

# Siemens Simatic i WinCC industrijsko rješenje za nadzor i upravljanje frekventnim pretvaračem i motornim pogonom

---

Jelić, Vedran

Master's thesis / Diplomski rad

2022

*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:200:548748>

*Rights / Prava:* [In copyright](#)/[Zaštićeno autorskim pravom.](#)

*Download date / Datum preuzimanja:* **2024-11-04**

*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**

**SIEMENS SIMATIC I WINCC INDUSTRIJSKO  
RJEŠENJE ZA NADZOR I UPRAVLJANJE  
FREKVENTNIM PRETVARAČEM I MOTORNIM  
POGONOM**

**DIPLOMSKI RAD**

**VEDRAN JELIĆ**

**OSIJEK, 2022.**

# SADRŽAJ

1.	UVOD .....	1
1.1.	ZADATAK I STRUKTURA RADA .....	2
2.	SIEMENS SIMATIC I WINCC INDUSTRIJSKO RJEŠENJE ZA NADZOR I UPRAVLJANJE FREKVENTNIM PRETVARAČEM I MOTORNIM POGONOM .....	3
2.1.	TEORIJSKI OSVRT NA PROBLEM I RJEŠENJE .....	3
2.2.	SKLOPOVSKO RJEŠENJE .....	15
2.3.	PROGRAMSKO RJEŠENJE .....	16
3.	REALIZACIJA I IMPLEMENTACIJA SUSTAVA.....	17
3.1.	OPIS KORIŠTENIH ALATA I UREĐAJA.....	17
3.2.	REALIZACIJA SKLOPOVSKOG RJEŠENJA.....	23
3.3.	REALIZACIJA PROGRAMSKOG RJEŠENJA.....	24
4.	TESTIRANJE I REZULTATI.....	32
4.2.	METODOLOGIJA TESTIRANJA .....	32
4.3.	REZULTATI TESTIRANJA .....	35
5.	ZAKLJUČAK.....	36
	LITERATURA.....	37
	SAŽETAK .....	39
	ABSTRACT .....	40
	ŽIVOTOPIS .....	41
	PRILOG 1. – MAIN .....	42
	PRILOG 2. – FCTN FUNKCIJA.....	43
	PRILOG 3. – FUNKCIJSKI BLOK ZA VACON KOMUNIKACIJU.....	45
	PRILOG 4. – PLC TAGS .....	49
	PRILOG 5. – HMI TAGS .....	50

## 1. UVOD

Elektromotorni pogoni su rasprostranjeni diljem svijeta i vrlo su važna karika današnjeg modernog života. Početkom prošlog stoljeća električna energija preuzela je većinu zahtjeva u industrijskim postrojenjima. Kako elektromotor pretvara električnu energiju u mehaničku, uz male gubitke i zbog njegove robusne izvedbe pronašli su svoju primjenu u postrojenjima. U početku su se motori koristili spojeni izravno na mrežu bez mogućnosti regulacije brzine vrtnje i momenta, no kako je industrija rasla i postavljali su se sve veći zahtjevi na preciznost i upravljanje, nastali su razni upravljački mehanizmi. Napretkom tehnologije napredovala je izrada električnih motora koji dobivaju sve više izvedbi različitih snaga, brzini vrtnje i uvjetima rada. Drugom polovicom prošlog stoljeća i razvojem digitalne elektronike nastali su i frekventni pretvarači, uređaji koji putem mijenjanja vrijednosti frekvencije i napona mijenjaju karakteristiku motora, odnosno postavljaju motor u različite radne točke, ovisno o potrebi i opterećenju mijenjaju mu moment i brzinu vrtnje. Današnji motori imaju visoku iskoristivost za razliku od prvobitnih što govori da već sama moderna tehnologija smanjuje gubitke u motoru. Važno je i ekonomično upravljati motorom što omogućava frekventni pretvarač. Razvojem računalnih mreža razvijene su i različiti komunikacijski protokoli koji omogućavaju komunikaciju između različitih uređaja u industriji radi poboljšanja kvalitete i povećavanja učinkovitosti. Napredak tehnologije omogućio je integraciju svih ranije spomenutih elemenata u jedan veliki sustav koji se može automatizirati i upravljati sa jednog mjesta. Takvi sustavi rade sa boljom učinkovitošću, preciznija su, pouzdanija, sigurnija za radnike i okolinu i na kraju jeftinija.

Stara i zastarjela postrojenja, kojih u Hrvatskoj ima mnogo, umjesto da se zatvaraju zbog prestanka proizvodnje ili ne učinkovitosti mogla bi se rekonstruirati i opremiti modernim sustavima upravljanja koji danas, zbog relativno niske cijene nisu nedostižni. Na taj način bi se očuvala radna mjesta, a i proizvodnja bi mogla ugoditi većim zahtjevima. Prilikom modernizacije zastarjelih postrojenja treba obratiti pažnju na odabir opreme i uređaja koji bi zadovoljili potrebe na prikladan način.

Ovim radom bi se trebalo uči više u problematiku implementacije cijelo ukupnog ustava, naučiti nova znanja iz područja programiranja industrijskih računala i konfiguracije komunikacije tako da ona radi na odgovarajući način kao i parametrisiranje frekventnog pretvarača zajedno sa njegovim regulatorima i zaštitnim svojstvima.

## **1.1. Zadatak i struktura rada**

Zadatak ovoga rada je izraditi pokazni sustav koji se sastoji od asinkronog motora i frekventnog pretvarača. Frekventni pretvarač se povezuje sa industrijskim računalom putem Profinet komunikacije. Na industrijsko računalo učitava se program koji šalje podatke frekventnom pretvaraču na koji način da upravlja motorom. Osim sustava upravljanja izrađuje se i sustav nadzora putem SCADA sustava. Na ekranu se prikazuje HMI grafičko sučelje koje prikazuje stanje i vrijednosti postrojenja.

U radu će se za početak objasniti teorijska podloga osnovnih elemenata koji će biti korišteni u radu kao što je asinkroni motor i frekvencijski pretvarač, a prikazane su sklopovske i programske strukture sustava. Nadalje opisuju se uređaji koji su korišteni kao i njihova implementacija zajedno sa pripadajućim programom i grafičkim sučeljem. Na samom kraju prikazuju se metode mjerenja i krajnji rezultati.

## **2. SIEMENS SIMATIC I WINCC INDUSTRIJSKO RJEŠENJE ZA NADZOR I UPRAVLJANJE FREKVENTNIM PRETVARAČEM I MOTORNIM POGONOM**

### **2.1. Teorijski osvrt na problem i rješenje**

U ovom poglavlju pobliže će se objasniti teorijska podloga glavnih uređaja korištenih u radu i problematika samih. Naime asinkroni motor je vrlo rasprostranjen u postrojenjima i pogonima diljem svijeta samim time je razvijena tehnika njegovog upravljanja na način da bude što bolje iskoristiva uz štednju energije. Električni motor priključen na krutu mrežu rotira konstantnom brzinom uz konstantan moment, stoga ovisno o opterećenju te vrijednosti se mijenjaju. Za upravljanje izmjeničnih motora koriste se frekventni pretvarači koji omogućavaju suvremeni nadzor i upravljanje motora odnosno ostvarivanje željenog momenta i brzine vrtnje. Kako bih se upravljanje automatiziralo, frekventni pretvarači se putem industrijskih komunikacija mogu povezati sa programibilnim logičkim kontrolerima odnosno industrijskim računalima koji omogućavaju izradu programa i direktnu komunikaciju sa frekventnim pretvaračima. Korištenje industrijskih računala omogućava izradu grafičkih SCADA sustava koji u stvarnom vremenu prikazuju vrijednosti i stanje pogona i putem njih se može vršiti upravljanje. U sljedećim poglavljkima pobliže će se objasniti uređaja.

#### **2.1.1. Izmjenični motor**

Električni stroj je pretvarač energije koji pretvara energiju pomoću elektromagnetske indukcije. Prilikom pretvorbe energije, najčešće je jedna od energija mehanička energija odnosno rotacijsko gibanje ako se govori o električnim motorima. Transformatori se isto nazivaju električni strojevi, no kod njih su energije na obje strane električne. Električna energija pretvara se u mehaničku ako stroj radi u motorskom režimu rada, a mehanička energija pretvara se u električnu ako stroj radi u generatorskom režimu rada. Naravno kao i kod svake vrste pretvorbe energije iz jednog oblika u drugi postoje gubici, odnosno količina ulazne energije ne može se u potpunosti pojaviti na izlazu.

Izmjenični motor je u posljednjem stoljeću najrasprostranjeniji električni stroj koji se koristi za pogon bilo to u kućanstvu ili industriji. Nikola Tesla ga je predstavio u svom revolucionarnom otkriću okretnog magnetskog polja pomoću tri namota, a krasi ga jednostavna izvedba i robusnost. Princip rada je dosta jednostavan, a temelji se na tri zakona elektromagnetizma: zakon elektromagnetskog protjecanja, zakon

indukcije i zakon sile u magnetskom polju. Ako promjenjivo magnetsko polje  $B$  djeluje na vodič, na krajevima vodiča inducira se napon. Proteče li struja  $I$  tim vodičem, na njega će djelovati sila  $F$ . Izmjenični motori dijele se na dvije vrste: sinkroni i asinkroni motori. Karakteristika sinkronog motora je takva da mu je brzina rotora  $n$  jednaka brzini okretnog magnetskog polja ako je motor priključen direktno na mrežu. Brzina okretnog magnetskog polja ovisi o frekvenciji mreže na koju je priključen stator i broju pari polova rotora, a naziva se sinkrona brzina. Konstrukcijom je sličan asinkronom motoru. Sinkroni motori se najčešće koriste kao generatori.

$$n = n_s = \frac{60 \cdot f}{p}$$

(2.1)

Jedan od korištenih elemenata rada je asinkroni motor, koji zapravo predstavlja glavni objekt upravljanja u ovom radu. Asinkroni motor sastoji se od dva glavna dijela, statora i rotora. Stator asinkronog motora izveden je tako da mu se unutar utora statora ubacuje višefazni namot. Najčešće je to simetrično raspoređen trofazni namot. Na isti način izrađuje se i stator sinkronog motora, jedina razlika između ta dva stroja je različita izvedba rotora. Kada se simetrično raspoređeni trofazni namot priključi na trofazni sustav napajanja, namotima protoku struje. Kako se izmjenično napajanje sastoji od tri faze, promjenjivog iznosa napona u vremenu i fazno pomaknute za  $120^\circ$ , stvara se protjecanje konstantnog iznosa i brzine vrtnje. Taj pojam naziva se okretno magnetsko polje.



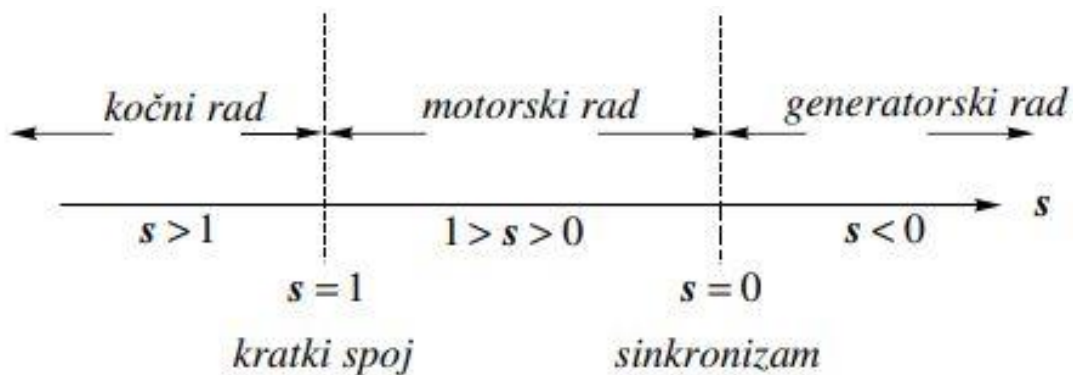
Slika 2.1 – Prikaz dva asinkronog motora (0.75kW i 2.2kW).

Rotor asinkronog motora izvodi se tako da mu se na obod raspoređuje namot. Rotorski namot se izrađuje kao višefazni, a broj faza ne mora odgovarati broju faza na statoru. Taj broj zna biti puno veći u odnosu na stator. Djelovanjem okretnog magnetskog polja, koje djeluje sa statora, silnice magnetskog polja presijecaju vodiče u namotu. Kako bi se u rotorskom namotu inducirao napon dolazimo do pojma klizanja. Klizanje  $s$  je razlika u brzini vrtnje rotora i brzine vrtnje okretnog magnetskog polja.

Brzina rotora različita od brzine vrtnje okretnog magnetskog polja, odnosno nešto je manja od sinkrone brzine u motornom režimu rada.

$$s = \frac{n_s - n}{n_s} \quad (2.2)$$

Upravo to kašnjenje rotora za sinkronom brzinom omogućava silnicama presijecanje vodiča i dolazi do induciranja napona u rotoru i tok struje kroz vodiče. U suprotnom ako bi se rotor gibao sinkronom brzinom, inducirani napon u rotorskom namotu bio bi jednak nuli. [1]



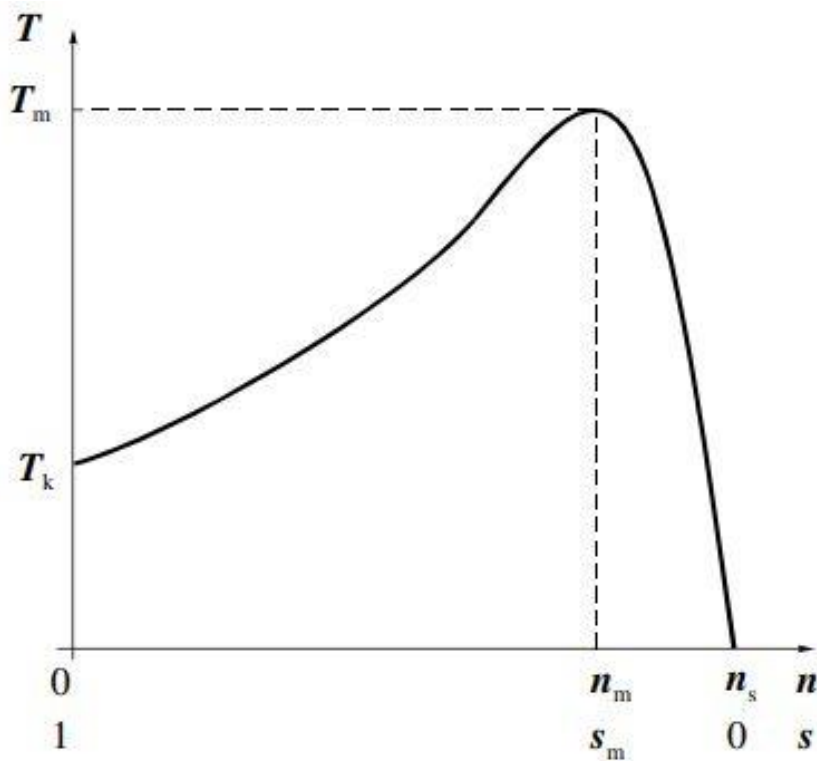
**Slika 2.2** – Prikaz režima rada elektromotora i vrijednosti klizanja [1]

Krajevi rotorskog namota mogu biti dovedeni na klizne prstene, preko kojih se u rotorski krug priključuju radni otpori, ili su ti krajevi namota kratko spojeni. Namot može biti izrađen od svitaka isto kao i statorski ili može biti izveden kao kavez. Ako je rotor motora izveden sa svitcima koji su spojeni na klizne prstene, onda se takav motor naziva klizno-kolutni. Kavezni asinkroni motor ili kavezni motor je druga izvedba asinkronog



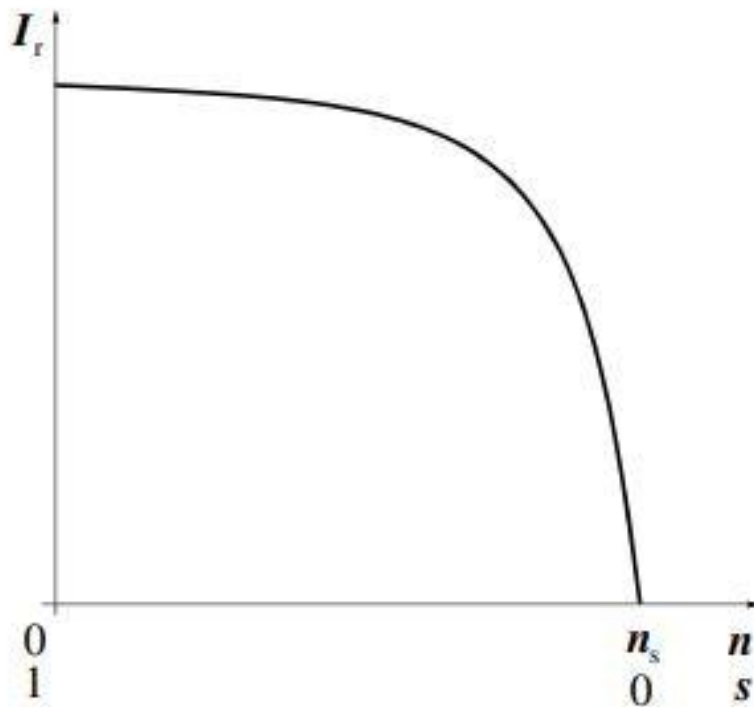
motora. U svakom utoru rotora, za razliku od svitaka, nalazi se po jedna vodič u obliku štapa. Sa svake strane rotora štapovi su kratko spojeni i čine višefazni namot, a fizički liči na kavez po čemu i dobiva naziv.

Rotor može imati proizvoljan broj faza jer su utori simetrični, a štapovi jednaki. Djelovanjem okretnog magnetskog polja na štapove, stvara se višefazni sustav induciranih napona i struja koji stvaraju moment. Važni parametri asinkronog motora su momentna karakteristika i strujna karakteristika. [1]



*Slika 2.3 – Momentna karakteristika asinkronog motora [1]*

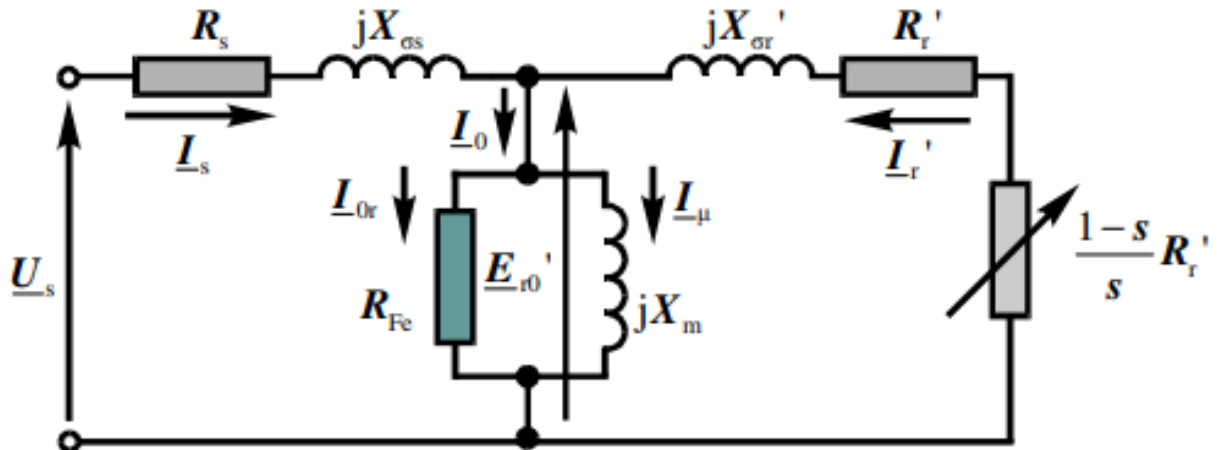
Na slici je prikazana momentna karakteristika  $T$  u odnosu na brzinu vrtnje  $n$  i klizanju  $s$ . Prilikom klizanju vrijednosti  $s = 1$  motor stoji. Moment se povećava kako motor ubrzava do maksimalne vrijednosti  $T_m$  odnosno prekretnog momenta. Moment nakon toga opada do vrijednosti 0 ako je motor u sinkronizmu  $s = 0$ . Početna vrijednost momenta  $T_k$  naziva se pokretni moment. Radno područje motora nalazi se između točke  $n_m$  i  $n_s$ . [1]



*Slika 2.4 – Karakteristika brzine vrtnje asinkronog motora [1]*

Na slici je prikazana strujna karakteristika u ovom slučaju rotorske struje. Struja ovisi o klizanju i proporcionalna je naponu. Struja će poprimiti maksimalnu vrijednost prilikom uklopa motora na mrežu. Velika pokretna struja stvara probleme prilikom pokretanja stroja. Struja može biti 5-10 puta veća od nazivne vrijednosti struje što dovodi do izbijanja zaštite ili oštećenja instalacija. Vrijednost 0 struja će imati kada je rotor u sinkronizmu  $s = 0$ . Karakteristika za statorsku struju je ista jedina je razlika što u sinkronizmu struja statora neće biti jednaka 0 zbog struje magnetiziranja jezgre.

Za proučavanje elektromagnetskih pojava asinkronog motora koristi se nadomjesna shema koja prikazuje elemente motora u električnoj shemi.

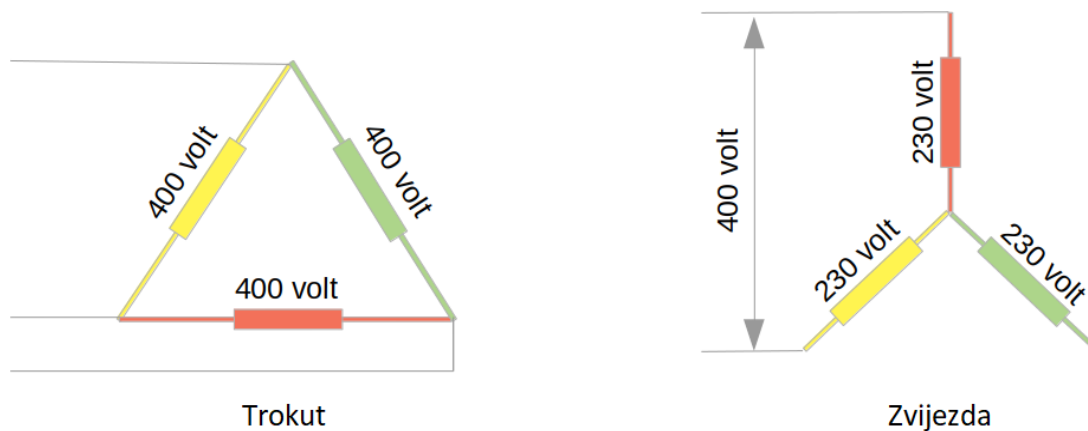


Slika 2.5 – Nadomjesna shema asinkronog motora [1]

Na slici su prikazani radni otpor  $R_s$  i rasipna reaktancija  $X_\sigma$  za statorski krug te isto tako transponirano za rotorski krug.  $X_m$  je glavna reaktancija (međuinuktivna), a  $R_{Fe}$  predstavlja nadomjesni otpor za gubitke u željezu. [1]

### 2.1.2. Frekventni pretvarači

Vratimo li se na ranije spomenutu veliku pokretnu struju asinkronog motora i fiksnu momentnu karakteristiku prilikom rada na mreži konstantne frekvencije, može se ustanoviti problem asinkronog motora. Potezna struja mogla se smanjiti promjenom spoja namota motora. Naime izmjenični motori, kako posjeduju tri namota na statoru, mogu se spajati na različite načine. Postoje dva spoja namota koja su najzastupljenija. Spoj u zvijezdu je spoj kod kojeg su radevi namota spojeni u jednu točku zvanu zvjezdište koja je na nultom potencijalu. Počeci namota priključeni su na linijski napon. Na ovaj način namoti su priključeni na fazni napon od 230V za  $\sqrt{3}$  manji od linijskog. Motor u zvijezda spoju povlači 3 puta manju struju što je pogodno za pokretanje motora, ali smanjuje se i potezni moment. Spoj u trokut je spoj kod kojeg su namoti spojeni u seriju. Kraj prvog spojen je na početak drugog, kraj drugog na početak trećeg i kraj trećeg na početak prvog. Na ovaj način namoti su priključeni na linijski napon od 400V. Ovaj spoj koristi se za trajni rad jačih motora preko 3KW. Postoje sklopke nazvane zvijezda-trokut sklopke gdje se motor zaliječe pri manjoj struji, te se preklapa u trokut spoj. [2]



Slika 2.6 – Prikaz vrste spoja namota trofaznog asinkronog motora. [3]

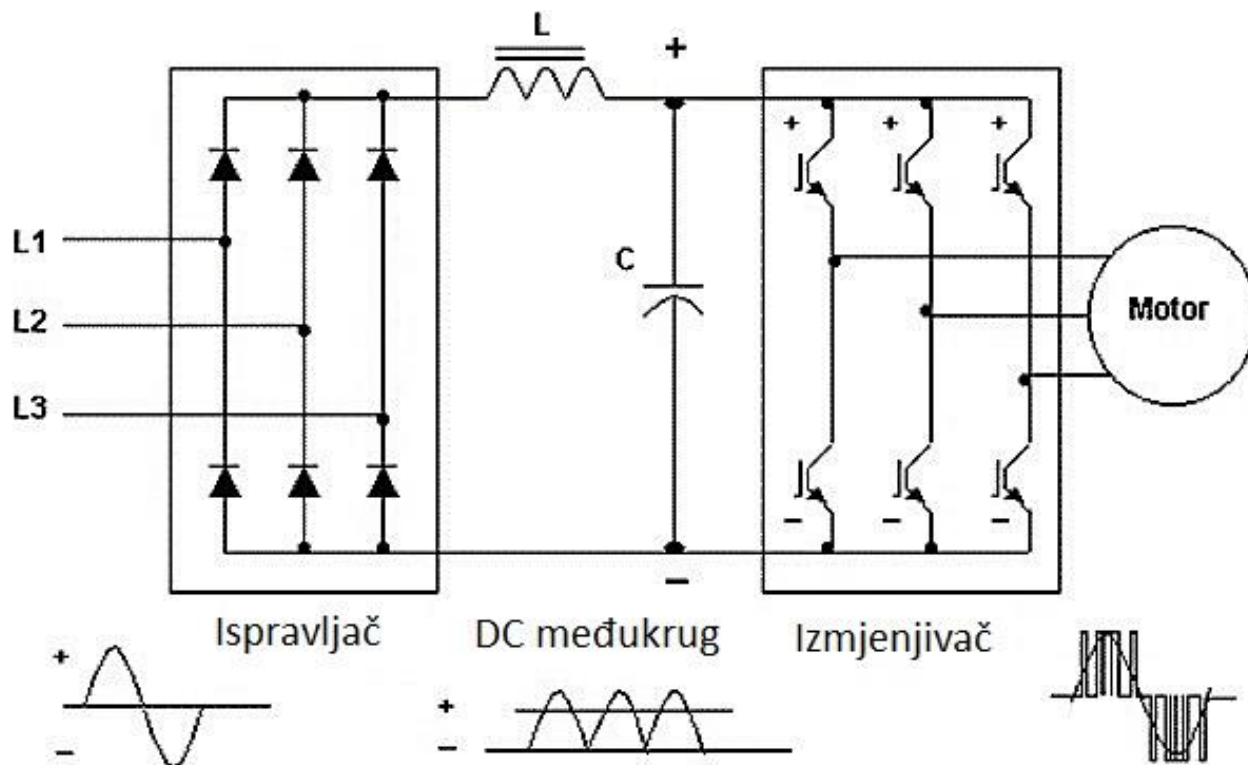
Osim smanjenja velike potezne struje javlja se zahtjev i za upravljanje brzinom vrtnje asinkronog motora na što efikasniji način. Razvojem elektronike i poluvodiča i taj zahtjev je našao svoje rješenje. Naime radi se o frekventnim pretvaračima, posebnim uređajima energetske elektronike koji se spajaju se između mreže i pogonskog motora. To su uređaji koji napon konstantnog iznosa i frekvencije mreže pretvaraju u promjenjivi napon i frekvenciju i tako upravljaju brzinom vrtnje izmjeničnog motora na način da zadrže željeni faktor snage uz smanjenje nepoželjnog zagrijavanja namota. Sastoji se od tri glavna dijela: ispravljača, istosmjernog međukruga i izmjenjivača.

Ispravljač je dio frekventnog pretvarača u kojem se izmjenični napon ispravlja u istosmjerni. Na ulazu u ispravljač priključuje se jednofazni ili trofazni napon mreže, a ispravljač na izlazu daje istosmjerni napon. Izvedbe ispravljača mogu biti različite, a izrađuju se pomoću dioda, spojenih u mosni spoj ili tiristora ako se upravlja kutom okidanja tiristora.

Istosmjerni međukrug skladišti električnu energiju. On motoru preko izmjenjivača predaje električnu energiju. Postoje izvedbe s promjenjivom strujom ili promjenjivim naponom. Kod promjenjivog napona može se konstantni ispravljeni napon pretvarati u promjenjivi ili se konstantni ispravljeni napon ispravljača filtrira i stabilizira kako bi se predao izmjenjivaču. Osim filtriranja napona prema izmjenjivaču, filtrira se i harmonijska distorzija koja može štetno utjecati na mrežu.

Izmjenjivač predaje motoru jednofazni ili trofazni napon promjenjivog napona i frekvencije. Svaka poluperioda sastoji se od niza pravokutnih impulsa koji su različite duljine. Izmjenjivač djeluje na osnovni harmonik izlaznog napona koji je promjenjive frekvencije i amplitude. Izrađen je najčešće od IGBT tranzistora (*engl. Insulated-Gate Bipolar Transistor*), kojima upravlja upravljačka jedinica. Frekvenciju izlaznog napona određuje izmjenjivač, dok u određivanju amplitude napona određuje izmjenjivač ili kombinacija sa istosmjernim međukrugom. Upravljača jedinica upravlja ventilima tako da dobiva informacije iz ispravljača, istosmjernog međukruga i izmjenjivača. Prema zadanim parametrima mijenja se omjer frekvencije i napona na taj način da se šalju impulsi za uključivanje i isključivanje poluvodičkih ventila. Upravljanje se na modernim frekvencijskim pretvaračima ostvaruje pomoću pulsno-širinske modulacije PWM (*engl. Pulse Width Modulation*), odnosno niz impulsa koji svojim širinom utječu na efektivnu vrijednost napona, a predznakom na frekvenciju. Naravno tu su svoju ulogu pronašli i mikroprocesori koji su svojim razvijanjem uvelike poboljšali elektromotorne pogone.

Frekventni pretvarači mogu sadržavati priključke na koje se spajaju kočni otpornici preko kojih se disipira energija prilikom kočenja motora ili mogu biti dvosmjerni gdje se energija kočenja može vraćati u mrežu. [4]



*Slika 2.7 – Prikaz sheme spoja frekventnog pretvarača sa istosmjernim međukrugom. [5]*

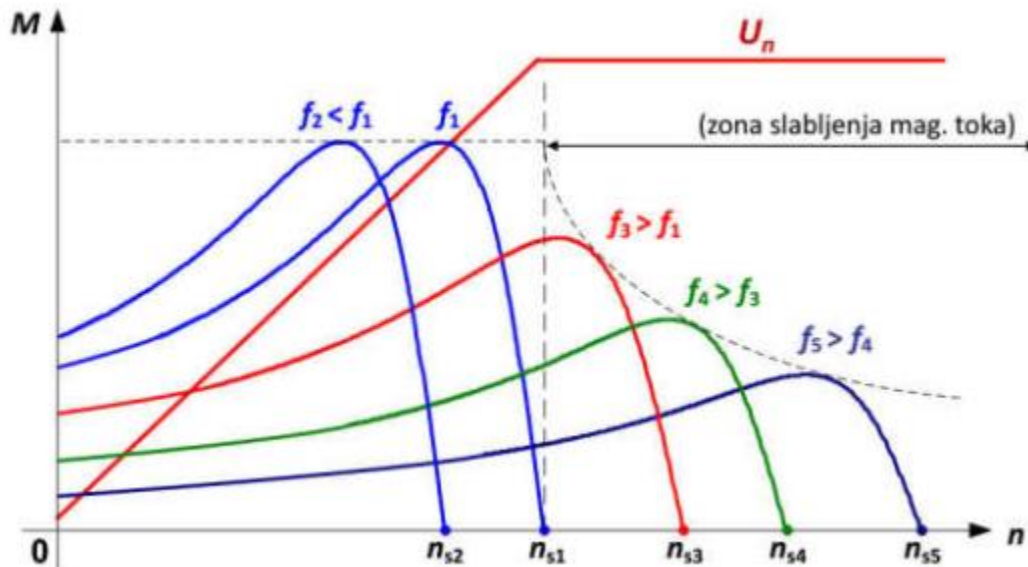
### 2.1.3. Upravljanje asinkronim motorom

U ovom dijelu ukratko će se objasniti načini upravljanja asinkronog motorom. Postoje dva glavna načina, a to su skalarno i vektorsko upravljanje.

Promjenu brzine vrtnje asinkronog motora moguće je ostvariti promjenom napona napajanja gdje se smanjuje nazivni moment napajanja ili promjenom frekvencije. Kako su obje fizikalne veličine skalarne ova metoda upravljanja naziva se skalarno upravljanje. Skalarno upravljanje je upravljanje brzinom vrtnje, magnetskim tokom i prekretnim momentom asinkronog promjenom napona i frekvencije. Pri istovremenoj i linearnoj promijeni frekvencije i napona ostvaruje se konstantni magnetski tok i prekretni moment.

Ovakav način upravljanja je dosta jednostavan, ali postoje ograničenja. Sa strane priključenog napona ograničenje je nazivna vrijednost napona koju proizvođač propisuje. Pređe li vrijednost napona nazivnu vrijednost dolazi do naprezanja i oštećenja izolacije motora. Pored toga uređaj preko kojeg se vrši skalarno upravljanje ograničen je naponom mreže. Povećanjem frekvencije ograničenje je mehaničke vrste,

jer se mijenja mehanička karakteristika zbog smanjenja magnetskog toka iznad nazivne frekvencije. Do nazivne vrijednosti napona omjer napona i frekvencije je konstantan te je magnetski tok konstantan, ali nakon što se dostigne nazivna vrijednost napona mijenja se samo frekvencija. To područje naziva se područje slabljenja magnetskog toka. U tom dijelu karakteristike opadaju pokretni i prekretni momenti. [6]



*Slika 2.8 – Prikaz momentne karakteristike asinkronog motora pri mijenjanju frekvencije [6]*

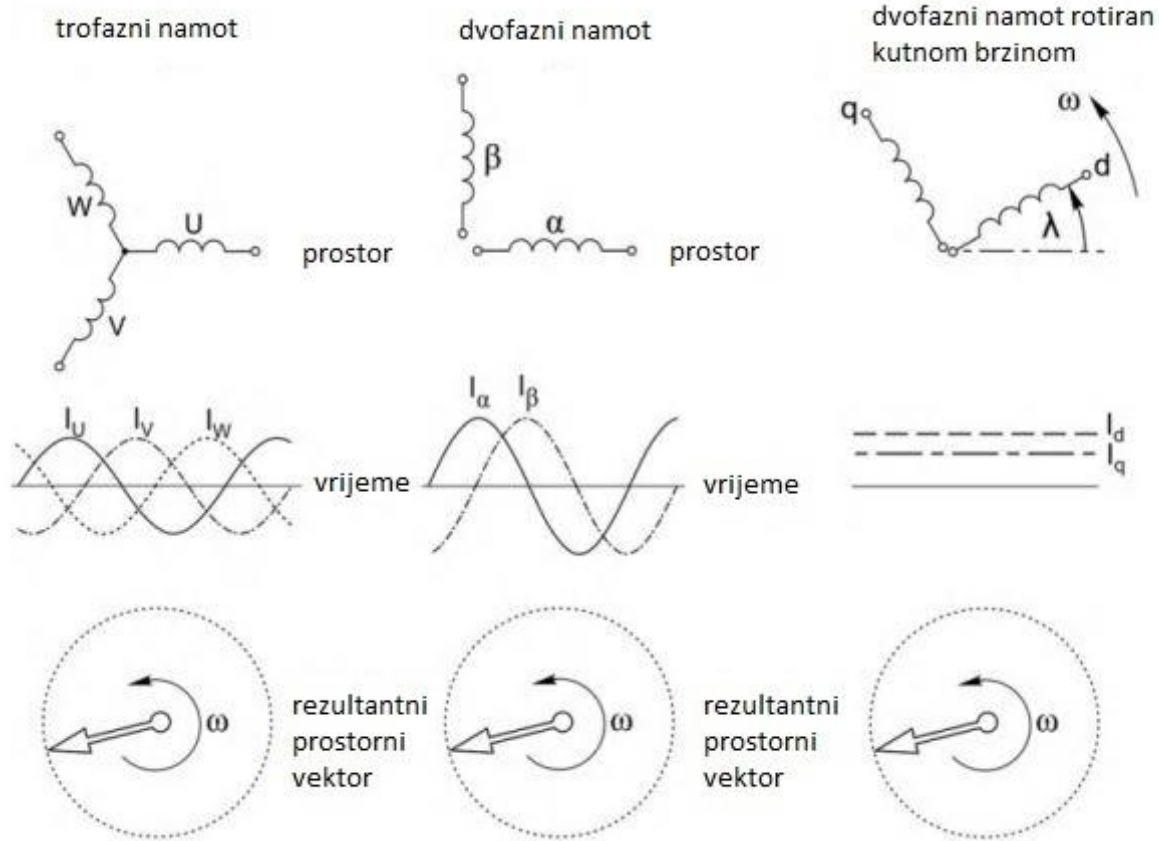
Skalarno upravljanje najčešće se koristi u pogonima gdje su jednostavniji zahtjevi kao što su pumpe, ventilatori ili pomične trake, gdje nije naglasak na prevelikoj preciznosti i dinamičkim svojstvima pogona. Upravljanje može biti u otvorenoj ili zatvorenoj petlji.

Pored skalarnog upravljanja objasniti će se ukratko i vektorsko upravljanje asinkronog motora. Nakon što su prikazani nedostaci i ograničenja skalarnog upravljanja, potrebe za preciznijom regulacijom dovele su do razvoja vektorskog upravljanja. Naime kod skalarnog upravljanja prijelazne pojave nisu uključene u promatranje, pa se u prijelaznim pojavama ne može dobiti konstantan magnetski tok, a s njime i moment. Magnetski tok i moment kod skalarnog upravljanja nisu raspregnuti, odnosno obje veličine ovise o frekvenciji i naponu, stoga njima se ne može upravljati odvojeno. Vektorsko upravljanje omogućava odvojeno upravljanje momentom i magnetskim tokom, a nudi kvalitetnije upravljanje kod sustava visokih zahtjeva.

Glavna pozadina ostvarivanja upravljanja, koje se traži prilikom zahtjevnijih pogona, je vektorsko upravljanje temeljeno na analogiji s istosmjernim motorom s nezavisnom uzbudom gdje se neovisno upravlja magnetskim tokom i momentom. Korišteni algoritmi koji se pritom koriste zahtijevaju brzo izvođenje računskih operacija i zasnivani su na brzom procesoru za obradu signala DSP-om (*engl. Digital Signal Processor*).

Istosmjerni motor ima mirujuće magnetsko polje uzrokovano permanentnim magnetima ili uzbudnim namotom na statoru i rotirajući armaturni namot na rotoru protjecan istosmjernom strujom. Magnetski tok i struja armature istosmjernog motora, o kojoj ovisi moment, mogu se izravno i nezavisno upravljati i lako mjeriti za razliku od izmjeničnog motora gdje te veličine nisu raspregnute i utječu jedna na drugu. Magnetski tokovi koji su stvoreni strujom armature i uzbude okomiti su jedan na drugi i raspregnuti. Struja  $i_u$  utječe samo na uzbudni tok  $\psi_u$ , a armaturna struja utječe samo na armaturni tok  $\psi_a$ . Kako bi se dobio matematički proračun i kako izmjenične veličine izmjeničnog motora prikazati kao istosmjerne, uvode se oznake  $d$  (uzdužna komponenta) i  $q$  (poprečna komponenta). Ako se izmjenični stroj promatra u  $d$ - $q$  koordinatnom sustavu koji rotira određenom brzinom  $\omega$  i ako su u tom sustavu promatraju sinusne veličine asinkronog stroja, promatrač koji se nalazi u tom sustavu i rotira zajedno s njim izmjenične veličine vidi kao istosmjerne. Ako komponente statorske struje  $i_s$  označimo s  $I_{ds}$  i  $I_{qs}$  (uzdužna i poprečna komponenta statorske struje) i promatramo ih kao izmjenične komponente magnetskog toka i izmjenične komponente momenta dobije se ekvivalencija prikazana na slici 2.9. Na ovom principu smišljena je  $d$ - $q$  transformacija gdje se sinusne veličine transformiraju u izmjenične kako bi se izmjenični motor upravljao kao istosmjerni sa odvojenom regulacijom momenta i magnetskog toka. Najprije se trofazne sinusne veličine Clark-ovom transformacijom prebacuju u dvofazne sinusne (alfa-beta), a Park-ovom transformacijom prebacuju u istosmjerne veličine (d-q). [6]

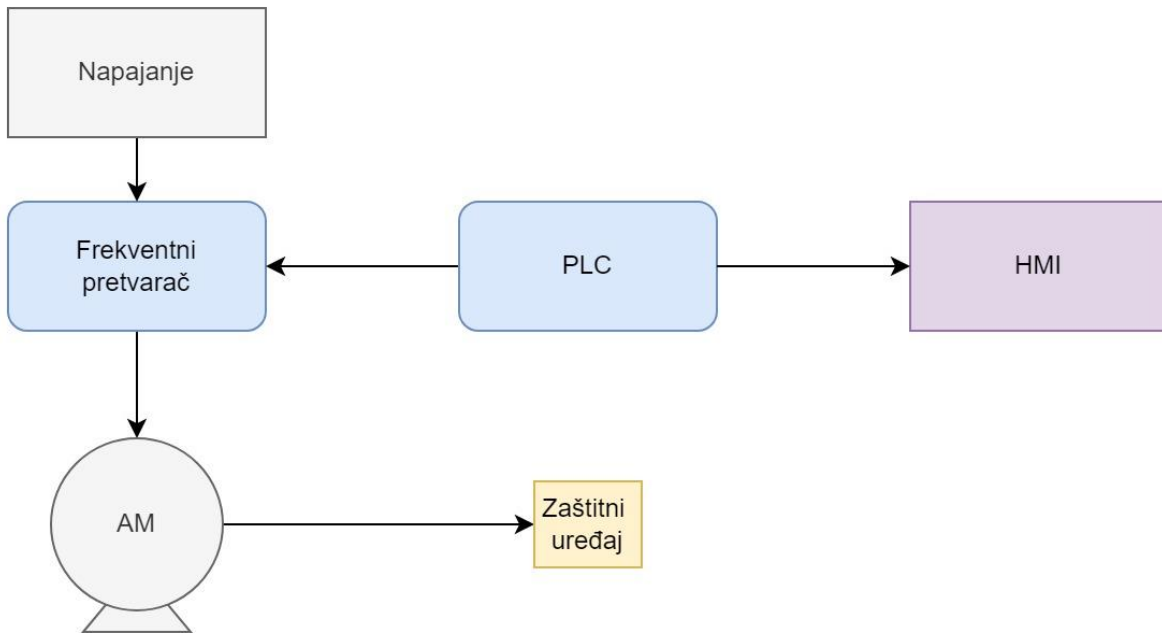




**Slika 2.9** – Prikaz prelaska iz trofaznog sustava u sinkrono rotirajući d-q sustav. [6]

Rješenje ovog zadatka bilo bi izrada sustava za upravljanje i nadzor asinkronog motora i frekventnog pretvarača putem programiranog industrijskog računala i pripadajućeg SCADA sustava. Nakon fizičkog povezivanja svih elementa, izradio bi se programski kod u kojem bi se uspostavila komunikacija između industrijskog računala i frekventijskog pretvarača koji bi upravljao motorom na zadani način što je prikazano u poglavljima 2.2 i 2.3.

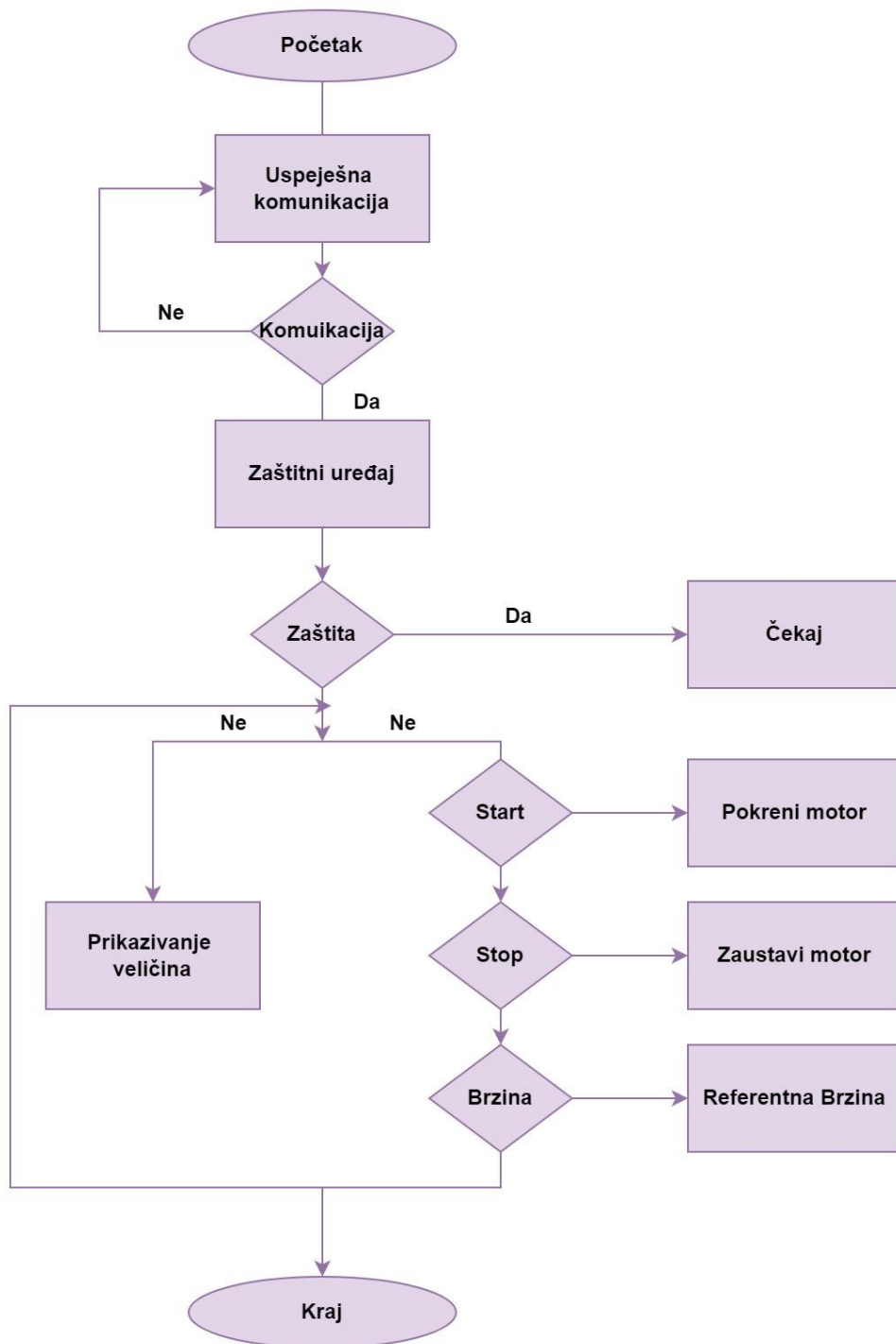
## 2.2. Sklopovsko rješenje



*Slika 2.10 – Prikaz načelne sheme sklopovskog rješenja.*

Frekventni pretvarač spaja se između motora i trofazne mreže vrijednosti napona 400V. Komunikacijskim kabelom frekventni pretvarač spaja se sa programibilnim logičkim kontrolerom, odnosno industrijskim računalom. Istom tom komunikacijom priključuje se PC na kojem se nalazi grafičko sučelje. Na digitalne ulaze frekventnog pretvarača priključuje se sklopka koja ima ulogu zaštitnog uređaja.

## 2.3. Programsko rješenje



Slika 2.11 – Prikaz blok sheme programskog rješenja.

### 3. REALZACIJA I IMPLEMENTACIJA SUSTAVA

#### 3.1. Opis korištenih alata i uređaja

U ovom poglavlju objasniti će se korištena oprema i alati koji su potrebni za ispravni rad, testiranje modela kao, program i parametrisiranje uređaja.

##### 3.1.1. Končar MES Elektromotor

U ovom radu koristio se kao glavni element asinkroni motor tvrtke Končar. Motor je robusan i relativno malih dimenzija te idealno odgovara u rad.

Tablica 3.1– Tehničke karakteristike asinkronog motora Končar.

	D	Y
HZ	50	50
KW	0,75	0,75
V	230	400
A	3	1,75
COS $\phi$	0,75	0,75
RPM	1410	1410



Slika 3.1 – Nazivna pločica motora.

### 3.1.2. Vacon 100 FLOW

VACON 100 FLOW je frekventni pretvarač razvijen kako bi poboljšao kontrolu elektromotora u automatiziranim postrojenjima. Ovaj model može pokretati motore u rasponu od 0.55kW – 2.2kW. Na izlazu pruža različite razine napona ovisno o spojenom uređaju. Razvijen je kako bi unaprijedio kontrolu, zaštitu i uštedu energije kod pumpi i ventilacijskih uređaja te poboljšao pouzdanost rada sustava. Sadrži razne programske zaštite što se tiče zaštite motora i namota. Ima ugrađenu termičku zaštitu zajedno sa dugotrajnim kondenzatorom u istosmjernom međukrugu. Predefimirani suvremeni kontroleri mogu kontrolirati npr. brzinu vrtnje pumpe koristeći signale koje prima sa senzora što omogućuje brže i točnije reagiranje na poremećaje ili na naredbu. Podržava razne industrijske komunikacije kao što su Mosbus RTU, Modbus TCP, Profinet itd. [7]

*Tablica 3.2 – Tehničke karakteristike frekvencijskog pretvarača VACON 100*

SERIJSKI BROJ PROIZVODA	VACON 0100-3L-0005-5-FLOW
SERIJA	VACON 100
PROIZVOĐAČ	VACON
IZLAZNA SNAGA	2.2KW
IZLAZNA STRUJA	5.6A
ZAŠTITA KUĆIŠTA	IP21
RADNI NAPON	380-500V
BROJ FAZA	3
DIMENZIJE	128X328X190MM
TEŽINA	6KG

### 3.1.3. Siemens SIMATIC ET200SP

Radi se o programabilnom logičkom kontroleru (*engl. PLC – Programmable Logic Controller*). Razvoj poluvodičkih komponenti omogućio je i razvoj industrijskih računala. PLC je kompaktno industrijsko računalo, bazirano na mikrokontroleru, dizajnirano za kontrolu i upravljanje procesima i pogonima putem računala. Koriste se u tvornicama i industrijskim postrojenjima kako bi upravljali motorima, pumpama, rasvjetom, ventilatorima, zaštitom i ostalim uređajima. Industrijska računala zamijenila su elektromehaničku analognu relejnu tehniku koja je skupa i ujedno veliki potrošač električne energije. Umjesto više upravljačkih ormara i operatera koji kontroliraju postrojenje, PLC omogućava upravljanje putem računala sa jednog mjesta. Procesni se podaci obrađuju tako da se program izvodi ciklički, očitavanjem vrijednosti senzorskih uređaja koji su priključeni na PLC putem ulazno-izlazne jedinice (*engl. I/O unit*). U odnosu na primljene podatke donose se odluke koje su zadane upisanim programom. Nakon obrade podataka, šalju se signali također putem I/O jedinice aktuatorima koji izmjenjuju postavke procesa i reguliraju ga. [8]

PLC se najčešće mogu naći kao modularni, ali i kao manji kompaktni uređaji. Sastoje se od napajanja, procesorske jedinice te ulazno izlazne jedinice. Procesorska jedinica sastoji se od mikroprocesora koji obrađuje ulazno izlazne podatke, memorije u kojoj se spremljen operacijski sustav (ROM) i promjenjive memorije u koju se upisuje program (RAM). RAM memorija koristi se i za mjerače vremena, brojila, stanje ulaznih i izlaznih jedinica i podatke. Postoje i ostale jedinice poput jedinica za komunikaciju za daljinsko komuniciranjem da udaljenim uređajima kao što su LAN, Modbus, Profinet itd.

Ulazno-izlazna jedinica služi za prikupljanje informacija od strane pogona i slanja informacija u pogon. Signali mogu biti digitalni ili analogni. Kod digitalnih ulaznih i izlaznih signala, moguća su samo dva stanja kao što su naredbe upali ili ugasi. Analogni signalu imaju veću rezoluciju očitavanja ili slanja signala jer se mjeri stvarna vrijednost struje ili napona od strane senzora ili aktuatora. Tipovi analognih senzora mogu biti osjetnici temperature, senzori pritiska, vlage ili protoka itd. Aktuatori mogu biti jednostavni svjetlosni indikatori, elektromehanički ventili ili frekventni pretvarači itd. [9]

Kao PLC u ovom radu koristiti će se SIEMENS SIMATIC DP CPU procesorska jedinica za ET200SP sustave. Upravljačka jedinica CPU (*engl. Central Processing Unit*) sastoji se od tri Profinet kanala za komunikaciju i programiranje. Raspon napona na kojem procesorska jedinica je od 10.2V – 28.8V DC. Što se tiče memorije, za programe jedinica raspolaže sa 200KB memorije dok je za podatkovnu pohranu

osigurano 1MB memorije. ET200SP modularan je program gdje se na upravljačku jedinicu dodaju analogni ili digitalni I/O moduli, komunikacijski moduli, moduli za pokretanje motora...itd. Ovakva upravljačka jedinica idealna je za kontrolu u decentraliziranim postrojenjima. [10]

Tablica 3.3 – Tehničke specifikacije Simens Simatic ET200SP

SERIJSKI BROJ PROIZVODA	6ES7512-1DK01-AB0
TIP PROIZVODA	CPU 1512SP-1 PN
DIMENZIJE	100X75X117MM
RADNI NAPON	24V
POTROŠNJA STRUJE	0.6A
NAJVEĆA STRUJA	4.7A
BROJ UTORA ZA MEM. KARTICU	1
RADNA MEMORIJA ZA PROGRAM	200KB
RADNA MEMORIJA ZA PODATKE	1MB

### 3.1.4. SCADA

Uvođenjem industrijskih računala u pogone i postrojenja te njihovo spajanje na lokalne mreže omogućilo je i razvoj SCADA sustava (*eng. Supervisory Control And Data Acquisition*) odnosno računalni sustav za nadzor, mjerenje i upravljanje industrijskih veličina. SCADA sustav su pogodni u automatiziranim pogonima gdje je moguće decentralizirano upravljanje. SCADA sustavi objedinjuju sustave prikupljanje podataka, sustave obrade podataka te sustave prikazivanja i praćenja procesnih veličina u stvarnom vremenu. Takvi sustavi dizajnirani su za prikupljanje procesnih veličina, slanje tih podataka do centralnog računala i prikazivanja tih vrijednosti u grafičkom ili tekstualnom obliku na ekranu. SCADA sustavi sastoje se od hardverskog i programskog dijela. Što se tiče hardverskog dijela koriste se MTU uređaj (*engl. Master Terminal Unit*) koji se koristi kao glavno ili centralno računalo. Uređaji poput PLC-a ili IED-a (*engl. Intelligent End Device*) te RTU-a (*engl. Remote Terminal Unit*) koriste se za kontroliranje lokalnih procesa, upravljanje aktuatorima i senzorima. Oni su raspoređeni po postrojenju i spojeni su sa senzorima i aktuatorima. MTU sprema i obrađuje podatke koje prima sa prostorno raspoređenih RTU-a koji prikupljaju podatke sa IED-a. [11]

Programski dio SCADA sustava sastoji se od sustava za prikazivanje trenutnih procesnih veličina i njihovo spremanje u bazu podataka, prikaz događaja i alarma. Umrežavanje i moderniziranje postrojenja omogućila su zamjenu starih kontrolnih panela sa modernijim displejima. Uvodi se pojam HMI (*engl. Human Machine Interface*) koji je dio SCADA sustava. On omogućuje operatoru postrojenja grafički prikaz stanja postrojenja ujedno sa direktnom kontrolom procesa. HMI grafičko sučelje može biti programirano tako da prikazuje važne parametre i kontrolne informacije operateru. [12]

### **3.1.5. Fieldbus komunikacija**

Automatizirana postrojenja posjeduju, kako je ranije navedeno, sustave kontrole koji obrađuju informacije senzora te ih šalju aktuatorima kako bih primijenili eventualne promijene u kontroli ili zadržali iste parametre u takozvanoj zatvorenoj petlji upravljanja. Sustavi upravljanja sa zatvorenom petljom su sustavi koji imaju povratu vezu, odnosno uvid u stvarno stanje procesa u stvarnom vremenu. Takvi sustavi jako ovise komunikaciji kako bi neometano i pogreške slali i primali informacije. SCADA sustavi također ovise o komunikaciji kako bih se informacije prikazivale na grafičkim sučeljima u kontrolnim sobama. Nakon obrade i donošenja odluke, informacije se šalju prema aktuatorima.

Serijski komunikacijski protokoli koriste se za upravljanje i kontrolu jer imaju manji podatkovni kapacitet, ali omogućuju konstantno podatkovno kašnjenje sa vrlo malim oscilacijama kao što su CAN, ControlNet, i PROFIBUS. Ethernet komunikacijski protokoli koriste se za praćenje i nadzor jer imaju nešto veći podatkovni kapacitet, ali mogu biti u mogućnosti tolerirati veće oscilacije u kašnjenju podataka kao što su Ethernet TCP/IP. [13]

Ethernet se najčešće koristi u lokanim LAN mrežama. Predstavlja protokole i hardver za prijenos podataka među čvorovima. Svaki čvor ima jedna pravo pristupa mreži. Uređaji komuniciraju na mreži putem mrežne kartice gdje svaka kartica ima svoju MAC adresu. Adresa se sastoji od 6 bajtova od kojih 3 identificiraju proizvođača, a ostatak karticu. Internet protokol IP omogućava usmjerivanje podataka različitih mreža. Svaki čvor koji komunicira ima jedinstvenu IP adresu od 32bita (IPv4) gdje jedan dio predstavlja mrežnu adresu, a drugi dio adresu čvora.

TCP protokol je komunikacijski protokol koji omogućuje komunikaciju računala uz pouzdan prijenos, mehanizme za detekciju pogrešaka i rasprostranjen je diljem interneta. [14]

U ovom radu za komunikaciju između frekventnog pretvarača, PLC-a i PC stanice koristit će se Profinet komunikacija. Profinet je industrijski komunikacijski standard temeljen na Ethernet tehnologiji, a razvijen od strane „*Profibus and Profinet International*“, i dio je IEC 61158 i IEC 61784 internacionalnog standarda za



komunikaciju sabirnica polja (*engl. Fieldbus*). Koristi PROFIBUS tehnologiju u pogledu brzine prijenosa podataka i korištenja informacijskih tehnologija. Profinet koristi podržava TCP/IP protokol za komuniciranje, konfiguraciju i dijagnosticiranje uređaja. Postoje tri inačice Profinet-a koje su se godinama razvijale, a prva inačica bazirana na Ethernet TCP/IP protokolu pri non-RT tehnologiji (*eng. non-Real Time*). Druga inačica dodaje vremenski-kritičnu komunikaciju te bi se zaobišli 3. i 4. ISO sloj. Treća inačica dodaje PROFIBUS IRT (*engl, PROFIBUS Isochronous Real Time*) koji omogućuje vrijeme ciklusa manjim od 1ms i bazira se na hardverskom rješenju koja je dodatak 2. ISO sloju. [15]

Profinet koristi tri razine protokola:

- TCP/IP koji se koristi za vremenski nekritične podatke, a vremena ciklusa su reda veličine 100ms. Ne može se koristiti u stvarnom vremenu. (Parametriranje uređaja)
- RT (Real Time) koji se koristi za komunikaciju u stvarnom vremenu vremenom ciklusa oko 10ms. (Ciklička komunikacija između uređaja i kontrolera)
- IRT (*engl. Isochronous Real Time*) koji se koristi za komunikaciju u stvarnom vremenu sa vremenom ciklusa od 1ms.

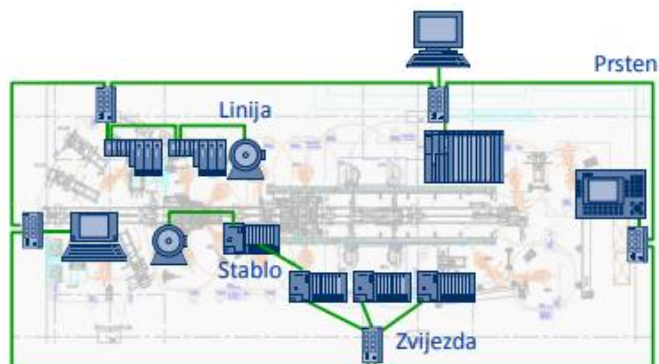
Prednosti Profinet-a su ti što omogućava komunikaciju u stvarnom vremenu, podržava postojeću infrastrukturu mreža (switch-ovi i router-i), brzina prijenosa podatak može doseći i 100Mb/s pri dvostrukoj komunikaciji. Podržava TCP/IP protokol i bežični prijenos podataka, broj čvorova u mreži je fleksibilan.

PROFINET definira i dva komunikacija protokola:

- PROFINET CBA
- PROFINET IO

PROFINET CBA je industrijski protokol koji se bazira na objektno orijentiranom programiranju

PROFINET IO se koristi u industrijskoj automatizaciji za integraciju na Ethernet mreži, a topologije mogu biti: zvijezda, stablo, prsten i linija.



Slika 3.2 – Prikaz vrsti umrežavanja sustava. [13]

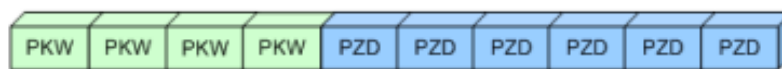
PROFINET IO mreža definirana su tri tipa uređaja:

- IO kontroler – upravlja IO uređajima (PLC)
- IO uređaj za nadzor - služi za puštanje u pogon ili podešavanje parametara IO uređaja (HMI)
- IO uređaji – uređaji koji procesiraju signale i prenose ih IO kontroleru ili uređaju za nadzor (Mjerni uređaji)

PROFINET IO koristi cikličku komunikaciju prilikom razmjene podataka između kontrolera i uređaja (RT, IRT). Aciklička komunikacija koristi se za konfiguraciju i parametriranje IO uređaja te slanje dijagnostičkih signala. Prilikom adresiranja na Profinet mreži, svaki uređaj ima svoju jedinstvenu MAC adresu, IP adresu i ime uređaja. IP adresa se može dodijeliti od strane uređaja za nadzor, kontrolera ili servera. IO kontroler adresira IO uređaj samo ako IO uređaj ima pridruženo ime koju mu daje IO uređaj za nadzor. Prije uspostavljanja komunikacije s IO kontrolerom u IO kontroler najprije se mora učitati konfiguracija IO uređaja, a konfiguracija je upisana u GSD datoteku. Nakon uspješne konfiguracije, uređaj i kontroler razmjenjuju podate ciklički.

PROFINET struktura telegrama mogu se podijeliti na dva dijela:

- PZD – dio telegrama s podacima iz procesa, koristi se za davanje upravljačkih signala, referentnih vrijednosti, čitanje statusa i trenutnih stanja.
- PKW – dio telegrama za parametriranje, koristi se za čitanje i pisanje parametara.



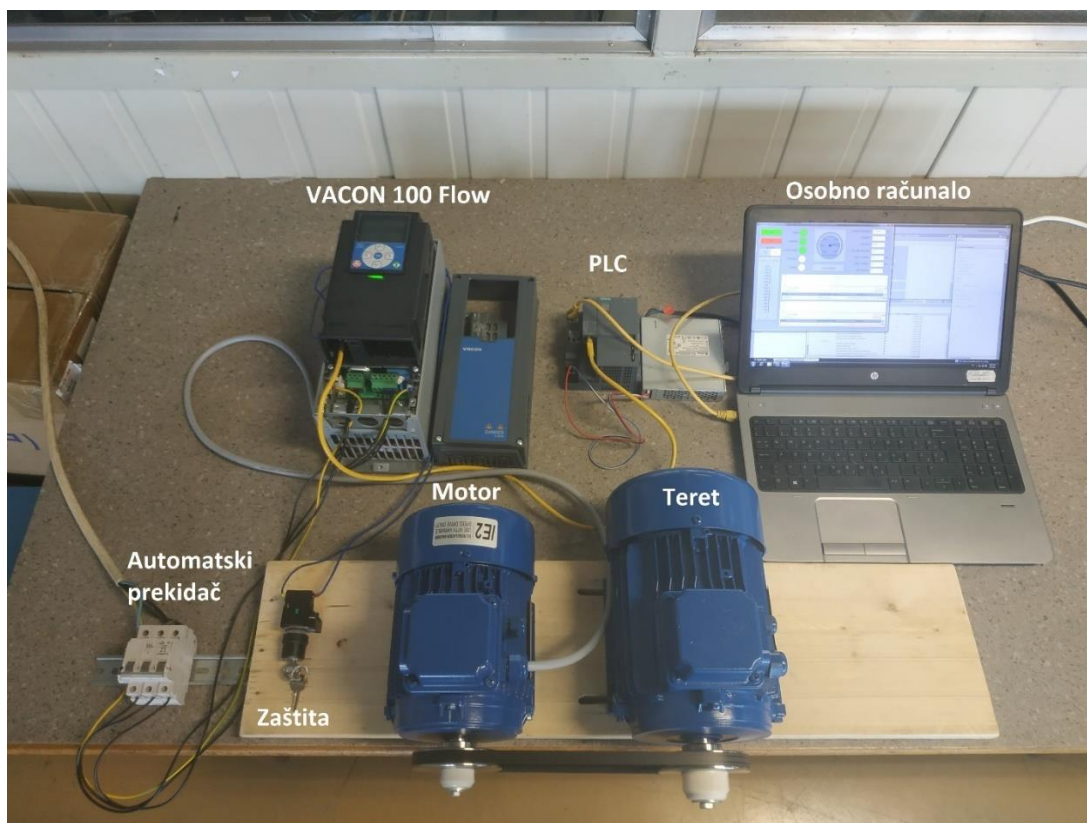
*Slika 3.3 – Struktura Profinet telegrama. [13]*

Odabirom odgovarajućeg telegrama definiraju se podaci koji se prenose između IO kontrolera i IO uređaja. Postoje predefrirani telegrami, specifični telegrami koje definira sami proizvođač i slobodni telegrami koje definira korisnik. [13,15]

### **3.2. Realizacija sklopovskog rješenja**

Sklop odnosno uređaji u i oprema objedinjeni su na način da je frekventni pretvarač priključen na trofaznu mrežu. Između mreže i frekventnog pretvarača nalaze se automatski prekidač kao zaštitni uređaj frekventnog pretvarača. Asinkroni motor priključen je na stezaljke frekventnog pretvarača. Industrijsko računalo odnosno PLC spojeno je sa frekventnim pretvaračem putem Ethernet-a odnosno putem Profinet komunikacije.

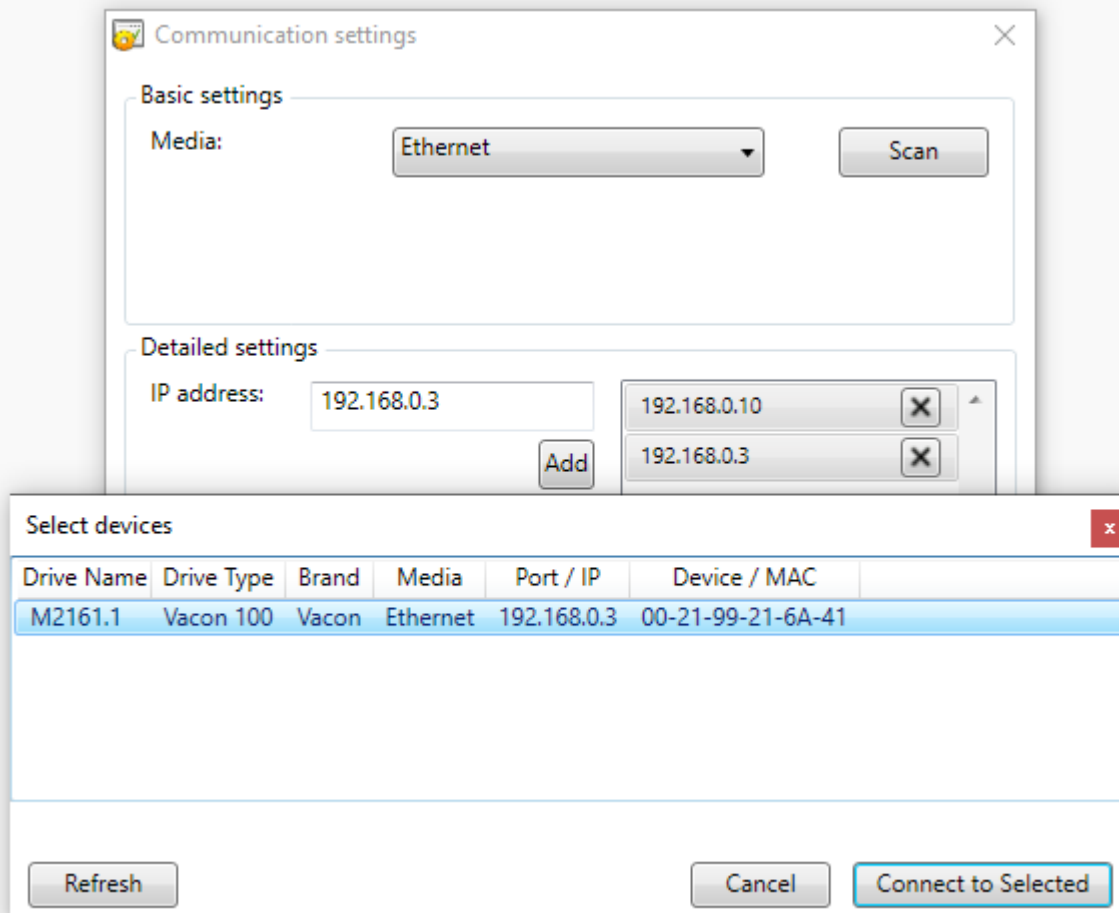
Prijenosno računalo, koje ujedno služi i za prebacivanje programa na PLC i kao HMI, spojeno je sa PLC-om putem Ethernet-a. Slika 3.6 prikazuje stvarno stanje spojenih elemenata. Kao opterećenje asinkronog motora koristi se drugi asinkroni motor od 2.2kW. Motori sadrže remenice na svojim osovinama, a spojeni su remenom. Na digitalni input frekventnog pretvarača priključena je sklopka sa ključem, koja služi kao zaštitni uređaj za blokiranje motora i aktiviranje greške.



*Slika 3.4 – Prikaz sklopovskog rješenja modela*

### **3.3. Realizacija programskog rješenja**

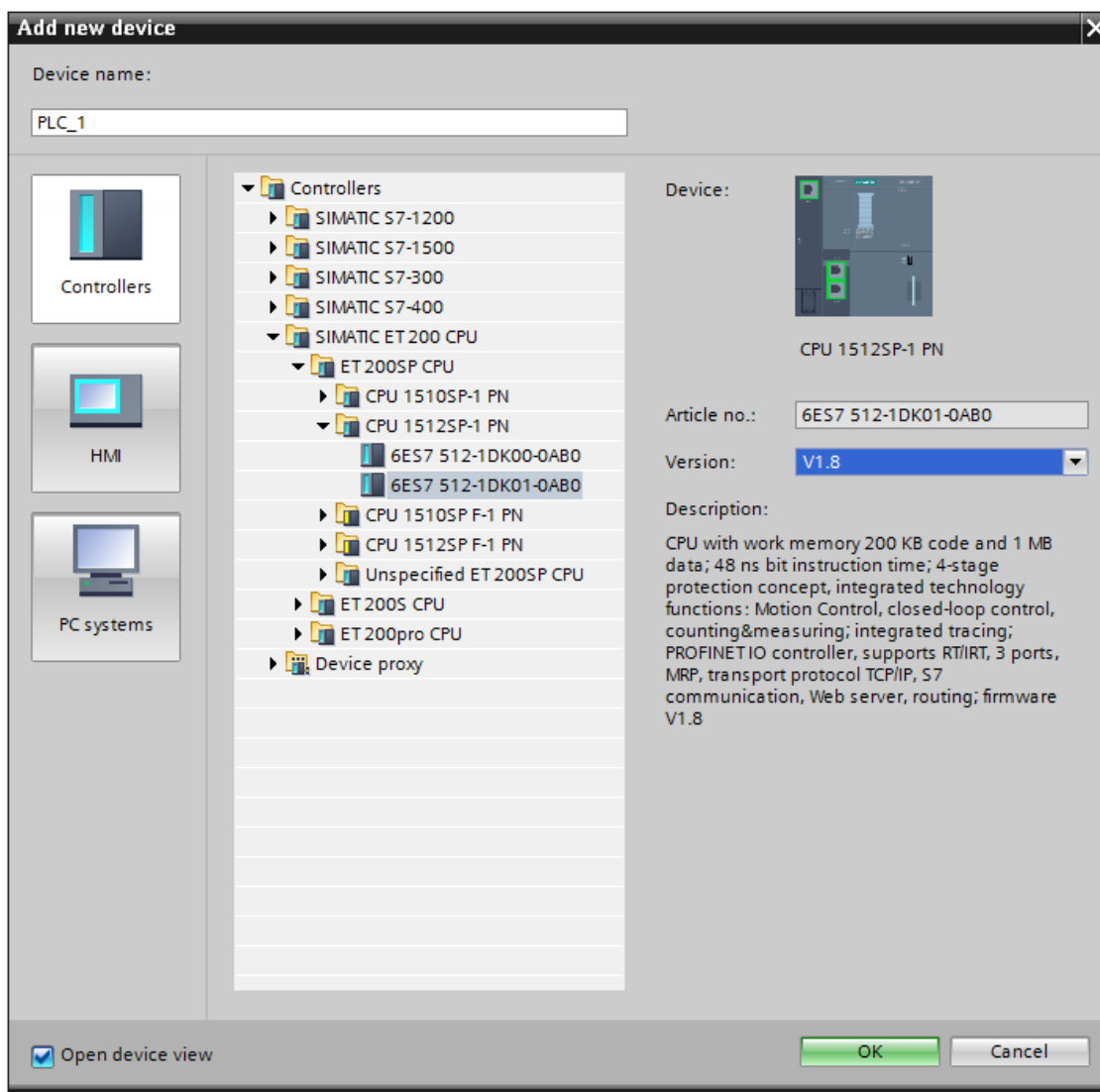
Za unos parametara motora unutar frekventnog pretvarača koristili se program pod nazivom „VACON Live PC Tool“. Vacon FLOW sadrži zaslon zajedno sa funkcijskim tipkama za unos parametara i kontrolu. Takav način unosa parametara pogodan je za pogone ili terenske radove, ali programski paket olakšava unos parametar putem osobnog računala. Vacon Live program prilikom instalacije automatski instalira drivere za komunikaciju. Nakon spajanja frekventnog pretvarača i računala putem mrežnog kabela u Vacon Live program odmah je prepoznao odgovarajući frekventni pretvarač i ponudio nam parametre koji se mogu unijeti.



*Slika 3.5 – Prikaz odabira frekventnog pretvarača unutar Vacon Live programa*

Prva grupa parametara za unos su parametri motora koji se mogu naći na nazivnoj pločici (slika 3.1). Pod karticom „Ethernet“ unijeli smo podatak da će se komunikacija vršiti putem Profinet komunikacije. Pod karticom „Fieldbus general“ isto je podešena Profinet komunikacija. „Start/Stop Setup kartica omogućuje unos podatka koji govori odakle će frekventni pretvarač primiti naredbe. Postavljen je parametar „Fieldbus Ctrl“. Frekventni pretvarač ima više vrsta rada u ovisnosti o pogonskim zahtjevima, u ovom slučaju postavljen je u „Standard Mode“. Slike programskog sučelja priložene su u prilogu.

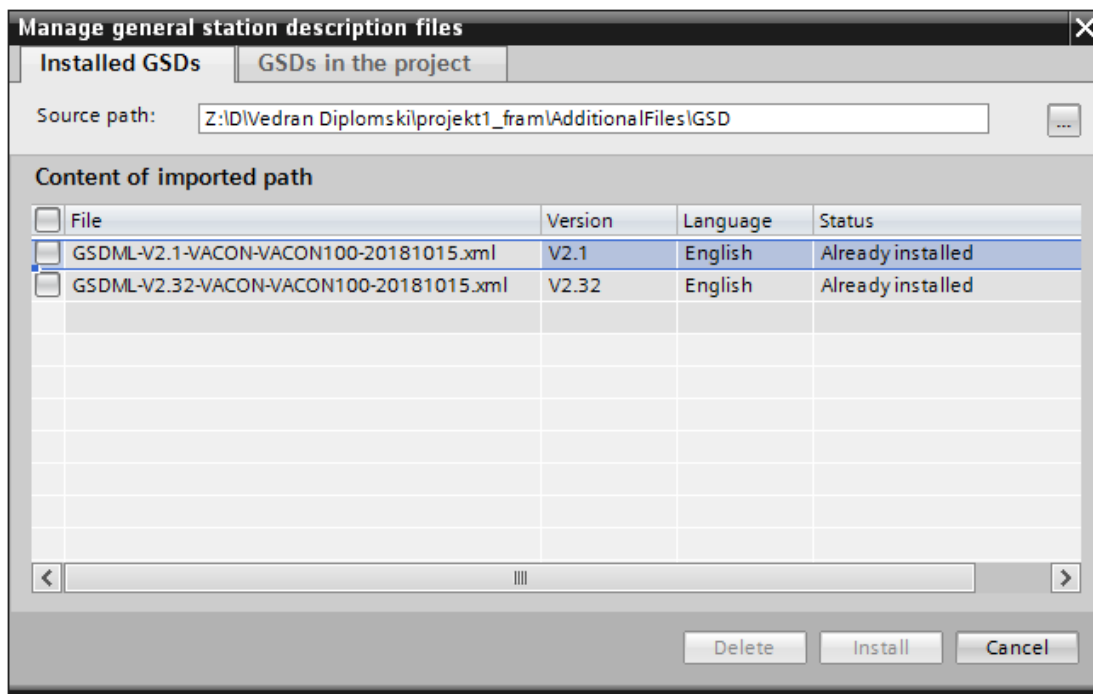
Siemens-ov program koji služi za parametrisiranje PLC-a, pisanje koda, izrada grafičkog sučelja i simulaciju naziva se TIA Portal (*engl. Totally Integrated Automation Portal*). U ovom radu koristila se verzija V14. Nakon otvaranja programa na računalu, odabrao se novi projekt. Na samom početku potrebno je definirati točno koji se uređaji koriste, njihove postavke i način komunikacije. U ovom slučaju uređaji koji se trebaju definirati u TIA Portalu su PLC, frekventni pretvarač i HMI sučelje.



*Slika 3.6 – Prikaz odabira PLC-a unutar TIA Portala.*

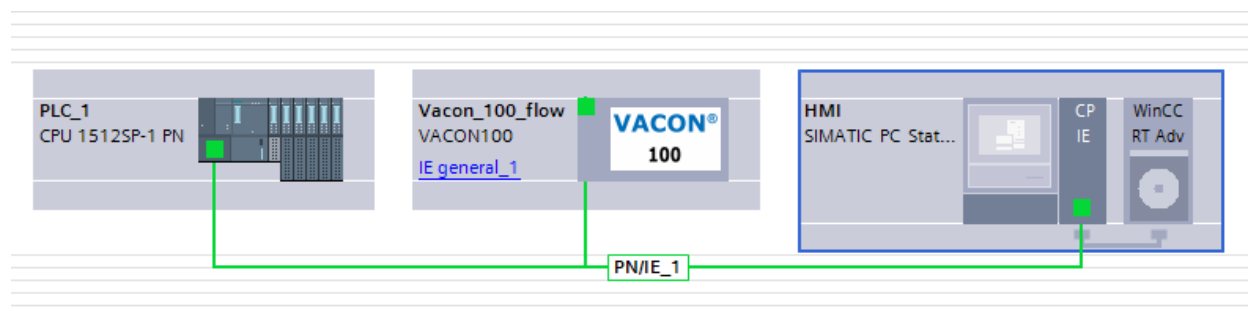
Nakon odabira točno određene procesorske jedinice sljedeći korak je uvesti GSD datoteku kako bi TIA portal prepoznao Vacon 100 Flow unutar programa. To se radi na način da se sa stranice proizvođača, u ovom slučaju „Danfoss“, preuzme datoteka koja se putem opcija u TIA portalu uveze u program. Nakon toga u

opcijama dodavanja uređaja u mrežu imamo mogućnost uvesti Vacon 100 Flow. Slika 3.9 prikazuje kako izgleda prozor gdje se GSD datoteka može uvesti.



*Slika 3.7 – Prikaz dodavanja GSD datoteke Vacon frekventnog pretvarača.*

Nakon dodavanja GSD datoteke Vacon 100 ostaje još samo dodati HMI sučelje. HMI sučelje dodajemo na način da u mrežu dodamo „SIMATIC PC Station“. PC Station-u potrebno je dodati Wincc modul kako bi mogli pokrenuti simulaciju i izraditi grafičko sučelje te dodati mu modul za Profinet komunikaciju. Nakon završenog dodavanja uređaja slika 3.8 prikazuje izgled mreže uređaja.



*Slika 3.8 – Prikaz mreže uređaja spojene Profinet komunikacijom.*

Kako bih se uspostavila komunikacija odnosno kako bih pravi podaci stizali uređajima potrebno je pronaći tablicu gdje je definirano točno koji Bit ili skup Bitova odgovara određenoj funkciji. Tablica 3.5, nalazi se u dokumentaciji koja dolazi sa Vacon 100 Flow uređajem, prikazuje koji Bit odgovara željenoj funkciji frekventnog pretvarača. Bit-ovi se šalju Profinet komunikacijom i njihov skup se naziva telegram. U ovom slučaju prilikom podešavanja komunikacije postavljen je telegram „Vendor 1“ što znači da PLC komunikacijom šalje 32bit-na riječ gdje je prvih 16bit-ova rezervirano za kontrolnu riječ, a drugih 16bit-a za vrijednost referentne brzine. Isto tako statusnu riječ frekventni pretvarač šalje PLC-u koja isto sadrži 16bit-ni zapis statusnih informacija i 16bit-ni zapis vrijednost trenutne brzine motora. Ovi telegrami su fiksni i ne mogu se mijenjati na način da izmjenjuju mjesta Bit-ova funkcija.

Ukratko ako se Bit 0 u telegramu kontrolne riječi koji se šalje pretvaraču putem Profinet komunikacije postavi u 1 tada će frekventni pretvarač dobiti informaciju da može pokrenuti motor. Tablica 3.4 prikazuje Bit-ove statusne riječi koje šalje frekventni pretvarač.

*Tablica 3.4 – Vacon 100 segmenti statusne riječi*

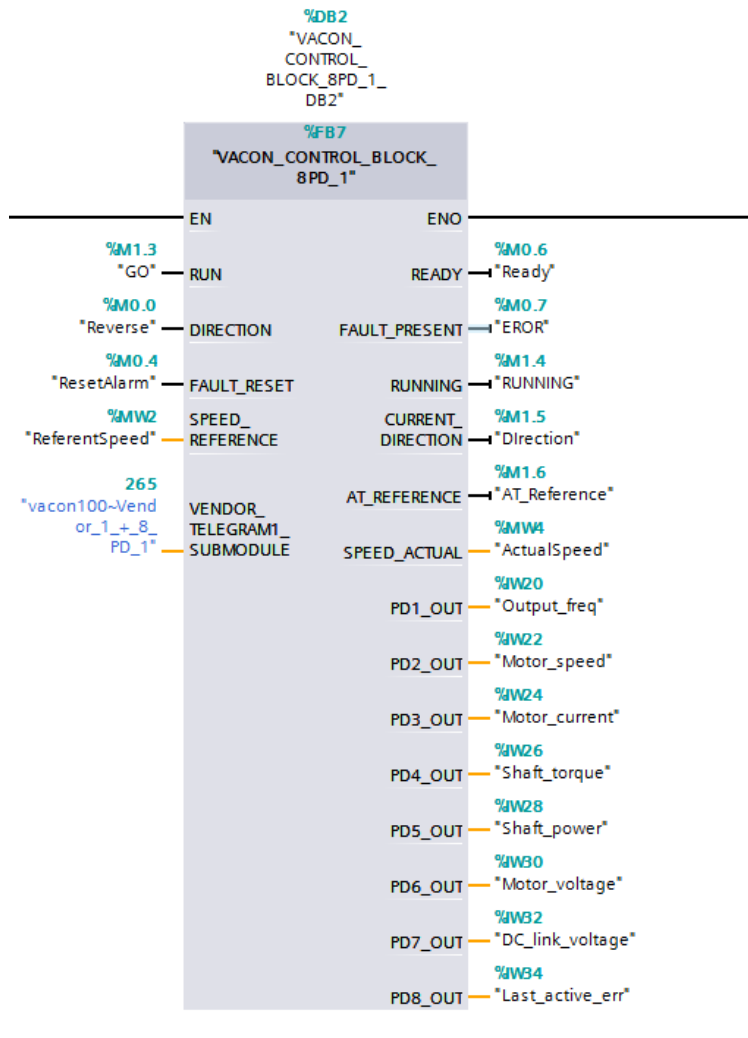
BIT	FUNKCIJA	OPIS	
0	SPREMAN	0	UREĐAJ NIJE SPREMAN
		1	UREĐAJ JE SPREMAN
1	U POGONU	0	MOTOR NIJE U POGONU
		1	MOTOR JE U POGONU
2	SMJER	0	SMJER JE NAPRIJED
		1	SMJER JE REVERZIRAN
3	GREŠKA	0	NEMA GREŠKE
		1	IMA GREŠKE
4	ALARM	0	NEMA ALARMA
		1	IMA ALARMA
5	NA REFERENTNOJ VRIJEDNOSTI	0	MOTOR NIJE NA REFERENTNOJ VRIJEDNOSTI
		1	MOTOR JE NA REFERENTNOJ VRIJEDNOSTI
6	BRZINA NA 0	0	MOTOR NIJE NA VRIJEDNOSTI BRZINE 0
		1	MOTOR JE NA VRIJEDNOSTI BRZINE 0
7	MAGNETNI TOK SPREMAN	0	MOTOR NIJE MAGNETIZIRAN
		1	MOTOR JE MAGNETIZIRAN
8-12	REZERVIRANO		

*Tablica 3.5 – Vacon 100 segmenti kontrolne riječi..*

BIT	FUNKCIJA	OPIS	
0	START/STOP	0	STOP
		1	KRENI
1	SMJER	0	NAPRIJED
		1	REVERZIRANJE
2	PONIŠTAVANJE GREŠKE	0	NIŠTA
		1	PONIŠTAVANJE GREŠKE
3	STOP MOD 1	0	NIŠTA
		1	STOP SA SLOBODNIM ZAUSTAVLJANJEM
4	STOP MOD 2	0	NIŠTA
		1	STOP SA ZAUSTAVLJANJEM PO RAMPI
5	BRŽE VRIJEME RAMPE	0	NORMALNO VRIJEME RAMPE
		1	KRAČE VRIJEME RAMPE
6	ZAMRZAVANJE PODEŠENE VRIJEDNOSTI	0	MIJENJANJE REFERENTNE BRZINE VRTNJE OD STRANE APLIKACIJE
		1	MIJENJANJE REFERENTNE BRZINE VRTNJE OD STRANE FIELDBUS-A
7	VRIJEDNOST BRZINE POSTAVI NA 0	0	VRIJEDNOST REFERENTNE BRZINE VRTNJE UZIMA SE SA FIELDBUS-A
		1	POSTAVLJANJE REFERENTNE BRZINE VRTNJE NA 0
8	ZATRAŽI KONTROLU SA FIELDBUS-A	0	KONTROLA POSTAVLJENA NA UREĐAJ
		1	KONSTROLA POSTAVLJANA NA FIELDBUS
9	ZATRAŽI REFERENTNU BRZINU SA FIELDBUS-A	0	REFERENTNA BRZINA POSTAVLJENA U UREĐAJU
		1	REFERENTNA BRZINA UZIMA SE SA FIELDBUS-A
10	IMPULSNI RAD 1	0	NIŠTA
		1	IMPULSNI MOD 1
11	IMPULSNI RAD 2	0	NIŠTA
		1	IMPULSNI MOD 2
12	BRZO ZAUSTAVLJANJE	0	NIŠTA
		1	ZAŠTITNO BRZO ZAUSTAVLJANJE
13-15	REZERVIRANO		

Kada je komunikacija definirana može se preći na programski kod. Unutar TIA Portala koriste se osnovna tri programska jezika, Ladder, FBD i SCL. U programskom kodu korištena je kombinacija Laddera i SCL-a. Programski kod u TIA portal-u moguće je pisati pomoću blokova odnosno pomoću osnovnih logičkih operatora, funkcija ili funkcijskih blokova. Unutar TIA Portal-a uvezen je funkcijskih blok preuzet sa „Danfoss“ stranice zadužen za čitanje i pisanje podataka između PLC-a i pretvarača. Unutar samog funkcijskog bloka nalazi se kod u SCL-u koji definira adrese na kojima se Bitovi nalaze i mogućnost njihove manipulacije ovisno o želji korisnika.





*Slika 3.9 – Funkcijski blok za upravljanje motorom.*

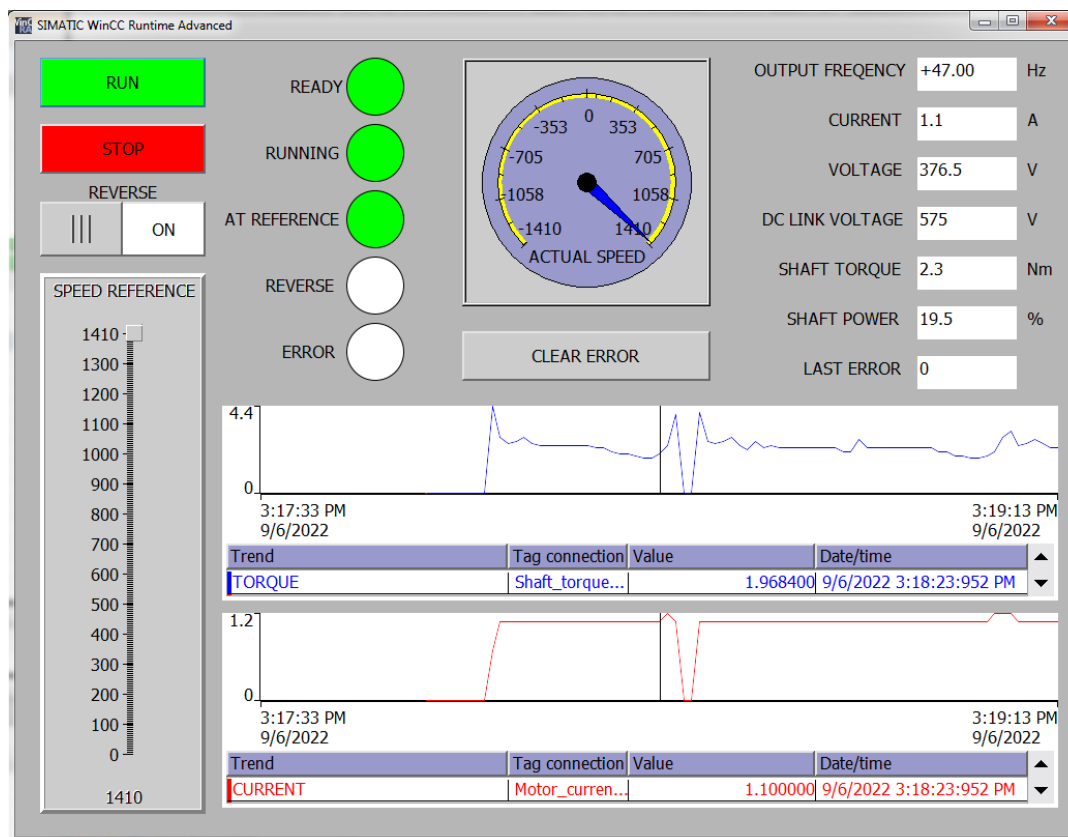
Na slici 3.9 prikazan je funkcijski blok koji se nalazi u „main“ programu. Kao ulaze funkcija prima naredbu za pokretanje, reverziranje, poništavanje greške i prima vrijednost željene brzine vrtnje. Funkcija ovisno o ulaznim stanjima kontrolnoj riječi mijenja Bit-ove i na taj način šalje frekventnom pretvaraču naredbu. Kao izlaze iz funkcijskog bloka postavljeni su signal spremnosti pretvarača, postoji li greška, motor je u pogonu, motor reverzira, motor je na željenoj brzini vrtnje, stvarna vrijednost brzine vrtnje. Dodanih 8 parametara koje pretvarač šalje PLC-u su: frekvencija, brzina motora, struja, moment osovine, snaga osovine, napon, napon istosmjernog međukruga i broj greške. Ostali dio koda napisan je u programskom jeziku Ladder, a sadrži većinom uvjete i podatke za ispravni rad grafičkog sučelja. Cijeli kod i funkcije priloženi su u prilogu.

Važna stvar za napomeniti su „PLC Tags“ odnosno PLC oznake. PLC oznake su dodjeljivanje podacima koji ulaze ili izlaze iz PLC-a memorijsku lokaciju kako bih se spremali i kako bih se njihova vrijednost mogla koristiti u programu. Tako postoje više vrsta PLC oznaka te se za ulazne podatke koriste jedna vrsta oznaka, za izlazne podatke druga vrsta kao i za ostale podatke, koji nisu niti ulaz niti izlaz nego samo rezultati logičkih operacija unutar samog programa, koristi se treća vrsta. Cijela lista PLC oznaka programa priložena je u prilogu.

## 4. TESTIRANJE I REZULTATI

### 4.2. Metodologija testiranja

Metoda testiranja prvo sadržava ispitivanje grafičkog sučelja, njegove funkcionalnosti prema napisanom kodu, ispitivanje komandi, ispitivanje prikazanih vrijednosti parametara motora. Nakon učitavanja programa na PLC, uključuje se grafičko sučelje na osobnom računalo, a njen izgled prikazuje slika 4.1.



Slika 4.1 – Izgled grafičkog sučelja.

Grafičko sučelje sastoji se od komandnih tipki za pokretanje i zaustavljanje motora. Signalne lampice signaliziraju spremnost motora i ispravnu komunikaciju, vrtanju motora, prikaz kada je motor na zadanoj brzini vrtnje, reverzira li motor i eventualno prisustvo greške. Na sredini ekrana nalazi se analogni prikaz brzine vrtnje, a ispod tipka za poništavanje greške. Sa lijeve strane nalazi se sklopka za suprotan smjer vrtnje i linearni potencijometar za reguliranje ili podešavanje željene brzine vrtnje. Na desnoj strani ispisuju se parametri

motora u stvarnom vremenu poput frekvencije, struje, napona, momenta, snage. Donji dio sučelja prikazuje grafove struje i momenta osovine, a ujedno se vrijednosti spremaju na računalo u CSV datoteku.

Iz Spremljene CSV datoteke izradili su se grafički prikazi vrijednosti struje i momenta pri različitim vremenima rampi sa i bez tereta. Vremena rampi mijenjala su se na pretvaraču putem Vacon Live programa.



*Slika 4.2 – Grafički prikaz struje i momenta sa rampom 5s bez tereta.*



*Slika 4.3 – Grafički prikaz struje i momenta sa rampom 5s i teretom.*



*Slika 4.4 – Grafički prikaz struje i momenta sa rampom 10s bez tereta.*



*Slika 4.5 – Grafički prikaz struje i momenta sa rampom 10s i teretom.*

### 4.3. Rezultati testiranja

Testiranje grafičkog sučelja vrlo je jednostavno s obzirom da se sve greške vrlo lako mogu ispraviti. Odziv pretvarača je brz s obzirom da je i Profinet komunikacija brza. Regulacija brzine reagira glatko pomicanjem linearnog potencijometra. Reverziranje motora funkcionira odlično, ako se pri nazivnoj brzini vrtnje aktivira suprotan smjer, motor prvo smanjuje brzinu na vrijednost 0 zatim se zaliječe u suprotnom smjeru do nazivne brzine vrtnje sa negativnim predznakom. Signalne lampice reagiraju na uvjete, prilikom pokretanja, ako je komunikacija uspješna lampica za spremnost pretvarača svijetli zeleno. Kada se postigne referentna brzina vrtnje signalna lampica svijetli zeleno isto tako i lampica za reverziranje pri suprotnom smjeru vrtnje. U slučaju gubitka komunikacije ili aktivacije sklopke sa ključem, koja u ovom slučaju služi kao sigurnosni element, lampica za grešku svijetli crveno i greška je aktivna sve dok ju korisnik ne otkloni pritiskom na tipku. Ispis parametara je točan i precizan. Pretvarač u svojoj dokumentaciji navodi vrste podataka koje šalje putem Profinet-a te se prema tome podaci prilagođavaju stvarnoj vrijednosti. Broj greške spada u „integer“ dok su struja, moment i snaga osovine realni brojevi koji se dijele sa 10.

Kako bih se testirao elektromotor snimila su se 4 grafa vrijednosti struje i momenta pri zaletu po rampi sa i bez tereta. Rampa u ovom slučaju ne utječe na rezultate, njome samo produljujemo vrijeme zaleta motora, odnosno zadajemo motoru željeno vrijeme zaleta od vrijednosti brzine 0 do nazivne brzine vrtnje. Rampom je inače moguće riješiti problem velike pokretne struje. Ono što se može primijetiti je moment motora sa i bez tereta. U slučaju kada se motor vrti u praznom hodu ili bez tereta vidimo manje vrijednosti nadvišenja vrijednosti momenta u odnosu na moment kada je motor opterećen. Razmotri li se struja može se vidjeti prava uloga frekventnog pretvarača gdje struja neovisno o momentu motora i opterećenju zadržava konstantnu vrijednost. Ovdje je primjer kako frekventni pretvarač zaliječe motor od 0 do nazivne brzine vrtnje mijenjajući omjer frekvencije i napona, a rezultat toga je željeni moment potreban za pokretanje i savladavanje tereta uz konstantan iznos struje bez naglih skokova ili velike struje pokretanja koja je nepoželjna i opasna.

## 5. ZAKLJUČAK

Ovaj rad pobliže objašnjava kako moderan pogon ili postrojenje koje sadrži elektromotore izgleda i na koji način se može izvesti upravljanje i nadzor. U prvom dijelu rada objašnjeno je zašto su asinkroni motori u kombinaciji sa frekventnim pretvaračima danas vrlo popularni i rasprostranjeni u realnim pogonima. Njihova efikasnost, pristupačnost i niska cijena ide tome u prilog. Objašnjena je i ostala oprema koja se koristi u stvarnim postrojenjima i kako se elementi implementiraju zajedno u cjelinu. Naravno naglasak je stavljen na Profinet komunikaciju koja je danas jako zastupljena jer se bazire na standardnoj Ethernet komunikaciji, ali je robusnija i prilagođena za industrijska postrojenja. Rad se sastojao od teorijske analize pojedinih dijelova i same problematike gdje je za cilj postavljeno njeno rješavanje. Frekventni pretvarač parametriran je iz Danfoss programskog paketa, koji pojednostavljuje unos parametara u pretvarač. Industrijsko računalo programirano je iz Siemensovog programskog paketa koje služi za parametriranje računala, pisanje koda i izrade grafičkog sučelja. Elemente objedinjuje Profinet komunikacija koja je brza i jednostavna za podešavanje. Pomoću nje moguće je promijeniti željene parametre frekventnog pretvarača putem grafičkog sučelja industrijskog računala. U ovom radu prikazana su mijenjanja samo nekih od parametara. Grafičko sučelje omogućava prikaz parametara pretvarača u stvarnom vremenu uz točnost i preciznost, a davanje kontrolnih signala pretvaraču je brzo. Profinet komunikacija omogućava upravljanje i nadzor pogona sa udaljenog mjesta što olakšava i pojednostavljuje upravljanje. Analizom sustava prikazan je način upravljanja asinkronim motorom putem frekventnog pretvarača preko snimljenih vrijednosti struje i momenta gdje se može vidjeti kako frekventni pretvarač omogućava motoru pružanje traženog momenta, ali uz kontroliranu struju. Također omogućeno je upravljanje brzinom vrtnje motora jednostavno i precizno. Sustav prikazuje samo neke od parametara koji se mogu mijenjati, ali mogućnosti su puno veće.

## LITERATURA

- [1] I. Mandić, V. Tomljenović, M. Pužar, Sinkroni i asinkroni električni strojevi, Priručnici Tehničkog Veleučilišta u Zagrebu, Zagreb, 2012.
- [2] Elektro Tehnički Portal, Spajanje trofaznog motora dostupno na: URL <https://www.elteh.net/el-instalacije/motori/spajanje-3f-motora.html/> [15.5.2022]
- [3] Slika preuzeta s interneta, dostupno na: URL <https://www.air-compressor-guide.com/articles/motor-startup-system-star-delta> [15.5.2022]
- [4] D. Srpak, I. Stanković, J. Huđek, Modernizacija postrojenja ugradnjom frekventnih pretvarača. Tehnički glasnik [Internet]. 2010;4(1-2):88-93, dostupno na: URL <https://hrcak.srce.hr/85924> [15.5.2022]
- [5] Slika preuzeta s interneta, dostupno na: URL [https://www.researchgate.net/figure/Internal-Circuit-of-Variable-Frequency-Drive\\_fig1\\_333661792](https://www.researchgate.net/figure/Internal-Circuit-of-Variable-Frequency-Drive_fig1_333661792) [18.5.2022]
- [6] I. Bašić, Predložci s predavanja kolegija Električni Strojevi, FERIT, Osijek, 2021.
- [7] 7Vacon 100 Industrial Manual, dostupno na : URL <https://files.danfoss.com/download/Drives/Vacon-100-INDUSTRIAL-Application-Manual-DPD00927J-UK.pdf/> [28.5.2022]
- [8] Practical guide to HMI selection, design and operation, AutomationDirect – HMI Handbook, dostuono na: URL <https://www.automationdirect.com/ebooks/hmi-handbook/> [29.5.2022]
- [9] P. Evans, Programable Logic Controller Basic Explained, dostupno na: URL <https://theengineeringmindset.com/PROGRAMABLE-LOGIC-CONTROLLERS-EXPLAINED/> [Pristup 21.5.2022]
- [10] Siemens Simatic ET200 SP User Manual, dostupno na: URL <https://support.industry.siemens.com/cs/document/84133942/simatic-et-200sp-manual-collection?dti=0&lc=en-US/> [29.5.2022]
- [11] J. Falco, K. Stouffer, K. Kent, Guide to Supervisory Control and Data Acquisition (SCADA) and Industrial Control Systems Security, National Institute of Standards and Technology, Initial Public Draft, Gaithersburg, 2006.



- [12] Practical guide to HMI selection, design and operation, AutomationDirect – HMI Handbook, dostuono na: URL <https://www.automationdirect.com/ebooks/hmi-handbook/>  
[29.5.2022]
- [13] M. Kutija, D. Sumina, Predlošci s predavanja kolegija Elektromotorni pogoni, FER, Zagreb, 2018.
- [14] Profinet University, Profinet communication channels, dostupno na: URL <https://profinetuniversity.com/profinet-basics/profinet-communication-channels/>  
[28.5.2022]
- [15] PI (Profibus & Profinet International), Profinet system description, technology and application, 2014.

## SAŽETAK

Ovaj rad pobliže objašnjava ulogu frekventnih pretvarača kod nadzora i upravljanja asinkronim motorom. Frekventni pretvarač implementiran je u sustav sa industrijskim računalom (*engl. PLC*) zajedno sa njegovim pripadajućim grafičkim sučeljem (*engl. HMI*), a sve se ostvaruje putem Profinet komunikacije. Pobliže je objašnjena struktura i princip rada asinkronih motora i zašto je danas njihovo upravljanje pojednostavljenom pojavom digitalne elektronike i frekventnih pretvarača. Objasnjeni su i frekventni pretvarači zajedno sa industrijskim računalima. Komunikacija koja ovaj sustav zajedno povezuje, a to je Profinet, pobliže je objašnjena i prikazani su načini podešavanja. Izrađen je program koji komunicira sa frekventnim pretvaračem, omogućava korisniku mijenjanje određenih parametara motora, a ujedno i očitavane trenutnih parametara. Za sami kraj odrađeno je testiranje sustava, gdje su se mjerili parametri motora.

**Ključne riječi:** asinkroni motor, frekventni pretvarač, PLC, HMI, Profinet, Siemens, TIA Portal, Danfoss, Vacon

## **ABSTRACT**

This paper explains in more detail the role of AC drives in monitoring and controlling an induction motor. The AC drive is implemented in a system with an industrial computer (PLC) together with its associated graphic interface (HMI), and everything is realized through Profinet communication. The structure and principle of operation of induction motors is explained in more detail and why their management is simplified with the advent of digital electronics and AC drives. AC drives are also explained together with industrial computers. The communication that connects this system together, which is Profinet, is explained in more detail and the adjustment methods are shown. A program was created that communicates with the AC drive, allows the user to change certain motor parameters, and at the same time read the current parameters. Finally, the system was tested, where the motor parameters were measured.

**Keywords:** induction motor, AC drive, PLC, HMI, Profinet, Siemens, TIA Portal, Danfoss, Vacon

## ŽIVOTOPIS

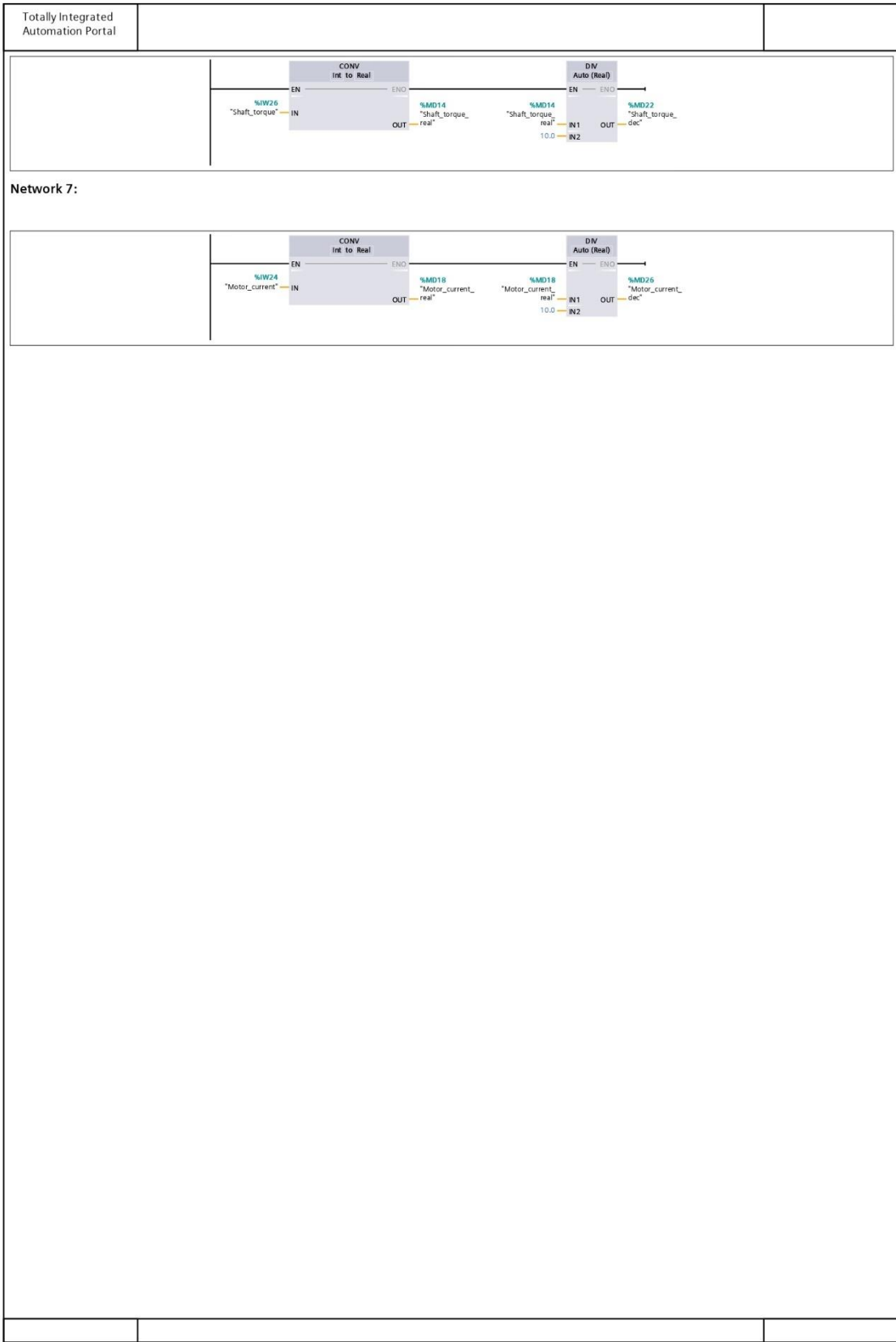
Rođen 2. 5. 1995. u Osijeku, trenutno prebivalište je u Belišću. Pohađao je osnovnu školu Ivan Kukuljević u Belišću koju završava sa odličnim uspjehom. Nakon osnovne škole upisuje smjer elektrotehničar u Srednjoj Školi Valpovo i završava ju sa vrlo dobrim uspjehom. 2014. godine. Završetkom srednje škole i stjecanjem zanimanja elektrotehničar, zapošljava se u tvrtki „TEO – Belišće d.o.o“. Prilikom radnog odnosa usavršava znanja stečena tokom školovanja i steče nova znanja u području elektrotehnike, automatike, industrijske informatike i strojarstva. 2016 godine prekida radni odnos i upisuje stručni studij, smjer Automatika na Fakultetu elektrotehnike, računarstva i informacijskih tehnologija u Osijeku. 2019. godine upisuje Razlikovnu godinu na istom fakultetu i nakon nje upisuje diplomski studij elektrotehnike, smjer industrijska elektroenergetika. Na fakultetu proširuje svoje znanje iz područja digitalne elektronike i signala, automatskog upravljanja pogona i postrojenja te programiranja.

# PRILOG 1. – MAIN PROGRAM

Totally Integrated Automation Portal		
<p>projekt1_fram_scada_final_v3 / PLC_1 [CPU 1512SP-1 PN] / Program blocks</p> <p><b>Main [OB1]</b></p>		
<b>Main Properties</b>		
<b>General</b>		
Name	Main	Number 1
Numbering	Automatic	Type OB
Language	LAD	
<b>Information</b>		
Title	"Main Program Sweep (Cycle)"	Author
Version	0.1	Comment
		Family
		User-defined ID
<b>Main</b>		
Name		Data type
▼ Input		Default value
Initial_Call	Bool	
Remanence	Bool	
Temp		
Constant		
Supervision		Comment
		Initial call of this OB
		=True, if remanent data are available
<b>Network 1:</b>		
<b>Network 2:</b>		

# PRILOG 2. – FCTN FUNKCIJA

Totally Integrated Automation Portal				
<p>projekt1_fram_scada_final_v3 / PLC_1 [CPU 1512SP-1 PN] / Program blocks</p> <p><b>Fctn Int [FC1]</b></p>				
<b>Fctn Int Properties</b>				
<b>General</b>				
Name	Fctn Int			
Number	1			
Type	FC			
Language	LAD			
<b>Information</b>				
Title	Author			
Version	0.1			
	User-defined ID			
	Comment			
	Family			
<b>Fctn Int</b>				
Name	Data type	Default value	Supervision	Comment
Input				
Output				
InOut				
▼ Temp				
NormalizedBrzina	Real			
Constant				
▼ Return				
Fctn Int	Void			
<b>Network 1: Pokretanje</b>				
<b>Network 2: Smjer</b>				
<b>Network 3: Aktivan Alarm</b>				
<b>Network 4: Resetiranje alarma</b>				
<b>Network 5: Pretvorba brzine</b>				
<b>Network 6: int to real</b>				



# PRILOG 3. – FUNKCIJSKI BLOK ZA VACON KOMUNIKACIJU

Totally Integrated Automation Portal									
projekt1_fram_scada_final_v3 / PLC_1 [CPU 1512SP-1 PN] / Program blocks									
VACON_CONTROL_BLOCK_8PD_1 [FB7]									
VACON_CONTROL_BLOCK_8PD_1 Properties									
General									
Name	VACON_CONTROL_BLOCK_8PD_1	Number	7	Type	FB				
Language	SCL								
Numbering	Manual								
Information									
Title		Author		Comment	Family				
Version	0.1	User-defined ID							
VACON_CONTROL_BLOCK_8PD_1									
Name	Data type	Offset	Default value	Accessible from HMI/OPC UA	Writ-able from HMI/OPC UA	Visible in HMI engineering	Setpoint	Supervision	Comment
▼ Input									
RUN	Bool	0.0	false	True	True	True	False		
DIRECTION	Bool	0.1	false	True	True	True	False		
RAMP	Bool	0.2	false	True	True	True	False		
FAULT_RESET	Bool	0.3	false	True	True	True	False		
SPEED_REFERENCE	Int	2.0	0	True	True	True	False		
VENDOR_TELEGRAM1_SUBMODULE	HW_SUBMODULE	4.0	0	True	True	True	False		
▼ Output									
READY	Bool	6.0	false	True	True	True	False		
FAULT_PRESENT	Bool	6.1	false	True	True	True	False		
RUNNING	Bool	6.2	false	True	True	True	False		
CURRENT_DIRECTION	Bool	6.3	false	True	True	True	False		
AT_REFERENCE	Bool	6.4	false	True	True	True	False		
SPEED_ACTUAL	Int	8.0	0	True	True	True	False		
PD1_OUT	Int	10.0	0	True	True	True	False		
PD2_OUT	Int	12.0	0	True	True	True	False		
PD3_OUT	Int	14.0	0	True	True	True	False		
PD4_OUT	Int	16.0	0	True	True	True	False		
PD5_OUT	Int	18.0	0	True	True	True	False		
PD6_OUT	Int	20.0	0	True	True	True	False		
PD7_OUT	Int	22.0	0	True	True	True	False		
PDB_OUT	Int	24.0	0	True	True	True	False		
InOut									
Static									
▼ Temp									
Vacon_FBFixedControlWord	Word	0.0							
▼ Vacon_FBFixedControlWord_BIT									
Vacon_FBFixedControlWord_BIT[0]	Bool	0.0							
Vacon_FBFixedControlWord_BIT[1]	Bool	0.1							
Vacon_FBFixedControlWord_BIT[2]	Bool	0.2							
Vacon_FBFixedControlWord_BIT[3]	Bool	0.3							
Vacon_FBFixedControlWord_BIT[4]	Bool	0.4							
Vacon_FBFixedControlWord_BIT[5]	Bool	0.5							
Vacon_FBFixedControlWord_BIT[6]	Bool	0.6							
Vacon_FBFixedControlWord_BIT[7]	Bool	0.7							
Vacon_FBFixedControlWord_BIT[8]	Bool	1.0							
Vacon_FBFixedControlWord_BIT[9]	Bool	1.1							
Vacon_FBFixedControlWord_BIT[10]	Bool	1.2							
Vacon_FBFixedControlWord_BIT[11]	Bool	1.3							
Vacon_FBFixedControlWord_BIT[12]	Bool	1.4							
Vacon_FBFixedControlWord_BIT[13]	Bool	1.5							
Vacon_FBFixedControlWord_BIT[14]	Bool	1.6							
Vacon_FBFixedControlWord_BIT[15]	Bool	1.7							
Vacon_FBFixedStatusWord	Word	2.0							
▼ Vacon_FBFixedStatusWord_BITS									
Vacon_FBFixedStatusWord_BITS	Bool	2.0							



Totally Integrated Automation Portal									
Name	Data type	Offset	Default value	Accessible from HMI/OPC UA	Writable from HMI/OPC UA	Visible in HMI engineering	Setpoint	Supervision	Comment
Vacon_FBFixedStatusWord_BITS[0]	Bool	2.0							
Vacon_FBFixedStatusWord_BITS[1]	Bool	2.1							
Vacon_FBFixedStatusWord_BITS[2]	Bool	2.2							
Vacon_FBFixedStatusWord_BITS[3]	Bool	2.3							
Vacon_FBFixedStatusWord_BITS[4]	Bool	2.4							
Vacon_FBFixedStatusWord_BITS[5]	Bool	2.5							
Vacon_FBFixedStatusWord_BITS[6]	Bool	2.6							
Vacon_FBFixedStatusWord_BITS[7]	Bool	2.7							
Vacon_FBFixedStatusWord_BITS[8]	Bool	3.0							
Vacon_FBFixedStatusWord_BITS[9]	Bool	3.1							
Vacon_FBFixedStatusWord_BITS[10]	Bool	3.2							
Vacon_FBFixedStatusWord_BITS[11]	Bool	3.3							
Vacon_FBFixedStatusWord_BITS[12]	Bool	3.4							
Vacon_FBFixedStatusWord_BITS[13]	Bool	3.5							
Vacon_FBFixedStatusWord_BITS[14]	Bool	3.6							
Vacon_FBFixedStatusWord_BITS[15]	Bool	3.7							
Vacon_Speed_Actual	Int	4.0							
OutputAddress	UDInt	6.0							
InputAddress	UDInt	10.0							
OutputAddressSize	UInt	14.0							
InputAddressSize	UInt	16.0							
RetVal	Int	18.0							
Constant									

```

0001 #RetVal := RD_ADDR(LADDR := #VENDOR_TELEGRAM1_SUBMODULE, PIADDR => #InputAddress, PICount => #InputAddressSize, PQADDR
=> #OutputAddress, PQCount => #OutputAddressSize);
0002
0003 IF #RetVal<>0 THEN
0004 // Statement section IF
0005 RETURN;
0006 ;
0007 END_IF;
0008
0009
0010 #Vacon_FBFixedStatusWord := PEEK_WORD(area:=16#81, byteOffset:=(#InputAddress), dbNumber:=16#0);
0011 #Vacon_FBFixedControlWord := 16#03;
0012 #Vacon_Speed_Actual := WORD_TO_INT(PEEK_WORD(area := 16#81, byteOffset := (#InputAddress + 2), dbNumber := 16#0));
0013 #PD1_OUT := WORD_TO_INT(PEEK_WORD(area := 16#81, byteOffset := (#InputAddress + 4), dbNumber := 16#0));
0014 #PD2_OUT := WORD_TO_INT(PEEK_WORD(area := 16#81, byteOffset := (#InputAddress + 6), dbNumber := 16#0));
0015 #PD3_OUT := WORD_TO_INT(PEEK_WORD(area := 16#81, byteOffset := (#InputAddress + 8), dbNumber := 16#0));
0016 #PD4_OUT := WORD_TO_INT(PEEK_WORD(area := 16#81, byteOffset := (#InputAddress + 10), dbNumber := 16#0));
0017 #PD5_OUT := WORD_TO_INT(PEEK_WORD(area := 16#81, byteOffset := (#InputAddress + 12), dbNumber := 16#0));
0018 #PD6_OUT := WORD_TO_INT(PEEK_WORD(area := 16#81, byteOffset := (#InputAddress + 14), dbNumber := 16#0));
0019 #PD7_OUT := WORD_TO_INT(PEEK_WORD(area := 16#81, byteOffset := (#InputAddress + 16), dbNumber := 16#0));
0020 #PD8_OUT := WORD_TO_INT(PEEK_WORD(area := 16#81, byteOffset := (#InputAddress + 18), dbNumber := 16#0));
0021
0022 IF #RUN = 1 THEN
0023 // Statement section IF
0024 #Vacon_FBFixedControlWord_BIT[8] := 1;
0025 ELSE
0026 #Vacon_FBFixedControlWord_BIT[8] := 0;
0027 ;
0028 END_IF;
0029
0030 IF #DIRECTION = 1 THEN
0031 #Vacon_FBFixedControlWord_BIT[9] := 1;
0032 ;
0033 ELSE
0034 #Vacon_FBFixedControlWord_BIT[9] := 0;
0035 ;
0036 END_IF;
0037
0038 IF #FAULT_RESET = 1 THEN
0039 // Statement section IF
0040 #Vacon_FBFixedControlWord_BIT[10] := 1;
0041 ;

```

Totally Integrated Automation Portal			
<pre> 0042 ELSE 0043   #Vacon_FBFixedControlWord_BIT[10] := 0; 0044 END_IF; 0045 0046 0047 IF #Vacon_FBFixedStatusWord_BITS[8] = 1 THEN 0048   // Statement section IF 0049   #READY := 1; 0050 ELSE 0051   #READY := 0; 0052   ; 0053 END_IF; 0054 0055 IF #Vacon_FBFixedStatusWord_BITS[9] = 1 THEN 0056   // Statement section IF 0057   #RUNNING := 1; 0058 ELSE 0059   #RUNNING := 0; 0060   ; 0061 END_IF; 0062 0063 IF #Vacon_FBFixedStatusWord_BITS[11] = 1 THEN 0064   // Statement section IF 0065   #FAULT_PRESENT := 1; 0066 ELSE 0067   #FAULT_PRESENT := 0; 0068   ; 0069 END_IF; 0070 0071 IF #Vacon_FBFixedStatusWord_BITS[10] = 1 THEN 0072   // Statement section IF 0073   #CURRENT_DIRECTION := 1; 0074 ELSE 0075   #CURRENT_DIRECTION := 0; 0076   ; 0077 END_IF; 0078 0079 IF #Vacon_FBFixedStatusWord_BITS[13] = TRUE THEN 0080   // Statement section IF 0081   #AT_REFERENCE := TRUE; 0082   ; 0083 ELSE 0084   #AT_REFERENCE := FALSE; 0085   ; 0086 END_IF; 0087 0088 0089 POKE(area := 16#82, 0090   byteOffset := #OutputAddress, 0091   dbNumber := 16#0, 0092   value := #Vacon_FBFixedControlWord); 0093 0094 POKE(area := 16#82, 0095   byteOffset := (#OutputAddress + 2), 0096   dbNumber := 16#0, 0097   value := INT_TO_WORD(#SPEED_REFERENCE)); 0098 0099 #SPEED_ACTUAL := #Vacon_Speed_Actual; </pre>			
Symbol	Address	Type	Comment
#AT_REFERENCE		Bool	
#CURRENT_DIRECTION		Bool	
#DIRECTION		Bool	
#FAULT_PRESENT		Bool	
#FAULT_RESET		Bool	
#InputAddress		UDInt	
#InputAddressSize		UInt	
#OutputAddress		UDInt	
#OutputAddressSize		UInt	
#PD1_OUT		Int	
#PD2_OUT		Int	
#PD3_OUT		Int	
#PD4_OUT		Int	
#PD5_OUT		Int	
#PD6_OUT		Int	
#PD7_OUT		Int	
#PD8_OUT		Int	
#READY		Bool	
#RetVal		Int	
#RUN		Bool	
#RUNNING		Bool	
#SPEED_ACTUAL		Int	
#SPEED_REFERENCE		Int	
#Vacon_FBFixedControlWord		Word	
#Vacon_FBFixedControlWord_BIT[8]		Bool	
#Vacon_FBFixedControlWord_BIT[9]		Bool	
#Vacon_FBFixedControlWord_BIT[10]		Bool	

Totally Integrated Automation Portal			
Symbol	Address	Type	Comment
#Vacon_FBFixedStatusWord		Word	
#Vacon_FBFixedStatusWord_BITS[8]		Bool	
#Vacon_FBFixedStatusWord_BITS[9]		Bool	
#Vacon_FBFixedStatusWord_BITS[10]		Bool	
#Vacon_FBFixedStatusWord_BITS[11]		Bool	
#Vacon_FBFixedStatusWord_BITS[13]		Bool	
#Vacon_Speed_Actual		Int	
#VENDOR_TELEGRAM1_SUBMODULE		HW_SUBMODULE	

# PRILOG 4. – PLC TAGS

Totally Integrated Automation Portal									
projekt1_fram_scada_final_v3 / PLC_1 [CPU 1512SP-1 PN]									
PLC tags									
Name	Data type	Address	Retain	Accessi-ble from HMI/OPC UA	Writable from HMI/OPC UA	Visible in HMI engi-neering	Supervision	Comment	
GO	Bool	%M1.3	False	True	True	True			
Start	Bool	%M1.0	False	True	True	True			
PowerOn	Bool	%M1.1	False	True	True	True			
Stop	Bool	%M1.2	False	True	True	True			
Reverse	Bool	%M0.0	False	True	True	True			
Reverse_Switch	Bool	%M0.1	False	True	True	True			
ResetAlarm	Bool	%M0.4	False	True	True	True			
RstAlarm	Bool	%M0.5	False	True	True	True			
ReferentSpeed	Int	%MW2	False	True	True	True			
Ready	Bool	%M0.6	False	True	True	True			
EROR	Bool	%M0.7	False	True	True	True			
RUNNING	Bool	%M1.4	False	True	True	True			
Direction	Bool	%M1.5	False	True	True	True			
AT_Reference	Bool	%M1.6	False	True	True	True			
ActualSpeed	Int	%MW4	False	True	True	True			
Potenciometar	Int	%MW6	False	True	True	True			
NormalizedSpeed	Real	%MD8	False	True	True	True			
Start_Fctn	Bool	%M1.7	False	True	True	True			
Output_freq	Int	%W20	False	True	True	True			
Motor_speed	Int	%W22	False	True	True	True			
Motor_current	Int	%W24	False	True	True	True			
Shaft_torque	Int	%W26	False	True	True	True			
Shaft_power	Int	%W28	False	True	True	True			
Motor_voltage	Int	%W30	False	True	True	True			
DC_link_voltage	Int	%W32	False	True	True	True			
Last_active_err	Int	%W34	False	True	True	True			
FBFixedSW	Int	%W0	False	True	True	True			
FBSpeedActual	Int	%W2	False	True	True	True			
FBFixedCW	Int	%QW0	False	True	True	True			
FBSpeedReference	Int	%QW2	False	True	True	True			
Output_freq_vendor	Int	%W4	False	True	True	True			
Motor_speed_vendor	Int	%W6	False	True	True	True			
Motor_current_vendor	Int	%W8	False	True	True	True			
Shaft_torque_vendor	Int	%W10	False	True	True	True			
Shaft_power_vendor	Int	%W12	False	True	True	True			
Motor_voltage_vendor	Int	%W14	False	True	True	True			
DC_link_voltage_vendor	Int	%W16	False	True	True	True			
Last_active_error_vendor	Int	%W18	False	True	True	True			
Tag_1	Word	%W3	False	True	True	True			
Alarm	Bool	%M12.0	False	True	True	True			
tmp	Bool	%M12.1	False	True	True	True			
Shaft_torque_real	Real	%MD14	False	True	True	True			
Motor_current_real	Real	%MD18	False	True	True	True			
Shaft_torque_dec	Real	%MD22	False	True	True	True			
Motor_current_dec	Real	%MD26	False	True	True	True			
tmp1	Bool	%M0.2	False	True	True	True			
ramp	Bool	%M0.3	False	True	True	True			

# PRILOG 5. – HMI TAGS

Totally Integrated Automation Portal					
projekt1_fram_scada_final_v3 / PC-System_1 [SIMATIC PC station] / HMI_RT_1 [WinCC RT Advanced]					
<b>HMI tags</b>					
<b>Start</b>					
Name	Start	Display name		Address	
Connection	HMI_Connection_1	Data type	Bool	Length	1
<b>Start</b>					
<b>Stop</b>					
Name	Stop	Display name		Address	
Connection	HMI_Connection_1	Data type	Bool	Length	1
<b>Stop</b>					
<b>Ready</b>					
Name	Ready	Display name		Address	
Connection	HMI_Connection_1	Data type	Bool	Length	1
<b>Ready</b>					
<b>Kreni</b>					
Name	Kreni	Display name		Address	
Connection	HMI_Connection_1	Data type	Bool	Length	1
<b>GO</b>					
<b>start1</b>					
Name	start1	Display name		Address	
Connection	HMI_Connection_1	Data type	Bool	Length	1
<b>Start_Fctn</b>					
<b>StvarnaBrzina</b>					
Name	StvarnaBrzina	Display name		Address	
Connection	HMI_Connection_1	Data type	Int	Length	2
<b>ActualSpeed</b>					
<b>Reverse</b>					
Name	Reverse	Display name		Address	
Connection	HMI_Connection_1	Data type	Bool	Length	1
<b>Reverse</b>					
<b>Potenciometar</b>					
Name	Potenciometar	Display name		Address	
Connection	HMI_Connection_1	Data type	Int	Length	2
<b>Potenciometar</b>					
<b>RUNNING</b>					
Name	RUNNING	Display name		Address	
Connection	HMI_Connection_1	Data type	Bool	Length	1
<b>RUNNING</b>					
<b>Direction</b>					
Name	Direction	Display name		Address	
Connection	HMI_Connection_1	Data type	Bool	Length	1
<b>Direction</b>					
<b>EROR</b>					
Name	EROR	Display name		Address	
Connection	HMI_Connection_1	Data type	Bool	Length	1
<b>EROR</b>					
<b>Reverse_Switch</b>					
Name	Reverse_Switch	Display name		Address	
Connection	HMI_Connection_1	Data type	Bool	Length	1
<b>Reverse_Switch</b>					
<b>AT_Reference</b>					
Name	AT_Reference	Display name		Address	
Connection	HMI_Connection_1	Data type	Bool	Length	1

Totally Integrated Automation Portal			
<b>AT_Reference</b>			
<b>Motor_speed</b>			
Name	Motor_speed	Display name	Address
Connection	HMI_Connection_1	Data type	Int
		Length	2
<b>Motor_speed</b>			
<b>Output_freq</b>			
Name	Output_freq	Display name	Address
Connection	HMI_Connection_1	Data type	Int
		Length	2
<b>Output_freq</b>			
<b>Motor_current</b>			
Name	Motor_current	Display name	Address
Connection	HMI_Connection_1	Data type	Int
		Length	2
<b>Motor_current</b>			
<b>Motor_voltage</b>			
Name	Motor_voltage	Display name	Address
Connection	HMI_Connection_1	Data type	Int
		Length	2
<b>Motor_voltage</b>			
<b>DC_link_voltage</b>			
Name	DC_link_voltage	Display name	Address
Connection	HMI_Connection_1	Data type	Int
		Length	2
<b>DC_link_voltage</b>			
<b>Shaft_torque</b>			
Name	Shaft_torque	Display name	Address
Connection	HMI_Connection_1	Data type	Int
		Length	2
<b>Shaft_torque</b>			
<b>Shaft_power</b>			
Name	Shaft_power	Display name	Address
Connection	HMI_Connection_1	Data type	Int
		Length	2
<b>Shaft_power</b>			
<b>Last_active_err</b>			
Name	Last_active_err	Display name	Address
Connection	HMI_Connection_1	Data type	Int
		Length	2
<b>Last_active_err</b>			
<b>RstAlarm</b>			
Name	RstAlarm	Display name	Address
Connection	HMI_Connection_1	Data type	Bool
		Length	1
<b>RstAlarm</b>			
<b>Shaft_torque_real</b>			
Name	Shaft_torque_real	Display name	Address
Connection	HMI_Connection_1	Data type	Real
		Length	4
<b>Shaft_torque_real</b>			
<b>Shaft_torque_dec</b>			
Name	Shaft_torque_dec	Display name	Address
Connection	HMI_Connection_1	Data type	Real
		Length	4
<b>Shaft_torque_dec</b>			
<b>Motor_current_dec</b>			
Name	Motor_current_dec	Display name	Address
Connection	HMI_Connection_1	Data type	Real
		Length	4
<b>Motor_current_dec</b>			