postoji puno alata s kojima je moguće prikazati određene slike (poput elektrotehničkih alata AutoCAD i Eplan). U ovome diplomskom radu za vizualizaciju će se koristiti Microsoftov Visio.

Programski alat Microsoft Visio se može upotrebljavati u razne svrhe. Osim za osnovne dijagrame, moguće je ovaj programski alat koristiti i za električne sheme, vremenske crte, tlocrte kuće i rasporede kućanskih elemenata, dijagrame tijeka, Ganttov-e dijagrame upravljanja projektima i mnoge druge. Slikom 3.5. pokazat će se izgled sučelja Microsoft Visia.



Slika 3.5. *Sučelje Microsoft Visia.*

Unutar programskog alata se nalazi široki opus potrebnih električnih simbola u više mapa (osnovni predmeti, kvalificirani simboli, poluvodiči i elektronske cijevi, sklopke i releji, te putevi prijenosa).

3.2. Korišteni PLC i frekvencijski pretvarač

PLC uređaj koji će biti korišten u ovome diplomskom radu je ET 200SP. Sastoji se iz dva dijela: glavnog dijela gdje se nalazi središnja procesna jedinica (*eng. Central Processing Unit – skraćeno CPU*) SIMATIC S7-1500 zajedno s otvorom za SIMATIC memorijsku karticu i modulskih jedinica koje je moguće spojiti na šinu pored glavnog dijela. Tablica 3.1. dat će uvid u osnove karakteristike PLC-a korištenog u diplomskom radu.

Tablica 3.1. *Karakteristike distribuiranog sustava ET 200SP, [12].*

Broj poizvoda	6ES7512-1DK01-AB0
Oznaka tipa proizvoda	CPU 1512SP-1 PN
Dimenzije [mm]	100 x 75 x 117
Napon napajanja [V DC]	24
Dozvoljeni raspon napona [V DC]	19,2 – 28,8

Struja potrošnje [A]	0,6
Najveća potisna struja [A]	4,7
Dolazna snaga na sabirnicu [W]	8,75
Gubici [W]	5,6
Broj otvora za memorijsku karticu	1
Zajednička radna memorija za program [kB]	200
Zajednička radna memorija za podatke [MB]	1
Najveća uključena radna memorija (SIMATIC memorijska kartica) [GB]	32
Broj distribuiranih I/O sustava[1]	32
Dozvoljeni broj modula na šini	80

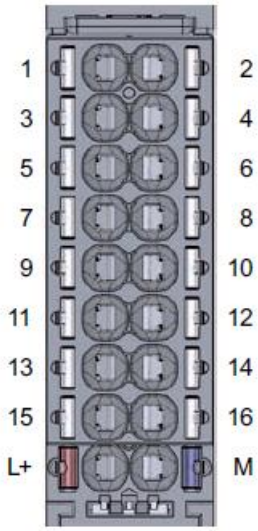
Drugi dio je komunikacijski modul pomoću kojega će se ostvariti komunikacijska veza između frekvencijskog pretvarača i PLC-a. Uočava se na slici 2.2. kako PLC na glavnom dijelu ima omogućenu komunikaciju preko PROFINET-a (modul BA 2xRJ45), odnosno LAN-a. Preko njega će se vršiti komunikacija između PLC-a i laptopa. Iz razloga što fekvencijski pretvarač podržava drugi oblik komunikacije dodat će se komunikacijski modul CM PtP. Karakteristike tog modula bit će prikazane idućom tablicom.

Tablica 3.2. *Karakteristike komunikacijskog modula CM PtP, [13]*.

Broj poizvoda	6ES7137-6AA00-BA0
Oznaka tipa proizvoda	ET 200SP CM PtP
Dimenzije [mm]	15 x 58 x 73
Napon napajanja [V DC]	24
Dozvoljeni raspon napona [V DC]	19,2 – 28,8
Struja potrošnje [mA]	29
Gubici [W]	0,7
Područje ulaza adresa [bajt]	8
Sučelja	RS 232, RS 422. RS 485
Najveća brzina prijenosa [kbit/s]	115,2
Najveća dužina kabela[m]	1200
Broj područja adresa	1 – 247; 1 – 65535 (prošireno)
Najveći broj *slave-a*	32

[1] Ulaza/Izlaza (*eng. Input/Output*)

RS 485 standard služi kao fizičko sučelje za ostvarivanje komunikacije. Moguće ga je pronaći u četverožilnom ili osmerožilnom kabelu. Koristit će se četverožilni kabel. Redoslijedom, na prvi pin (crna žica) dovodi se 5 V, drugi pin (crvena žica) posjeduje izlaz za terminal B-, treći pin (zelena žica) posjeduje izlaz za terminal A+, te se na četvrtom pinu (žuta žica) nalazi uzemljenje. Prvi pin se ne spaja, tj. kraj žice će biti izoliran. Na komunikacijskom modulu nalaze se terminali na koje je potrebno spojiti RS 485 standard. Iduća slika prikazuje terminale na komunikacijskom modulu.



Slika 3.6. *Terminali komunikacijskog modula CM PtP, [13]*.

Terminali bitni za rad, u odnosu na RS 485 standard, su terminal 12 (prima/šalje podatke, T(A)/R(A), dvožični način), terminal 14 (prima/šalje podatke, T(B)/R(B), dvožični način), te terminali 15 i 16 (uzemljenje).

Kako bi PLC funkcionirao, potrebno mu je dovesti napajanje. To se radi preko ispravljača SIMATIC S7 gdje se mrežni napon pretvara na napon napajanja PLC uređaja 24 V DC. Slikom 3.7. prikazat će se navedeni ispravljač.

Slika 3.7. *Ispravljač SIMATIC S7.*

Ispravljač SIMATIC S7 sadrži sljedeće karakteristike.

***Tablica* 3.3.** *Karakteristike ispravljača SIMATIC S7, [14].*

Naziv proizvoda	SIMATIC S7-300 reguirani izvor napajanja
Broj proizvoda	6ES7307-1EA00-0AA0
Opskrbni napon [V]	120, 230 i automatski odabir raspona
Raspon napona [V]	85 – 132; 170 – 264
Prekonaponski otpor [Ω]	$1{,}3 \cdot U_{ul,n}$ (1,3 ms)
Nazivna frekvencija [Hz]	50; 60
Raspon frekvencije [Hz]	47 – 63
Ulazna struja [A]	2,3 (za 120 V); 1,2 (za 230 V)
Najveći limit struje uklopa (+25℃) [A]	20
Ugrađeni osigurač (nije pristupačan)	T 3,15 A/250 V
Izlaz	kontrolirani, izolirani DC napon
Nazivni izlazni napon [V DC]	24
Tolerancija, statična [%]	± 3
Dizanje napona [ms]	10
Raspon izlazne struje [A]	0 – 5
Aktivna snaga napajanja [W]	120
Korisnost [%]	87
Gubici [W]	18
Zaštita	IP20

Može se primijenitit kako je građen i za europsko (230 V) i za američko (120 V) tržište, s vrlo visokom korisnošću.

Osim PLC-a, potreban je bio i frekvencijski pretvarač. On se izabire po nazivnoj struji motora, a struja pretvarača mora biti ili jednaka ili veća od nazivne struje motora. Uzet je EURA E800 frekvencijski pretvarač, te su karakteristike navedene u tablici 3.4. ispisane iz korisničkog priručnika koji je došao uz frekvencijski pretvarač.

Tablica 3.4. *Karakteristike frekvencijskog pretvarača EURA E800.*

Model	E800-0007S2
Opcijske funkcije	E1U1F2AE03B1R3
Ulazni broj faza	1
Ulazni napon [V]	220-240
Ulazna frekvencija [Hz]	50-60
Izlazni broj faza	3
Izlazni napon [V]	0-240
Izlazna struja [A]	4,5
Izlazna frekvencija [Hz]	0,5 - 650
Izlazna snaga [kW]	0,75
Stupanj zaštite	IP20

Iz modela frekvencijskog pretvarača može se vidjeti serija proizvoda (E800), izlazna snaga (0007) i broj ulaznih faza (S2 što znači monofazan priključak 220 – 240 V AC). Unutar opcijskih funkcija F2 zapravo znači da se MODBUS spaja preko priključnica, dok E1 označava englesku verziju frekvencijskog pretvarača bez potenciometra. Izgled korištenog frekvencijskog pretvarača dat će se sljedećom slikom.

Slika 3.8. *Frekvencijski pretvarač EURA E800.*

Može se uočiti kako frekvencijski pretvarač posjeduje i panel s tipkama s kojima se može upravljati. To će biti zamijenjeno „udaljenim“ grafičkim sučeljem unutar TIA Portala. Može se uočiti kako u svome podnožju frekvencijski pretvarač ima oznake. Prva slijeva je za uzemljenje, *L1* i *L2* služe za ulaz, odnosno napajanje s mreže (230 V). Sljedeće dvije, *P* i *B*, služe za spajanje otpornika za kočenje motora, dok se na zadnje tri: *U*, *V* i *W*, spaja trofazni asinkroni motor.

Iako ovaj frekvencijski pretvarač posjeduje više od 300 funkcijskih parametara, sljedećom tablicom dat će se 10 njihovih grupacija koji će ih sve objediniti.

Tablica 3.5. *Grupacija funkcijskih parametara, [15].*

Ime grupacije	**Raspon funkcijskog parametra**
Osnovni parametri	F100 – F160
Radno upravljanje	F200 – F230
Multifunkcionalni ulazni/izlazni priključak	F300 – F330
Ulazni i izlazni analogni signali	F400 – F480
Višestepenasto upravljanje brzine	F500 – F580
Pomoćne funkcije	F600 – F650

Pogreške i zaštita	F700 – F760
Parametri motora	F800 – F850
Komunikacijski parametri	F900 – F930
PID parametri	FA00 – FA70

Također, sa strane frekvencijskog pretvarača uočavaju se određeni priključci. Najznačajniji su *CM, A+* i *B-* priključci. *CM* (*eng. Common port* ili zajednički ulaz) služi za uzemljenje napajanja od 24 V i ostalih kontrolnih signala. Priključci *A+* (pozitivni polaritet diferencijalnog signala) i *B-* (negativni polaritet diferencijalnog signala) su komunikacijski priključci za RS 485 standard, te MODBUS komunikacijski protokol.

Nakon što se uključi frekvencijski pretvarač, potrebno je unijeti parametre motora s natpisne pločice. Zato što je korišten motor starije proizvodnje, poznato je bio samo naziv proizvođača, dok je struja parametrirana „prema iskustvu“, uz pomoć inženjera elektrotehnike s dužim radnim stažom.

Prvo je potrebno unijeti osnovne parametare (funkcijski parametri F800 – F805 i F810). Također je moguće ugađanje moguće u vidu rotacionog (F800=1) ili stacionarnog (F800=2), no za rotaciono je potrebno isključiti teret.

Način upravljanja motorom moguće je odrediti preko funkcijskog parametra F106 i to: 0 za bezsenzorsko vektorsko upravljanje, 1 za rezervirano upravljanje, 2 za skalarno U/f upravljanje i 3 za vektorsko upravljanje 1 (uz to ima i opciju 6 koja bezsenzorsko vektorski upravlja sinkronim motorima s permanentnim magnetima).

Funkcijskim parametrom F113 upravlja se frekvencija motora te ju je moguće postaviti na željenu. No ta željena mora biti unutar raspona minimalne (određeno funkcijim parametrom F112) i maksimalne frekvencije (određeno funkcijim parametrom F111).

Uz pomoć funkcijskih parametara F114 – F117 i F277 – F280 moguće je upravljati ubrzavanjem i usporavanjem motora.

Prikaz trenutnih vrijednosti moguće je ostvariti odabirom F131 funkcijskog parametra . Odabire se određeni broj (potencija broja dva), a najvažniji brojevi su: 0 za trenutnu izlaznu frekvenciju, 1

za izlaznu rotacijsku brzinu, 2 za izlaznu struju, 4 za izlazni napon, 2048 za izlaznu snagu, te 4096 za izlazni moment. Ukoliko se želi prikazati više veličina odjednom, odabire se zbroj tih vrijednosti, npr. zbroj 2056 znači da je odabran prikaz izlazne struje, izlaznog napona i izlazne snage.

Način kompenzacije momenta moguće je odabrati kroz funkcijski parametar F137: linearna kompenzacija (0), kvadratna kompenzacija (1), kompenzacija po želji korisnika odabirom više točaka (2) i automatska kompenzacija (3). Za ovaj slučaj uzet će se linearna kompenzacija, jer je najbolje rješenje kod univerzalnih tereta s konstantnim momentom.

Funkcijski parametri F200 i F201 omogućuju način unošenja naredbe za pokretanje i zaustavljanje. On može biti preko tipkovnice (0), priključnica (1), tipkovnice i priključnica zajedno (2), MODBUS-a (3) i svega zajedno (4).

Uz pomoć funkcijskog parametra F203 moguće je odabirom broja 10 izabrati da je MODBUS glavni nositelj frekvencije, te je on kompatibilan sa svim dodatnim izvorima frekvencije.

Komunikacijska adresa postavlja se u funkcijskom parametru F900. Odabirom brojeva od 1 do 255 postavlja se adresa pretvarača, dok 0 označava *broadcast* adresu. Način komunikacije postavlja se u F901 (1 – ASCII, 2 – RTU), a paritetna provjera u F903 (0 – nevažeći, 1 – neparni, 2 – parni). Broj bauda ili brzinu prijenosa podataka moguće je naći u funkcijskom parametru F904, te se određuje različiti raspon odabirom brojeva 0 – 6 (0 – 1200, 1 – 2400, 2 – 4800, 3 – 9600, 4 – 19200, 5 – 38400, 6 – 57600). Preporučena brzina prijenosa je 9600 Bd (bit/s).

Između frekvencijskog pretvarača i motora je spojena motorna zaštitna sklopka ISKRA DME 25 radi zaštite motora. Treba naglasiti kako se u većini slučajeva sklopnik postavlja između mreže i frekvencijskog pretvarača, no za ovaj sustav je dovoljan sklopnik i u ovoj poziciji. Ova motorna zaštitna sklopka štiti od napona većih od 500 V i struja 2,5 A. Posjeduje termički relej te ima osigurač parametriran na 2,5 A. Slika 3.10. prikazuje motornu zaštitnu sklopku.

Slika 3.9. Motorna zaštitna sklopka Iskra DME 25.

Motor koji će biti pogonjen ovim integriranim sustavom bit će dan sljedećom slikom.

Slika 3.10. Korišteni motor.

Kako se može vidjeti, motor je starije izrade, te je natpisna pločica s nazivnim podacima otpala. Ostala je samo natpisna pločica s nazivom tvornice: Otto Wottova Zagreb.

3.3. MODBUS protokol pri realizaciji sustava

Detaljno je, teoretski, MODBUS protokol opisan u potpoglavlju 2.5., no ovdje će se pojasniti što je sve potrebno za komunikaciju u stvarnom sustavu. Unutar priručnika za korisnike za frekvencijski pretvarač EURA E800 zapisane su potrebne adrese kako bi se olakšao dio komunikacije za neke značajnije radnje, a to je samo jedan dio koji je potreban pri realizaciji komunikacije. Idućom tablicom dat će se adrese registara koji će se koristiti, ali je potrebno naglasiti kako su one u heksadecimalnom formatu.

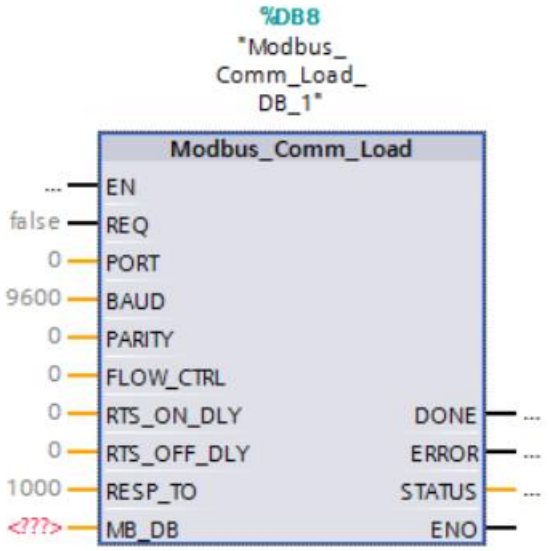
Tablica 3.6. *Adrese registara frekvencijskog pretvarača za MODBUS, [15].*

Registar adrese parametra	Opis adrese parametra
1000	Iščitavanje izlazne frekvencije
1001	Iščitavanje izlaznog napona
1002	Iščitavanje izlazne struje
1005	Iščitavanje statusa pretvarača 0X00: Čekanje 0X01: Naprijed 0X02: Reverziranje 0X0E, 0X0F, 0X10, 0X11: Greške 1 – 4
1007	Iščitavanje temperature grijača pretvarača
1017	Iščitavanje senzora trenutne brzine vrtnje
1018	Iščitavanje senzora snage na prvu decimalu
2000	Upisivanje komandi: 0001: Naprijed (bez parametara) 0002: Reverziranje (bez parametara) 0003: Zaustavljanje deceleracijom 0008: Pokreni (bez smjera) 0009: Poništavanje greške

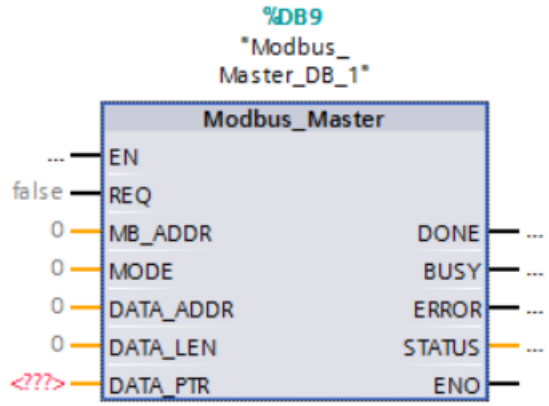
Uočava se kako registri adresa koje su heksadecimalno 1000 služe za iščitavanje određenih podataka, dok one koje su heksadecimalno 2000 za upisivanje. Ukoliko se želi promijeniti neki

od funkcijskih parametara, tada se njegova adresa pretvara na sljedeći način: funkcijski parametar F114 postaje heksadecimalni 010E.

Tu je još i fizičko sučelje u vidu RS 485 kabela, a za komunikaciju MODBUS protokolom potrebno je, u TIA Portalu, imati dva funkcijska blok dijagrama koji će biti dani sljedećim slikama.



Slika 3.11. *Komunikacijski blok za učitavanje komunikacije.*



Slika 3.12. *Master komunikacijski blok.*

Svrha komunikacijskog bloka za učitavanje „Modbus_Comm_Load“ je konfiguracija ulaza za modul CM PtP. Ulaz *EN* omogućava izvođenje bloka kada je istinit, dok se *REQ* izvršava kada očita rastući brid signala. *PORT* služi za postavljanje ID-a modula (259 – *Local~CM_PtP_1*), a u *BAUD* se unosi brzina prijenosa podataka (9600). *PARITY* služi za paritetnu provjeru koju se može

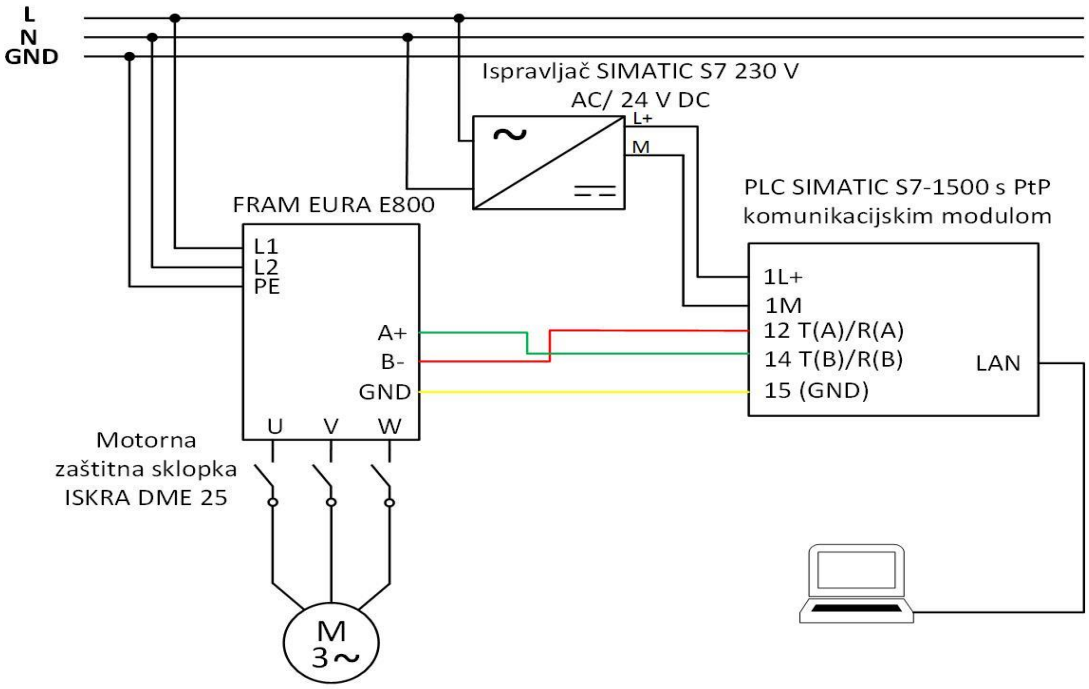
isključiti (0), ali može i biti neparna (1) ili parna (2), a za ovaj primjer bit će postavljena parna paritetna provjera. Iduća tri ulaza (*FLOW_CTRL*, *RTS_ON_DLY* i *RTS_OFF_DLY*) služe se kontrolu toka podataka, ali se postavljaju na 0. Ulaz *RESP_TO* je vrijeme odgovora *slave* jedinice *master* jedinici koje se ostavlja na početnu vrijednost 1000 ms. Posljednji ulaz *MB_DB* služi kao referenca „MB_Master“ bloku, te nakon unošenja („Modbus_Master_DB“.MB_DB) pojavljuje se identifikator bloka podataka u padajućoj listi „MB_DB“ prozora veze. Postoje i tri izlaza koji ukazuju na stanje izvođenja bloka. *DONE* prikazuje da se blok izvršio bez grešaka, a ukoliko to nije riječ, tada će izlaz *ERROR* izbaciti logičku jedinicu. Izlaz *STATUS* prikazuje status bloka te se preko njega isto može vidjeti radi li komunikacija ispravno ili pod greškom. Nakon što se veza između modula i frekvencijskog pretvarača ostvari, ovaj blok završava sa svojim dijelom posla te ga prepušta bloku „Modbus_Master“.

Blok „Modbus_Master“ predstavlja *master* jedinicu u komunikaciji i to je PLC uređaj, dok frekvencijski pretvarač predstavlja *slave* jedinicu. Izvršava se nakon bloka „Modbus_Comm_Load“ i služi i za slanje i primanje podataka (čitanje i pisanje). Ulaz *EN* omogućava izvođenje bloka, dok *REQ* ako je istinit zahtjeva prijenos podataka *slave* jedinici. *MB_ADDR* je adresa *slave* jedinice, odnosno frekvencijskog pretvarača, koja se postavi u funkcijskom parametru F900. *MODE* ulaz predstavlja način rada bloka, a to može biti, od potrebnih za ovaj diplomski rad, iščitavanje bitova, registara ili riječi (0), upisivanje jednog ili više bita ili registara (1), dijagnostika, tj. iščitavanje statusa greške (80). *DATA_ADDR* funkcionira po principu da se zbrajaju adrese za čitanje/pisanje (30001 – 39999 / 40001 – 49999 i 400001 – 465535) i željenog registara (čija se adresa pretvara iz heksadecimalnog u decimalni zapis) te se krajnja vrijednost upisuje. *DATA_LEN* je duljina podataka, odnosno broj registara do kojeg će „Modbus_Master“ čitati ili pisati neki podatak. U ulaz *DATA_PTR* unosi se blok podataka ili memorijska adresa bita na CPU-u odakle se podaci iščitavaju ili pišu. Funkcije izlaza *DONE*, *ERROR* i *STATUS* jednaki su kao i u prethodnom funkcijskom bloku, dok novi izlaz *BUSY* pokazuje ima li u danom trenutku prijenosa ili ne.

Princip rada protokola zasniva se na tomu da se komunikacija odvija serijski, odnosno u danom trenutku MODBUS može ili samo pisati ili samo čitati podatke, inače nastane greška.

3.4. Integracija sustava

Sustav, tj. shema sustava, koji se sastoji od frekvencijskog pretvarača, ispravljača, asinkronog motora, motorne zaštitne sklopke, PLC-a i laptopa je moguće vidjeti sljedećom slikom.



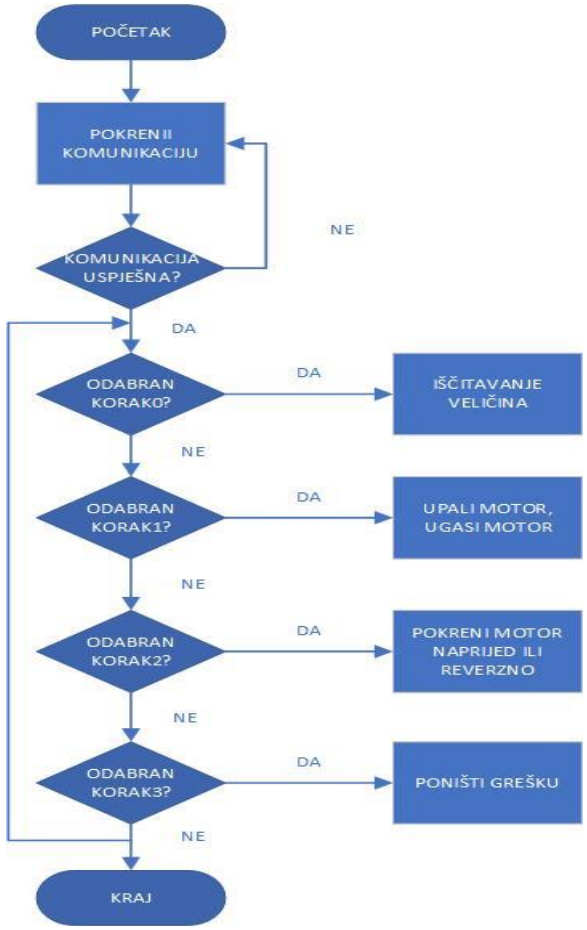
Slika 3.13. *Shema integriranog sustava.*

Kako je vidljivo, na mrežni napon spajaju se stezaljke frekvencijskog pretvarača. Motor se spaja preko sklopnika ISKRA, na stezaljke *U, V* i *W*. Ispravljač Simatic S7 služi za napajanje PLC uređaja njegovih modula. Preko LAN kabla laptop sa SCADA sučeljem se povezuje na PLC uređaj, dok se preko četverožilnog kabla (RS 485) spajaju frekvencijski pretvarač i PLC uređaj, s naglaskom da crni žila kabela ostaje izolirana i ne spaja se na PLC.Ovo će poglavlje još dati uvid u dijagrame tijeka kojima će se opisati tijek izvođenja radnji unutar PLC-a kojim će se programirati funkcioniranje frekvencijskog pretvarača, ali i princip rada grafičkog sučelja kojim će se približiti proces vizualno.

Ovo će poglavlje još dati uvid u dijagrame tijeka kojima će se opisati tijek izvođenja radnji unutar PLC-a kojim će se programirati funkcioniranje frekvencijskog pretvarača, ali i princip rada grafičkog sučelja kojim će se približiti proces vizualno.

3.4.1. PLC programsko rješenje

Ovo potpoglavlje daje na uvid tok informacija u PLC programskom kodu. Prikaz će se odviti preko dijagrama tijeka koji će biti dan idućom slikom.

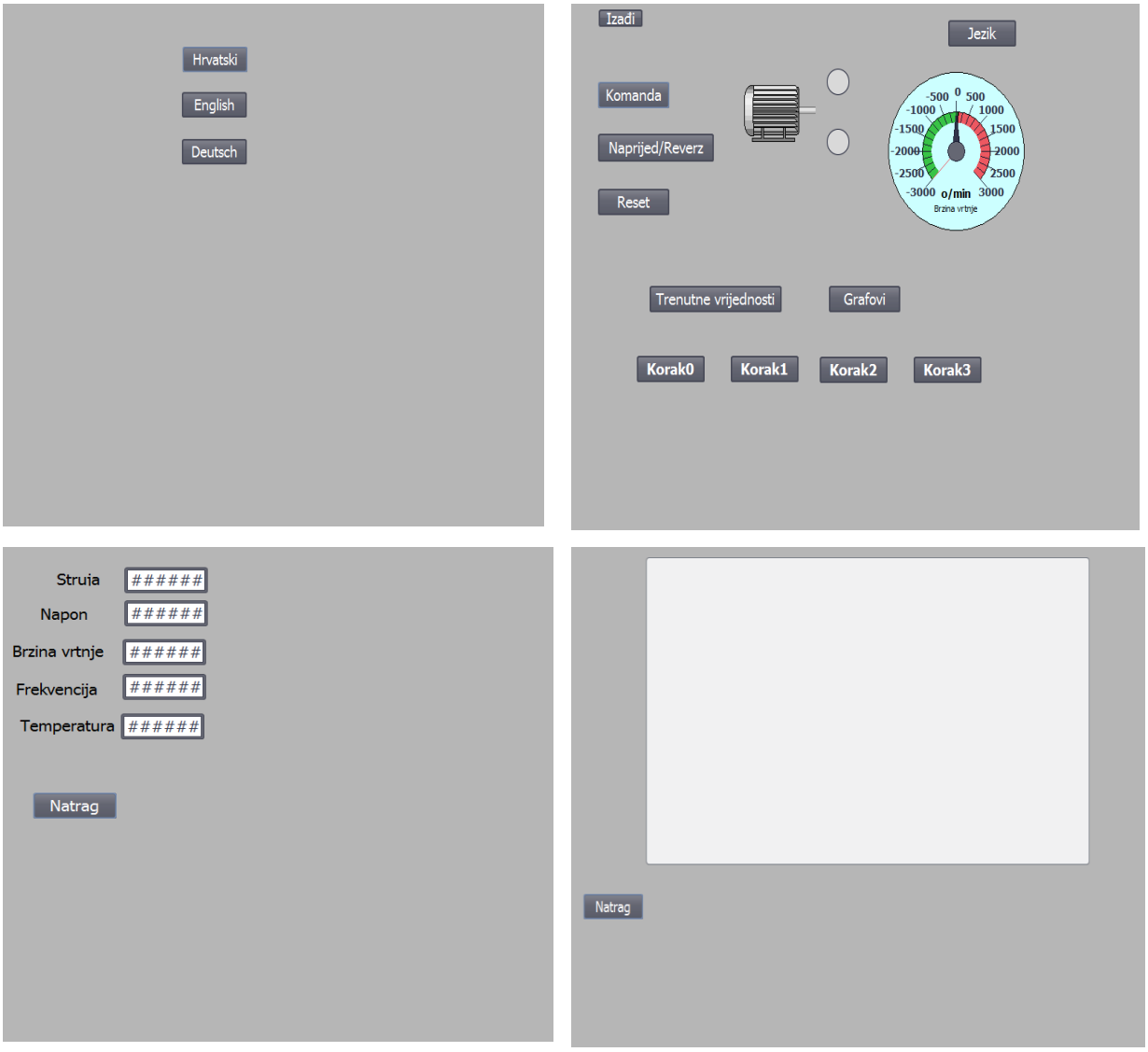


Slika 3.14. *Dijagram tijeka PLC programskog rješenja.*

Kako je vidljivo prethodnom slikom PLC programsko rješenje bazira se na izboru koraka koji svaki obavlja neku radnju. Program počinje uspostavljanjem komunikacije. Ukoliko se ona ne uspostavi, ponavlja se sve dok se ne uspostavi. Nakon toga korisnik odabere određeni korak za obavljanje određene radnje i taj dio programa se odvija ciklično. Svaki korak je predodređen za neku radnju, npr. ukoliko korisnik želi pokrenuti motor, tada će pritisnuti *Korak1* i odabrati komandu za paljenje motora, odnosno ugasit će motor ponovnim pritiskom na komandu. Prilozi 1. i 2. prikazuju programski kod (FBD i SCL), dok Prilog 3. prikazuje sve statičke varijable korištene unutar rješenja. Prilog 4. daje prikaz svih PLC tagova koji su korišteni.

3.4.2. Realizacija grafičkog sučelja (SCADA-e)

Grafičko sučelje unutar diplomskog rada koncipirano je na tri osnovna dijela. Osim što je na početnom izborniku omogućen izbor jezika, grafičko sučelje sastoji se od glavnog izbornika gdje su postavljene najvažnije komande, te dva podizbornika: podizbornik trenutnih vrijednosti ,gdje će se prikazati trenutne vrijednosti određenih veličina, i podizbornik grafa gdje će se tražene veličine prikazati grafički. Idućom slikom bit će prikazani izgledi izbornika i podizbornika na hrvatskom jeziku.



Slika 3.15. *Prikazi izbornika jezika (gore lijevo), glavnog izbornika (gore desno), podizbornika trenutnih vrijednosti (dolje lijevo) i podizbornika grafa (dolje lijevo).*

Dakle, početni zaslon daje na izbor korisniku koji jezik želi koristiti. Ukoliko on izabere engleski ili njemački, tada će mu se otvoriti glavni izbornici na tom jeziku, te će i podizbornici biti isto na stranom. Prilozi 5. i 6. prikazuju izgled HMI sučelja na engleskom (Prilog 5.), odnosno njemačkom jeziku (Prilog 6.), a Prilog 7. korištene HMI tagove. Ukoliko se naknadno želi promijeniti jezik, to je moguće odabirom tipke *Jezik* ili *Language* ili *Sprache* te tamo odabrati željeni jezik.

Glavni izbor omogućava davanje komande, odnosno gašenje motora, pokretanje motora naprijed ili u drugu stranu (reverziranje), te poništavanje greške. Te tri tipke nalaze se lijeve strane izbornika. U sredini je motor i dva kružića koji pomažu brzinskom čitanju stanja motora. Motor treperi kada je motor u pogonu, a i gornji kružić je zelen. Donji kružić pokazuje grešku u radu ili komunikaciji te treperi crveno jednom kada se greška pojavila. S desne strane nalazi se analogni prikaz brzine vrtnje, isto radi brzinskom iščitavanja stanja pogona. Iznad svega toga je tipka koja omogućuje izlaz iz HMI sučelja (*Izađi, Esc* i *Flucht*). Ispod svega toga stoje dvije tipke koje korisnika vode u podizbornike trenutnih vrijednosti i grafa. I na kraju su tipke *Korak0, Korak1, Korak2* i *Korak3* koje korisniku omogućuju radnje poput upisivanja u registre ili iščitavanja registara, a koje je detaljno objašnjeno u prošlom potpoglavlju dijagramom tijeka.

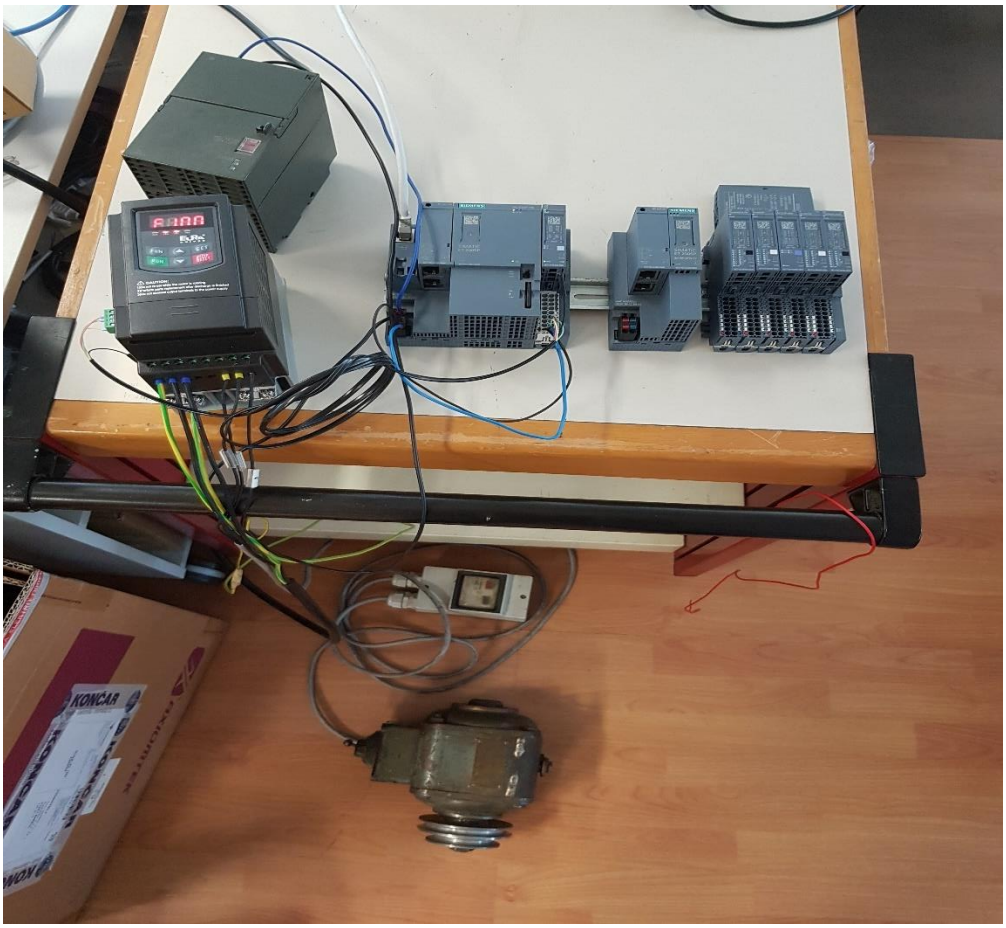
Podizbornik trenutnih vrijednosti daje korisniku digitalni prikaz trenutnih vrijednosti struje, napona, brzine vrtnje, frekvencije i temperature grijača frekvencijskog pretvarača. Podizbornik prikazuje grafički struju, napon i brzinu vrtnje, dakle to je skup trenutnih vrijednosti u vremenu.

4. TESTIRANJE

Ovo je najbitnije poglavlje u ovom radu iz razloga što će biti predstavljeni rezultati testiranja. Prikazat će se izgled integriranog sustava, te opisan postupak osposobljavanja istog. Uz to će se analizirati njegov rad i opisani problemi na koje je studene naišao tijekom izrade i pokretanja sustava.

4.1. Rezultati testiranja u realnom pogonu

Sustav je spojen prema shemi na slici 3.13. Na mrežu preko produžnog kabela spojeni su frekvencijski pretvarač i ispravljač. Laptop nije prikazan pošto je udaljen od ispitnog mjesta, ali je spojen preko Ethernet kabla na LAN ulaz PLC uređaja. Slika 4.1. daje završni prikazan ispitivanog integriranog sustava.



Slika 4.1. *Integrirani sustav u realnom pogonu.*

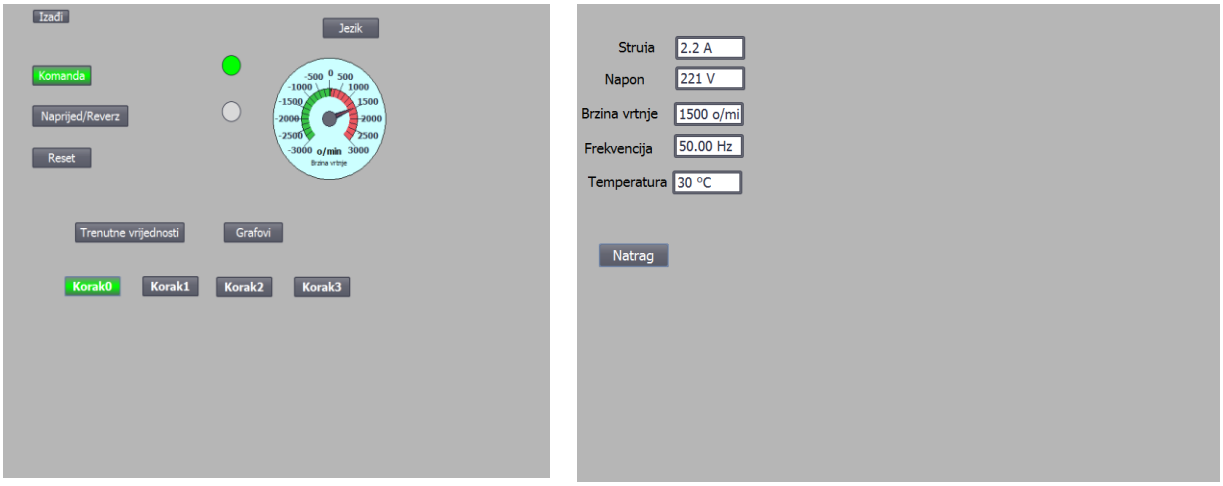
Iako je više modula spojeni na šini s PLC uređajem, oni nisu spojeni.

Nakon što je ostvarena komunikacija između PLC-a i laptopa, moglo se upravljati preko HMI sučelja. Ukoliko nešto ne bi funkcioniralo, moguće je bilo pokrenuti nadgledanje unutar samog programa u PLC-u tako provjeriti tok signala. Moguće je preko nadzora provjeriti stanje varijabla na isti način kao i signale. Time se lakše uklanjaju pogreške. Prvi korak je činilo iščitavanje veličina. Pritiskom na tipku *Korak0* i dalje na podizbornik *Trenutne vrijednosti* uvidjelo se kako iščitavanje veličina funkcionira. Sljedeća slika prikazuje izgled podizbornika za vrijeme prvog koraka.

Slika 4.2. *Iščitavanje veličina u podizborniku trenutnih vijednosti.*

Samo se iščitava vrijednost temperature grijača frekvencijskog pretvarača, dok su ostale vrijednosti nula.

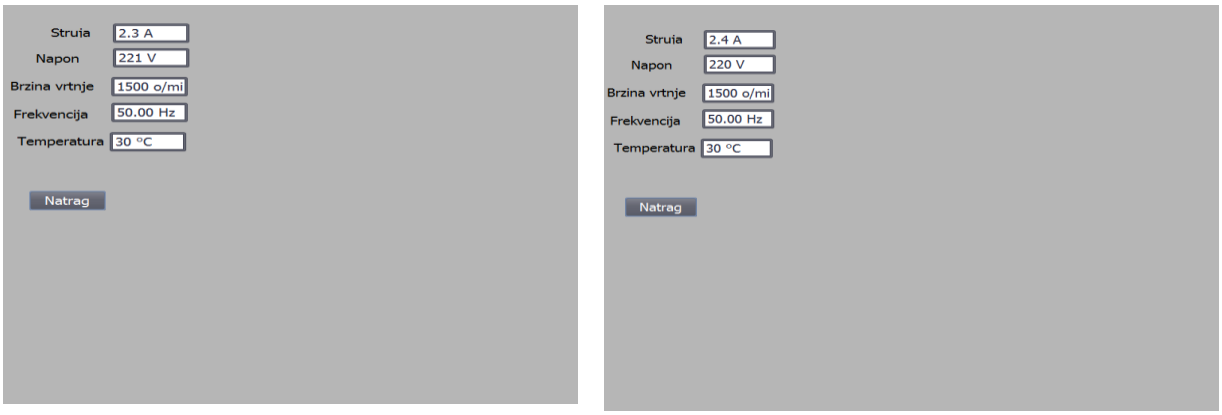
Idući korak je bilo upisivanje u motor da ga se pokrene, odnosno zaustavi. Prvo je potrebno pritisnuti tipku *Korak1*, a nakon toga tipku *Komanda*. Motor se pokrene, te sada korisnika zanimaju veličine. To je moguće tako da se ponovno pritisne tipka *Korak1*, te je sada potrebno pritisnuti *Korak0* za iščitavanje veličina. Moguće je vidjeti da s odstojanjem od nekih 4 s prikaz brzine vrtnje prikazuje brzinu vrtnje, a pritiskom na podizbornik trenutnih veličina dobiva se i detaljniji prikaz o ostalim veličinama. Slika 4.3. prikazuje kako je to izgledalo u HMI sučelju tijekom testiranja kombinacije drugog i prvog koraka.



Slika 4.3. *Izgled glavnog izbornika (lijevo) i podizbornika trenutnih vrijednosti (desno) za kombinaciju drugog i prvog koraka.*

Treba naglasiti kako je u trenutku slikanja zaslona motor nestao, no on treperi.

Treći korak je pokretanje motora naprijed ili reverzno, te su tu u kombinaciji tri koraka. *Korak2* služi da bi se motor pokrenuo, odnosno reverzirao. *Korak0* iščitava veličine, koje su iste i za naprijed i za reverz pošto se koristi UInt varijabla za podatke (*eng. Unsigned Integer*). Također treperi motor za rad, te svijetli zeleno. Gašenje motora iz ovog koraka nije moguće, već je potrebno da tipka *Komanda* ima bit 0, te pritiskom na tipku *Korak1* motor se gasi deceleracijom. Slika 4.4. prikazuje očitavanje veličina za naprijed i reverzni rad motora.



Slika 4.4. *Prikaz veličina za rad motora naprijed (lijevo) i reverz (desno).*

Poništavanje greške funkcionira da se pojavljivanjem greške pali trepereći (donji) kružić. Pregleda se status i dio programa zašto je došlo do greške, te kada se greška otklonila pritisne se *Korak3* i tako se poništi greška i motor je spreman za daljnji rad.

4.2. Analiza realnog pogona

Prvi potreban korak da se sve uspješno spoji je odabrati odgovarajuće korištene uređaje u kartici *Devices & Network*. Nakon toga se odgovarajuće spoje virtualno uređaji. Unutar svojstava virtualnog PLC uređaja moguće je omogućiti korištenje sat memorijskih bajtova koji će poslužiti prilikom brojanja sekundi ili neke druge vremenske jedinice. Potrebno je i postaviti IP adresu laptopa, koju je moguće naći unutar upravljačke ploče laptopa (za ovaj slučaj je 192.168.0.2). Potrebno je i prevesti (*eng. compile*) u strojni jezik, te se tek tada povezuje laptop i PLC uređaj (uz uvjet da je u fizičkome svijetu sve dobro spojeno).

Naravno, tu je potrebno i postaviti komunikacijske parametre (F900 – F904) i osnovne parametre motora (F800 – F805 i F810) u frekvencijskom pretvaraču s ciljem besprijekornog spajanja.

Sam PLC uređaj sastavljen je od CPU-a i PtP modula. Modul je potrebno dobro isparametrirati, te se u svojstvima pod sučelje postavlja polusmjerna dvožična RS 485 komunikacija. Tu se postavljaju i osnovni parametri poput brzine prijenosa, paritetne provjere, broja podatkovnih bitova i stop bitova. Identifikator sklopovlja govori koja je adresa modula, te će ona biti upisana u ulaz *PORT* funkcijskog blok dijagrama „Modbus_Comm_Load“. Ako se program ne uspije prevesti, moguće je preko prevođenja unutar opcije *Online tools* ponovno izgraditi (*eng. rebuild all*) programsku podršku ili sklopovlje. Osim adresu za PLC potrebno je namjestiti i adresu za HMI sučelje, preko protokola IPv4 preko upravljačke ploče (za ovaj slučaj 192.168.0.1). Tijekom uspostavljanja veze odabire se odgovarajući adapter za Ethernet koje je moguće provjeriti u *Device Manageru*.

Za funkcionalnost programa još je izrazito važno promijeniti *MODE* parametar funkcijskog blok dijagrama „Modbus_Comm_Load_DB“. To je moguće napraviti unutar kartice *System blocks* → *Program resources* gdje se upisuje za početnu vrijednost 16#04 što znači da će se odvijati polusmjerna dvožična RS 485 komunikacija. Nakon toga sve što preostaje je napraviti smisleni program za ono što se želi postići.

S obzirom na to da se ciklus PLC-a izvodi vrlo brzo, vrlo često je dolazilo do ispreplitanja koraka, no uz određeno vremensko odstupanje na kraju bi program odradio sve poslove koji su mu dani. To bi se događalo kada bi se prebacivalo iz *Koraka1* ili *Koraka2* u *Korak0* kako bi se omogućilo iščitavanje veličina prilikom rada motora. Također slično se odvijalo kada bi se trebalo ugasiti motor, te se prebacivalo iz *Koraka2* s radom naprijed ili reverznim radom motora u *Korak1*. Sve je ovisilo koliko je komunikacija bila zauzeta. S koracima pojedinačno nije bilo vremenskog odstupanja.

5. ZAKLJUČAK

Ovaj diplomski rad upoznao je studenta dodatno s digitalnim regulatorima, te je prošireno znanje stečeno kroz kolegije na Fakultetu. Osim digitalnih regulatora i teorije iza njihovog principa rada, student se pobliže se upoznao i s teorijom MODBUS komunikacijom te funkcioniranjem iste. Uočeno je kako ima dosta komplikacija prilikom parametriranja i da se treba na štošta paziti. Pobliže su proučene i funkcije frekvencijskog pretvarača, te opcije registara korištenih za MODBUS komunikaciju.

Unutar TIA Portala pobliže je student upoznat s dodatnim funkcijama koje su potrebne za osposobiti MODBUS komunikaciju. S HMI sučeljem nije imao puno problema, te su napravljene određene korekcije u skladu s funkcioniranjem programa. Glavni problem je bio ostvariti komunikaciju, te je uz razne preinake to i ostvareno. Nakon što je komunikacija ostvarena, problem unutar same komunikacije se nije pojavljivao, te ona nije nijednom „pukla“.

Rezultati koji su dobiveni su zadovoljavajući pošto se komunikacijom ostvarilo ono što je traženo, iako uz mala vremenska odstupanja. Sve je funkcioniralo iz prve, te je motor mogao biti pokrenut, ugašen, reverziran, a bilo je moguće i iščitavati tražene veličine, te poništiti grešku. Ta odstupanja nastala su prilikom izvođenja kompleksnijih radnji (korištenja drugih koraka uz prvotni). Vrlo bitno je naglasiti kako je ovo realna situacija u pogonu, te se tako student mogao susresti s izazovima koji ga očekuju u budućnosti.

Ono što je moguće doraditi u ovome programskom rješenju je da se potpuno automatiziraju koraci (ono što u pokušaju nije uspjelo, jer se ciklus previše brzo odvije da bi mogao odraditi predloženo programsko rješenje), te dolazi do raznih pogrešaka. Također je moguće bolje obraditi signale, pa npr. za negativnu brzinu vrtnje postaviti da ide u negativnu vrijednost. Moguće je i namjestiti da se motor pokrene na određenu brzinu preko frekvencije. Praktički, svaki od parametara unutar frekvencijskog pretvarača moguće je postaviti preko MODBUS komunikacije, no za to ipak treba više iskustva.

LITERATURA

[1] Wikipedija: Automatizacija, https://bit.ly/2KejEs5 (pristup ostvaren 16.10.2017.)

[2] Wikipedia: Automation, https://bit.ly/1B8wgXR (pristup ostvaren 18.10.2017.)

[3] F. Belić, Vizualizacija industrijskog procesa, diplomski rad, Osijek, 2008.

[4] D. Slišković, Procesna automatizacija, skripta s predavanja, Elektrotehnički fakultet u Osijeku, Osijek, 2009.

[5] M. Živčić, MODBUS, seminarski rad, FER, Zagreb, 2007.

[6] Wikipedia: Modbus, https://bit.ly/1T11959 (pristup ostvaren 25.01.2018.)

[7] I. Flegar, Elektronički energetski pretvarači, Kigen, Zagreb, 2010.

[8] Gozuk: What is Frequency Converter? How it works?, https://bit.ly/2tA3z5Q (pristup ostvaren 18.01.2018.)

[9] E-elektro, Trofazni asinkroni motor s kaveznim rotorom, https://bit.ly/2KeEN5i (pristup ostvaren 23.01.2018.)

[10] Z. Valter, Osnove električnih strojeva, ETF Osijek, 2007.

[11] Siemens, TIA Portal, https://sie.ag/2r2kzhm (pristup ostvaren 6.12.2017.)

[12] Distribuirani sustav ET 200SP CPU 1512SP-1 PN, priručnik, https://bit.ly/2tBHWSB (pristup ostvaren 28.06.2018.)

[13] Modul CM PtP, priručnik, https://sie.ag/2KuqNR5 (pristup ostvaren 28.06.2018.)

[14] Simatic S7 PS307, priručnik, https://sie.ag/2D4gFiR (pristup ostvaren 15.09.2018.)

[15] Eura Drives: Frekvencijski pretvarač EURA E800, priručnik

[16] H. Berger, SIMATIC automatizacijski sustavi, SIEMENS, Graphis, Zagreb, 2013.

SAŽETAK

U ovome radu je ukratko opisan proces automatizacije. Objašnjeni su općenito programabilni logički kontroleri (PLC-ovi), SCADA sustavi (HMI sučelje), komunikacijski protokoli (s naglaskom na MODBUS), te frekvencijski pretvarači. Također, dan je kratki opis asinkronog motora. Nakon toga je napravljen program u programskom paketu TIA Portal, s HMI sučeljem, te je sve spojeno preko MODBUS RTU/RS485 protokola na osobno računalo. Preko osobnog računala se upravlja asinkroni motor, te je moguće očitavati trenutne vrijednosti struje, snage i brzine vrtnje.

Ključne riječi: asinkroni motor, frekvencijski pretvarač, komunikacijski protokoli, HMI, MODBUS, PLC, TIA Portal

ABSTRACT

Program solution for data exchange between frequency converter and SIMATIC 1500 PLC based on MODBUS RTU/RS485 protocol

In this paper automation process has been shorty described. Generally, programmable logic controllers (PLCs), SCADA systems (HMI interface), communication protocols (focusing on MODBUS) and frequency converters have been explained. As well, short description of asynchronous motor has been given. Afterwards, program has been made in software package TIA Portal, with HMI interface, and all have been connected via MODBUS RTU/RS485 protocol on PC. Via PC asynchronous motor has been controlled and instant current, power and rotation speed values could have been determined.

Keywords: asynchronous motor, communication protocols, frequency converter, HMI, MODBUS, PLC, TIA Portal

ŽIVOTOPIS

Rođen 14.08.1993. u Našicama, trenutno prebivalište je u Osijeku. Pohađao je osnovnu školu Jagode Truhelke u Osijeku te završio s odličnim uspjehom. Nakon toga upisuje zagrebačku XV. gimnaziju koju završava s vrlo dobrim uspjehom. 2012. godine upisuje preddiplomski studij elektrotehnike na Elektrotehničkom fakultetu u Osijeku, a 2013. se usmjerava na elektroenergetiku, koju završava 2015. s vrlo dobrim uspjehom. Iste godine na sveučilišnom diplomskom studiju upisuje izborni blok industrijska elektroenergetika na Fakultetu elektrotehnike, računarstva i informacijskih tehnologija u Osijeku. Dobro se služi engleskim jezikom, te ima osnovno znanje njemačkog. Od računalnih vještina ima dobro vladanje Microsoft Office alatima, te osnovna znanja u programskim paketima Matlab, Ansys, TIA Portal i EPLAN. Nastupao je na Festivalima znanosti s temama „Pretvaranje svjetlosnog elektromagnetskog vala u električnu energiju“ (2014.), „Solarne elektrane“ (2015.), „Znanost i umjetnost nanotvari“ (2016.), „Kuća novog vremena“ (2017.), te „Otkrića Nikole Tesle“ (2018.). 2017. g. je odradio praksu od pet tjedana u firmi „Teo-Belišće d.o.o.“ pod vodstvom mentora Filipa Belića u kojoj se bolje upoznao s programskim paketom TIA Portal. Kroz seminarski radi iz kolegija „Elementi automatike“ stekao je osnovna znanja iz lemljenja elektroničkih komponenti, te se upoznao s mikrokontrolerom Arduino. U slobodno vrijeme se bavi košarkom i vozi bicikl.

Student:
Luka Šarić

_____

PRILOG 1. FBD DIO KODA U TIA PORTALU

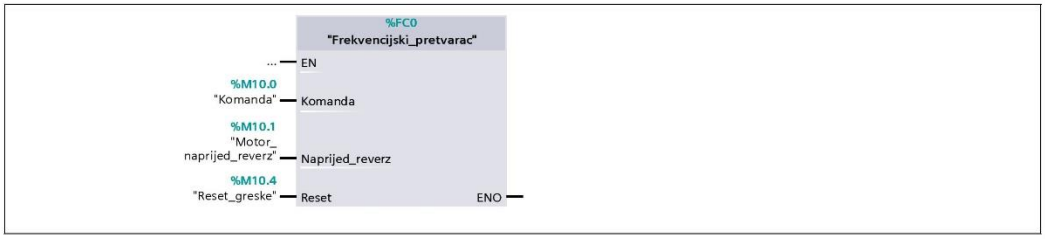
Totally Integrated Automation Portal		

Main [OB1]

Main Properties					
General					
Name	Main	**Number**	1	**Type**	OB
Language	FBD	**Numbering**	Automatic		
Information					
Title	"Main Program Sweep (Cycle)"	**Author**		**Comment**	
Family		**Version**	0.1	**User-defined ID**	

Name	Data type	Default value	Supervision	Comment
Input				
Initial_Call	Bool			Initial call of this OB
Remanence	Bool			=True, if remanent data are available
Temp				
Constant				

Network 1:



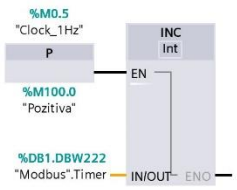
Diplomski_rad_Saric_Luka_4 / PLC_1 [CPU 1512SP-1 PN] / Program blocks

Frekvencijski_pretvarac [FC0]

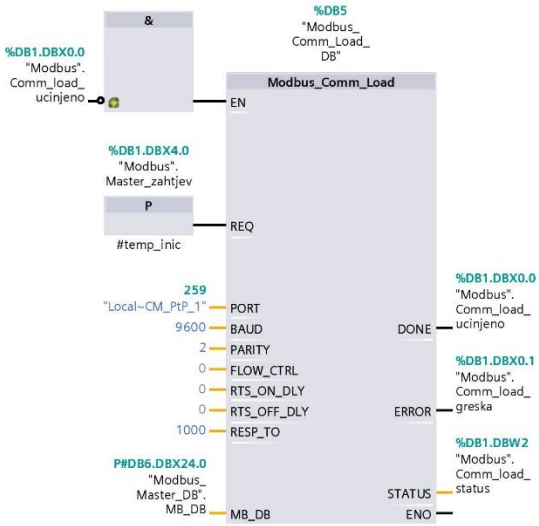
Frekvencijski_pretvarac Properties					
General					
Name	Frekvencijski_pretvarac	**Number**	0	**Type**	FC
Language	FBD	**Numbering**	Manual		
Information					
Title		**Author**		**Comment**	
Family		**Version**	0.1	**User-defined ID**	

Frekvencijski_pretvarac				
Name	**Data type**	**Default value**	**Supervision**	**Comment**
▼ Input				
Komanda	Bool			
Naprijed_reverz	Bool			
Reset	Bool			
Output				
InOut				
▼ Temp				
temp_inic	Bool			
temp1	Bool			
Constant				
▼ Return				
Frekvencijski_pretvarac	Void			

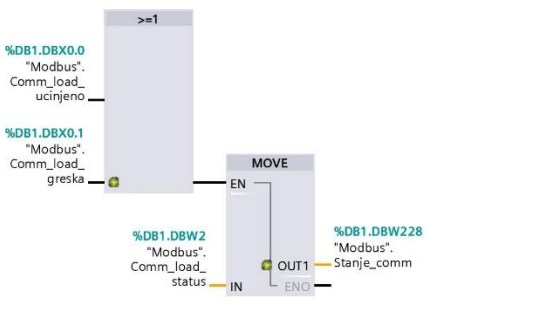
Network 1:



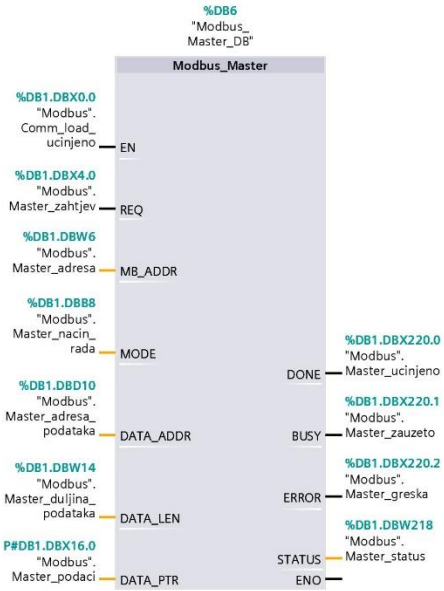
Network 2:



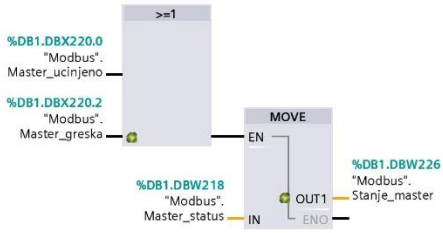
Network 3:



Network 4:



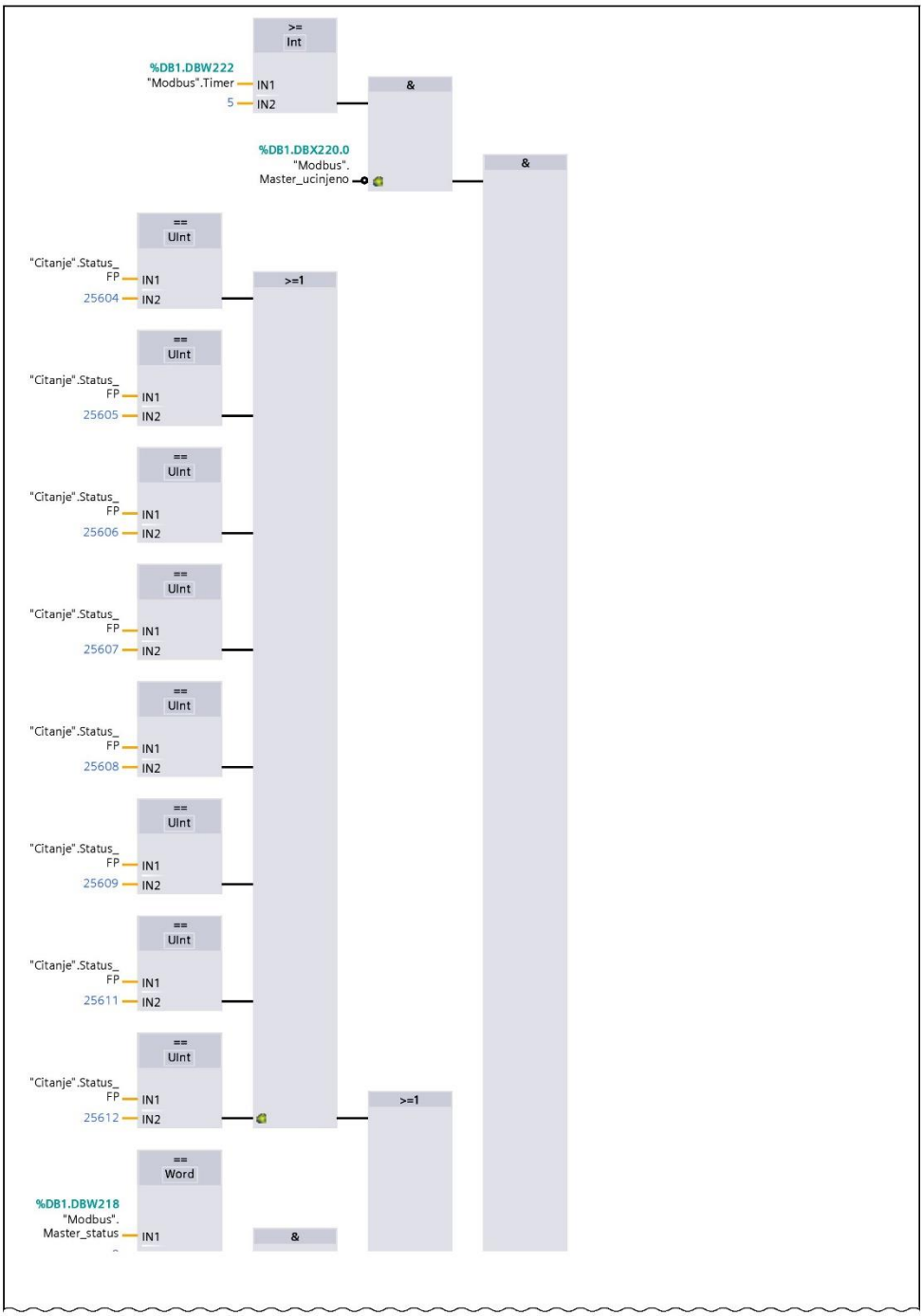
Network 5:



Network 6:

Network 7:

Network 7: (1.1 / 2.1)

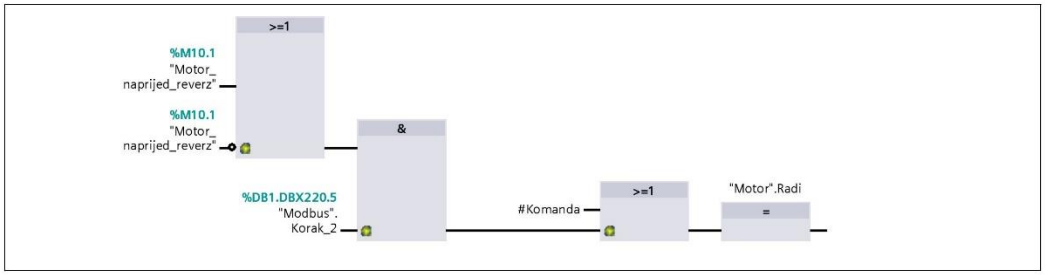


2.1 (Page1 - 5)

Network 7: (2.1 / 2.1)

Totally Integrated Automation Portal		

Network 8:



PRILOG 2. SCL DIO KODA U TIA PORTALU

Totally Integrated Automation Portal

Diplomski_rad_Saric_Luka_4 / PLC_1 [CPU 1512SP-1 PN] / Program blocks

SCL dio programa [FC1]

SCL dio programa Properties					
General					
Name	SCL dio programa	**Number**	1	**Type**	FC
Language	SCL	**Numbering**	Manual		
Information					
Title		**Author**		**Comment**	
Family		**Version**	0.1	**User-defined ID**	

SCL dio programa				
Name	**Data type**	**Default value**	**Supervision**	**Comment**
▼ Input				
Komanda	Bool			
Naprijed_reverz	Bool			
Reset	Bool			
Output				
InOut				
▼ Temp				
temp	Bool			
Constant				
▼ Return				
SCL dio programa	Void			

```
0001 "Motor".Komanda:= #Komanda;
0002 "Motor".Iskljucen := NOT #Komanda;
0003 "Motor".Naprijed := NOT #Naprijed_reverz;
0004 "Motor".Reverz := #Naprijed_reverz;
0005
0006 REGION "Citanje velicina"
0007   IF "Modbus".Korak = TRUE THEN
0008     "Modbus".Master_zahtjev := TRUE;
0009     "Modbus".Master_adresa := 1;
0010     "Modbus".Master_nacin_rada := 0;
0011     "Modbus".Master_adresa_podataka := 44097;
0012     "Modbus".Master_duljina_podataka := 24;
0013     "Citanje".Frekvencija := "Modbus".Master_podaci[0];
0014     "Citanje".Napon := "Modbus".Master_podaci[1];
0015     "Citanje".Struja := "Modbus".Master_podaci[2];
0016     "Citanje".Broj_polova_kontrola := "Modbus".Master_podaci[3];
0017     "Citanje".Napon_sabirnice := "Modbus".Master_podaci[4];
0018     "Citanje".Status_FP := "Modbus".Master_podaci[5];
0019     "Citanje".Moment := "Modbus".Master_podaci[6];
0020     "Citanje".Temperatura_grijaca_FP := "Modbus".Master_podaci[7];
0021     "Citanje".PID_dana_vrijednost := "Modbus".Master_podaci[8];
0022     "Citanje".PID_vrijednost_PV := "Modbus".Master_podaci[9];
0023     "Citanje".Snaga_int := "Modbus".Master_podaci[10];
0024     "Citanje".OP_status_terminala := "Modbus".Master_podaci[11];
0025     "Citanje".Izlazni_status_terminala := "Modbus".Master_podaci[12];
0026     "Citanje".AI1 := "Modbus".Master_podaci[13];
0027     "Citanje".AI2 := "Modbus".Master_podaci[14];
```

```
0028      "Citanje".AI3 := "Modbus".Master_podaci[15];
0029      "Citanje".Rezervirano := "Modbus".Master_podaci[16];
0030      "Citanje".Rezervirano1 := "Modbus".Master_podaci[17];
0031      "Citanje".Rezervirano2 := "Modbus".Master_podaci[18];
0032      "Citanje".Faza_brzine := "Modbus".Master_podaci[19];
0033      "Citanje".Vanjski_brojac := "Modbus".Master_podaci[20];
0034      "Citanje".AO1 := "Modbus".Master_podaci[21];
0035      "Citanje".AO2 := "Modbus".Master_podaci[22];
0036      "Citanje".Brzina := "Modbus".Master_podaci[23];
0037      "Citanje".Snaga := "Modbus".Master_podaci[24];
0038      "Modbus".Timer := 0;
0039    END_IF; //Citanje
0040    IF "Modbus".Korak = TRUE AND ("Modbus".Master_ucinjeno = TRUE OR "Mod-
      bus".Master_greska = TRUE) THEN
0041      "Modbus".Master_zahtjev := FALSE;
0042   END_IF;
0043 END_REGION
0044
0045 REGION "Pisanje: upali/ugasi "
0046    IF "Motor".Komanda = TRUE AND "Modbus".Korak_1 = TRUE  THEN
0047      "Modbus".Master_zahtjev := TRUE;
0048      "Modbus".Master_adresa := 1;
0049      "Modbus".Master_nacin_rada := 1;
0050      "Modbus".Master_adresa_podataka := 48193;
0051      "Modbus".Master_duljina_podataka := 1;
0052      "Modbus".Master_podaci[0] := 8;//Pisanje: upali motor
0053      "Modbus".Timer := 0;
0054
0055
0056    ELSIF "Motor".Iskljucen = TRUE AND "Modbus".Korak_1 = TRUE THEN
0057      "Modbus".Master_zahtjev := TRUE;
0058      "Modbus".Master_adresa := 1;
0059      "Modbus".Master_nacin_rada := 1;
0060      "Modbus".Master_adresa_podataka := 48193;
0061      "Modbus".Master_duljina_podataka := 1;
0062      "Modbus".Master_podaci[0] := 3; //Pisanje: iskljuci motor deceleracijom
0063      "Modbus".Timer := 0;
0064    END_IF;
0065
0066    IF "Modbus".Korak_1 = TRUE AND ("Modbus".Master_ucinjeno = TRUE OR "Mod-
      bus".Master_greska = TRUE) THEN
0067      "Modbus".Master_zahtjev := FALSE;
0068    END_IF;
0069 END_REGION
0070 REGION "Pisanje": Motor naprijed/reverz"
0071    IF "Motor".Naprijed = TRUE AND "Modbus".Korak_2 = TRUE THEN
0072      "Modbus".Master_zahtjev := TRUE;
0073      "Modbus".Master_adresa := 1;
0074      "Modbus".Master_nacin_rada := 1;
0075      "Modbus".Master_adresa_podataka := 48193;
0076      "Modbus".Master_duljina_podataka := 1;
0077      "Modbus".Master_podaci[0] := 1; //Pisanje: upali motor naprijed
0078      "Modbus".Timer := 0;
0079
0080    ELSIF "Motor".Reverz = TRUE AND "Modbus".Korak_2 = TRUE THEN
0081      "Modbus".Master_zahtjev := TRUE;
0082      "Modbus".Master_adresa := 1;
0083      "Modbus".Master_nacin_rada := 1;
0084      "Modbus".Master_adresa_podataka := 48193;
```

```
0085      "Modbus".Master_duljina_podataka := 1;
0086      "Modbus".Master_podaci[0] := 2; //Pisanje: upali motor reverz
0087      "Modbus".Timer := 0;
0088    END_IF;
0089    IF "Modbus".Korak_2 = TRUE AND ("Modbus".Master_ucinjeno = TRUE OR "Mod-
      bus".Master_greska = TRUE) THEN
0090      "Modbus".Master_zahtjev := FALSE;
0091    END_IF;
0092  END_REGION
0093  REGION "Pisanje: reset greske"
0094    IF "Reset_greske" = TRUE AND "Modbus".Korak_3 = TRUE THEN
0095      "Modbus".Master_zahtjev := TRUE;
0096      "Modbus".Master_adresa := 1;
0097      "Modbus".Master_nacin_rada := 1;
0098      "Modbus".Master_adresa_podataka := 48193;
0099      "Modbus".Master_duljina_podataka := 1;
0100      "Modbus".Master_podaci[0] := 5; //Pisanje: reset greske
0101      "Modbus".Timer := 0;
0102      "Modbus".Korak := 0;
0103    END_IF;
0104    IF "Modbus".Korak_3 = TRUE AND ("Modbus".Master_ucinjeno = TRUE OR "Mod-
      bus".Master_greska = TRUE) THEN
0105      "Modbus".Master_zahtjev := FALSE;
0106      "Modbus".Greska := FALSE;
0107    END_IF;
0108  END_REGION
0109
```

Symbol	Address	Type	Comment
"Citanje".AI1		UInt	
"Citanje".AI2		UInt	
"Citanje".AI3		UInt	
"Citanje".AO1		UInt	
"Citanje".AO2		UInt	
"Citanje".Broj_polo-va_kontrola		UInt	
"Citanje".Brzina		UInt	
"Citanje".Faza_brzine		UInt	
"Citanje".Frekvencija		UInt	
"Citanje".Izlazni_sta-tus_terminala		UInt	
"Citanje".Moment		UInt	
"Citanje".Napon		UInt	
"Citanje".Napon_sabir-nice		UInt	
"Citanje".OP_status_ter-minala		UInt	
"Citanje".PID_dana_vri-jednost		UInt	
"Citanje".PID_vrijed-nost_PV		UInt	
"Citanje".Rezervirano		UInt	
"Citanje".Rezervirano1		UInt	
"Citanje".Rezervirano2		UInt	
"Citanje".Snaga		UInt	
"Citanje".Snaga_int		UInt	
"Citanje".Status_FP		UInt	
"Citanje".Struja		UInt	
"Citanje".Tempera-tura_grijaca_FP		UInt	

Totally Integrated Automation Portal

Symbol	Address	Type	Comment
"Citanje".Vanjski_brojac		UInt	
"Modbus".Greska	%DB1.DBX224.0	Bool	
"Modbus".Korak	%DB1.DBX220.3	Bool	
"Modbus".Korak_1	%DB1.DBX220.4	Bool	
"Modbus".Korak_2	%DB1.DBX220.5	Bool	
"Modbus".Korak_3	%DB1.DBX220.6	Bool	
"Modbus".Master_adresa	%DB1.DBW6	Word	
"Modbus".Master_adresa_podataka	%DB1.DBD10	UDInt	
"Modbus".Master_duljina_podataka	%DB1.DBW14	UInt	
"Modbus".Master_greska	%DB1.DBX220.2	Bool	
"Modbus".Master_nacin_rada	%DB1.DBB8	USInt	
"Modbus".Master_podaci[0]		UInt	
"Modbus".Master_podaci[1]		UInt	
"Modbus".Master_podaci[2]		UInt	
"Modbus".Master_podaci[3]		UInt	
"Modbus".Master_podaci[4]		UInt	
"Modbus".Master_podaci[5]		UInt	
"Modbus".Master_podaci[6]		UInt	
"Modbus".Master_podaci[7]		UInt	
"Modbus".Master_podaci[8]		UInt	
"Modbus".Master_podaci[9]		UInt	
"Modbus".Master_podaci[10]		UInt	
"Modbus".Master_podaci[11]		UInt	
"Modbus".Master_podaci[12]		UInt	
"Modbus".Master_podaci[13]		UInt	
"Modbus".Master_podaci[14]		UInt	
"Modbus".Master_podaci[15]		UInt	
"Modbus".Master_podaci[16]		UInt	
"Modbus".Master_podaci[17]		UInt	
"Modbus".Master_podaci[18]		UInt	
"Modbus".Master_podaci[19]		UInt	
"Modbus".Master_podaci[20]		UInt	
"Modbus".Master_podaci[21]		UInt	
"Modbus".Master_podaci[22]		UInt	

Symbol	Address	Type	Comment
"Modbus".Master_poda-ci[23]		UInt	
"Modbus".Master_poda-ci[24]		UInt	
"Modbus".Master_ucinje-no	%DB1.DBX220.0	Bool	
"Modbus".Master_zaht-jev	%DB1.DBX4.0	Bool	
"Modbus".Timer	%DB1.DBW222	Int	
"Motor".Iskljucen		Bool	
"Motor".Komanda		Bool	
"Motor".Naprijed		Bool	
"Motor".Reverz		Bool	
"Reset_greske"	%M10.4	Bool	
#Komanda		Bool	
#Naprijed_reverz		Bool	

PRILOG 3. STATIČKE VARIJABLE KORIŠTENE U PROGRAMU

Totally Integrated Automation Portal

Diplomski_rad_Saric_Luka_4 / PLC_1 [CPU 1512SP-1 PN] / Program blocks

Citanje [DB2]

Citanje Properties					
General					
Name	Citanje	**Number**	2	**Type**	DB
Language	DB	**Numbering**	Manual		
Information					
Title		**Author**		**Comment**	
Family		**Version**	0.1	**User-defined ID**	

Citanje

Name	Data type	Start value	Retain	Accessible from HMI/OPC UA	Writable from HMI/OPC UA	Visible in HMI engineering	Set-point	Supervision	Comment
Static									
Frekvencija	UInt	0	False	True	True	True	False		
Napon	UInt	0	False	True	True	True	False		
Struja	UInt	0	False	True	True	True	False		
Broj_polova_kontrola	UInt	0	False	True	True	True	False		
Napon_sabirnice	UInt	0	False	True	True	True	False		
Status_FP	UInt	0	False	True	True	True	False		
Moment	UInt	0	False	True	True	True	False		
Temperatura_grijaca_FP	UInt	0	False	True	True	True	False		
PID_dana_vrijednost	UInt	0	False	True	True	True	False		
PID_vrijednost_PV	UInt	0	False	True	True	True	False		
Snaga_int	UInt	0	False	True	True	True	False		
OP_status_terminala	UInt	0	False	True	True	True	False		
Izlazni_status_terminala	UInt	0	False	True	True	True	False		
AI1	UInt	0	False	True	True	True	False		
AI2	UInt	0	False	True	True	True	False		
AI3	UInt	0	False	True	True	True	False		
Rezervirano	UInt	0	False	True	True	True	False		

Totally Integrated Automation Portal		

Name	Data type	Start value	Retain	Accessible from HMI/OPC UA	Writable from HMI/OPC UA	Visible in HMI engineering	Setpoint	Supervision	Comment
Rezervirano1	UInt	0	False	True	True	True	False		
Rezervirano2	UInt	0	False	True	True	True	False		
Faza_brzine	UInt	0	False	True	True	True	False		
Vanjski_brojac	UInt	0	False	True	True	True	False		
AO1	UInt	0	False	True	True	True	False		
AO2	UInt	0	False	True	True	True	False		
Brzina	UInt	0	False	True	True	True	False		
Snaga	UInt	0	False	True	True	True	False		
Izlazna_struja_visoki	Byte	16#0	False	True	True	True	False		
Izlazna_struja_niski	Byte	16#0	False	True	True	True	False		
Spremnost_pretvaraca	Byte	16#0	False	True	True	True	False		
Status_FP_cekanje	Byte	16#0	False	True	True	True	False		
Status_FP_naprijed	Byte	16#0	False	True	True	True	False		
Status_FP_reverz	Byte	16#0	False	True	True	True	False		
Status_FP_prekostruja	Byte	16#0	False	True	True	True	False		
Status_FP_DC_prekostruja	Byte	16#0	False	True	True	True	False		
Status_FP_gubitak_ulazne_faze	Byte	16#0	False	True	True	True	False		
Status_FP_nadopterecenje_frekvencije	Byte	16#0	False	True	True	True	False		
Status_FP_podnapon	Byte	16#0	False	True	True	True	False		
Status_FP_pregrijavanje	Byte	16#0	False	True	True	True	False		
Status_FP_nadopterecenje_motora	Byte	16#0	False	True	True	True	False		
Status_FP_interferencija	Byte	16#0	False	True	True	True	False		
Status_FP_LL	Byte	16#0	False	True	True	True	False		
Status_FP_vanjski_kvar	Byte	16#0	False	True	True	True	False		
Status_FP_greska1	Byte	16#0	False	True	True	True	False		

Totally Integrated Automation Portal		

Name	Data type	Start value	Retain	Accessible from HMI/OPC UA	Writable from HMI/OPC UA	Visible in HMI engineering	Setpoint	Supervision	Comment
Status_FP_greska2	Byte	16#0	False	True	True	True	False		
Status_FP_greska3	Byte	16#0	False	True	True	True	False		
Status_FP_greska4	Byte	16#0	False	True	True	True	False		
Status_FP_OC1	Byte	16#0	False	True	True	True	False		
Status_FP_PF0	Byte	16#0	False	True	True	True	False		
Status_FP_zastita_analognog_iskljucivanja	Byte	16#0	False	True	True	True	False		
Status_FP_EP3	Byte	16#0	False	True	True	True	False		
Status_FP_zastita_podopterecenja	Byte	16#0	False	True	True	True	False		
Status_FP_PP	Byte	16#0	False	True	True	True	False		
Status_FP_zastita_kontrole_pritiska	Byte	16#0	False	True	True	True	False		
Status_FP_pogreska_postavljanja_PID_parametara	Byte	16#0	False	True	True	True	False		

Diplomski_rad_Saric_Luka_4 / PLC_1 [CPU 1512SP-1 PN] / Program blocks

Modbus [DB1]

Modbus Properties					
General					
Name	Modbus	**Number**	1	**Type**	DB
Language	DB	**Numbering**	Manual		
Information					
Title		**Author**		**Comment**	
Family		**Version**	0.1	**User-defined ID**	

Modbus

Name	Data type	Offset	Start value	Retain	Accessible from HMI/OPC UA	Writable from HMI/OPC UA	Visible in HMI engineering	Setpoint	Supervision	Comment
▼ Static										
Comm_load_ucinjeno	Bool	0.0	false	False	False	False	False	False		
Comm_load_greska	Bool	0.1	false	False	False	False	False	False		
Comm_load_status	Word	2.0	16#0	False	False	False	False	False		
Master_zahtjev	Bool	4.0	false	False	False	False	False	False		
Master_adresa	Word	6.0	16#0	False	False	False	False	False		
Master_nacin_rada	USInt	8.0	0	False	False	False	False	False		
Master_adresa_podataka	UDInt	10.0	0	False	False	False	False	False		
Master_duljina_podataka	UInt	14.0	0	False	False	False	False	False		
▼ Master_podaci	Array[0..100] of UInt	16.0		False	False	False	False	False		
Master_podaci[0]	UInt	16.0	0	False	False	False	False	False		
Master_podaci[1]	UInt	18.0	0	False	False	False	False	False		
Master_podaci[2]	UInt	20.0	0	False	False	False	False	False		
Master_podaci[3]	UInt	22.0	0	False	False	False	False	False		
Master_podaci[4]	UInt	24.0	0	False	False	False	False	False		
Master_podaci[5]	UInt	26.0	0	False	False	False	False	False		
Master_podaci[6]	UInt	28.0	0	False	False	False	False	False		

Name	Data type	Offset	Start value	Retain	Accessible from HMI/OPC UA	Writable from HMI/OPC UA	Visible in HMI engineering	Setpoint	Supervision	Comment
Master_podaci[7]	UInt	30.0	0	False	False	False	False	False		
Master_podaci[8]	UInt	32.0	0	False	False	False	False	False		
Master_podaci[9]	UInt	34.0	0	False	False	False	False	False		
Master_podaci[10]	UInt	36.0	0	False	False	False	False	False		
Master_podaci[11]	UInt	38.0	0	False	False	False	False	False		
Master_podaci[12]	UInt	40.0	0	False	False	False	False	False		
Master_podaci[13]	UInt	42.0	0	False	False	False	False	False		
Master_podaci[14]	UInt	44.0	0	False	False	False	False	False		
Master_podaci[15]	UInt	46.0	0	False	False	False	False	False		
Master_podaci[16]	UInt	48.0	0	False	False	False	False	False		
Master_podaci[17]	UInt	50.0	0	False	False	False	False	False		
Master_podaci[18]	UInt	52.0	0	False	False	False	False	False		
Master_podaci[19]	UInt	54.0	0	False	False	False	False	False		
Master_podaci[20]	UInt	56.0	0	False	False	False	False	False		
Master_podaci[21]	UInt	58.0	0	False	False	False	False	False		
Master_podaci[22]	UInt	60.0	0	False	False	False	False	False		
Master_podaci[23]	UInt	62.0	0	False	False	False	False	False		
Master_podaci[24]	UInt	64.0	0	False	False	False	False	False		
Master_podaci[25]	UInt	66.0	0	False	False	False	False	False		
Master_podaci[26]	UInt	68.0	0	False	False	False	False	False		
Master_podaci[27]	UInt	70.0	0	False	False	False	False	False		
Master_podaci[28]	UInt	72.0	0	False	False	False	False	False		
Master_podaci[29]	UInt	74.0	0	False	False	False	False	False		
Master_podaci[30]	UInt	76.0	0	False	False	False	False	False		
Master_podaci[31]	UInt	78.0	0	False	False	False	False	False		

Name	Data type	Offset	Start value	Retain	Accessible from HMI/OPC UA	Writable from HMI/OPC UA	Visible in HMI engineering	Setpoint	Supervision	Comment
Master_podaci[32]	UInt	80.0	0	False	False	False	False	False		
Master_podaci[33]	UInt	82.0	0	False	False	False	False	False		
Master_podaci[34]	UInt	84.0	0	False	False	False	False	False		
Master_podaci[35]	UInt	86.0	0	False	False	False	False	False		
Master_podaci[36]	UInt	88.0	0	False	False	False	False	False		
Master_podaci[37]	UInt	90.0	0	False	False	False	False	False		
Master_podaci[38]	UInt	92.0	0	False	False	False	False	False		
Master_podaci[39]	UInt	94.0	0	False	False	False	False	False		
Master_podaci[40]	UInt	96.0	0	False	False	False	False	False		
Master_podaci[41]	UInt	98.0	0	False	False	False	False	False		
Master_podaci[42]	UInt	100.0	0	False	False	False	False	False		
Master_podaci[43]	UInt	102.0	0	False	False	False	False	False		
Master_podaci[44]	UInt	104.0	0	False	False	False	False	False		
Master_podaci[45]	UInt	106.0	0	False	False	False	False	False		
Master_podaci[46]	UInt	108.0	0	False	False	False	False	False		
Master_podaci[47]	UInt	110.0	0	False	False	False	False	False		
Master_podaci[48]	UInt	112.0	0	False	False	False	False	False		
Master_podaci[49]	UInt	114.0	0	False	False	False	False	False		
Master_podaci[50]	UInt	116.0	0	False	False	False	False	False		
Master_podaci[51]	UInt	118.0	0	False	False	False	False	False		
Master_podaci[52]	UInt	120.0	0	False	False	False	False	False		
Master_podaci[53]	UInt	122.0	0	False	False	False	False	False		
Master_podaci[54]	UInt	124.0	0	False	False	False	False	False		
Master_podaci[55]	UInt	126.0	0	False	False	False	False	False		
Master_podaci[56]	UInt	128.0	0	False	False	False	False	False		

Name	Data type	Offset	Start value	Retain	Accessible from HMI/OPC UA	Writable from HMI/OPC UA	Visible in HMI engineering	Setpoint	Supervision	Comment
Master_podaci[57]	UInt	130.0	0	False	False	False	False	False		
Master_podaci[58]	UInt	132.0	0	False	False	False	False	False		
Master_podaci[59]	UInt	134.0	0	False	False	False	False	False		
Master_podaci[60]	UInt	136.0	0	False	False	False	False	False		
Master_podaci[61]	UInt	138.0	0	False	False	False	False	False		
Master_podaci[62]	UInt	140.0	0	False	False	False	False	False		
Master_podaci[63]	UInt	142.0	0	False	False	False	False	False		
Master_podaci[64]	UInt	144.0	0	False	False	False	False	False		
Master_podaci[65]	UInt	146.0	0	False	False	False	False	False		
Master_podaci[66]	UInt	148.0	0	False	False	False	False	False		
Master_podaci[67]	UInt	150.0	0	False	False	False	False	False		
Master_podaci[68]	UInt	152.0	0	False	False	False	False	False		
Master_podaci[69]	UInt	154.0	0	False	False	False	False	False		
Master_podaci[70]	UInt	156.0	0	False	False	False	False	False		
Master_podaci[71]	UInt	158.0	0	False	False	False	False	False		
Master_podaci[72]	UInt	160.0	0	False	False	False	False	False		
Master_podaci[73]	UInt	162.0	0	False	False	False	False	False		
Master_podaci[74]	UInt	164.0	0	False	False	False	False	False		
Master_podaci[75]	UInt	166.0	0	False	False	False	False	False		
Master_podaci[76]	UInt	168.0	0	False	False	False	False	False		
Master_podaci[77]	UInt	170.0	0	False	False	False	False	False		
Master_podaci[78]	UInt	172.0	0	False	False	False	False	False		
Master_podaci[79]	UInt	174.0	0	False	False	False	False	False		
Master_podaci[80]	UInt	176.0	0	False	False	False	False	False		
Master_podaci[81]	UInt	178.0	0	False	False	False	False	False		

Name	Data type	Offset	Start value	Retain	Accessible from HMI/OPC UA	Writable from HMI/OPC UA	Visible in HMI engineering	Setpoint	Supervision	Comment
Master_podaci[82]	UInt	180.0	0	False	False	False	False	False		
Master_podaci[83]	UInt	182.0	0	False	False	False	False	False		
Master_podaci[84]	UInt	184.0	0	False	False	False	False	False		
Master_podaci[85]	UInt	186.0	0	False	False	False	False	False		
Master_podaci[86]	UInt	188.0	0	False	False	False	False	False		
Master_podaci[87]	UInt	190.0	0	False	False	False	False	False		
Master_podaci[88]	UInt	192.0	0	False	False	False	False	False		
Master_podaci[89]	UInt	194.0	0	False	False	False	False	False		
Master_podaci[90]	UInt	196.0	0	False	False	False	False	False		
Master_podaci[91]	UInt	198.0	0	False	False	False	False	False		
Master_podaci[92]	UInt	200.0	0	False	False	False	False	False		
Master_podaci[93]	UInt	202.0	0	False	False	False	False	False		
Master_podaci[94]	UInt	204.0	0	False	False	False	False	False		
Master_podaci[95]	UInt	206.0	0	False	False	False	False	False		
Master_podaci[96]	UInt	208.0	0	False	False	False	False	False		
Master_podaci[97]	UInt	210.0	0	False	False	False	False	False		
Master_podaci[98]	UInt	212.0	0	False	False	False	False	False		
Master_podaci[99]	UInt	214.0	0	False	False	False	False	False		
Master_podaci[100]	UInt	216.0	0	False	False	False	False	False		
Master_status	Word	218.0	16#0	False	False	False	False	False		
Master_ucinjeno	Bool	220.0	false	False	False	False	False	False		
Master_zauzeto	Bool	220.1	false	False	False	False	False	False		
Master_greska	Bool	220.2	false	False	False	False	False	False		
Korak	Bool	220.3	false	False	True	True	True	False		
Korak_1	Bool	220.4	false	False	True	True	True	False		

Name	Data type	Offset	Start value	Retain	Accessible from HMI/OPC UA	Writable from HMI/OPC UA	Visible in HMI engineering	Setpoint	Supervision	Comment
Korak_2	Bool	220.5	false	False	True	True	True	False		
Korak_3	Bool	220.6	false	False	True	True	True	False		
Timer	Int	222.0	0	False	True	True	True	False		
Greska	Bool	224.0	false	False	True	True	True	False		
Stanje_master	Word	226.0	16#0	False	True	True	True	False		
Stanje_comm	Word	228.0	16#0	False	True	True	True	False		

Diplomski_rad_Saric_Luka_4 / PLC_1 [CPU 1512SP-1 PN] / Program blocks

Motor [DB4]

Motor Properties					
General					
Name	Motor	**Number**	4	**Type**	DB
Language	DB	**Numbering**	Manual		
Information					
Title		**Author**		**Comment**	
Family		**Version**	0.1	**User-defined ID**	

Motor

Name	Data type	Start value	Retain	Accessible from HMI/OPC UA	Writable from HMI/OPC UA	Visible in HMI engineering	Setpoint	Supervision	Comment
▼ Static									
Komanda	Bool	false	False	True	True	True	False		
Iskljucen	Bool	false	False	True	True	True	False		
Naprijed	Bool	false	False	True	True	True	False		
Reverz	Bool	false	False	True	True	True	False		
Radi	Bool	false	False	True	True	True	False		

Diplomski_rad_Saric_Luka_4 / PLC_1 [CPU 1512SP-1 PN] / Program blocks

Pisanje [DB3]

Pisanje Properties					
General					
Name	Pisanje	**Number**	3	**Type**	DB
Language	DB	**Numbering**	Manual		
Information					
Title		**Author**		**Comment**	
Family		**Version**	0.1	**User-defined ID**	

Pisanje

Name	Data type	Start value	Retain	Accessible from HMI/OPC UA	Writable from HMI/OPC UA	Visible in HMI engineering	Set-point	Supervision	Comment
▼ Static									
Motor_naprijed	Bool	false	False	True	True	True	False		
Motor_reverz	Bool	false	False	True	True	True	False		
Stop_deceleracija	Bool	false	False	True	True	True	False		
Stop_nagli	Bool	false	False	True	True	True	False		
Start_naprijed_jog	Bool	false	False	True	True	True	False		
Stop_naprijed_jog	Bool	false	False	True	True	True	False		
Ukljucen	Bool	false	False	True	True	True	False		
Reset_greske	Bool	false	False	True	True	True	False		
Stop_naprijed_jog_1	Bool	false	False	True	True	True	False		
Stop_reverz_jog	Bool	false	False	True	True	True	False		
Zadana_frekvencija	UInt	0	False	True	True	True	False		

PRILOG 4. PLC TAGOVI

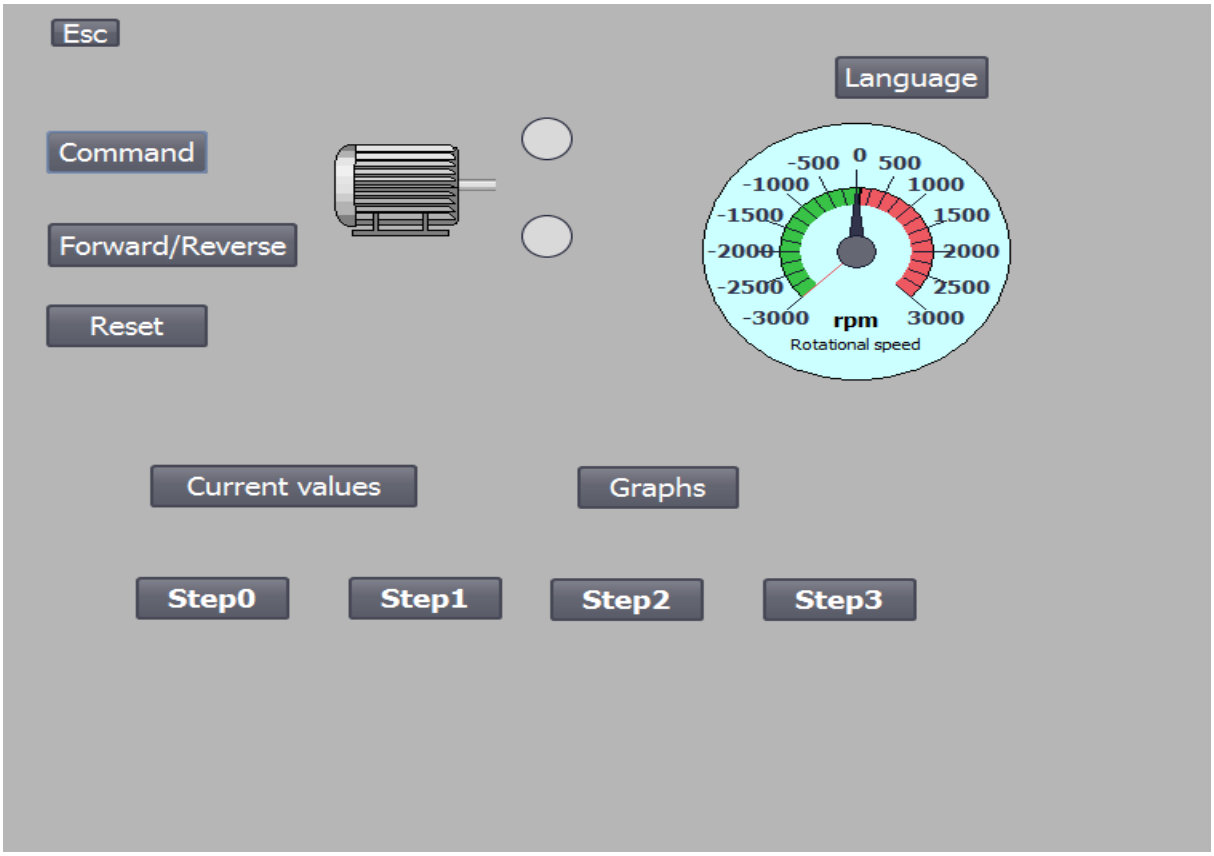
Totally Integrated Automation Portal

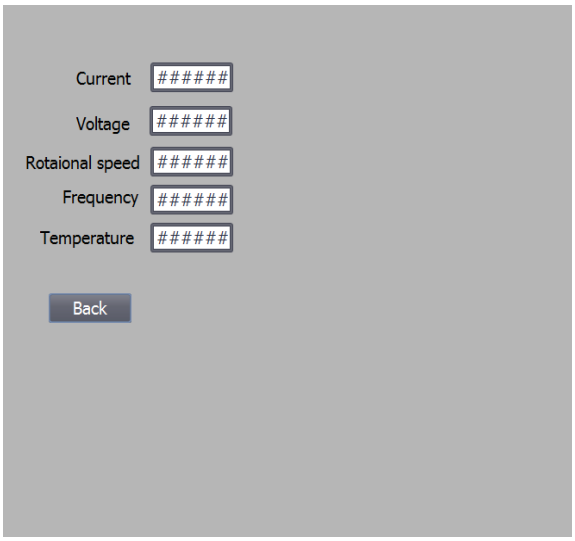
Diplomski_rad_Saric_Luka_4 / PLC_1 [CPU 1512SP-1 PN]

PLC tags

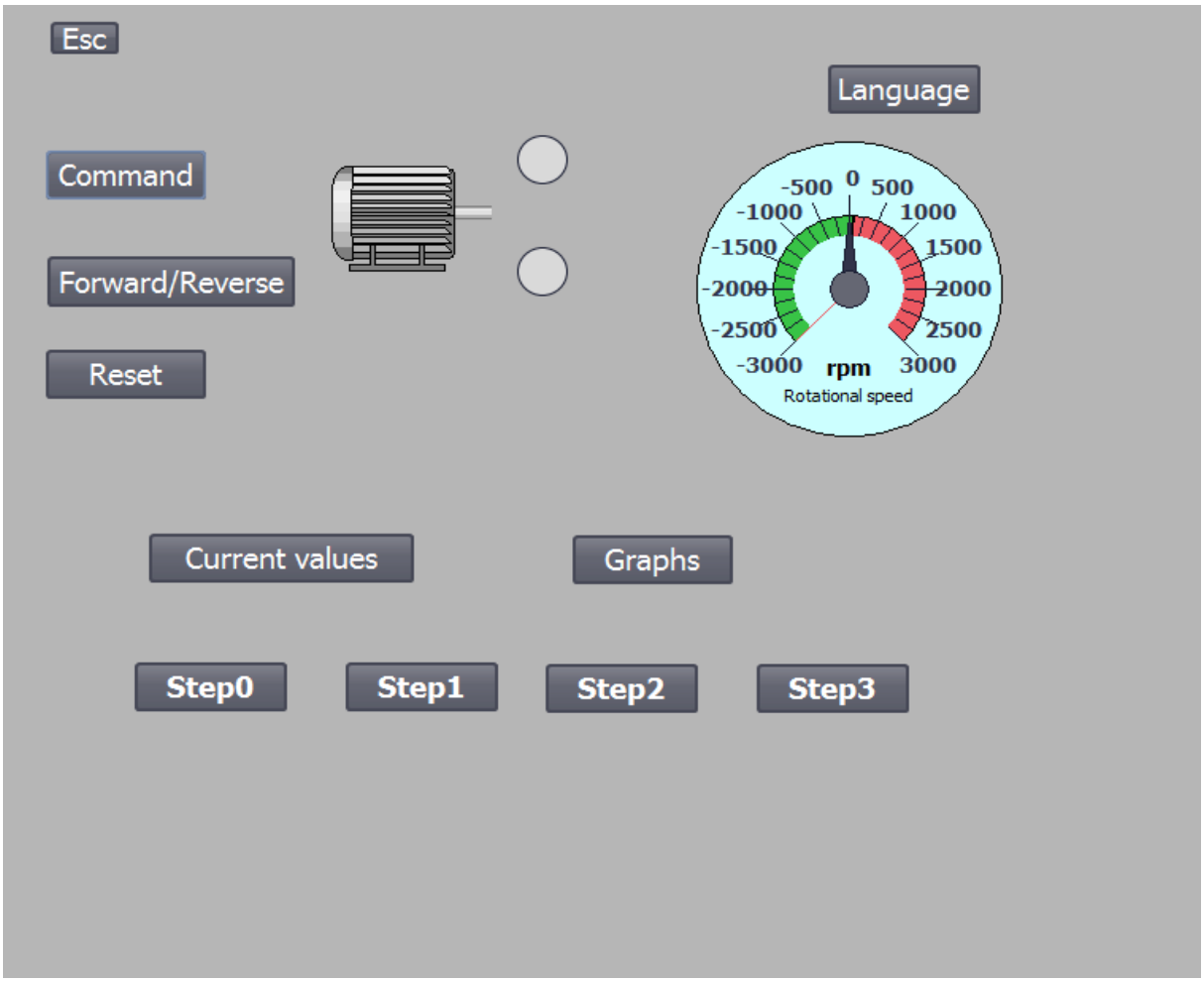
	Name	Data type	Address	Retain	Acces-sible from HMI/OPC UA	Writa-ble from HMI/OPC UA	Visi-ble in HMI engi-neer-ing	Supervision	Comment
	System_Byte	Byte	%MB1	False	True	True	True		
	FirstScan	Bool	%M1.0	False	True	True	True		
	DiagStatusUpdate	Bool	%M1.1	False	True	True	True		
	AlwaysTRUE	Bool	%M1.2	False	True	True	True		
	AlwaysFALSE	Bool	%M1.3	False	True	True	True		
	Clock_Byte	Byte	%MB0	False	True	True	True		
	Clock_10Hz	Bool	%M0.0	False	True	True	True		
	Clock_5Hz	Bool	%M0.1	False	True	True	True		
	Clock_2.5Hz	Bool	%M0.2	False	True	True	True		
	Clock_2Hz	Bool	%M0.3	False	True	True	True		
	Clock_1.25Hz	Bool	%M0.4	False	True	True	True		
	Clock_1Hz	Bool	%M0.5	False	True	True	True		
	Clock_0.625Hz	Bool	%M0.6	False	True	True	True		
	Clock_0.5Hz	Bool	%M0.7	False	True	True	True		
	Komanda	Bool	%M10.0	False	True	True	True		
	Motor_naprijed_re-verz	Bool	%M10.1	False	True	True	True		
	Reset_greske	Bool	%M10.4	False	True	True	True		
	Pozitiva	Bool	%M100.0	False	True	True	True		

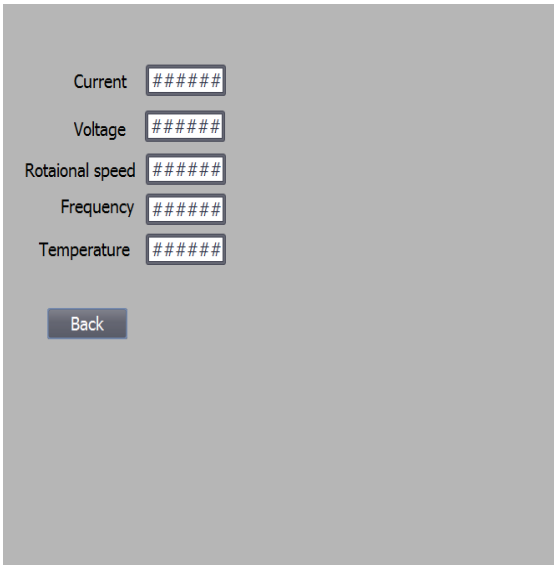
PRILOG 5. HMI SUČELJE NA ENGLESKOM JEZIKU





PRILOG 6. HMI SUČELJE NA NJEMAČKOM JEZIKU





PRILOG 7. HMI TAGOVI

Totally Integrated Automation Portal		

Diplomski_rad_Saric_Luka_4 / Luka's PC [SIMATIC PC station] / HMI_RT_1 [WinCC RT Advanced]

HMI tags

Komanda

General			
Name	Komanda	**Display name**	
Connection	HMI_Connection_1	**Data type**	Bool
Array elements	0	**Length**	1
Address	%M10.0	**Access mode**	<absolute access>
Coding	Binary	**PLC name**	PLC_1
Settings			
Acquisition cycle	100 ms	**Acquisition mode**	Cyclic continuous
Limits			
Upper 2		**Upper 1**	
Lower 1		**Lower 2**	
Linear scaling			
Linear scaling	Unchecked	**PLC value range end value**	10
PLC value range start value	0	**HMI device value range end value**	100
HMI device value range start value	0		
Values			
ID tag		**Start value**	
Comment			
Comment		**Source comment**	
Multiplexing			
Multiplexing	Unchecked	**Index tag**	
Logging			
Data log			
GMP (Good Manufacturing Practice)			
Confirmation type	None	**GMP relevant**	Unchecked
Comment required	Unchecked		

Komanda

Motor_naprijed_reverz

General			
Name	Motor_naprijed_reverz	**Display name**	
Connection	HMI_Connection_1	**Data type**	Bool
Array elements	0	**Length**	1
Address		**Access mode**	<symbolic access>
Coding	Binary	**PLC name**	PLC_1
Settings			
Acquisition cycle	100 ms	**Acquisition mode**	Cyclic continuous
Limits			
Upper 2		**Upper 1**	
Lower 1		**Lower 2**	
Linear scaling			
Linear scaling	Unchecked	**PLC value range end value**	10
PLC value range start value	0	**HMI device value range end value**	100

HMI device value range start value	0		
Values			
ID tag		Start value	
Comment			
Comment		Source comment	
Multiplexing			
Multiplexing	Unchecked	Index tag	
Logging			
Data log			
GMP (Good Manufacturing Practice)			
Confirmation type	None	GMP relevant	Unchecked
Comment required	Unchecked		

Motor_naprijed_reverz

Reset_greske

General			
Name	Reset_greske	Display name	
Connection	HMI_Connection_1	Data type	Bool
Array elements	0	Length	1
Address		Access mode	<symbolic access>
Coding	Binary	PLC name	PLC_1
Settings			
Acquisition cycle	100 ms	Acquisition mode	Cyclic continuous
Limits			
Upper 2		Upper 1	
Lower 1		Lower 2	
Linear scaling			
Linear scaling	Unchecked	PLC value range end value	10
PLC value range start value	0	HMI device value range end value	100
HMI device value range start value	0		
Values			
ID tag		Start value	
Comment			
Comment		Source comment	
Multiplexing			
Multiplexing	Unchecked	Index tag	
Logging			
Data log			
GMP (Good Manufacturing Practice)			
Confirmation type	None	GMP relevant	Unchecked
Comment required	Unchecked		

Reset_greske

Brzina

General			
Name	Brzina	Display name	
Connection	HMI_Connection_1	Data type	UInt
Array elements	0	Length	2
Address		Access mode	<symbolic access>
PLC tag	Citanje.Brzina	Coding	Binary
PLC name	PLC_1		

Settings			
Acquisition cycle	100 ms	Acquisition mode	Cyclic continuous
Limits			
Upper 2		Upper 1	
Lower 1		Lower 2	
Linear scaling			
Linear scaling	Unchecked	PLC value range end value	10
PLC value range start value	0	HMI device value range end value	100
HMI device value range start value	0		
Values			
ID tag		Start value	
Comment			
Comment		Source comment	
Multiplexing			
Multiplexing	Unchecked	Index tag	
Logging			
Data log			
GMP (Good Manufacturing Practice)			
Confirmation type	None	GMP relevant	Unchecked
Comment required	Unchecked		

Motor_radi

General			
Name	Motor_radi	Display name	
Connection	HMI_Connection_1	Data type	Bool
Array elements	0	Length	1
Address		Access mode	<symbolic access>
PLC tag	Motor.Radi	Coding	Binary
PLC name	PLC_1		
Settings			
Acquisition cycle	100 ms	Acquisition mode	Cyclic continuous
Limits			
Upper 2		Upper 1	
Lower 1		Lower 2	
Linear scaling			
Linear scaling	Unchecked	PLC value range end value	10
PLC value range start value	0	HMI device value range end value	100
HMI device value range start value	0		
Values			
ID tag		Start value	
Comment			
Comment		Source comment	
Multiplexing			
Multiplexing	Unchecked	Index tag	
Logging			
Data log			
GMP (Good Manufacturing Practice)			
Confirmation type	None	GMP relevant	Unchecked
Comment required	Unchecked		

Napon

General			
Name	Napon	**Display name**	
Connection	HMI_Connection_1	**Data type**	UInt
Array elements	0	**Length**	2
Address		**Access mode**	<symbolic access>
PLC tag	Citanje.Napon	**Coding**	Binary
PLC name	PLC_1		
Settings			
Acquisition cycle	100 ms	**Acquisition mode**	Cyclic continuous
Limits			
Upper 2		**Upper 1**	
Lower 1		**Lower 2**	
Linear scaling			
Linear scaling	Unchecked	**PLC value range end value**	10
PLC value range start value	0	**HMI device value range end value**	100
HMI device value range start value	0		
Values			
ID tag		**Start value**	
Comment			
Comment		**Source comment**	
Multiplexing			
Multiplexing	Unchecked	**Index tag**	
Logging			
Data log			
GMP (Good Manufacturing Practice)			
Confirmation type	None	**GMP relevant**	Unchecked
Comment required	Unchecked		

Struja

General			
Name	Struja	**Display name**	
Connection	HMI_Connection_1	**Data type**	UInt
Array elements	0	**Length**	2
Address		**Access mode**	<symbolic access>
PLC tag	Citanje.Struja	**Coding**	Binary
PLC name	PLC_1		
Settings			
Acquisition cycle	100 ms	**Acquisition mode**	Cyclic continuous
Limits			
Upper 2		**Upper 1**	
Lower 1		**Lower 2**	
Linear scaling			
Linear scaling	Unchecked	**PLC value range end value**	10
PLC value range start value	0	**HMI device value range end value**	100
HMI device value range start value	0		
Values			
ID tag		**Start value**	
Comment			
Comment		**Source comment**	

Multiplexing			
Multiplexing	Unchecked	Index tag	
Logging			
Data log			
GMP (Good Manufacturing Practice)			
Confirmation type	None	GMP relevant	Unchecked
Comment required	Unchecked		

Moment

General			
Name	Moment	Display name	
Connection	HMI_Connection_1	Data type	UInt
Array elements	0	Length	2
Address		Access mode	<symbolic access>
PLC tag	Citanje.Moment	Coding	Binary
PLC name	PLC_1		
Settings			
Acquisition cycle	100 ms	Acquisition mode	Cyclic continuous
Limits			
Upper 2		Upper 1	
Lower 1		Lower 2	
Linear scaling			
Linear scaling	Unchecked	PLC value range end value	10
PLC value range start value	0	HMI device value range end value	100
HMI device value range start value	0		
Values			
ID tag		Start value	
Comment			
Comment		Source comment	
Multiplexing			
Multiplexing	Unchecked	Index tag	
Logging			
Data log			
GMP (Good Manufacturing Practice)			
Confirmation type	None	GMP relevant	Unchecked
Comment required	Unchecked		

Snaga

General			
Name	Snaga	Display name	
Connection	HMI_Connection_1	Data type	UInt
Array elements	0	Length	2
Address		Access mode	<symbolic access>
PLC tag	Citanje.Snaga	Coding	Binary
PLC name	PLC_1		
Settings			
Acquisition cycle	100 ms	Acquisition mode	Cyclic continuous
Limits			
Upper 2		Upper 1	
Lower 1		Lower 2	
Linear scaling			
Linear scaling	Unchecked	PLC value range end value	10
PLC value range start value	0	HMI device value range end value	100

HMI device value range start value	0		
Values			
ID tag		Start value	
Comment			
Comment		Source comment	
Multiplexing			
Multiplexing	Unchecked	Index tag	
Logging			
Data log			
GMP (Good Manufacturing Practice)			
Confirmation type	None	GMP relevant	Unchecked
Comment required	Unchecked		

Temperatura

General			
Name	Temperatura	Display name	
Connection	HMI_Connection_1	Data type	UInt
Array elements	0	Length	2
Address		Access mode	<symbolic access>
PLC tag	Citanje.Temperatura_grijaca_FP	Coding	Binary
PLC name	PLC_1		
Settings			
Acquisition cycle	100 ms	Acquisition mode	Cyclic continuous
Limits			
Upper 2		Upper 1	
Lower 1		Lower 2	
Linear scaling			
Linear scaling	Unchecked	PLC value range end value	10
PLC value range start value	0	HMI device value range end value	100
HMI device value range start value	0		
Values			
ID tag		Start value	
Comment			
Comment		Source comment	
Multiplexing			
Multiplexing	Unchecked	Index tag	
Logging			
Data log			
GMP (Good Manufacturing Practice)			
Confirmation type	None	GMP relevant	Unchecked
Comment required	Unchecked		

Frekvencija

General			
Name	Frekvencija	Display name	
Connection	HMI_Connection_1	Data type	UInt
Array elements	0	Length	2
Address		Access mode	<symbolic access>
PLC tag	Citanje.Frekvencija	Coding	Binary
PLC name	PLC_1		
Settings			
Acquisition cycle	100 ms	Acquisition mode	Cyclic continuous

Limits			
Upper 2		Upper 1	
Lower 1		Lower 2	
Linear scaling			
Linear scaling	Unchecked	PLC value range end value	10
PLC value range start value	0	HMI device value range end value	100
HMI device value range start value	0		
Values			
ID tag		Start value	
Comment			
Comment		Source comment	
Multiplexing			
Multiplexing	Unchecked	Index tag	
Logging			
Data log			
GMP (Good Manufacturing Practice)			
Confirmation type	None	GMP relevant	Unchecked
Comment required	Unchecked		

Greska

General			
Name	Greska	Display name	
Connection	HMI_Connection_1	Data type	Bool
Array elements	0	Length	1
Address		Access mode	<symbolic access>
PLC tag	Modbus.Greska	Coding	Binary
PLC name	PLC_1		
Settings			
Acquisition cycle	100 ms	Acquisition mode	Cyclic continuous
Limits			
Upper 2		Upper 1	
Lower 1		Lower 2	
Linear scaling			
Linear scaling	Unchecked	PLC value range end value	10
PLC value range start value	0	HMI device value range end value	100
HMI device value range start value	0		
Values			
ID tag		Start value	
Comment			
Comment		Source comment	
Multiplexing			
Multiplexing	Unchecked	Index tag	
Logging			
Data log			
GMP (Good Manufacturing Practice)			
Confirmation type	None	GMP relevant	Unchecked
Comment required	Unchecked		

Zadana_frekvencija

General			
Name	Zadana_frekvencija	Display name	
Connection	HMI_Connection_1	Data type	UInt

Array elements	0	**Length**	2
Address		**Access mode**	<symbolic access>
PLC tag	Pisanje.Zadana_frekvencija	**Coding**	Binary
PLC name	PLC_1		
Settings			
Acquisition cycle	100 ms	**Acquisition mode**	Cyclic continuous
Limits			
Upper 2		**Upper 1**	
Lower 1		**Lower 2**	
Linear scaling			
Linear scaling	Unchecked	**PLC value range end value**	10
PLC value range start value	0	**HMI device value range end value**	100
HMI device value range start value	0		
Values			
ID tag		**Start value**	
Comment			
Comment		**Source comment**	
Multiplexing			
Multiplexing	Unchecked	**Index tag**	
Logging			
Data log			
GMP (Good Manufacturing Practice)			
Confirmation type	None	**GMP relevant**	Unchecked
Comment required	Unchecked		

Clock_1Hz

General			
Name	Clock_1Hz	**Display name**	
Connection	HMI_Connection_1	**Data type**	Bool
Array elements	0	**Length**	1
Address		**Access mode**	<symbolic access>
Coding	Binary	**PLC name**	PLC_1
Settings			
Acquisition cycle	100 ms	**Acquisition mode**	Cyclic continuous
Limits			
Upper 2		**Upper 1**	
Lower 1		**Lower 2**	
Linear scaling			
Linear scaling	Unchecked	**PLC value range end value**	10
PLC value range start value	0	**HMI device value range end value**	100
HMI device value range start value	0		
Values			
ID tag		**Start value**	
Comment			
Comment		**Source comment**	
Multiplexing			
Multiplexing	Unchecked	**Index tag**	
Logging			
Data log			
GMP (Good Manufacturing Practice)			
Confirmation type	None	**GMP relevant**	Unchecked
Comment required	Unchecked		

Clock_1Hz

Korak0

General			
Name	Korak0	**Display name**	
Connection	HMI_Connection_1	**Data type**	Bool
Array elements	0	**Length**	1
Address		**Access mode**	<symbolic access>
PLC tag	Modbus.Korak	**Coding**	Binary
PLC name	PLC_1		
Settings			
Acquisition cycle	100 ms	**Acquisition mode**	Cyclic in operation
Limits			
Upper 2		**Upper 1**	
Lower 1		**Lower 2**	
Linear scaling			
Linear scaling	Unchecked	**PLC value range end value**	10
PLC value range start value	0	**HMI device value range end value**	100
HMI device value range start value	0		
Values			
ID tag		**Start value**	
Comment			
Comment		**Source comment**	
Multiplexing			
Multiplexing	Unchecked	**Index tag**	
Logging			
Data log			
GMP (Good Manufacturing Practice)			
Confirmation type	None	**GMP relevant**	Unchecked
Comment required	Unchecked		

Korak1

General			
Name	Korak1	**Display name**	
Connection	HMI_Connection_1	**Data type**	Bool
Array elements	0	**Length**	1
Address		**Access mode**	<symbolic access>
PLC tag	Modbus.Korak_1	**Coding**	Binary
PLC name	PLC_1		
Settings			
Acquisition cycle	100 ms	**Acquisition mode**	Cyclic in operation
Limits			
Upper 2		**Upper 1**	
Lower 1		**Lower 2**	
Linear scaling			
Linear scaling	Unchecked	**PLC value range end value**	10
PLC value range start value	0	**HMI device value range end value**	100
HMI device value range start value	0		
Values			
ID tag		**Start value**	

Comment			
Comment		Source comment	
Multiplexing			
Multiplexing	Unchecked	Index tag	
Logging			
Data log			
GMP (Good Manufacturing Practice)			
Confirmation type	None	GMP relevant	Unchecked
Comment required	Unchecked		

Korak2

General			
Name	Korak2	Display name	
Connection	HMI_Connection_1	Data type	Bool
Array elements	0	Length	1
Address		Access mode	<symbolic access>
PLC tag	Modbus.Korak_2	Coding	Binary
PLC name	PLC_1		
Settings			
Acquisition cycle	100 ms	Acquisition mode	Cyclic in operation
Limits			
Upper 2		Upper 1	
Lower 1		Lower 2	
Linear scaling			
Linear scaling	Unchecked	PLC value range end value	10
PLC value range start value	0	HMI device value range end value	100
HMI device value range start value	0		
Values			
ID tag		Start value	
Comment			
Comment		Source comment	
Multiplexing			
Multiplexing	Unchecked	Index tag	
Logging			
Data log			
GMP (Good Manufacturing Practice)			
Confirmation type	None	GMP relevant	Unchecked
Comment required	Unchecked		

Korak3

General			
Name	Korak3	Display name	
Connection	HMI_Connection_1	Data type	Bool
Array elements	0	Length	1
Address		Access mode	<symbolic access>
PLC tag	Modbus.Korak_3	Coding	Binary
PLC name	PLC_1		
Settings			
Acquisition cycle	100 ms	Acquisition mode	Cyclic in operation
Limits			
Upper 2		Upper 1	
Lower 1		Lower 2	

Linear scaling			
Linear scaling	Unchecked	**PLC value range end value**	10
PLC value range start value	0	**HMI device value range end value**	100
HMI device value range start value	0		
Values			
ID tag		**Start value**	
Comment			
Comment		**Source comment**	
Multiplexing			
Multiplexing	Unchecked	**Index tag**	
Logging			
Data log			
GMP (Good Manufacturing Practice)			
Confirmation type	None	**GMP relevant**	Unchecked
Comment required	Unchecked		