

Automatizacija uređaja za zavarivanje

Turković, Karlo

Undergraduate thesis / Završni rad

2020

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **Bjelovar University of Applied Sciences / Veleučilište u Bjelovaru**

Permanent link / Trajna poveznica: <https://um.nsk.hr/um:nbn:hr:144:307831>

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

Download date / Datum preuzimanja: **2025-02-20**



Repository / Repozitorij:

[Digital Repository of Bjelovar University of Applied Sciences](#)



VELEUČILIŠTE U BJELOVARU
PREDDIPLOMSKI STRUČNI STUDIJ MEHATRONIKA

AUTOMATIZACIJA UREĐAJA ZA ZAVARIVANJE

Završni rad br. 17/MEH/2019

Karlo Turković

Bjelovar, srpanj 2020.



Veleučilište u Bjelovaru
Trg E. Kvaternika 4, Bjelovar

1. DEFINIRANJE TEME ZAVRŠNOG RADA I POVJERENSTVA

Kandidat: **Turković Karlo** Datum: 30.09.2019.

Matični broj: 001903

JMBAG: 0035198994

Kolegij: **RAČUNALNO VOĐENJE I UPRAVLJANJE PROCESIMA**

Naslov rada (tema): **Automatizacija uređaja za zavarivanje**

Područje: **Tehničke znanosti**

Polje: **Elektrotehnika**

Grana: **Automatizacija i robotika**

Mentor: **Zoran Vrhovski, mag.ing.el.techn.inf.**

zvanje: **viši predavač**

Članovi Povjerenstva za ocjenjivanje i obranu završnog rada:

1. **Tomislav Pavlic, mag.ing.mech., predsjednik**
2. **Zoran Vrhovski, mag.ing.el.techn.inf., mentor**
3. **Danijel Radočaj, mag.Inž.meh., član**

2. ZADATAK ZAVRŠNOG RADA BROJ: 17/MEH/2019

U radu je potrebno:

1. Opisati postojeća rješenja na tržištu uređaja za zavarivanje
2. Opisati senzore i aktuatore uređaja za zavarivanje
3. Odabrati i opisati PLC uređaj i ostalu opremu potrebnu za uautomatizaciju uređaja za zavarivanje
4. Izraditi i opisati programsko rješenje PLC i HMI uređaja za automatizaciju uređaja za zavarivanje
5. Opisati shemu spajajna elemenata sustava automatizacije uređaja za zavarivanje

Zadatak uručen: 30.09.2019.

Mentor: **Zoran Vrhovski, mag.ing.el.techn.inf.**



Zahvala

Zahvaljujem svome mentoru dr.sc. Zoranu Vrhovskom na savjetima, pomoći pri izradi završnog rada, te utrošenom vremenu i trudu. Također, zahvaljujem firmi Nopus d.o.o., gospodinu Niki Nikiću i gospodinu Hassanu Lakhnatiju na ustupanju opreme za izradu završnog rada, ukazanom povjerenju i svim savjetima koji su mi pomogli u izradi završnoga rada.

SADRŽAJ

1. UVODI

2. OPIS STROJA N1 I NJEGOVIH ZAHTJEVA2

3. SLIČNA RJEŠENJA NA TRŽIŠTU4

4. RUKOVANJE STROJEM N15

4.1 Podizanje sustava5

4.2 Početna stranica6

4.3 Procedura za reset7

4.4 Postavke sustava9

4.4.1 Status senzora11

4.4.2 Status daljinskog upravljača11

4.4.3 Postavke stroja12

4.4.4 Status stroja13

4.4.5 V90 status14

4.4.6 Gumb za održavanje15

4.5 Unos programa15

4.6 Zavarivački program16

4.6.1 Parametri programa17

4.6.2 Parametri pozicija18

4.6.3 Vremenski parametri19

4.6.4 Parametri sektora21

4.6.5 Parametri zavarivanja23

4.7 Ručno upravljanje26

4.8 Glavna lista programa27

4.9 Postavljanje radnog komada u stroj28

5. OPIS OPREME I KOMPONENATA STROJA29

5.1 Električni dio29

5.1.1 PLC30

5.1.2 SINAMICS V90 PN servo upravljački sustavi35

5.1.3 HMI uređaj36

5.1.4 EWM Tetrix BUSINT X11 modul36

5.1.5 Kabeli36

5.2 STL37

5.2.1 Senzori37

5.2.2 Aktuatori37

5.3 SCAR38

5.3.1 Senzori38

5.3.2 Aktuatori38

5.4 Daljinski upravljač38

6. KONFIGURACIJA V90 SERVO UPRAVLJAČKIH SUSTAVA40

7. PROGRAMSKO RJEŠENJE PLC UREĐAJA47

7.1 Organizacijski blokovi50

7.1.1 Glavni organizacijski blok50

7.1.2 Ciklički prekid51

7.2 Funkcije52

- 7.2.1 Automatski ciklus zavarivanja52
- 7.2.2 Izlazi stroja59
- 7.2.3 Referenciranje60
- 7.2.4 Referenciranje rotacijske osi63
- 7.2.5 Rotacijska os65
- 7.2.6 Linearne osi73
- 7.2.7 BUSINT75
- 7.2.8 Status stroja78
- 7.2.9 Brojači vremena81
- 7.2.10 Alarmi82
- 7.2.11 Spremanje pozicije83
- 7.2.12 Rampe struje zavarivanja84
- 7.3 *Funkcijski blokovi*86
 - 7.3.1 BUSINT TETRIX X11 komunikacija86
 - 7.3.2 AVC kontrola87
 - 7.3.3 Kalkulacija parametara za zavarivanje89
 - 7.3.4 Kalkulacija zadnjeg aktivnog sektora91
- 7.4 *PLC tagovi*92
- 7.5 *PLC podatkovne strukture*93

8. KORISNIČKO SUČELJE95

- 8.1 *Objekti korisničkog sučelja*95
 - 8.1.1 Postavke96
 - 8.1.2 Animacije97
 - 8.1.3 Događaji99
 - 8.1.4 Tekstovi99
- 8.2 *Ekrani*100
- 8.3 *Upravljanje ekranima*102
- 8.4 *HMI tagovi*103
- 8.5 *Konekcija*106
- 8.6 *HMI alarmi*107
- 8.7 *Recepti*108
- 8.8 *Tekstualne i grafičke liste*110
- 8.9 *Korisnička administracija*111

9. PREZENTACIJA RADA STROJA N113

10. ZAKLJUČAK115

11. LITERATURA117

12. OZNAKE I KRATICE119

13. SAŽETAK120

14. ABSTRACT121

1. UVOD

Tema ovog završnog rada je automatizacija zavarivačkog procesa. Rad će pobliže opisati programsko rješenje za automatizaciju Nopus N1 automatiziranog stroja za orbitalno zavarivanje (u daljnjem tekstu stroj N1). Opisat će se električne komponente koje su postavljene na stroj N1, senzori i aktuatori koji su korišteni za automatizaciju, te će se pružiti uvid u softversko rješenje upravljačkog i korisničkog dijela programskog koda.

Konačni rezultat je potpuno automatizirano softversko rješenje pozicioniranja gorionika za zavarivanje u radnu poziciju, rotacija radnog komada koji se zavaruje i prilagođavanje parametara za zavarivanje ovisno o sektoru zavarivanja u kojem se radni komad nalazi.

U drugom poglavlju opisan je stroj N1 i njegove funkcije. Slična rješenja na tržištu opisana su u trećem poglavlju. Četvrto poglavlje opisuje rukovanje strojem N1. U petom poglavlju nalazi se opis opreme i komponenata stroja N1. Šesto poglavlje opisuje konfiguraciju V90 servo upravljačkih sustava korištenih na stroju N1. Programsko rješenje PLC uređaja opisano je u poglavlju broj sedam, a korisničko sučelje u poglavlju broj osam.

2. OPIS STROJA N1 I NJEGOVIH ZAHTJEVA

Stroj N1 služi za orbitalno zavarivanje, odnosno zavarivanje okruglih cijevi međusobno ili zavarivanje priрубnica na okrugle cijevi i prikazan je na slici 2.1.



Slika 2.1: Stroj za automatizirano zavarivanje

Najčešća namjena ovakvih strojeva je zavarivanje priрубnica na cijevi. Za zavarivanje cijevi na cijev u praksi se koriste drugačija rješenja poput traktorskih sistema kod kojih se okrugla vodilica postavi direktno na cijev i sustav za zavarivanje putuje po njoj. Razlog je taj što u cjevarskoj industriji najčešće treba zavarivati cijevi velikog promjera i većih duljina pa bi i veličina i cijena stroja bile znatno veće.

Specifičnost stroja N1 je u tome što je namijenjen za zavarivanje komada različite dužine (maksimalne dužine 600 mm) i različitih promjera (maksimalnog promjera 600 mm). Zbog toga stroj ima po jednu glavnu horizontalnu i vertikalnu os koje stroj dovode u poziciju, i po dvije pomoćne horizontalne i dvije pomoćne vertikalne osi koje se prilagođavaju radnom komadu, tj. pozicijama koje se zavaruju.

Posebnost ovakvih strojeva je preciznost i ponovljivost procesa. Automatizirane naprave poput stroja N1 ne koriste se kod zavarivačkih procesa gdje se ne zahtijeva velika preciznost zavarenog komada jer bi izrada ovakve naprave za jednostavnije primjene bila preskupa. Proizvod za koji je stroj N1 konstruiran i programiran su mjerni uređaji za

kemijsku i naftnu industriju. Takvi uređaji zahtijevaju vrlo preciznu metodu zavarivanja s ponovljivošću od 99 % jer svaka greška u zavaru može prouzročiti greške u mjernom procesu. Greška od 1 % mora se uzeti u obzir, ali je zanemariva za proizvođača automatiziranog rješenja jer nije prouzrokovana automatizacijom. Problem se javlja kod postavljanja pozicije ulaska dodatnog materijala u radni komad. Ulazak dodatnog materijala u rastaljeni bazen osnovnog materijala određuje se manualno pomoću 4-osnog držača postavljenog na gorionik. Budući da operater manualno postavlja poziciju ulaska žice, stvara se mogućnost za pogreškom koja direktno utječe na svojstvo ponovljivosti. Kako je ta greška uzrokovana od strane operatera, proizvođač automatiziranog rješenja ne može nikako utjecati na njezino smanjenje.

Korijen zavara na unutarnjoj strani cijevi ne smije biti veći od 0,5 mm. Veći korijen zavara mogao bi unijeti nepotrebne turbulencije u strujanje fluida unutar mjernog uređaja i time znatno poremetiti mjerni proces i njegov krajnji rezultat. Zavar ne smije sadržavati poroznost koja znatno utječe na čvrstoću zavarenog spoja. Fluid kroz mjerni uređaj teče pod velikim tlakom i zavareni spoj mora imati zagarantiranu čvrstoću kako ne bi došlo do pukotina u zavaru. Lice zavara mora se prelitati preko vanjskog brida kanala kako bi se rastalio u jednu masu s cijevi i pribudnicom. Također lice zavara mora biti konkavnog oblika i estetski prihvatljivo jer kod automatiziranih zavarivačkih procesa nema zahtjeva za naknadnom obradom odvajanjem čestica.

Kada je radni komad zavaren šalje se na daljnju montažu mjerne elektronike i direktno u prodaju. Ovisno o specifikaciji proizvoda, zavareni komad može ići na plastifikaciju prije montaže mjerne elektronike.

3. SLIČNA RJEŠENJA NA TRŽIŠTU

Postojeća rješenja na tržištu slična stroju N1 nisu vrlo česta. Strojevi za ovakvu namjenu najčešće su specijalizirane naprave izrađene po specifikaciji kupca. Kod automatiziranih naprava za orbitalno zavarivanje univerzalnim dijelovima mogu se nazvati samo čelična konstrukcija i okretni stol s čeljusnom steznom glavom. Sve ostale komponente na stroju određene su specifikacijom kupca. Upravo zbog te specifikacije kupca i izrade posebno naručene naprave, čest je slučaj da tvrtke koje izrađuju napravu moraju potpisati ugovor o tajnosti. U dostupnoj literaturi nema puno informacija o sličnim strojevima.

Najsličniji standardni proizvod koji se može naći na tržištu je stroj za zavarivanje kružnog šava (eng. *Circumferential Seam Welding Machine*) [1, 2]. Standardna izvedba ovisi od proizvođača do proizvođača, no ono što je standardizirano na svakom stroju za zavarivanje kružnog šava je automatizirani okretni stol i barem jedna upravljiva os za gorionik za zavarivanje, bilo manualna ili automatizirana. Kao što je već navedeno u prethodnom odlomku, sve ostale funkcije naprave određene su specifikacijom kupca. Najčešće dodatne funkcije su dodavanje još jedne osi za drugi gorionik, prihvat slobodnog kraja radnog komada dodatnom okretnom glavom ili pneumatskim, manualnim ili motoriziranim konjićem i dodavanje pomoćnih osi ili zglobnih prihвата za dodatnu manipulaciju položaja gorionika.

Neki od poznatijih proizvođača standardnih i specijalnih automatiziranih naprava za zavarivanje su američki *Milleri Lincoln Electric*, njemački *MüllerOpladeni Förster*, francuski *Polysoude* i *SODEC*, švedski *ESAB*, te engleski *BODE*.

4. RUKOVANJE STROJEM N1

Za rukovanje strojem potreban je minimalno jedan operater koji će u stroj stavljati radne komade i pokretati automatski ciklus zavarivanja pomoću tipke „*START*“ na daljinskom upravljaču. No prije nego što je stroj spreman za rad, potrebno ga je pustiti u pogon i iskonfigurirati.

4.1 Podizanje sustava

Sustav se uključuje okretanjem glavne sklopke za prednju ugradnju u radni položaj. Potrebno je 60 sekundi kako bi se sustav u potpunosti podigao i bio spreman za rad. Kada je sustav spreman za rad, tipka „*RESET*“ (eng. *Reset Cycle*) početak će blicati crvenom bojom, a tipka „*POTVRDI*“ (eng. *Acknowledgment*) zelenom bojom. Na naslovnoj stranici korisničkog sučelja pisat će „*Status stroja: Potrebno resetirati stroj*“ (eng. *Machine status: Reset Request*).

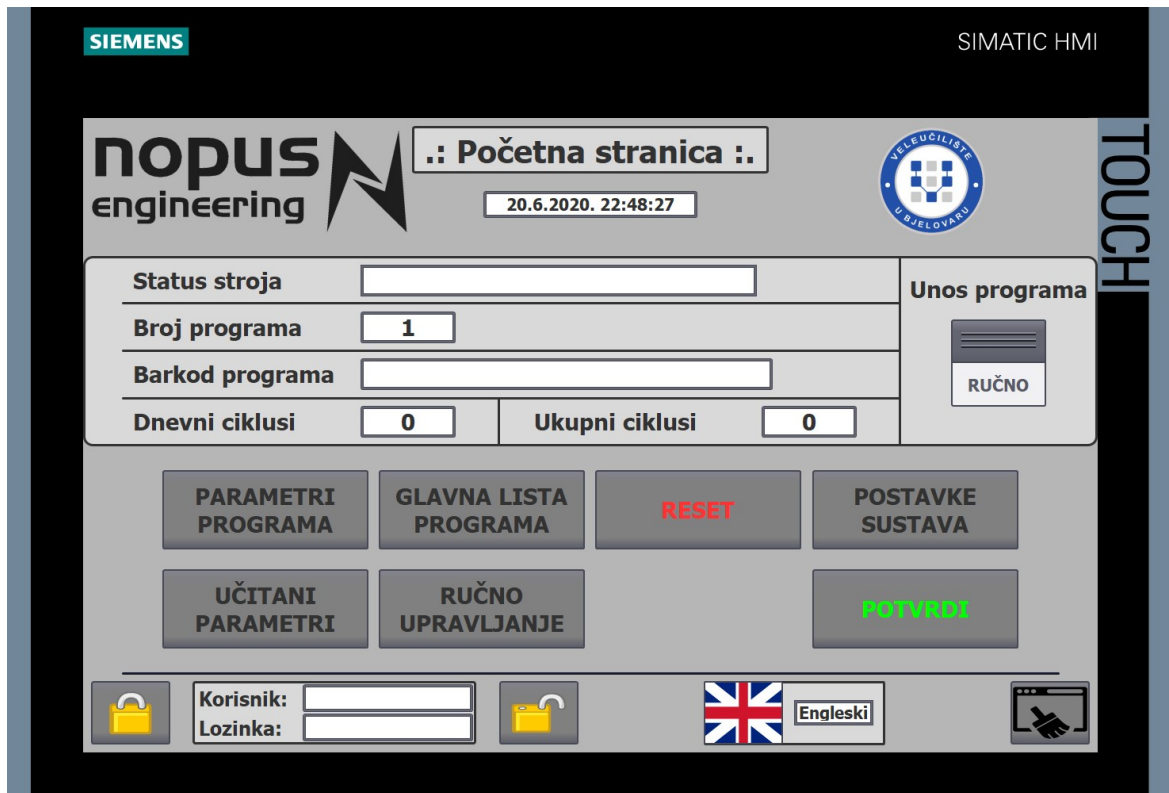
Prva stvar koju operater mora napraviti jest pustiti napajanje na V90 servo upravljačke sustave i omogućiti rad s njima. Procedura aktivacije je sljedeća:

1. pritisnuti plavu tipku „*Emergency Reset*“ koja se nalazi na gornjoj strani upravljačkog ormara desno od HMI uređaja
2. pritisnuti tipku „*POTVRDI*“ na početnoj stranici HMI uređaja
3. pritisnuti crveno gljivasto tipkalo
4. otpustiti crveno gljivasto tipkalo
5. ponovno pritisnuti tipku „*Emergency Reset*“
6. ponovno pritisnuti tipku „*POTVRDI*“ na početnoj stranici HMI uređaja.

Kada je procedura pravilno izvršena, napajanje će se propustiti na V90 servo upravljačke sustave i omogućit će se rad s njima u programskom kodu. Ovu proceduru potrebno je napraviti nakon svakog uključivanja stroja N1 ili nakon što se stroj zaustavi pritiskanjem crvenog gljivastog tipkala. Kada je procedura aktivacije uspješno izvršena, strojem N1 moguće je upravljati manualno pomoću daljinskog upravljača. Automatske cikluse nije moguće pokrenuti prije nego što stroj N1 odradi proceduru za reset, tj. pronađe referentne točke.

4.2 Početna stranica

„Početna stranica“ je prva stranica koja se prikaže na HMI uređaju kada se sustav podigne, a prikazana je na slici 4.1.



Slika 4.1: Početna stranica korisničkog sučelja

Na gornjem djelu stranice nalazi se 5 polja informativnog sadržaja i jedan prekidač za unos podataka:

1. „*Status stroja*“ prikazuje korisniku u kojem načinu rada se stroj nalazi.
2. „*Broj programa*“ prikazuje broj trenutno unesenog programa u sustav.
3. „*Barkod programa*“ prikazuje barkod trenutno unesenog programa u sustav.
4. „*Dnevni ciklusi*“ prikazuje broj završenih ciklusa od trenutka kada je stroj upaljen, odnosno od kada se brojač zadnji puta postavljen na vrijednost 0.
5. „*Ukupni ciklusi*“ prikazuje broj kompletiranih ciklusa od trenutka kada je stroj postavljen u pogon.
6. „*Unos programa*“ je prekidač za odabir načina unos programa u sustav. Program se može upisati ručno upisom broja programa u za to predviđeno polje ili očitavanjem barkoda radnog komada pomoću barkod skenera.

Središnji dio početne stranice sastoji se od 8 tipki :

1. „*PARAMETRI PROGRAMA*“ vodi korisnika na stranicu gdje se mogu podesiti svi parametri trenutno učitanoz zavarivačkog programa i upisati njegove osnovne informacije.
2. „*GLAVNA LISTA PROGRAMA*“ je stranica koja sadržava tablični prikaz osnovnih informacija o svih 300 programa spremljenih u memoriju stroja.
3. „*RESET*“ tipka služi za pokretanje procedure za reset.
4. „*POSTAVKE SUSTAVA*“ vodi korisnika na stranicu za podešavanje postavki sustava i provjeru osnovnih informacija o stroju.
5. „*UČITANI PARAMETRI*“ vodi korisnika na stranicu za čitanje i provjeru parametara trenutno unesenog programa u sustav.
6. „*RUČNO UPRAVLJANJE*“ služi za ručno upravljanje osima bez pomoći komandne palice na daljinskomu upravljaču.
7. „*PARKING*“ služi za pokretanje „*Parking procedure*“, odnosno pozicioniranje stroja u parking poziciju trenutno upisanog programa u sustav.
8. „*POTVRDI*“ tipka služi za potvrđivanje grešaka i upozorenja sustava koji se prikazu na HMI uređaju.

Na donjem dijelu stranice nalaze se tri grupe parametara:

1. prozor za prijavu korisnika u sustav s tipkama za odjavu i prijavu
2. tipka za promjenu jezika sustava i indikator trenutnog jezika
3. tipka „*Očisti ekran*“ (eng. *Screen Clean*) koja na 10 sekundi na korisničkom sučelju isključuje funkciju osjetljivosti na dodir kako bi ga korisnik mogao očistiti.

Ovisno s kojim korisničkim imenom se korisnik prijavi u sustav, neke funkcije stroja će mu biti omogućene, odnosno onemogućene. Naime, svako korisničko ime ima drugačiji stupanj autorizacije jer iste funkcije stroja ne smiju biti dozvoljene svim korisnicima stroja. Detaljnije o tome biti će prikazano u poglavlju 7.10.

4.3 Procedura za reset

Ako operater želi omogućiti rad s automatskim ciklusima zavarivanja na stroju N1, prvo mora napraviti proceduru za reset. Enkoderi na svim motorima stroja su

inkrementalni pa se pozicije svih motora izbrišu iz memorije svaki put kada na stroju nestane napajanje. Kada se stroj upali, niti jedan V90 servo upravljački sustav ne zna točnu poziciju svojeg motora u prostoru sve dok se ne referencira na referentnim krajnjim prekidačima. Nakon referenciranja svaki V90 servo upravljački sustav pamti referentne točke dok god ima napajanje. Referenciranje stroja obavljase pomoću procedure za reset koja započinje pritiskom tipke „RESET“ na početnoj stranici korisničkog sučelja. Kada je procedura za reset završena i sve točke su referencirane (pronađene su im nultočke na referentnim krajnjim prekidačima), na korisničkom sučelju pisat će: *Status stroja: Automatski način*(eng. *Machine Status: Automatic Mode*) ili *Status stroja: Manualni način*(eng. *Machine Status: Manual Mode*), ovisno u kojem položaju se nalazi prekidač za odabir načina rada na daljinskom upravljaču.

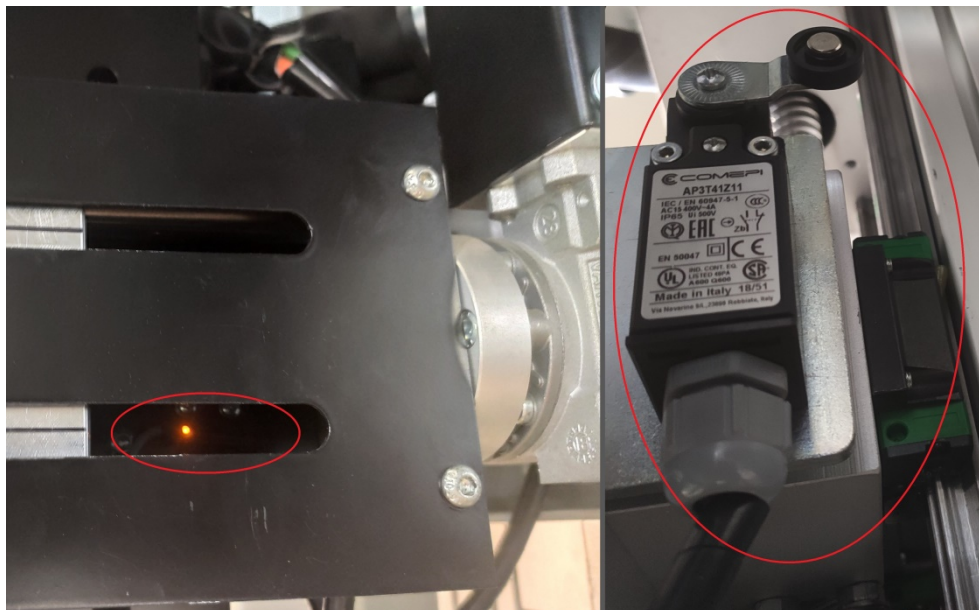
Procedura za reset odvija se tako da se prvo referenciraju glavna vertikalna os i dvije pomoćne vertikalne osi (u daljnjem tekstu AVC osi). Nakon što su vertikalne osi referencirane, započinje referenciranje glavne horizontalne osi i dvije pomoćne horizontalne osi (u daljnjem tekstu oscilatori). Ovakvim načinom referenciranja, stroj se miče iz radnog prostora prije referenciranja horizontalnih osi i time eliminira mogućnost kolizije s nepokretnim dijelovima stroja.

Referenciranje svih linearnih osi odvija se u 7 koraka:

1. os se kreće prema referentnom krajnjem prekidaču
2. os čeka 1.5 s na referentnom krajnjem prekidaču
3. os putuje 2 s u smjeru suprotnom od referentnog krajnjeg prekidača
4. os čeka 1.5 s na poziciji izvan utjecaja referentnog krajnjeg prekidača
5. os se predefiniranom brzinom vraća na referentni krajnji prekidač
6. os staje kada aktivira referentni krajnji prekidač
7. os se referencira.

Brzine u koracima 1. i 3. definira korisnik pod varijablom „Reset brzina“ na stranici „Postavke stroja“ na korisničkom sučelju. Vremena u koracima 2., 3. i 4. definirana su u programskom kodu i korisnik ih ne može mijenjati. U koraku 5. os se vraća na krajnji prekidač brzinom definiranom u programskom kodu. Ta brzina nije dostupna korisniku kako bi se osiguralo da se stroj uvijek referencira u istoj poziciji. Promjena brzine referenciranja uzrokovala bi i promjenu referentne točke što nikako nije poželjno u bilo kojem automatiziranom sustavu. Naime, svaki sustav ima svoje vrijeme odziva, pa tako i referentni krajnji prekidač. Uzmimo za primjer da vrijeme odziva i trajanje putovanja

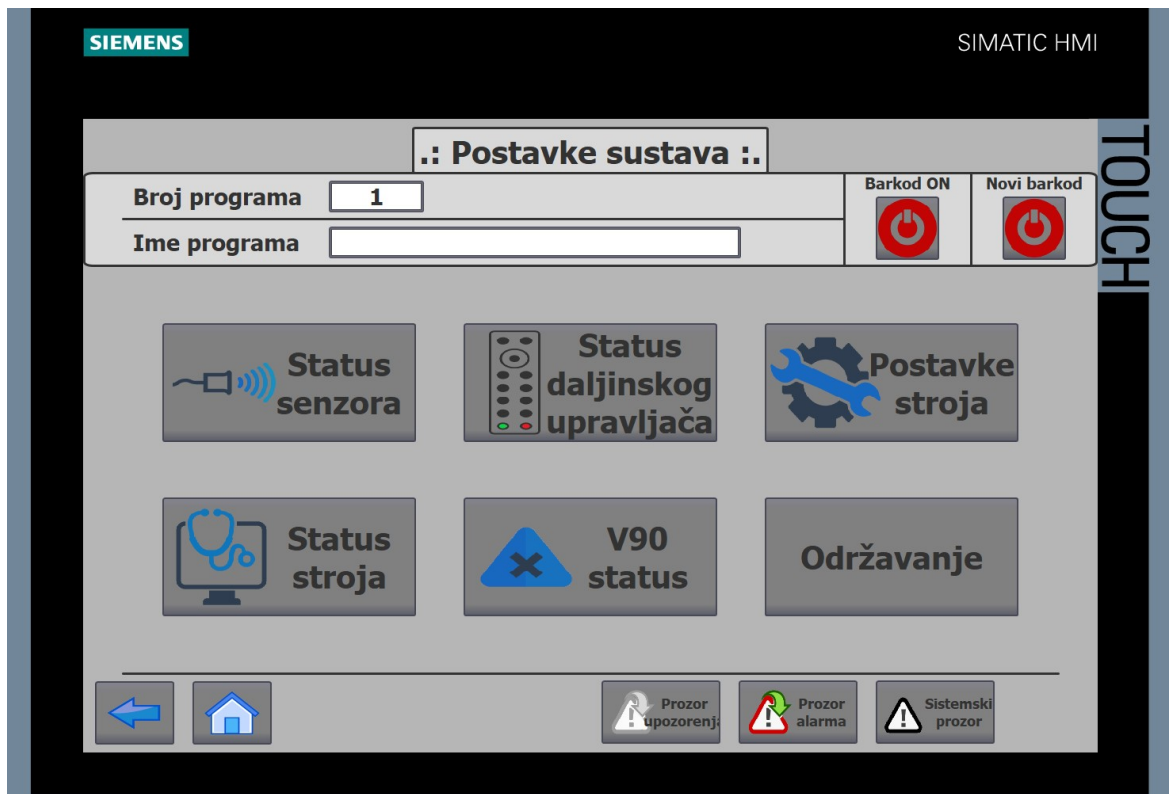
signala od referentnog krajnjeg prekidača do PLC uređaja iznosi 20 ms. Ako bi referenciranje osi izvodili pri različitim brzinama, vrijeme odziva referentnog krajnjeg prekidača uvijek bi bilo isto, ali put koji bi os prevalila u tom vremenu bi varirao. Manja brzina rezultirala bi manjim prevaljenim putom, a veća brzina većim. Svaki taj prevaljeni put zapravo predstavlja udaljenost od mjesta okidanja referentnog krajnjeg prekidača do referentne točke. Budući da je mjesto okidanja referentnog krajnjeg prekidača uvijek na istome mjestu, jer prekidač stoji na definiranom mjestu na stroju, može se zaključiti kako je jedino udaljenost referentne točke varijabilan podatak koji ovisi o brzini osi koja se referencira. Promjena referentne točke ne smije se dešavati kod referenciranja jer bi se onda zahtijevala prilagodba radnih i parking pozicija prema novoj referentnoj točki u svih 300 zavarivačkih programa. Postavljanje referentne točke uvijek na istom mjestu u prostoru je osnova od koje automatizacija stroja N1 kreće, jer bez konstantnih referentnih točaka cijeli proces automatizacije gubi svoj smisao. Na slici 4.2 nalaze se referentni krajnji prekidač desne osi oscilatora (lijevo) i referentni krajnji prekidač glavne vertikalne osi (desno).



Slika 4.2: Referentni krajnji prekidači na stroju

4.4 Postavke sustava

Na stranici „*Postavke sustava*“ podešavaju se svi parametri za ispravan rad stroja N1 i mogu se vidjeti trenutni statusi tipki na daljinskom upravljaču, krajnjih prekidača i stroja općenito te podesiti parametri stroja poput brzina osi. Stranica je prikazana na slici 4.3.



Slika 4.3: Stranica Postavke sustava stroja N1

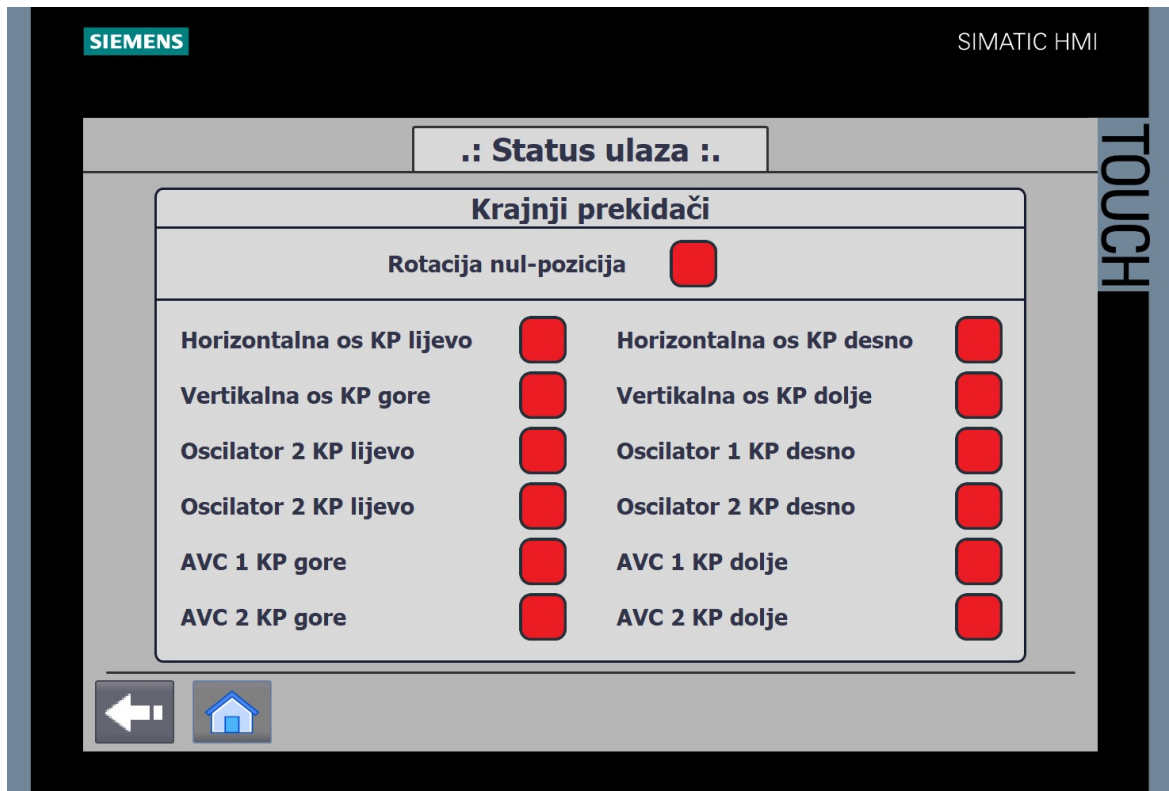
Na stranici „*Postavke sustava*“ nalaze se tipke za sljedeće stranice:

1. Status senzora
2. Status daljinskog upravljača
3. Postavke stroja
4. Status stroja
5. V90 status
6. Gumb za održavanje

Na donjem dijelu stranice nalaze se dvije vrste tipki. Tipke na lijevoj strani služe za navigaciju kroz stranice korisničkog sučelja: prva tipka vraća operatera na prethodnu stranicu dok druga tipka vodi operatera direktno na početnu stranicu sustava. Tipke na desnoj strani aktiviraju skočne prozore s tabličnim prikazom trenutno aktivnih alarma, upozorenja i sistemskih grešaka stroja. Svaka tipka otvara zaseban skočni prozor sa svojim funkcijama.

4.4.1 Status senzora

„Status senzora“ je stranica na kojoj korisnik može vidjeti trenutno stanje svih krajnjih prekidača na stroju, a prikazana je na slici 4.4. Korisnik na stranici ne može unijeti niti jedan parametar, već samo može vidjeti je li određeni krajnji prekidač aktivan ili neaktivan.



Slika 4.4: Stranica Status senzora

4.4.2 Status daljinskog upravljača

„Status daljinskog upravljača“ je stranica na kojoj korisnik može vidjeti trenutno stanje svih tipki i komandne palice na daljinskom upravljaču. Korisnik na stranici ne može unijeti niti jedan parametar, već samo može vidjeti je li određena tipka ili smjer komandne palice aktivan. Stranica je prikazana na slici 4.5.

Ako operater smatra da je određena tipka neispravna ili da su se odspojili kontakti na tipkama daljinskog upravljača, samostalno može dijagnosticirati kvar. Ako operater uspješno detektira kvar, ali ne može intervenirati kako bi taj kvar uklonio, mora upozoriti djelatnike iz odjela održavanja na prirodu nastalog kvara i zatražiti njihovu asistenciju u otklanjanju nastalog kvara.

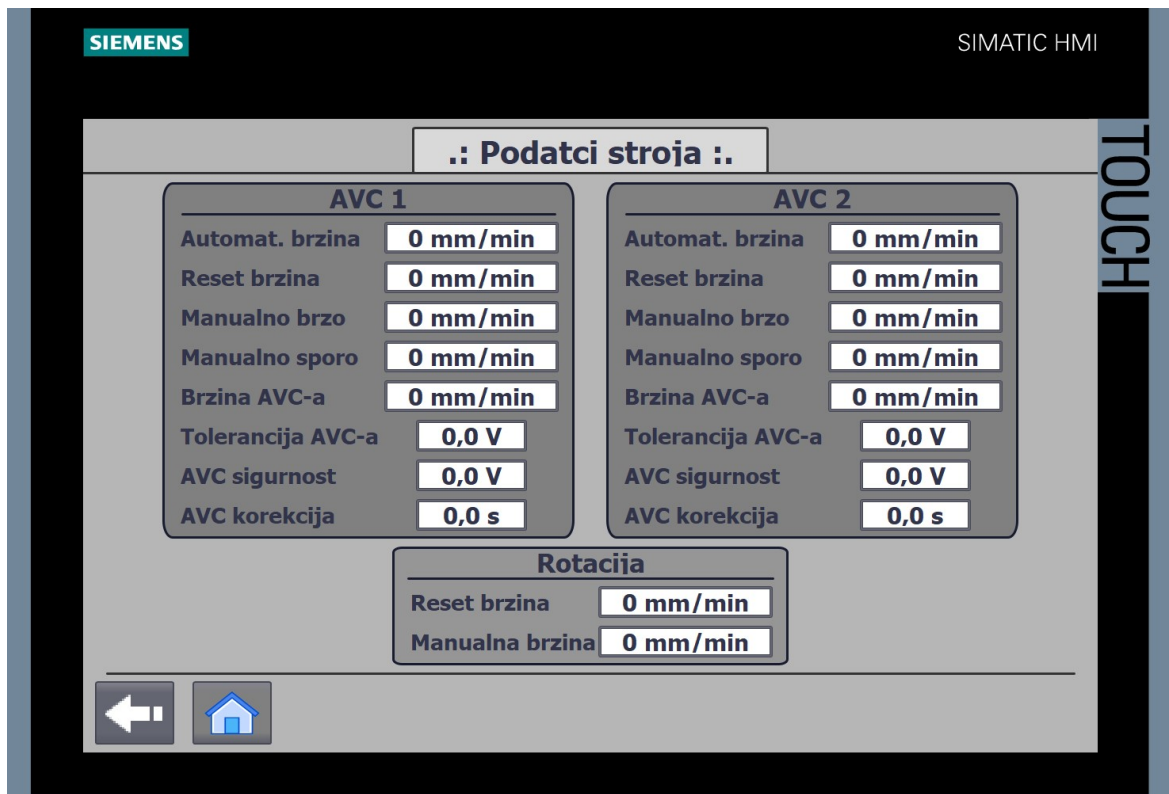


Slika 4.5: Stranica Status daljinskog upravljača

4.4.3 Postavke stroja

„*Postavke stroja*“ sastoje se od dvije stranice na kojima se nalaze postavke brzine svih osi i postavke AVC funkcije. Stranica s postavkama AVC osi prikazana je na slici 4.6.

Na prvoj stranici nalaze se postavke brzina glavne horizontalne i vertikalne osi te postavke brzina oscilatora. Na drugoj stranici nalaze se postavke brzina AVC osi i postavke AVC funkcije. Korisnik ima mogućnost podešavanja svih parametara na stranicama „*Postavke stroja*“, no vrijednosti koje može upisati su limitirane. Budući da se korisniku daje pravo korekcije brzina stroja, maksimalne brzine moraju se ograničiti zbog sigurnosti operatera, ali i samoga stroja. Korisnik također ima i mogućnost unosa tolerancije pozicije koja uvijek treba biti pozitivan broj. Zbog toga su sva I/O polja na stranicama „*Postavke stroja*“ limitirana maksimalnom i minimalnom vrijednošću koje svako polje može primiti. Čim korisnik krene pisati novu vrijednost u polje, iznad polja prikazat će mu se minimalna i maksimalna vrijednost koju može upisati.



Slika 4.6: Postavke AXC osi i AXC funkcije

4.4.4 Status stroja

„Status stroja“ je stranica koja sadržava informacije o ciklusima stroja, vrijeme zadnjeg automatskog ciklusa i brojače napravljenih automatskih ciklusa.

Gornja grupa podataka prikazuje informacije o sekvencijalnim koracima automatskog ciklusa zavarivanja i procedure za reset. Ako se dogodi da automatski ciklus zavarivanja ili reset ciklus uđu u grešku, korisnik može pročitati u kojem sekvencijalnom koraku je određeni ciklus stao i lakše dijagnosticirati uzrok greške. Primjerice, ako stroj stane, a „Korak automatskog ciklusa“ prikazuje broj 40 znači da stroj čeka da istekne brojač vremena za odgodu paljenja električnog luka. Greška se najvjerojatnije dogodila jer je korisnik unio preveliku vrijednost vremenskog parametra za odgodu paljenja električnog luka.

Donja grupa podataka prikazuje vrijeme posljednjeg automatskog ciklusa zavarivanja i brojače završenih automatskih ciklusa. Parametar „Brojač gotovih ciklusa“ prikazuje sveukupan broj gotovih ciklusa na stroju i korisnik ne može resetirati vrijednost ovog brojača na 0. Varijabla „Brojač dnevnih ciklusa“ prikazuje ukupan broj gotovih ciklusa od trenutka kada je stroj zadnji puta upaljen i korisnik ima mogućnost resetiranja vrijednosti brojača na 0. Varijable „Zavarivač X“ prikazuju broj odrađenih automatskih ciklusa svakog

od 4 zavarivača koji imaju pristup stroju. Nakon što se korisnik prijavi u sustav, brojač s njegovim korisničkim imenom bit će aktivan na stroju. Svi dnevni brojači ciklusa korisnika „Zavarivač X” mogu se resetirati na vrijednost 0, dok ukupne brojače ciklusa kod korisnika „Zavarivač X” nije moguće resetirati na vrijednost 0. Stranica „Status stroja” prikazana je na slici 4.7.

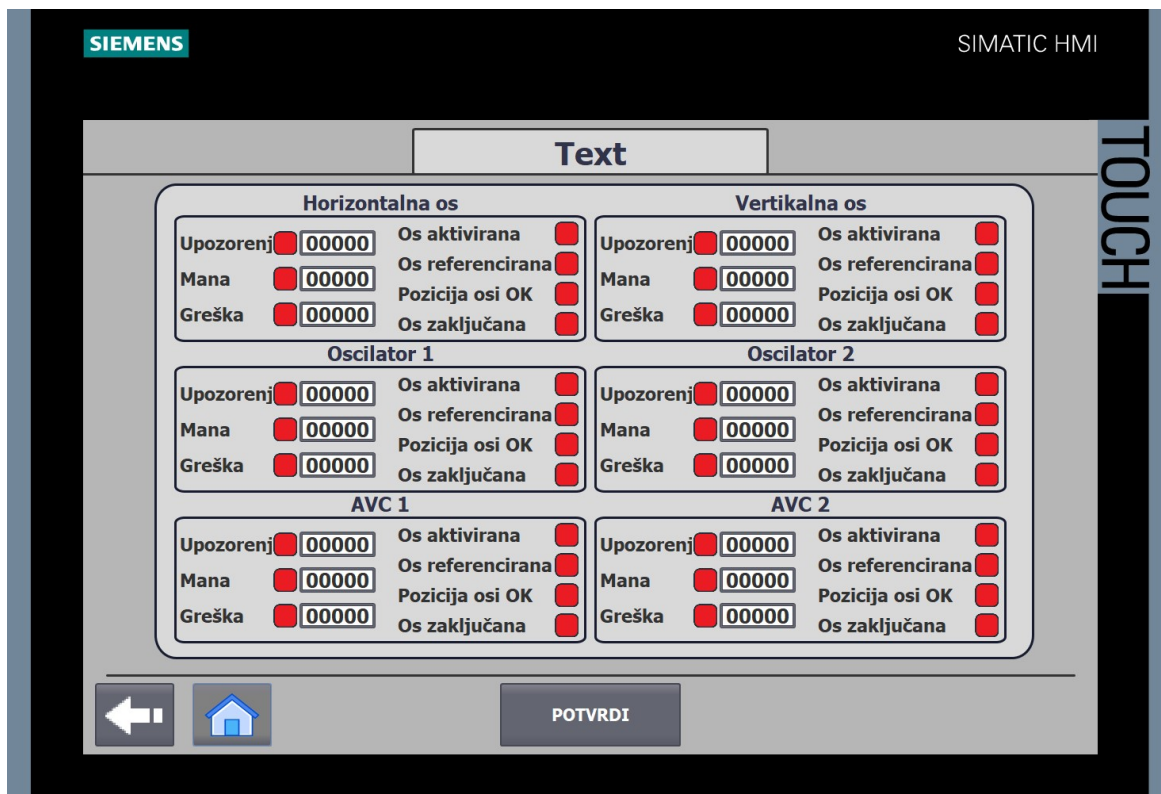


Slika 4.7: Stranica Status stroja

4.4.5 V90 status

„V90 status“ je stranica koja sadržava informacije s V90 servo upravljačkih sustava linearnih osi. Za svaku linearnu os korisnik na stranci može pročitati status i vrijednost upozorenja, mane i greške te informacije je li os aktivirana, referencirana, zaključana i je li pozicija osi u redu. Ako operater uoči neke nepravilnosti u automatskom radu stroja, na ovoj stranici može lako dijagnosticirati je li problem nastao u V90 servo upravljačkim sustavima i njihovim motorima. Ako bilo koji V90 servo upravljački sustav javlja upozorenje (eng. *Warning*), radnik kvar može otkloniti pritiskom na tipku „POTVRDI“. Upozorenja, kao što im i samo ime kaže, upozoravaju na neke nepravilnosti koje obično ne ometaju rad sustava pa vrlo često radnik neće ni primijetiti da su upozorenja aktivna. Upozorenja se najčešće javljaju ako neki parametar V90 servo upravljačkog sustava ili

motora izađe iz zadanih okvira koje smo mu zapisali u memoriju V90 servo upravljačkog sustava. Ako se na V90 servo upravljačkom sustavu javi mana (eng. *Fault*), operater ponovno mora stisnuti tipku „*POTVRDI*“. Ako tipka „*POTVRDI*“ ne otkloni nastali problem, operater mora pozvati djelatnike iz održavanja kako bi oni dijagnosticirali uzrok problema. Na slici 4.8 prikazana je stranica „*V90 status*“



Slika 4.8: Stranica V90 status

4.4.6 Gumb za održavanje

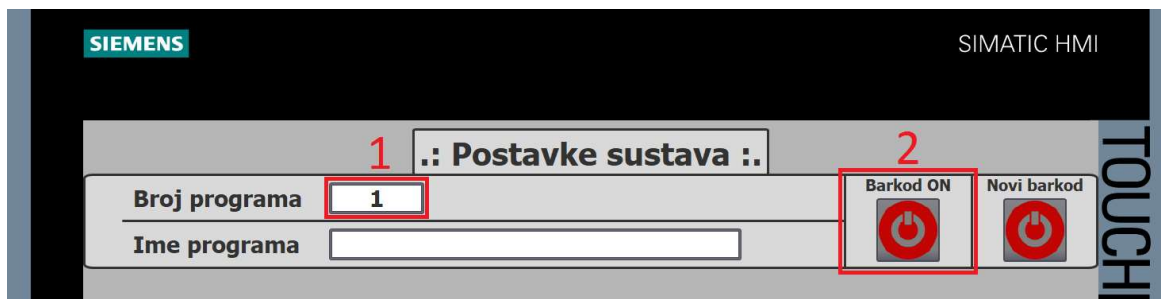
„*Gumb za održavanje*“ je tipka koja služi za aktivaciju načina rada održavanja i inspekcije kvarova na stroju. Aktivacijom ove tipke, tipka će postati crvena kako bi se korisniku dalo do znanja da je tipka aktivirana i programski se onemogućuje rad sa strojem, tj. pokretanje bilo kakvih ciklusa stroja. Manualni rad sa strojem također se onemogućava zbog sigurnosti radnika koji obavljaju održavanje.

4.5 Unos programa

Nakon što je sustav referenciran i spreman za rad, operater mora unijeti broj zavarivačkog programa koji želi da stroj obavi u automatskom ciklusu.

To može odraditi na dva načina:

1. ručno upisom broja zavarivačkog programa
2. automatski pomoću barkod skenera.



Slika 4.9: Mogućnosti unosa zavarivačkog programa

Na slici 4.9 prikazane su mogućnosti unosa zavarivačkog programa u sustav. Ako se program unosi ručnim načinom u polje „Program“, nakon unosa potrebno je na dnu ekrana pritisnuti tipku „Učitaj“ kako bi se program učitao u PLC. Način unosa programa barkod skenerom učitava odabrani broj programa u PLC uređaj automatski. Dok se zavarivački program učitava u sustav, na ekranu će se pojaviti tekst „Pričekaj“ koji naizmjenično blica crvenom i žutom bojom kako bi se operatera obavijestilo da je učitavanje zavarivačkog programa u tijeku.

4.6 Zavarivački program

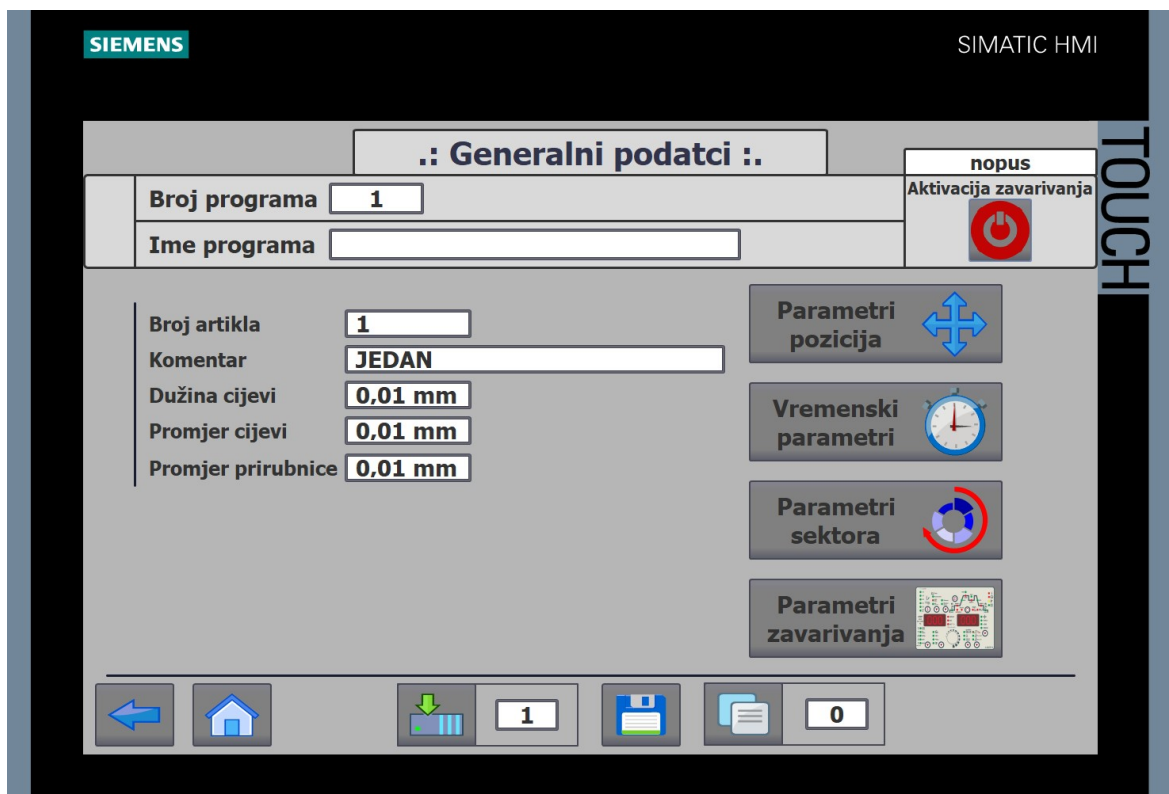
Svaki zavarivački program sastoji se od 611 parametara raspoređenih u sljedeće skupine:

- generalni podatci (6 parametara)
- parametri pozicija (12 parametara)
- parametri vremena (8 parametara)
- parametri sektora (40 parametara)
- parametri zavarivanja:
 - univerzalni parametri zavarivanja (5 parametara)
 - parametri po sektorima (27 parametara)
 - univerzalni parametri sektora (3 parametra)
 - gorionik 1 (12 parametara)
 - gorionik 2 (12 parametara).

Dio parametara bitan je za pravilno izvođenje automatskog ciklusa zavarivanja pa ih korisnik obavezno mora upisati u program. Drugi dio parametara ovisi o radnom komadu i specifikaciji zavarivanja te je upisivanje tih parametara u program opcionalno. Automatski ciklus zavarivanja može maksimalno sadržavati do 20 različitih setova zavarivačkih parametara koji se nazivaju sektori zavarivanja.

4.6.1 Parametri programa

„Parametri programa“ glavna je stranica za podešavanje parametara zavarivanja, a prikazana je na slici 4.10.



Slika 4.10: Parametri zavarivačkog programa

Na desnoj strani stranice nalaze se osnovne informacije o učitanoj programu:

1. broj artikla
2. komentar
3. dužina cijevi
4. promjer cijevi
5. promjer prirubnice.

Na lijevoj strani nalaze se tipke koje vode na stranice za podešavanje parametara:

1. parametri pozicija
2. vremenski parametri
3. parametri sektora
4. parametri zavarivanja.

4.6.2 Parametri pozicija

„Parametri pozicija“ je stranica koja sadrži parametre radne i parking pozicije zavarivačkog ciklusa. Na slici 4.11 prikazana je stranica „Parametri pozicija“.



4.11: Parametri pozicija zavarivačkog programa

Spremanje radne pozicije radi se preko tipke „STORING“ na daljinskom upravljaču. Prije spremanja pozicije korisnik mora napraviti dvije stvari. Prvo mora aktivirati tipku „Aktivacija spremanja pozicije“ na HMI uređaju. Ova tipka služi kao sigurnosna funkcija, odnosno aktivacija funkcije spremanja pozicije. Ovime se izbjegava da korisnik tijekom rada slučajnim pritiskom tipke „STORING“ ne spremi neku neželjenu poziciju u program. Prije spremanja programa korisnik prvo svjesno mora aktivirati tipku „Aktivacija spremanja pozicije“. Pritiskom na tipku „STORING“ na daljinskom upravljaču, tipka

„Aktivacija spremanja pozicije“ se deaktivira. Druga stvar koju korisnik mora definirati je položaj prekidača „POZICIJA“. Ako se prekidač nalazi u lijevoj ili „RADNA“ poziciji, onda se pritiskom na tipku „STORING“ trenutna pozicija stroja sprema u program kao radna pozicija. Ako se prekidač nalazi u desnoj ili „PARK“ poziciji, onda se trenutna pozicija stroja sprema u program kao parking pozicija. Obadvije pozicije vidljive su na stranci „Parametri pozicija“ i ažuriraju se automatski svaki puta nakon što se pritisne tipka „STORING“ na daljinskom upravljaču.

4.6.3 Vremenski parametri

„Vremenski parametri“ je stranica koja služi za definiranje vremenski određenih procesa u automatskom ciklusu.

Tablica 4.1: Definicije vremenskih parametara

Početni plin	Trajanje zaštitnog plina prije početka zavarivanja
Vrijeme formir plina 1	Trajanje plina za oblikovanje 1 koji se pušta u radni komad
Vrijeme formir plina 2	Trajanje plina za oblikovanje 2 koji se pušta u radni komad
Odgoda paljenja el. luka	Nakon koliko vremena od pozicioniranja stroja u radnu poziciju se uspostavlja električni luk za zavarivanje
Vrijeme porasta struje	Vrijeme podizanja struje na radnu vrijednost (rampa)
Odgoda početka rotacije	Nakon koliko vremena od uspostave električnog luka za zavarivanje se pokreće rotacija radnog komada
Vrijeme pada struje	Vrijeme spuštanja struje na nulu (rampa)
Završni plin	Trajanje zaštitnog plina nakon završetka zavarivanja

Tablica 4.1 ukratko opisuje funkcije svakog parametra na stranici „Vremenski parametri“. Svaka funkcija detaljnije je opisana u daljnjem tekstu.

Parametar „Početni plin“ pušta se kroz gorionik kako bi se mjesto zavarivanja očistilo od nečistoća i kako bi se oko njega stvorila zaštitna atmosfera. Zaštitni plin također propuhuje i čisti gorionik. Bez početnog plina početak zavara bio bi izrazito crn, nepravilan, šupljikav i oko zavara bi bio velik broj prskotina.

Parametri „*Vrijeme formir plina 1*“ i „*Vrijeme formir plina 2*“ koriste se kako bi se unutar radnog komada pustio plin za oblikovanje koji služi za formiranje zavara s unutarnje strane cijevi. On stvara određeni pritisak unutar cijevi kako na mjestu zavarivanja ne bi došlo do kapanja rastaljenog materijala. Premala količina plina za oblikovanje uzrokuje kapanje rastaljene mase, a prevelika količina poroznost u zavaru i prskanje budući da plin krene izlaziti iz radnog komada kroz rastaljenu masu, što je prikazano na slici 4.12.



Slika 4.12: Posljedice prevelike količine plina za oblikovanje

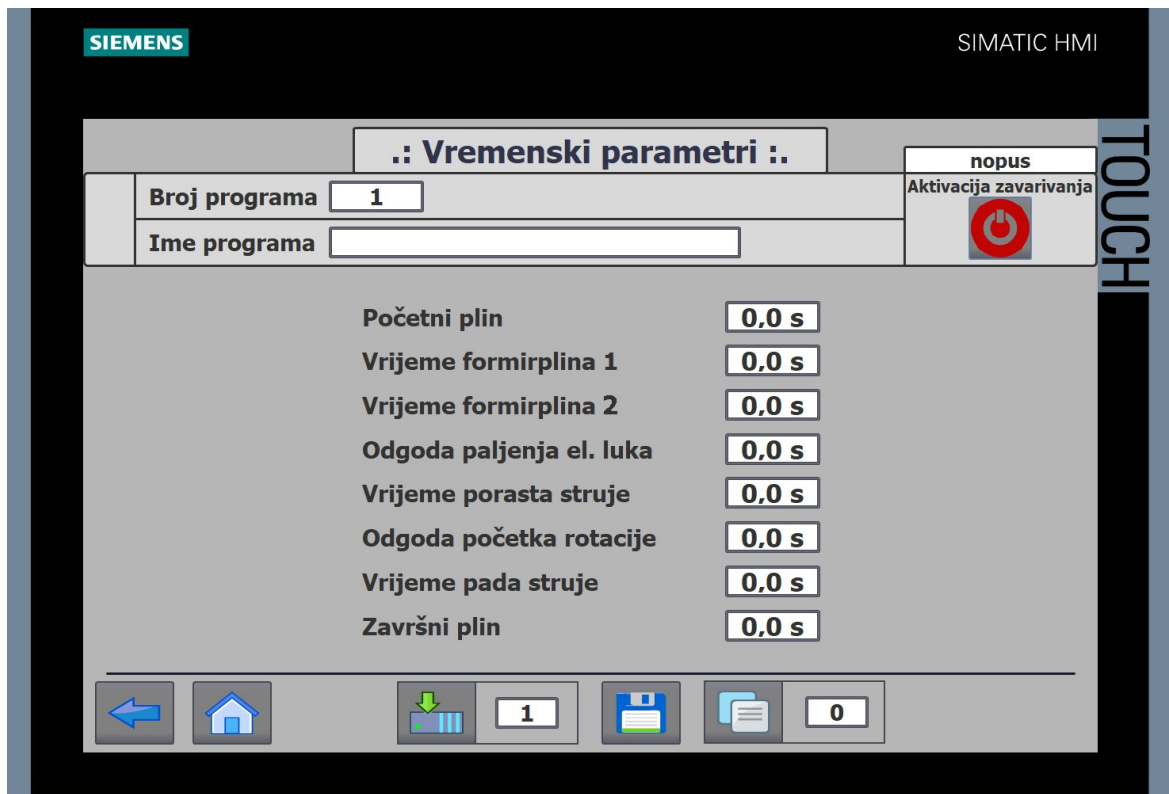
Parametar „*Odgoda paljenja električnog luka*“ koristi se kako bi se stvorila dostatna količina zaštitne atmosfere oko mjesta zavarivanja. U pravilu su vrijeme početnog plina i vrijeme odgode paljenja el. luka iste vrijednosti, no u ovome programskom rješenju su razdvojeni kako bi korisnik mogao svaku funkciju kontrolirati zasebno.

Parametar „*Vrijeme porasta struje*“ i „*Vrijeme pada struje*“ su vremenske vrijednosti koje definiraju uzlaznu, odnosno silaznu putanju rampe za struju zavarivanja.

Parametar „*Odgoda početka rotacije*“ je vremenska varijabla koja počinje teći nakon uspostave električnog luka i služi kako bi se stvorio bazen rastaljenog materijala i kako bi struja zavarivanja imala dovoljno vremena da se podigne na radnu vrijednost, te da prođe u radni komad.

Parametar „*Završni plin*“ počinje teći na gorioniku tek kada iščezne električni luk, odnosno kada završi vrijeme pada struje i služi kao zaštita vruće volframove elektrode i vrućeg završnog zavara od nečistoća iz zraka.

Vremenski parametri prikazani su na slici 4.13.



Slika 4.13: Vremenski parametri zavarivačkog programa

4.6.4 Parametri sektora

Stroj N1 ima mogućnost spremanja 300 različitih programa s definiranim položajima i parametrima zavarivanja. Svaki program dodatno je podijeljen u 20 sektora koji služe kao setovi parametara unutar programa.

Tek kada se prvi sektor aktivira, na ekranu će se prikazati sljedeći sektor i tako sve do sektora 20. Time se operatera sprječava da definira sektore napreskokce što bi rezultiralo neispravnim radom stroja. Također svaki sljedeći sektor kao minimalnu vrijednost unosa konačne pozicije ima vrijednost prethodnog sektora. Drugim riječima, ako operater kao krajnju vrijednost prvog sektora definira 180° , kao krajnja vrijednost drugog sektora ne može se upisati vrijednost manja od 180° . Time se osigurava da krajnje vrijednosti sektora uvijek rastu uzlaznom putanjom.

Sektoru su kod automatiziranog zavarivanja vrlo bitan faktor jer omogućuju unos različitih parametara zavarivanja u jedan program. Primjerice kod komada s debljom stjenkom jedan prolaz (rotacija za 360°) nije dovoljan da popuni pripremu zavara (kanal u kojem se radi zavarivanje). Proces je moguće izvesti s manjom brzinom zavarivanja i većom brzinom žice kako bi se cijeli kanal ispunio, no onda se ne bi postigla odgovarajuća

kvaliteta korijenskog zavara koja je ključan faktor dobrog zavarenog spoja. Zbog toga se u automatizirane zavarivačke procese uvode sektori.



4.14: Parametri sektora zavarivačkog programa

Na slici 4.14 prikazana je stranica „*Parametri sektora*“. Svakom sektoru se definira krajnja vrijednost pozicije. Prvi sektor uvijek je korijenski zavar i definira ga se od 0° do primjerice 363° . Korijenski zavar uvijek mora biti za par stupnjeva veći od punog kruga, kako bi se dobio lijep prijelaz, tj. kako bi se početak zavara i kraj zavara stopili u jednoličnu masu. Taj postupak naziva se preklapanje (eng. *Overlapping*). Ako bi se prvi sektor napravio rotacijom od 360° , jasno bi se vidjela granica između početka i kraja zavara, odnosno završetak zavara. Prvi ili korijenski prolaz najbitniji je prolaz u orbitalnom zavarivanju jer je on nositelj cijelog zavarenog spoja i svi ostali prolazi imaju samo funkciju popunjavanja materijala i povećavanja čvrstoće spoja. Nakon toga definiraju se parametri za druge sektore koji se nazivaju ispuna. Sektori ispune obično imaju veću brzinu žice i brzinu rotacije od korijenskog zavara kako bi se prije ispunio kanal za zavarivanje. Tijekom zavarivanja korijenskog zavara u radni komad unesena je određena količina topline koja ostaje u radnom komadu i tijekom ispune. Kako se u radni komad ne bi unijela prevelika količina topline koja može uzrokovati distorziju i promjenu

molekularne strukture radnog komada, u sektoru ispune obično se smanjuje jakost struje zavarivanja. Stupnjevi sektora ispune definiraju se ovisno o radnom komadu i obično nema potrebe raditi preklapanja između sektora. Primjerice, kod manjih komada obično se za ispunu od 360° radi 8 ili više sektora. Mali promjer cijevi i tanka stjenka znači i mali volumen materijala pa se već nakon par desetaka stupnjeva u radni komad unese poprilična količina topline. Zbog toga je potrebno definirati više sektora u kojima se postepeno smanjuje struja zavarivanja kako bi se smanjio unos topline u radni komad. Dodatna žica će se i dalje normalno topiti jer, iako je smanjen unos topline kroz električni luk, sam komad ima dostatnu količinu energije za adekvatno rastaljivanje dodatne žice. S druge strane, kod velikih promjera cijevi s debelom stjenkom potrebno je napraviti par rotacija kako bi se priprema zavara kvalitetno ispunila. Svaka rotacija može biti zasebni sektor, ali nije rijedak slučaj da se jedan sektor ispune kod ovakvih komada definira i za više od 360°.

Nakon sektora ispune radi se završni sektor koji se još naziva i završni prolaz. U završnom prolazu ponovno je potrebno preklapanje kako bi zavar bio estetski i metalurški prihvatljiv. Ako bi se završni prolaz radio bez preklapanja i spuštanja struje sa radne vrijednosti na 0 A, na mjestu gdje se ugasio električni luk zavarivanja pojavila bi se poroznost u zavaru što se nikako ne smije dogoditi.

4.6.5 Parametri zavarivanja

„*Parametri zavarivanja*” je stranica koja služi za podešavanje parametara potrebnih za zavarivački proces.

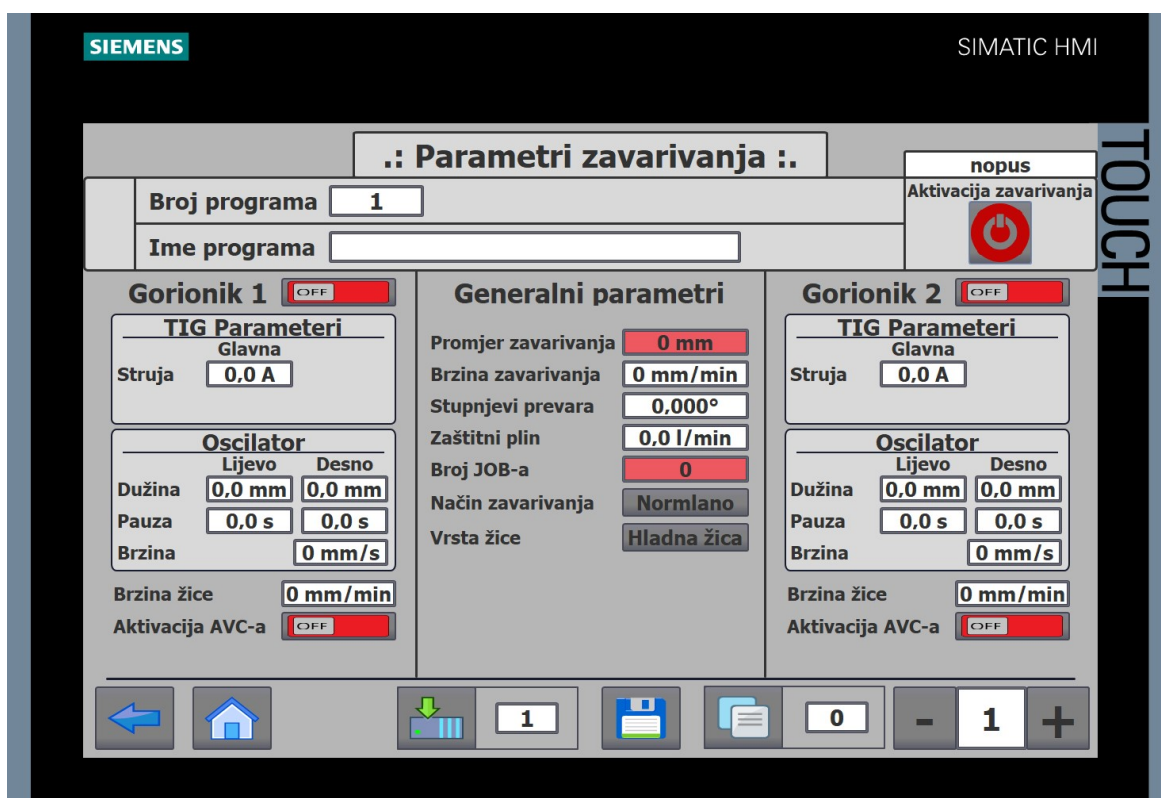
Podijeljeni su u 3 glavne sekcije, od čega su dvije dodatno podijeljene u 3 podsekcije:

1. *Generalni parametri*
2. *Gorionik 1 (TIG Parametri, Oscilator i Parametri žice i AVC-a)*
3. *Gorionik 2 (TIG Parametri, Oscilator i Parametri žice i AVC-a).*

„*Generalni parametri*“ su zajednički parametri za obadva gorionika, a parametri „*Promjer zavarivanja*“, „*Broj JOB-a*“ i „*Stupanj prevara*“ konstantni su kroz sve sektore. Parametri „*Brzina zavarivanja*“, „*Zaštitni plin*“, „*Broj programa*“ i „*Način zavarivanja*“ mogu se mijenjati kroz sektore. Krucijalne vrijednosti za pravilno izvršavanje zavarivačkog procesa u nedefiniranom zavarivačkom programu imaju crvene pozadine I/O polja pri vrijednostima 0. Time se operateru želi vizualizirati da je unos vrijednosti u to polje obavezan. Bez tih parametara nema zavarivačkog procesa. Svi drugi parametri samo utječu

na zavarivački proces, te na izgled i kvalitetu zavara, ali zavarivački proces bez njih može funkcionirati.

Na slici 4.15 prikazani su parametri zavarivanja. „Gorionik 1“ i „Gorionik 2“ imaju identične nazive parametara, stoga će ih se opisati zajedno pod nazivom Gorionik X. Korisnik ima opciju rada sa samo jednim gorionikom ili obadva gorionika istovremeno. Željeni gorionik aktivira se prekidačem *Gorionik X*. Svaki gorionik ima parametre podijeljene u tri podsektora.



Slika 4.15: Parametri zavarivanja zavarivačkog programa

Prva podsekcija naziva se „TIG parametri“ i služi za podešavanje iznosa jakosti struje zavarivanja i njezinih trajanja. Ako operater koristi normalni način zavarivanja tada mu je na HMI uređaju prikazano samo polje za unos glavne struje zavarivanja jer mu je taj parametar jedini potreban za odabrani način rada. Ako korisnik radi s pulsним načinom zavarivanja, tada mu se uz polje za unos glavne struje zavarivanja pojave i kućice za unos iznosa jakosti pulsne struje zavarivanja i kućice za trajanje svake struje zavarivanja. U pulsnom načinu rada struja pulsira, odnosno određeno vrijeme zavaruje s iznosom glavne struje zavarivanja, a određeno vrijeme s pulsnom strujom zavarivanja. Obadvije vremenske varijable trajanja struja definira operater. U pravilu je iznos pulsne struje zavarivanja

uvijek manji od iznosa glavne struje zavarivanja. Ovaj način koristi se kod materijala koji imaju zahtjev za malim unosom topline, malih promjera cijevi ili cijevi s tankom stjenkom. Pulsni način zavarivanja može se usporediti s pulsno-širinskom modulacijom ili PWM-om (eng. *Pulse-width modulation*) u elektrotehnici, jedina razlika je u tome što vrijednost pulsne struje nije 0, nego neka konstantna vrijednost. Kao što se kod PWM-a visinom amplitude i širinom signala nazivna vrijednost smanjuje na vrijednost izlaznog signala, tako se kod zavarivanja pulsним načinom rada smanjuje vrijednost topline koja se unosi u radni komad.

Druga podsekcija naziva se „*Oscillator*“ i služi za podešavanje oscilacije osi oscilatora tijekom zavarivačkog procesa. Radna pozicija gorionika uvijek se uzima kao centralna točka iz koje oscilacija kreće, i od nje se definiraju parametri „*Dužina Lijevo*“ i „*Dužina Desno*“. Ovakvim načinom omogućeno je da oscilacija bude asimetrična jer neki radni komadi to zahtijevaju. Prilikom spajanja prirubnice i cijevi, priprema zavara ne mora biti zrcalno simetrična na spojenim bridovima. U tom slučaju ni spoj bridova, koji definira radnu točku i korijen zavara, ne nalazi se točno u centru pripreme zavara. Zbog toga u sektorima ispune jedna strana mora oscilirati više kako bi se priprema zavara ispunila u cijelosti.

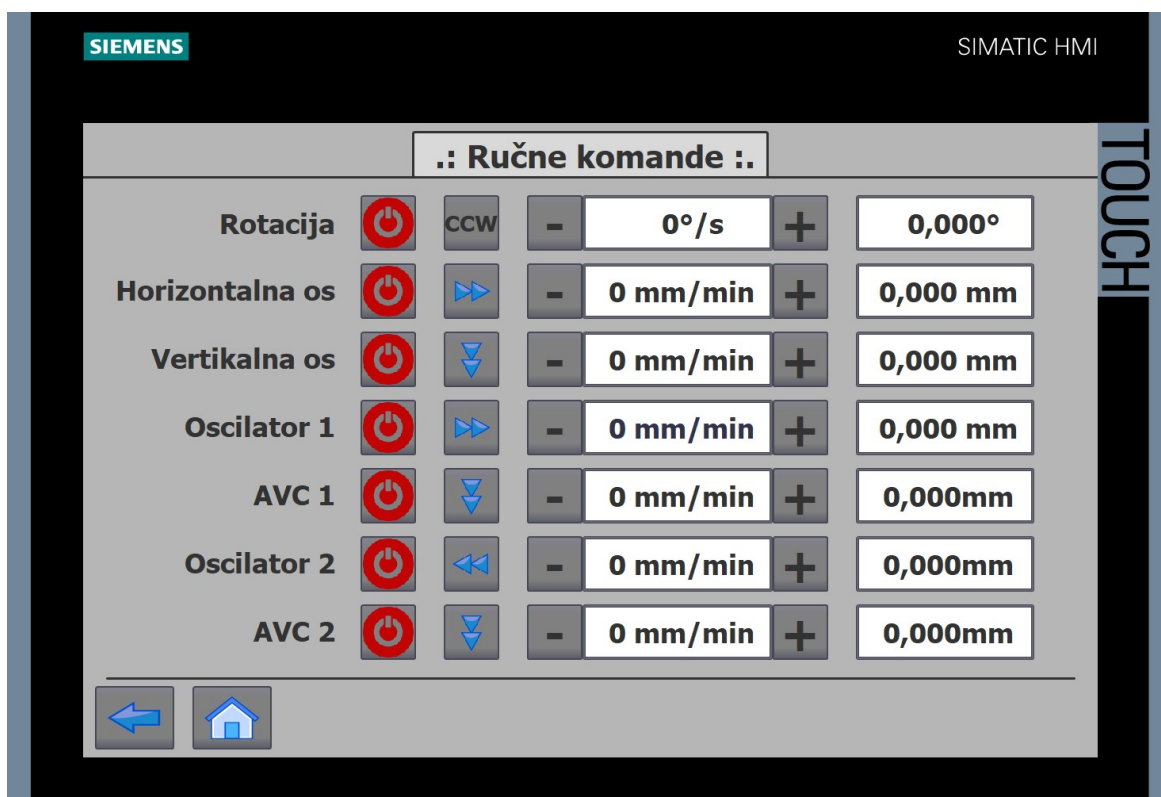
Treća podsekcija naziva se „*Parametri žice i AVC-a*“ i služi za podešavanje brzine žice za zavarivanje, te aktivaciju i podešavanje funkcije AVC-a. Brzina žice parametar je koji se mora pažljivo uskladiti s brzinom rotacije radnog komada. Ako je brzina žice prespora, žica se neće moći taliti konstantno te će ju struja zavarivanja paliti i topiti u kuglicama. S druge strane, ako je brzina žice prevelika rastalit će se prevelika količina žice koja će se izliti iz pripreme zavara po radnom komadu. AVC ili automatska kontrola napona električnog luka (eng. *Automatic Voltage Control*) služi za distanciranje gorionika od radnog komada u automatskom zavarivačkom ciklusu. Operater mora aktivirati prekidač „*Aktivacija AVC-a*“ i podesiti „*Setpoint AVC-a*“ vrijednost kako bi aktivirao AVC funkciju. Aparat za zavarivanje u automatskom zavarivačkom procesu očitava vrijednost napona na gorioniku koja raste s udaljavanjem gorionika od radnog komada, odnosno s povećanjem električnog luka i izmjerenu vrijednost šalje na PLC uređaj. AVC funkcija drži gorionik u tolerancijskom polju vrijednosti definirane pod „*Setpoint AVC-a*“ podizanjem i spuštanjem pozicije gorionika. Dobrim parametriziranjem AVC funkcije i sektora zavarivanja na stroju bi se moglo zavarivati i kvadratne i pravokutne cijevi jer bi AVC funkcija spuštanjem i podizanjem pozicije gorionika mogla pratiti promjenu brida radnog komada.

4.7 Ručno upravljanje

„Ručno upravljanje“ je stranica pomoću koje korisnik može manualno upravljati svim osima stroja bez korištenja komandne palice na daljinskom upravljaču.

Prvi stupac sadržava tipke za aktivaciju osi. Drugi stupac sadržava tipke za određivanje smjera kretanja osi. Treći stupac sadržava brzine osi koje se unose ručno. Tipke minus i plus služe za korekciju upisane vrijednosti. Četvrti stupac sadržava prikaze trenutnih pozicija osi.

Operater treba aktivirati os ili osi koje želi koristiti te odrediti njihov smjer i brzinu kretanja. Pritiskom na tipku „START“ na daljinskom upravljaču, sve aktivirane osi početi će se kretati brzinom definiranom u trećem stupcu odabrane osi, te u definiranom smjeru kretanja osi. Operater može upravljati s više osi istovremeno, no to nije preporučljivo budući da se povećava rizik od kolizije stroja s okolinom i stroja samim sobom. Preporuka je svaku os kontrolirati zasebno kako bi se rizik od kolizije sveo na minimum. Kada korisnik otpusti tipku „START“ stroj će stati u pozicijama u kojima se našao kada je tipka otpuštena.



Slika 4.16: Manualno upravljanje strojem N1

Manualno upravljanje služi kao sigurnost u slučaju kvara na komandnoj palici daljinskog upravljača, ali i kao stranica za provjeru trenutnih pozicija. Ako se operateru čini da se stroj ne pozicionira dobro, može napraviti simulacijski zavarivački program. Na stranici „Manualno upravljanje“ zatim može provjeriti odgovaraju li trenutne pozicije linearnih osi pozicijama koje je operater upisao u simulacijski zavarivački program. Stranica „Manualno upravljanje“ prikazana je na slici 4.16.

4.8 Glavna lista programa

„Glavna lista programa“ je stranica koja u tabličnom prikazu sadržava generalne parametre svih 300 zavarivačkih programa na stroju. Na lijevoj strani stranice nalazi se broj programa, a desno od njega generalni parametri programa s tim brojem. Stranica „Glavna lista programa“ služi kako bi korisnik mogao pronaći odgovarajući program za radni komad u slučaju da nema barkod tog radnog komada. Stranica „Glavna lista programa“ je prikazana na slici 4.17.

Pr. N°	Ar. N°	Dužina	Cijev Φ	Pri. Φ	Barkod	Komentar
1	0	000,0 mm	0,0 mm	0,0 mm		
2	0	000,0 mm	0,0 mm	0,0 mm		
3	0	000,0 mm	0,0 mm	0,0 mm		
4	0	000,0 mm	0,0 mm	0,0 mm		
5	0	000,0 mm	0,0 mm	0,0 mm		
6.	0	000,0 mm	0,0 mm	0,0 mm		
7.	0	000,0 mm	0,0 mm	0,0 mm		
8.	0	000,0 mm	0,0 mm	0,0 mm		
9.	0	000,0 mm	0,0 mm	0,0 mm		
10.	0	000,0 mm	0,0 mm	0,0 mm		
11.	0	000,0 mm	0,0 mm	0,0 mm		
12.	0	000,0 mm	0,0 mm	0,0 mm		
13.	0	000,0 mm	0,0 mm	0,0 mm		
14.	0	000,0 mm	0,0 mm	0,0 mm		
15.	0	000,0 mm	0,0 mm	0,0 mm		

Slika 4.17: Glavna lista programa

4.9 Postavljanje radnog komada u stroj

Operater prvo postavlja prihvate za radni komad u steznu glavu i na pneumatski konjić te ih steže. Prihvati za radni komad u sebi imaju probušenu sredinu kako bi se osigurao protok plina za oblikovanje kroz radni komad tijekom zavarivanja. Nakon što su prihvat za radni komad stegnuti, operater može postaviti komad u prihvati na lijevoj strani stroja, odnosno na prihvati postavljen u steznu glavu. Operater za postavljanje radnog komada ima slobodnu samo jednu ruku, pa se preporučuje ručno postavljanje komada do 10 kilograma. Sve teže radne komade operater bi u stroj trebao postavljati pomoću kрана ili neke druge vrste mehanizma za pomoć pri podizanju tereta. Druga ruka operateru mora biti slobodna za pozicioniranje pneumatskog konjića. Pneumatski konjić postavlja se pomoću dvije pneumatske sklopke. Prva sklopka služi za aktiviranje pneumatske kočnice. U negativnom stanju pneumatska kočnica je zakočena, a kako bi ju se otpustilo potrebno je na nju narinuti zrak pritiska minimalno 5 bara prebacivanjem prve sklopke u pozitivno stanje. Odabirom ovakve vrste kočnice osigurava se nesmetanost izvođenja automatskog ciklusa u slučaju pada tlaka ili čak i nestanka tlaka u cijelom sustavu. Druga sklopka služi za pomicanje pneumatskog cilindra s prihvatom za radni komad. Postavljanje sklopke lijevo odvest će pneumatski cilindar u lijevu krajnju točku, odnosno radnu točku, a postavljanje sklopke u desno, odvesti će ga u desnu krajnju ili nultočku. Operater kod pozicioniranja pneumatskog konjića mora paziti na dvije stvari: kočnica uvijek mora biti zakočena prije nego što se krene upravljati pneumatskim cilindrom i tlak cilindra ne smije biti veći od tlaka kočnice, odnosno 5 bara. Ako se bilo koje od ovih pravila zanemari, rezultat će biti odgurivanje pneumatskog konjića u desnu stranu i gubitak njegove pozicije te u konačnici nemogućnost pokretanja automatskog ciklusa zavarivanja.

Kada operater pravilno postavi radni komad u stroj, ispred radnog komada mora postaviti zaštitni prozor sa zatamnjenim staklom za zavarivanje (DIN 11). Pritiskom na gumb „*START*“ na daljinskom upravljaču stroj odrađuje potpuno automatizirani proces zavarivanja s parametrima prethodno unesenim u korisničko sučelje. Nakon što je proces zavarivanja gotov, stroj se vraća u parking poziciju kako bi operater mogao zavareni komad izvaditi iz stroja i ponoviti proces s novim komadom.

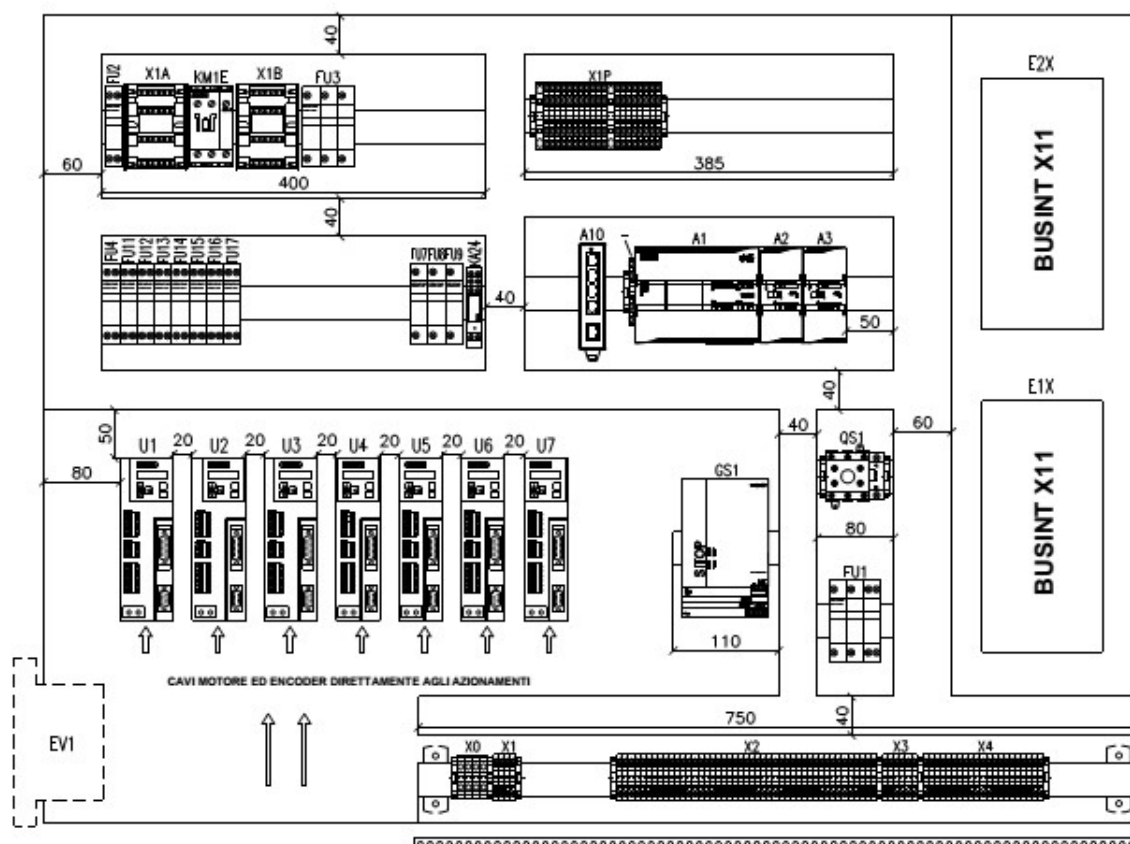
5. OPIS OPREME I KOMPONENATA STROJA

Stroj je sastoji od električnog i mehaničkog dijela, koji je dodatno podijeljen u dvije cjeline: SCAR i STL. Svi dijelovi pobliže su opisani u sljedećim potpoglavljima.

5.1 Električni dio

Električni dio nalazi se u upravljačkom ormaru s lijeve strane stroja. Upravljački ormar povezan je s 13 senzora i 7 motora na stroju pomoću kablova koji s desne strane upravljačkog ormara izlaze prema stroju. Iz upravljačkog ormara također izlazi i signalni kabel za daljinski upravljač te dva kabela koji povezuju *EWM BUSINT TETRIX X11* modul za komunikaciju s *EWM Tetricx 552 Synergic RC HW* aparatima za TIG zavarivanje.

Upravljački ormar napaja se preko trofaznog kabela spojenog preko glavne sklopke za prednju ugradnju na *SIEMENS SITOP 24 V DC – 20 A* jedinicu za napajanje. Sa SITOP-a napajanje se preko osigurača spaja na ostale uređaje u ormaru. Shema električnog ormara prikazana je na slici 5.1.



Slika 5.1 Shema električnog ormara stroja N1

5.1.1 PLC

Najvažniji element upravljačkog ormara je *PLC SIMATIC S7-1200 CPU1215 DC/DC/DC* na kojemu se nalazi najveći dio upravljačkog programa (pozicioniranje osi nalazi se na V90 servo upravljačkim sustavima).



Slika 5.1: PLC SIMATIC S7-1200 6ES7215-1AG40-0XB0 [3]

Kako bi se lakše shvatio princip rada PLC uređaja, opisać će se njegova arhitektura koja se sastoji od:

- centralna procesna jedinica/mikroprocesor
- ulazno/izlazni sklopovi
- memorija
- sabirnice.

PLC uređaj napaja se pomoću 24 V DC. Na prednjoj strani uređaja nalazi se LED diode koje signaliziraju stanje izlaznih, odnosno ulaznih sklopova. Konektor za napajanje i ulazne stezaljke nalaze se na gornjoj strani uređaja, a dva PROFINET konektora i izlazne stezaljke s donje strane uređaja. Komunikacijski moduli (eng. *Communication Modules, CM*) uvijek se dodaju s desne strane PLC uređaja i služe za serijsku komunikaciju PLC uređaja s drugim uređajima (npr. posebnim sensorima, vizijskim sustavima ili sigurnosnim modulima). PLC uređaj stroja N1 nema na sebe dodan niti jedan komunikacijski modul. Signalni moduli (eng. *Signal Modules, SM*) uvijek se dodaju s lijeve strane PLC uređaja i

služe za proširenje ulaza i/ili izlaza PLC uređaja . Ovaj model PLC uređaja nema dovoljan broj potrebnih ulaza za predviđenu namjenu stoga su na njega dodana 2 ulazna modula:

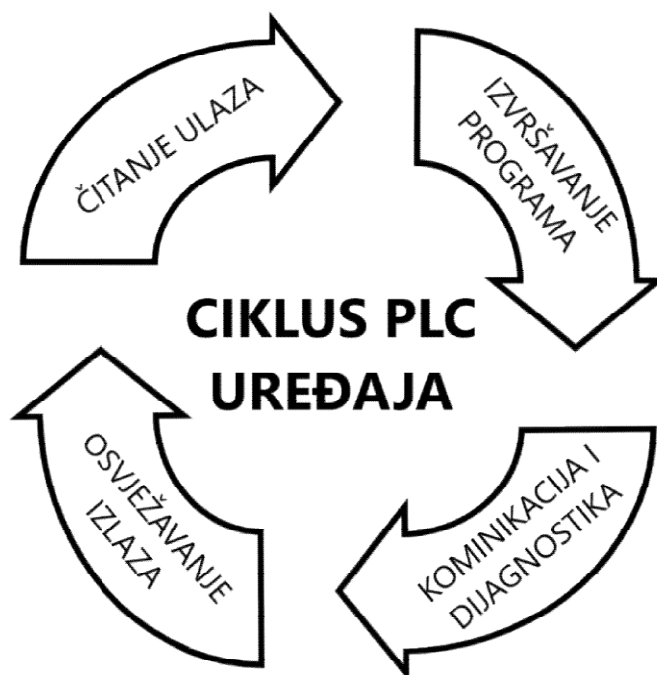
- *SM 1221-16DI* (signalni modul sa 16 ulaza)
- *SM 1221-8DI* (signalni modul s 8 ulaza).

Tablica 5.1: Karakteristike SIMATIC S7-1200 CPU1215 DC/DC/DC/ PLC uređaja

Dimenzije		130x100x75 mm
Težina		500 g
Napon napajanja	Radna vrijednost	24.0 V DC
	Donja granica	20.4 V DC
	Gornja granica	28.8 V DC
Memorija	Memorija za učitavanje	4 Mb
	Radna memorija	125 kb
	Memorija sustava	10 kb
Ulazno/izlazni sklopovi	Digitalni ulazi/izlazi	14/10
	Analogni ulazi/izlazi	2/0
Brzina izvršavanja operacija	S bitovima	0.08 μ s
	S vrijednostima riječi	1.7 μ s
	Aritmetičke operacije s vrijednostima s pomičnim zarezom	2.3 μ s
Mogućnosti proširenja hardvera	Komunikacijski moduli	3
	Signalne pločice	1
	Signalni moduli	8

Signalni moduli na sebi imaju konektore koji se priključe na PLC uređaj ili na drugi signalni modul tako da nema dodatnog ožičenja. Ako je korisniku potreban mali broj dodatnih ulaza ili izlaza, PLC uređaj ima opciju umetanja signalne pločice (*eng. Signal Board, SB*) s prednje strane uređaja. PLC uređaj prima signale s 14 digitalnih i 2 analogna ulaza integriranih u uređaj i 24 digitalna ulaza na signalnim modulima s kojima je PLC uređaj proširen. Nakon što PLC uređaj dobije ulazne signale, CPU jedinica određuje izlazne signale na temelju logike zapisane u programskom kodu CPU memorije. Izlazni

signali šalju se na aktuatora preko 10 digitalnih izlaza i/ili PROFINET komunikacije. Na slici 5.2 grafički je prikazan ciklus PLC uređaja.



Slika 5.2: Ciklus PLC uređaja

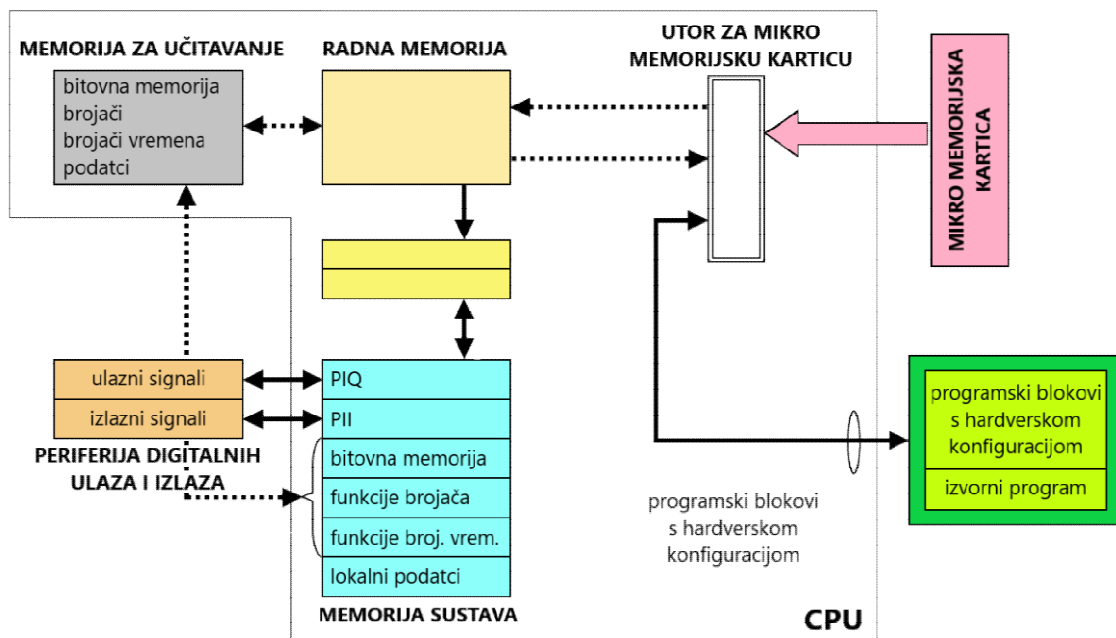
Memorija CPU jedinice sastoji se od sistemske memorije i korisničke memorije koja je dodatno podijeljena u radnu memoriju i memoriju za učitavanje.

Memorija za učitavanje (eng. *Load memory*) je memorija u koju se zapisuje programski kod, konfiguracija hardvera, komentari i drugo. Izvedena je kao EEPROM (eng. *Electrically Erasable Programmable Read Only Memory*) memorija u koju se podatci spremaju trajno neovisno o naponu napajanja. Ugrađena je u CPU jedinicu i može biti dodatno proširena umetanjem *SIMATIC MMC* (eng. *Micro Memory Card*) mikro memorijske kartice. Memorija za učitavanje ostaje nepromijenjena unutar PLC uređaja, moguće ju je promijeniti jedino učitavanjem novog programskog koda.

Radna memorija (eng. *Work memory*) je memorija u koju se, nakon pokretanja PLC uređaja, programski kod prebacuje u obliku strojnog koda (npr. bez komentara, naziva varijabli) iz memorije za učitavanje. Izvedena je kao RAM (eng. *Random Access Memory*) memorija u potpunosti integrirana u CPU modul bez mogućnosti proširenja. U radnoj memoriji nalaze se samo oni podatci koji su bitni za izvođenje programskog koda čime se osigurava brz pristup memoriji i brzo izvođenje programskog koda. Podatci smješteni u radnu memoriju brišu se u slučaju prestanka napona napajanja.

Memorija sustava (eng. *Retentive Memory*) je memorija koja ostane zapisana unutar memorijskih lokacija PLC uređaja kada na njemu nema napajanja. U nju su smješteni podaci koji su bitni za izvođenje programskog koda poput stanja digitalnih ulaza i izlaza. Podatci u sistemskoj memoriji, ovisno o njihovoj primjeni, grupirani su u određena memorijska područja s identifikatorima područja :

- procesna slika ulaza I
- procesna slika izlaza Q
- bitovna memorija M
- podatkovni blokovi DB
- lokalna memorija L.



Slika 5.3: Arhitektura memorije PLC uređaja

Arhitektura memorijskih lokacija PLC uređaja prikazana je na slici 5.3. Svaki podatak koji je upisan u memoriju mora biti dostupan programeru kako bi ga mogao pozvati u programskom kodu. Također, svaki podatak zapisan u memoriju mora imati vlastitu memorijsku lokaciju i adresu kojom se pristupa toj lokaciji. Kako bi svaka memorijska lokacija bila jedinstvena, njezina adresa sastoji se od sljedećih podataka:

- identifikatora memorijskog područja
- vrste podatka kojoj se pristupa
- bajt adresa podatka.

Tablica 5.2: Memorijska područja, vrste podataka i identifikatori

Memorijsko područje	Vrsta podatka		Identifikator
	hrvatski	engleski	
Procesna slika ulaza	Ulazni bit	(Input BIT)	I
	Ulazni bajt	(Input BYTE)	IB
	Ulazna riječ	(Input WORD)	IW
	Ulazna dvostruka riječ	(Input DWORD)	ID
Procesna slika izlaza	Izlazni bit	(Output BIT)	Q
	Izlazni bajt	(Output BYTE)	QB
	Izlazni riječ	(Output WORD)	QW
	Izlazni dvostruka riječ	(Output DWORD)	QD
Bitovna memorija	Memorijski bit	(Memory BIT)	M
	Memorijski bajt	(Memory BYTE)	MB
	Memorijska riječ	(Memory WORD)	MW
	Memorijska dvostruka riječ	(Memory DWORD)	MD
Podatkovni blok	Podatkovni bit	(Data BIT)	DBX
	Podatkovni bajt	(Data BYTE)	DBB
	Podatkovna riječ	(Data WORD)	DBW
	Podatkovna dvostruka riječ	(Data DWORD)	DBD
Lokalna memorija	Lokalni bit	(Local BIT)	L
	Lokalni bajt	(Local BYTE)	LB
	Lokalna riječ	(Local WORD)	LW
	Lokalna dvostruka riječ	(Local DWORD)	LD

Tablica 5.2 prikazuje memorijske lokacije, vrste podataka i identifikatore koje PLC koristi. Podatku se pristupa tako da se prvo upiše identifikator memorijskog područja, zatim vrsta podatka kojoj se pristupa i na posljetku bajt adresa podatka. Ako bi se željelo pristupiti podatku u bitovnoj memoriji koji je vrste dvostruka riječ i započinje šestim

bajtom, odgovarajuća adresa podatka bila bi %MD6. Isti način adresiranja koristi se i za bajt i riječ vrste podataka. Ako bi se pak željelo pristupiti podatku u procesnoj slici ulaza koji je vrste bit, započinje nultim bajtom te se nalazi na trećem mjestu u bajtu, odgovarajuća adresa podatka bila bi %I0.2. Mjesta rezervirana za podatke u memoriji raspoređena su u bajtovnom obliku. Zbog toga za vrste podataka tipa *bajt* (veličine 1 bajt), *riječ* (veličine 2 bajta) i *dvostruka riječ* (veličine 4 bajta) potrebno je definirati samo bajtovnu (dekadsku) vrijednost adrese budući da će one zauzeti jedno, dva ili četiri cjelokupna bajtovna mjesta. Svako bajtovno mjesto memorije sastoji se od 8 bitova, pa je stoga kod vrste podataka tipa bit potrebno uz bajtovnu vrijednost upisati i bitovnu (decimalnu) vrijednost adrese.

Maksimalno vrijeme izvođenja jednog ciklusa postavljeno je automatski na 150 ms. Ako ciklus premaši zadano vrijeme, PLC javi grešku (eng. *error mode*) i onemogućiti izvršavanje koda. Kako bi se PLC vratio u radni način (eng. *RUN mode*) potrebno je pritisnuti tipku RUN na PLC-u ili pomoću softvera TIA Portal u kojemu se piše programski kod prebaciti PLC u radni način. Kako bi se PLC pomoću TIA Portala prebacio u radni način, računalo na kojemu je upaljen softver i otvoren programski kod mora biti ethernet kablom (RJ45 – RJ45) povezano s PLC uređajem.

5.1.2 SINAMICS V90 PN servo upravljački sustavi

V90 servo upravljački sustav je inteligentni sustav upravljanja S-1FL6 serijom servo motora. Sveukupno 7 V90 PN servo upravljačkih sustava povezano je PROFINET komunikacijom u serijsku vezu s PLC uređajem. Jedan V90 servo upravljački sustav odgovoran je za jednu motoriziranu os stroja, odnosno jedan V90 servo upravljački sustav pogoni samo jedan SIMOTICS S-1FL6 motor. V90 servo upravljački sustav nova je generacija SIEMENS-ovih servo upravljačkih sustava koja na sebe preuzima jedan dio programskih funkcija. Sustav je sposoban na temelju parametara poslanih s PLC uređaja preko PROFINET komunikacije samostalno odraditi određene funkcije, poput referenciranja, držanja konstantne brzine ili preciznog pozicioniranja. Željene parametre, npr. apsolutnu poziciju u definiranoj mjernoj jedinici, potrebno je poslati u V90 servo upravljački sustav i on će na temelju parametara zapisanih u svoju ROM memoriju željenu os pozicionirati s preciznošću od maksimalno 0.001 mm. Cjelokupno pozicioniranje V90 preuzima na sebe, PLC uređaj mu samo šalje ulazne parametre i s njega prima izlazne parametre poput trenutne brzine, trenutne pozicije, eventualnih grešaka i drugo. Prije nego

što se V90 servo upravljački sustav pusti u pogon potrebno ga je konfigurirati što je objašnjeno u poglavlju 6.

5.1.3 HMI uređaj

HMI (eng. *Human Machine Interface*) je uređaj pomoću kojeg se odvija interakcija korisnika sa strojem i vizualizira proces zapisan u kod PLC uređaja. Na stroju N1 koristi se *SIMATIC TP15000 Comfort Panel*, tankoslojni tranzistorski zaslon s tekućim kristalima veličine 15“ (eng. *TFT LCD*) osjetljiv na dodir. Rezolucija ekrana je 1280x800 piksela s mogućnošću prikaza 16.777.216 različitih boja. Ekran se spaja na 24 V DC, a radni napon smije mu varirati između 19.2 i 28.8 volta. Korisnički podatci mogu se zapisivati direktno u memoriju HMI uređaja i za to je predviđeno 24 MB interne memorije.

5.1.4 EWM Tetrix BUSINT X11 modul

EWM Tetrix BUSINT X11 je komunikacijski modul pomoću kojeg PLC uređaj komunicira s *EWM Tetrix 552 Synergic RC HW* aparatom za TIG zavarivanje. BUSINT pretvara ulazne signale koji mu dolaze s PLC uređaja i konvertira ih u oblik prikladan sustavu *EWM Tetrix 552 Synergic RC HW* aparata za TIG zavarivanje. On također procesne veličine s aparata za zavarivanje konvertira u oblik prikladan PLC uređaju i šalje ih na PLC uređaj.

5.1.5 Kabeli

U električnom ormaru nalaze se kabeli opisani u tablici 5.3.

Tablica 5.3: Popis kabela u upravljačkom ormaru

BROJ	NAZIV	SERIJSKI BROJ	KOM.
1.	SIEMENS signalni kabel za enkoder 10m		7
2.	SIEMENS kabel za napajanje 10m		7
3.	Signalni kabel s 15 žica	/	1
4.	Signalni kabel sa 17 žica	/	1
5.	Signalni kabel s 19 žica	/	1
6.	Profinet RJ45-RJ45 C5 FTP 0.5m	QBX 2005035	7

7.	Profinet RJ45-RJ45 C5 FTP 1m	QBX 2005031	3
8.	Profinet RJ45-RJ45 C5 FTP 3m	QBX 2005033	3
9.	EWM 7-polni komunikacijski kabel	FRV 7POL 5 m	2

5.2 STL

STL je naziv za donji dio glavne konstrukcije stroja na kojem se nalazi rotacijska os i pneumatski konjić. Na njemu se nalazi jedan senzor i 8 aktuatora od čega jedan motor i 7 elektroventila. Motor rotira čeljusnu steznu glavu koja, uz pomoć pneumatskog konjića, drži radni komad u poziciji. Budući da kroz radni komad u zavarivačkom procesu mora prolaziti plin za oblikovanje (eng. *Forming gas*), motor se ne može nalaziti u centru čeljusne stezne glave jer je on rezerviran za cijev za protok plina za oblikovanje. Zbog toga je čeljusna stezna glava pričvršćena na ploču s planetarnim zupčaničkim prijenosnim omjerom 4/15. Na nju se nastavlja mehanički prijenos prijenosnog omjera 1/10 na kojeg se nastavlja još jedan mehanički prijenos prijenosnog omjera 1/40 te u konačnici motor koji pokreće rotaciju. Cjelokupni prijenosni omjer STL-a iznosi 1/1500, što znači da motor mora napraviti 1500 punih rotacija kako bi čeljusna stezna glava napravila jednu punu rotaciju.

5.2.1 Senzori

Senzor koji se nalazi na STL-u je mehaničko tipkalo s kotačićem koje služi kao referentna točka za postavljanje pozicije rotacijske glave. Na rotacijskoj glavi se nalazi okidač koji pomoću kotačića na tipkalu pritisne tipkalo i okine senzor koji PLC-u pošalje signal.

5.2.2 Aktuatori

Od aktuatora na PLC-u nalazi se jedan motor *SIMATIC1FL60342AF211AA1* koji pogoni rotacijsku glavu i 6 elektroventila koji služe za puštanje plina za formiranje unutar radnog komada. Tri ventila služe za puštanje plina kroz čeljusnu glavu, a tri ventila za puštanje plina kroz pneumatski konjić.

5.3 SCAR

SCAR je naziv za glavnu konstrukciju stroja na kojoj se nalaze sve motorizirane linearne osi. Na njemu se nalazi 6 aktuatora i 12 senzora.

5.3.1 Senzori

Senzori na SCAR-u podijeljeni su u dvije skupine: mehanički i induktivni. Svi senzori služe kao krajnji prekidači i nalaze se na krajnjim pozicijama osi. Svaka os na sebi ima 2 senzora.

Mehanički senzori, kojih je 4, nalaze se na glavnim osima za određivanje krajnjih radnih pozicija stroja. Spojeni su na ulazni modul PLC uređaja i softverski je blokirana svaka kretnja u smjeru senzora kada je senzor aktiviran.

Osam induktivnih senzora nalaze se u unutrašnjosti osi oscilatora (4 komada, 2 po jednoj osi) i AVC osi (4 komada, 2 po jednoj osi). Konjići koji se nalaze na linearnim vodilicama unutar svake osi imaju po jedan duži vijak koji aktivira senzor.

5.3.2 Aktuatori

Aktuatori koji se nalaze na SCAR-u su motori, od čega 2 *SIMATIC IFL60342AF211AA1* koji pogone glavne osi i 4 *SIMATIC IFL60242AF211AA1* motora koji pogone 4 pomoćne osi.

5.4 Daljinski upravljač

Daljinski upravljač povezan je 5 metarskim kabelom s upravljačkim ormarom. Na njemu se nalazi 8 normalno otvorenih ili NO (eng. *Normally open*) tipkala, 1 normalno zatvoreno ili NC (eng. *Normally closed*) tipkalo, 2 dvopozicijska prekidača, 1 tropozicijski prekidač i 1 komandna palica:

1. „*Fast/Slow Mode*“ dvopozicijski prekidač služi za prebacivanje između brzog i sporog načina u manualnom načinu rada.
2. „*SCAR/Torch 1/Torch 2*“ tropozicijski prekidač služi da odabir linearnih osi koje će se pomicati pomoću komandne palice:
 - a. lijeva pozicija - glavne osi se upravljaju pomoću komandne palice
 - b. srednja pozicija - pomoćne osi Gorionika 1 se upravljaju pomoću komandne palice

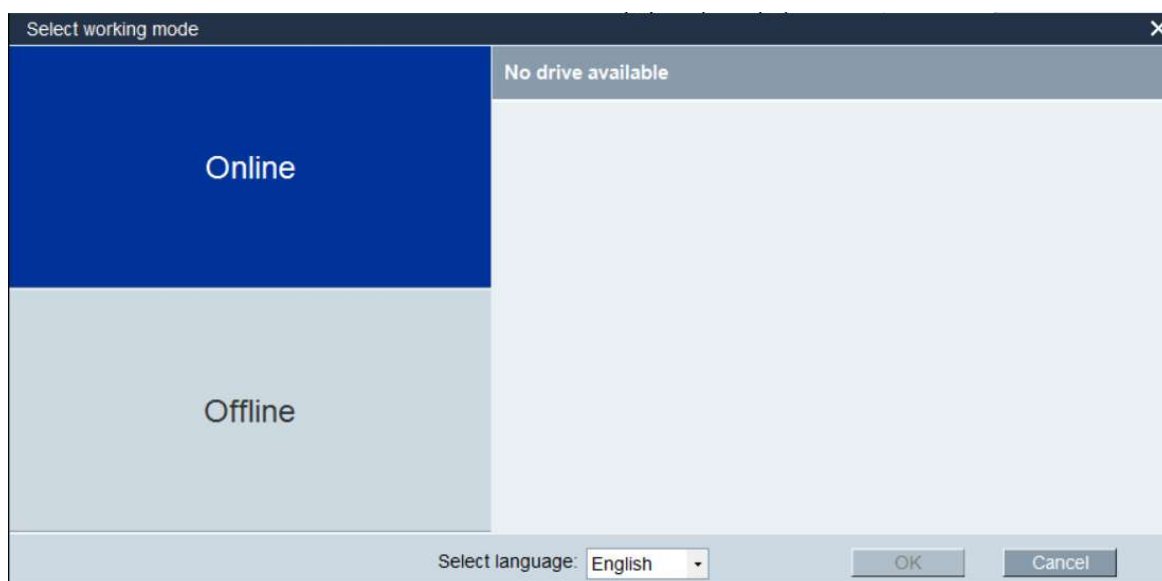
c. desna pozicija - pomoćne osi Gorionika 2 se upravljaju pomoću komandne palice.

3. „*Komandna palica*“ služi za pomicanje odabrane osi i moguće ga je pomicati u 4 smjera.
4. „*STORING*“ NO tipkalo služi za spremanje radne pozicije kada se stroj nalazi u ručnom načinu rada.
5. „*Auto/Man*“ dvopozicijski prekidač služi za odabir automatskog (pozicija 0) i ručnog (pozicija 1) načina rada.
6. „*Speed -*“ NO tipkalo, nije u upotrebi.
7. „*Speed +*“ NO tipkalo, nije u upotrebi.
8. „*Current -*“ NO tipkalo, nije u upotrebi.
9. „*Current +*“ NO tipkalo, nije u upotrebi.
10. „*Wire -*“ NO tipkalo, nije u upotrebi
11. „*Wire +*“ NO tipkalo, nije u upotrebi.
12. „*START*“ NO tipkalo sa zelenom LED lampom služi za pokretanje automatskog ciklusa u automatskom načinu rada ili za pomicanje osi u manualnom modu uz komande s HMI uređaja.
13. „*STOP*“ NC tipkalo služi za zaustavljanje automatskog ciklusa.

6. KONFIGURACIJA V90 SERVO UPRAVLJAČKIH SUSTAVA

Kao što je već navedeno, SINAMICS V90 su pametni servo upravljački sustavi koji određene funkcije mogu izvršavati sami. No, kako bi V90 servo upravljački sustav svoje funkcije izvršavao što preciznije i u što kraćem roku potrebno ga je dobro konfigurirati. Za to se koristi SIEMENS-ov softver V-ASSISTANT [5, 6] koji se besplatno može preuzeti sa SIEMENS-ove stranice.

U V-ASSISTANT-u prvo se odabire način rada: ONLINE (povezani na servo upravljački sustav) ili OFFLINE načinu rada. Ako se odabere ONLINE način rada, softver učitava sve parametre s V90 servo upravljačkog sustava s kojim je povezan u novi projekt. Odabirom OFFLINE načina rada, korisnik bira između novog projekta ili nekog postojećeg projekta. Novi OFFLINE projekt sadrži generički zadane parametre i korisnik sam mora odabrati vrstu V90 servo upravljačkog sustava i motora spojenog na njega. Ako se odabere kriva vrsta sustava, projekt se neće moći povezati s fizičkim sustavom i spremiti parametre na njega. Početna stranica V-ASSISTANT softvera prikazana je na slici 6.1.

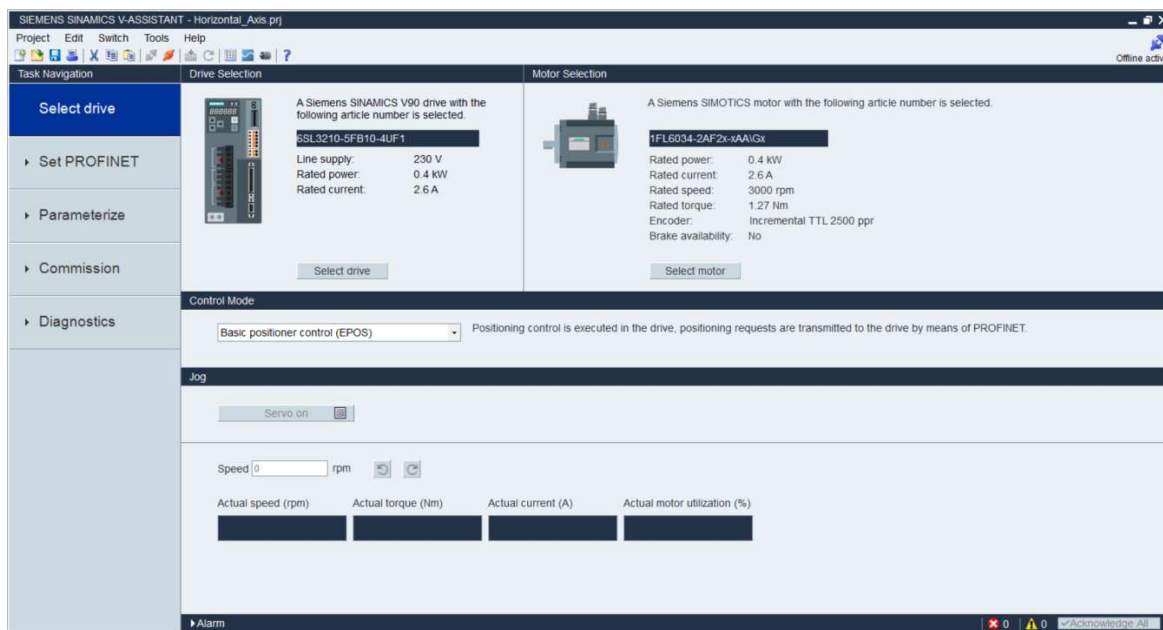


Slika 6.1: Početna stranica V-ASSISTANT-a

Kada se otvori projekt, u gornjem dijelu prozora nalazi se alatna traka koja je aktivna na svim karticama isto kao i traka grešaka i upozorenja na donjem dijelu stranice.

Na početnoj stranici nalazi se 5 kartica, što je prikazano na slici 6.2. Na prvoj kartici „Odabir sustava“ (eng. *Select drive*) odabire se vrsta V90 servo upravljačkog sustava i vrstu motora koji je povezan na sustav. Također, ovdje se odabire i jedan od dva moguća načina rada: „EPOS“ ili „S“ način rada. „Osnovna kontrola pozicije“ (eng. *Basic Position*

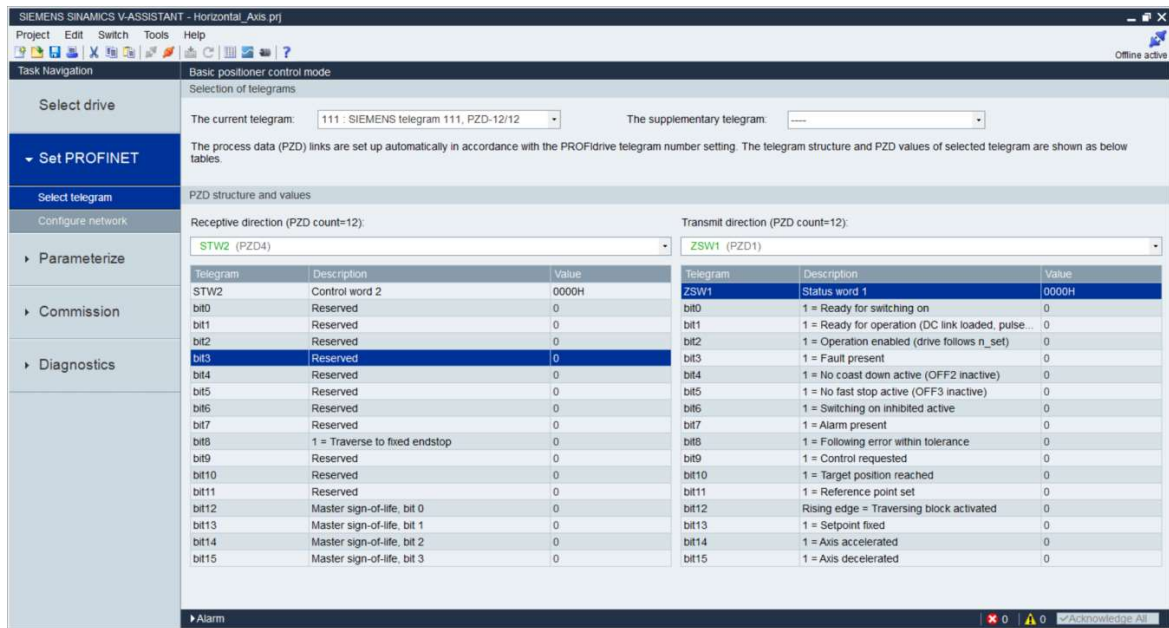
Control) ili „EPOS“ je način rada u kojemu sustav na sebe preuzima funkciju pozicioniranja. Drugi način rada je „Kontrola brzine“ (eng. *Speed Control*) ili „S“ način rada i u njemu sustav na sebe preuzima funkciju održavanja konstantne brzine. Treća funkcija na ovoj kartici je „Jog“, odnosno testiranje kretanja motora u obadva smjera i očitavanje trenutnih parametara motora.



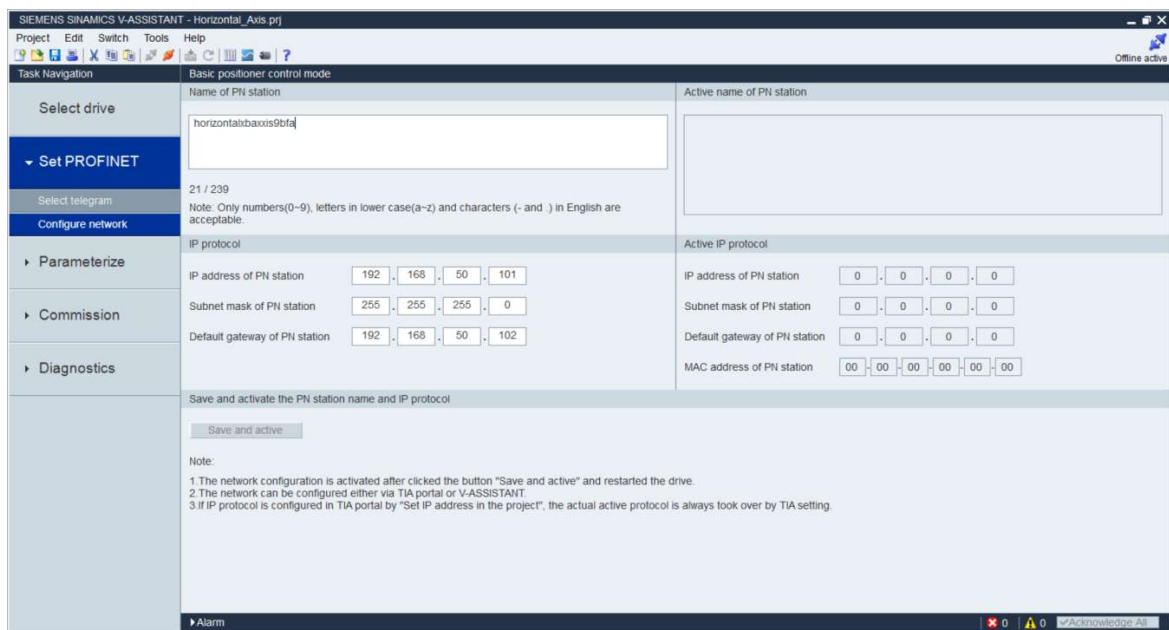
Slika 6.2: Početna stranica projekta u V-ASSISTANT-u

Druga kartica zove se „Postavi PROFINET“ (eng. *Set PROFINET*) i sastoji se od dvije potkartice. Prva potkartica „Odabir telegrama“ (eng. *Select telegramm*) prikazana je na slici 6.3 i služi za odabir vrste telegrama pomoću kojega se parametri s PLC uređaja šalju na V90 servo upravljački sustav i obratno. Telegrami su unaprijed definirani skupovi informacija od strane SIEMENS-a koji služe za komunikaciju PLC uređaja sa servo upravljačkim sustavima. U slučaju korištenja „EPOS“ načina rada odabiremo Telegram 111. Druga potkartica „Konfiguriraj mrežu“ (eng. *Configure Network*) prikazana je na slici 6.4 i služi za postavljanje imena i IP adrese V90 servo upravljačkog sustava na PROFINET komunikaciji. Ime V90 servo upravljačkog sustava može se također postaviti i direktno iz TIA Portalu, no više se preporuča je to raditi na ovaj način. Ako se ime definira u TIA Portalu, ono se preko PROFINET komunikacije šalje iz projekta na računalu direktno na sustav pa postoji mogućnost da se željeno ime pošalje na krivu adresu, odnosno krivi sustav. Ako se ime postavlja u projektu V-ASSISTANT-a, onda se fizički mora spojiti na

željeni V90 servo upravljački sustav kako bi se podatci spremili direktno u njega čime se smanjuje mogućnost pogreške.



Slika 6.3: Postavljanje telegrama u V-ASSISTANT-u



Slika 6.4: Postavljanje IP adrese u V-ASSISTANT-u

Treća kartica „Parametriziraj“ (eng. *Parameterize*) sastoji se od sedam potkartica na kojima se nalaze parametri koji su spremljeni u memoriji V90 servo upravljačkog sustava. Na prvoj stranici „Postavi mehanizam“ (eng. *Set mechanism*) postavlja se vrsta fizičkog

mehanizma kojeg motor pokreće. Parametri poput vrste prijenosa, prijenosnog omjera i , u ovom slučaju, koraka navojne šipke u dužinskim jedinicama (eng. *Length Units*) bitni su za određivanje pozicije u kojoj se motor trenutno nalazi. Prema unesenim podacima V90 servo upravljački sustav računa konverzijski faktor za enkoder pomoću kojeg izračunava svoju točnu poziciju u prostoru. Dužinske jedinice posebnost su ovog softvera. Ako je na motor spojena navojna šipka s korakom od 10 mm po jednom okretu, a korisnik želi postići preciznost od 1 μm , u softver se unosi 10000 (parametar p29247). To znači da jedan okret navojne šipke rezultira pomakom od 10000 dužinskih jedinica. Stranica parametriziranja prikazana je na slici 6.5.



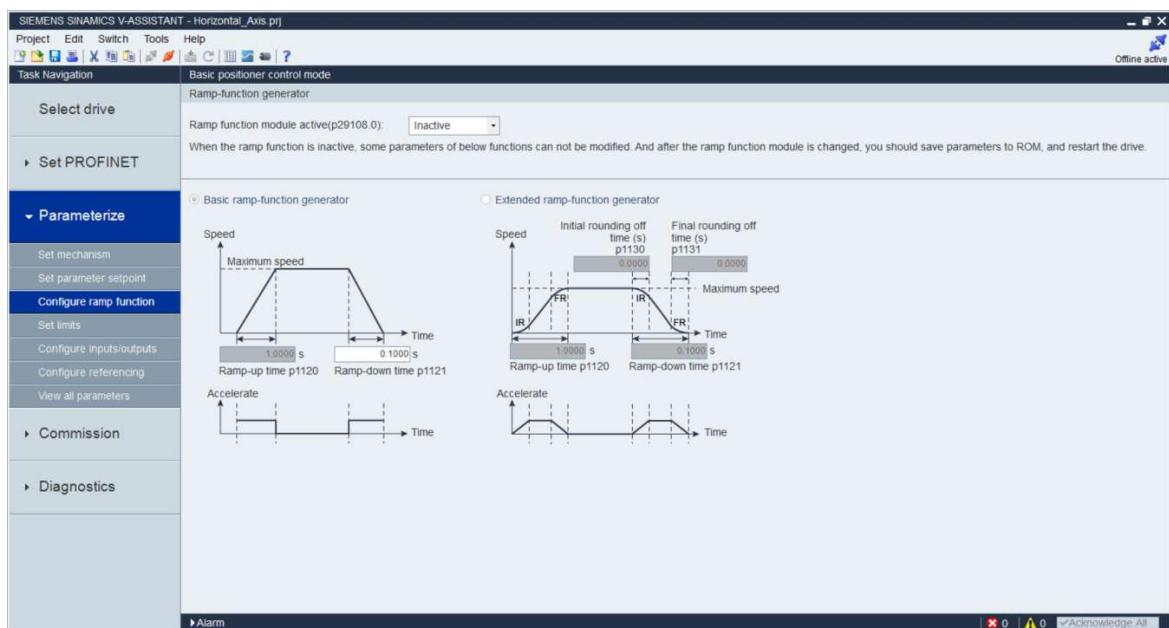
Slika 6.5 Mehanički parametri V90 servo upravljačkog sustava

Sljedeća kartica koja se koristi pri konfiguriranju V90 servo upravljačkih sustava je „Konfiguriranje funkcije rampe“ (eng. *Configure Ramp Functions*). Na njoj se određuju parametri funkcije rampe za postizanja zadane brzine motora. Određuju se vremena potrebna da motor ubrza na maksimalnu brzinu i uspori s maksimalne brzine na nulu. U V90 servo upravljačkom sustavu moguće je aktivirati dvije vrste rampi:

- generator obične rampe (eng. *Basic ramp-function generator*) i
- generator produžene rampe (eng. *Extended ramp-function generator*).

Generator obične rampe je standardni način u kojemu se definira vrijeme dizanja i vrijeme spužtanja rampe na zadanu brzinu. Generator produžene rampe je način u kojemu

se, uz standardna vremena dizanja i spuštanja rampe za zadanu brzinu, definiraju i vremena spuštanja i dizanja rampe za akceleraciju. Time se dobivaju blaži prijelazi između konstantnih brzina i rampi. Kod malih brzina korištenje generatora produžene rampe za postizanje zadane brzine nije potrebno. No kod velikih brzina generator produžene rampe je preporučljivo koristiti jer se unosi manje vibracija u sustav i produžuje se životni vijek motora i njegove popratne opreme (navojnih šipki, trapezних vretena, zupčanika, matica i drugo). Ako bi korisnik bez rampe želio ubrzati na neku veću brzinu, postoji velika mogućnost da se napravi materijalna šteta na sustavu poput trganja motora, zupčanika, navojne šipke i raznih prihvata. Konfiguriranje generatora rampe prikazano je na slici 6.5.

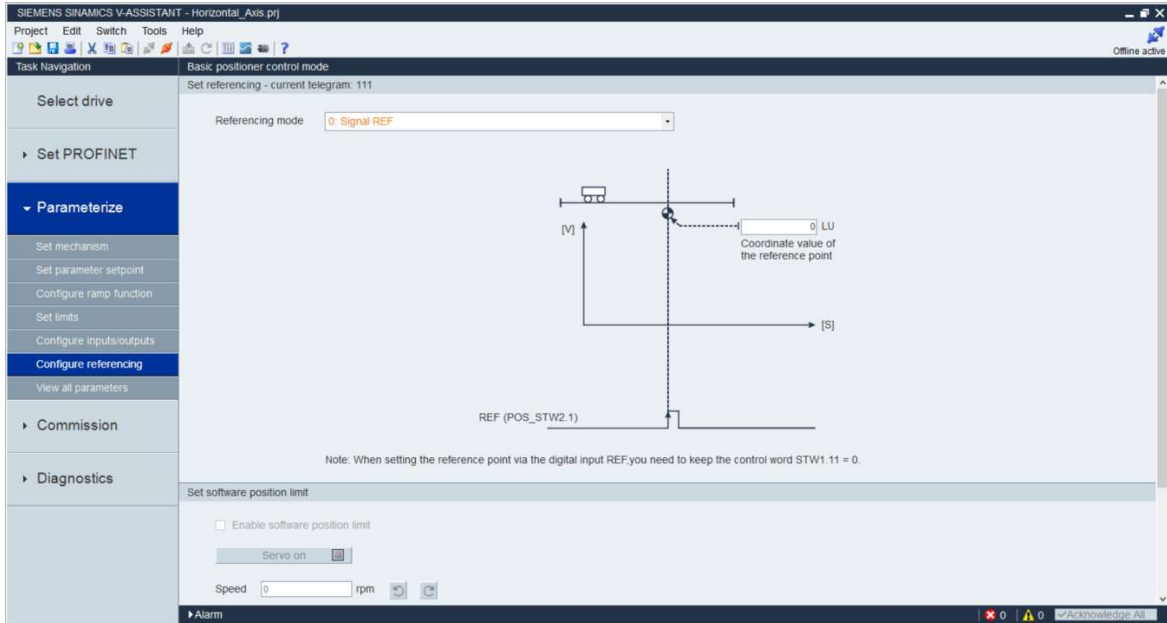


Slika 6.6: Parametri generatora rampe V90 servo upravljačkog sustava

Na slici 6.6 prikazana je treća kartica koja se zove „Konfiguriranje referenciranja“ (eng. *Configure referencing*). Na njoj se odabire jedan od tri moguća načina referenciranja motora:

- referentni signal (eng. *Signal REF*)
- referentni signal i nul-oznaka enkodera (eng. *Reference cam (signal REF) and encoder zero mark*)
- samo nul-oznaka enkodera (eng. *Encoder zero mark only*).

U svim V90 servo upravljačkim sustavima na stroju N1 korišten je način referenciranja „Referentni signal“ u kojemu PLC šalje informaciju servo upravljačkom sustavu kada se treba referencirati.



Slika 6.6: Parametri referenciranja V90 servo upravljačkog sustava

Nakon što se definiraju svi potrebni parametri, na kartici „Pogledaj sve parametre“ (eng. *View all parameters*) korisnik može provjeriti jesu li svi parametri dobro uneseni.

Group filter	All Parameter	Find	Factory default	Save changes					
Group	Parameter No	Name	Value	Unit	Range	Factory setting	Effect type		
App	p29000	Motor ID	54	N.A.	[0, 65535]	0	immediately		
App	p29001	Reversal of motor direction	0 : Direction normal	N.A.	--	0	immediately		
App	p29002	BOP display selection	0 : Speed	N.A.	--	0	immediately		
App	p29003	Control mode	1 : EPOS	N.A.	--	2	reset		
App	p29005	Brake resistor capacity percentage alar...	100.0000	%	[1, 100]	100.0000	immediately		
App	p29006	Line supply voltage	230	V	[200, 480]	400	immediately		
App	p29020[0]	Tuning: Dynamic factor : One-button tuni...	18	N.A.	[1, 35]	18	immediately		
App	p29021	Tuning: Mode selection	0 : Disable	N.A.	--	0	immediately		
App	p29022	Tuning: Ratio of total inertia moment to ...	1.0000	N.A.	[1, 10000]	1.0000	immediately		
App	p29023	Tuning: One-button auto tuning configur...	0007H	N.A.	--	0007H	immediately		
App	p29024	Tuning: Real-time auto tuning configurati...	004CH	N.A.	--	004CH	immediately		
App	p29025	Tuning: Configuration overall	001CH	N.A.	--	0004H	immediately		
App	p29026	Tuning: Test signal duration	2000	ms	[0, 5000]	2000	immediately		
App	p29027	Tuning: Limit rotation of motor	0	°	[0, 30000]	0	immediately		
App	p29028	Tuning: Pre-control time constant	7.5000	ms	[0, 60]	7.5000	immediately		
App	p29035	VIBSUP activation	0 : Disable	N.A.	--	0	immediately		
App	p29050[0]	Torque limit upper : Torque limit upper 0	300.0000	%	[-150, 300]	300.0000	immediately		
App	p29051[0]	Torque limit lower : Torque limit lower 0	-300.0000	%	[-300, 150]	-300.0000	immediately		
App	p29070[0]	Speed limit positive : Speed limit positiv...	5000.0000	rpm	[0, 210000]	210000.0000	immediately		
App	p29071[0]	Speed limit negative : Speed limit negati...	-5000.0000	rpm	[-210000, 0]	-210000.0000	immediately		
App	p29080	Overload threshold for output signal trng	100.0000	%	[10, 300]	100.0000	immediately		
App	p29108	Function module activate	00000000H	N.A.	--	00000000H	immediately		
App	p29110	Position loop gain	1.8000	1000/min	[0, 300]	1.8000	immediately		
App	p29111	Speed pre-control factor (feed forward)	0.0000	%	[0, 200]	0.0000	immediately		
App	p29120	Speed loop gain	0.0140	Nms/rad	[0, 999999]	0.3000	immediately		
App	p29121	Speed loop integral time	15.0000	ms	[0, 100000]	15.0000	immediately		
App	p29150	User defined PZD receive	0 : No function	N.A.	--	0	immediately		
App	p29151	User defined PZD send	0 : No function	N.A.	--	0	immediately		
App	p29230	MDI direction selection	0 : MDI shortest di...	N.A.	--	0	immediately		

Slika 6.7: Popis svih parametara V90 servo upravljačkog sustava

Također, na ovoj kartici nalaze se i parametri koji nisu svrstani niti u jednu prethodnu karticu, poput parametra p29001 koji određuje smjer motora. Slika 6.7 prikazuje sve parametre V90 servo upravljačkog sustava.

Kada je korisnik siguran da su svi parametri dobro uneseni, mora ih spremi na ROM memoriju V90 servo upravljačkog sustava. Ako je projekt rađen u OFFLINE načinu rada, na alatnoj traci nalazi se narančasta ikona kojom se povezuje na V90 servo upravljački sustav. Nakon što je V-ASSISTANT povezan s fizičkim V90 servo upravljačkim sustavom, pokraj ikone za povezivanje na sustav nalazi se ikona za spremanje parametara na ROM sustava. Druga opcija je u alatnoj traci odabrati „Alati“ > „Spremi parametre u ROM“ (eng. *Tools>Save Parameters to ROM*).

Na kartici „Dijagnostika“ (eng. *Diagnostics*) korisnik može provjeriti ponašanje motora u radu i pročitati trenutne vrijednosti parametara motora. Ako utvrdi da su neki parametri podešeni krivo, ponavlja proces parametriziranja V90 servo upravljačkog sustava sve dok ne bude zadovoljan s rezultatima. Dijagnostika je prikazana na slici 6.8.

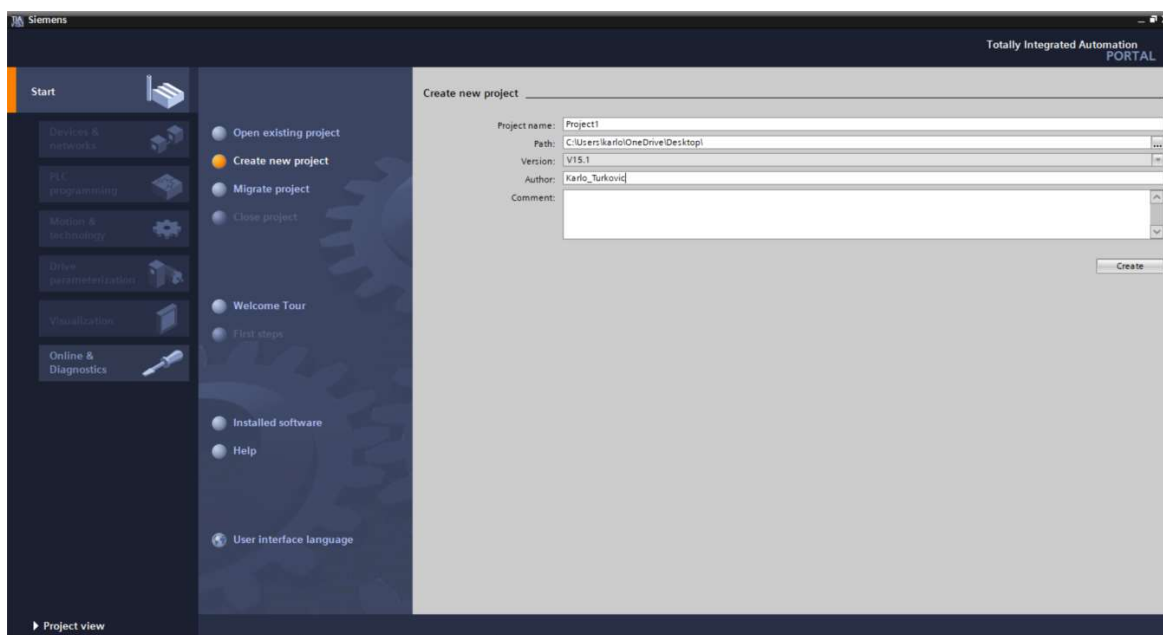


Slika 6.8: Dijagnosticiranje V90 servo upravljačkih sustava

V-ASSISTANT ima dodatnih mogućnosti koje nisu korištene pri izradi ovog projekta pa sukladno tome nisu ni opisane u ovom poglavlju

7. PROGRAMSKO RJEŠENJE PLC UREĐAJA

Cjelokupno programsko rješenja napravljeno je u softveru TIA Portal V15.1 [7]. To je softver koji objedinjuje programsko rješenje i vizualizaciju programa. U prošlosti SIEMENS je koristio softver STEP 7 za programsko rješenje i WinCC za vizualizaciju koji su kasnije objedinjeni u softveru TIA Portal. Početna stranica TIA Portala prikazana je na slici 7.1.

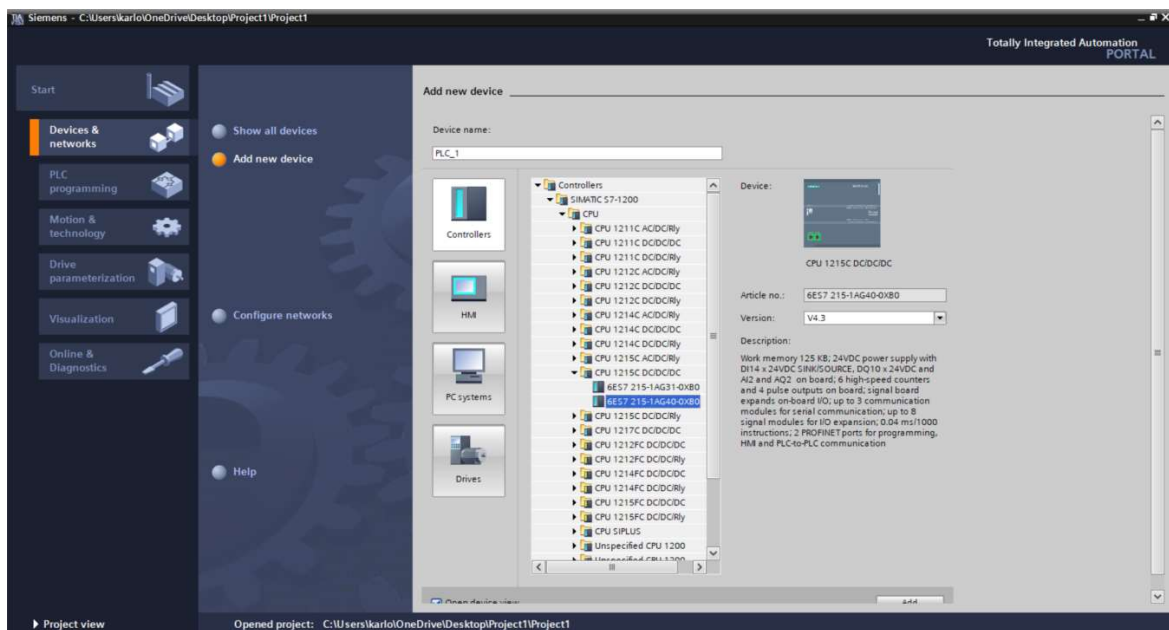


Slika 7.1: Početna stranica programa TIA Portal

Kada se otvori TIA Portal, korisnik može birati neki od postojećih projekata koje je već radio ili napraviti potpuno novi projekt. Odabere li korisnik novi projekt, potrebno je dodati sve SIEMENS-ove uređaje koji se koriste u projektu, a dostupni su u izborniku softvera. To se, naravno, odnosi samo na opremu koja se može programirati ili konfigurirati, tj. svi moduli s kojima PLC može komunicirati. Oprema se može dodati i naknadno, no jednostavnije je taj dio odraditi na samome početku projekta. Sklopnici, releji i ostala električna oprema se ne dodaje u softver. U ovom slučaju to su PLC, HMI, V90 servo upravljački sustavi, komunikacijski modul barkod čitača i komunikacijski moduli EWM aparata za zavarivanje. Za početak u projekt su dodani samo PLC i HMI jer su oni trenutno jedini dostupni u izborniku.

U ovoj verziji softvera SINAMICS V90 PN V1.0 servo upravljački sustavi koji se koriste nisu dodati u izbornik pa ih se mora dodati ručno pomoću GSD ili GSDML

datoteka. To su vrste datoteka koje unose vrstu i broj ulazno-izlaznih jedinica hardverskih uređaja u TIA Portal projekt. U suštini GSD i GSDML datoteke definiraju nepoznati uređaj u TIA Portalu i tako omogućavaju komunikaciju između PLC uređaja i dodanog uređaja. Uglavnom se koriste za definiranje uređaja od drugih proizvođača, no u ovom slučaju vidljivo je kako su nekada potrebni i za definiranje opreme od samog SIEMENS-a. Komunikacijski modul barkod čitača i EWM aparata za zavarivanje također se unosi u projekt pomoću GSD ili GSDML datoteka. Na slici 7.2 prikazano je dodavanje modula u projekt.



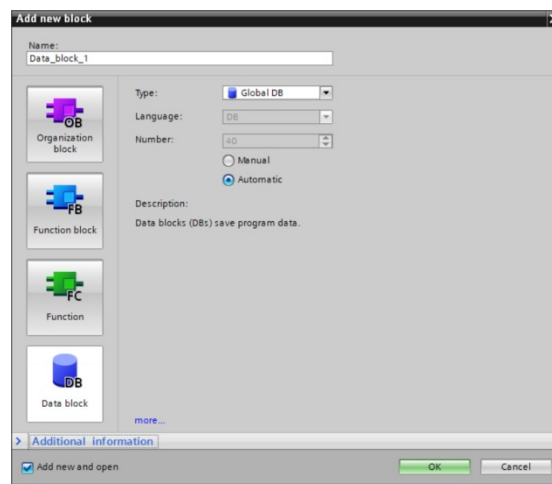
Slika 7.2 Dodavanje uređaja u projekt

Kada su svi uređaji dodani u projekt potrebno ih je konfigurirati. Svi uređaji nalaze se u glavnom izborniku projekta na lijevoj strani ekrana pod karticom Uređaji i mreže (eng. *Devices & Networks*). Vrsta komunikacije između PLC uređaja i V90 servo upravljačkog sustava je PROFINET komunikacija, što se vidi i iz kraticе PN u imenu SINAMICS V90 PN V1.0. Tu vrstu komunikacije potrebno je dodatno definirati postavljanjem IP adrese i povezivanjem svih uređaja u mrežu. Svaki uređaj mora imati jedinstvenu IP adresu kako bi mogao primiti i slati informacije PROFINET komunikacijom. Svi uređaji koji su fizički međusobno povezani PROFINET komunikacijom moraju biti povezani u istu mrežu i u projektu. U ovome projektu ta mreža generički je nazvana PN/IE_1 i označena je zelenom bojom koja je standarda boja prikaza PROFINET komunikacije.

Druga stvar koju je potrebno definirati je način cikličke izmjene podataka između PLC uređaja i V90 servo upravljačkih sustava. Načini cikličke izmjene kod SIEMENS-a nazivaju se telegrami (eng. *Telegramm*) i to su predefinirani skupovi kodnih riječi koji se šalju između PLC uređaja i V90 servo upravljačkih sustava. Telegrami korisnik odabire ovisno o funkciji koju želi dodijeliti V90 servo upravljačkom sustavu. U ovome programskom rješenju svi V90 servo upravljački sustavi koriste se za pozicioniranje pomoću „*SinaPos*“ [4] naredbe za što se koristi telegram 111. Ako bi se V90 servo upravljački sustav koristio za drugu namjenu, na primjer drugi način pozicioniranja ili za održavanje konstantne brzine, telegram 111 morao bi se zamijeniti s telegramom koji odgovara odabranom načinu rada.

Na slici 7.3 prikazane su četiri vrste blokova koje je moguće odabrati u TIA Portalu:

- organizacijski blokovi (eng. *Organization block*) koji tvore vezu između operativnog sustava i upravljačkog programa, odnosno blokovi koji se ciklički pozivaju u operativni sustav
- funkcijski blokovi (eng. *Function block*) su blokovi koji trajno spremaju svoje vrijednosti u posebne podatkovne blokove koji su im predodređeni (eng. *instance data blocks*) pa podatci ostaju dostupni i nakon što se blok izvrši
- funkcije (eng. *Function*) su blokovi bez „memorije“ jer svoje vrijednosti ne spremaju u predodređene podatkovne blokove, već im se sve vrijednosti moraju zadati kada se blok pozove
- podatkovni blokovi (eng. *Data block*) su blokovi koji sadrže vrijednosti koje se pozivaju u upravljački program.



Slika 7.3: Vrste blokova u TIA Portalu

Upravljački dio programskog rješenja nalazi se zapisan u memoriji SIMATIC S1200 CPU1215 PLC uređaja. Kod se na PLC zapisuje u memorijsku lokaciju od 4 KB u kojoj u ovom slučaju zauzima 25% prostora pa se može zaključiti kako je veličina koda na PLC-u 1 KB. Slika 7.4 prikazuje raspored memorijskih lokacija unutar PLC uređaja.

Name	Objects	Load memory	Work memory	Retain memory	I/O	DI	DO	AI	AQ
		25 %	45 %	60 %	95 %	63 %	25 %	23 %	
	Total:	4 MB	128000 bytes	10240 bytes	Configured:	40	16	114	112
	Used:	1066891 bytes	57428 bytes	6154 bytes	Used:	38	10	28	26
6	OB	72360 bytes	4230 bytes						
7	FC	642213 bytes	24740 bytes						
8	FB	166917 bytes	14782 bytes						
9	DB	142907 bytes	13686 bytes	6154 bytes					
10	Objects for Motion Technology			0 bytes					
11	Data types	32376 bytes							
12	PLC tags	10118 bytes							

Slika 7.4: Raspored memorije u projektu

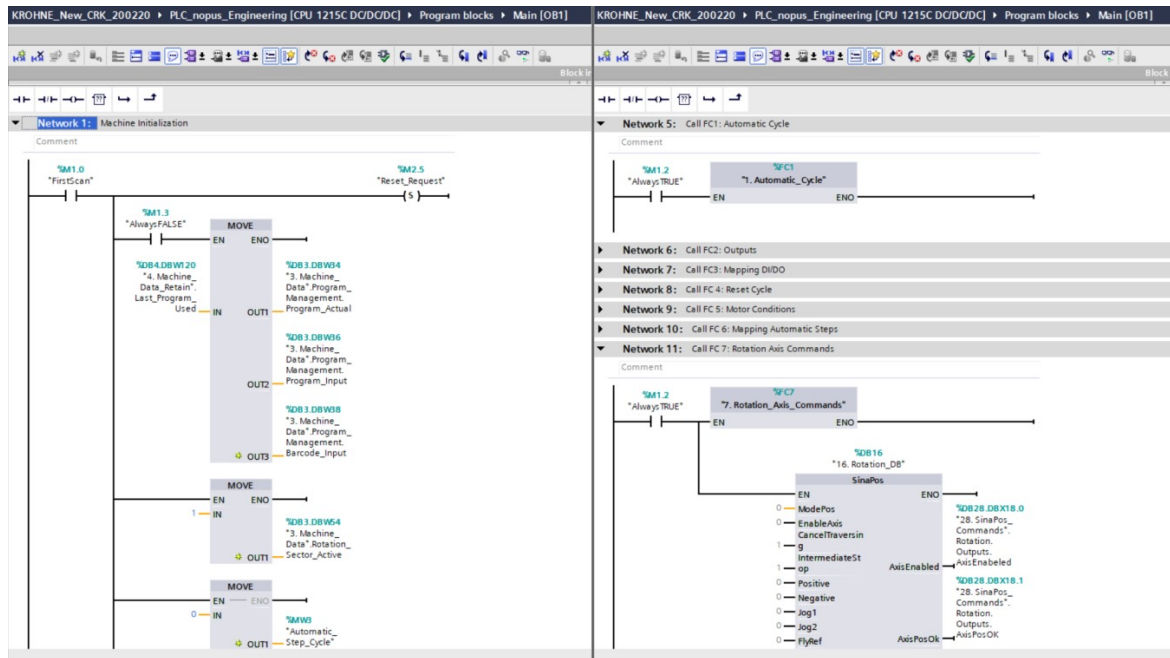
7.1 Organizacijski blokovi

Organizacijski blokovi imaju predodređene funkcije i dodaju se u program po potrebi, osim glavnog organizacijskog bloka „Main [OBI]“ koji služi za pozivanje svih drugih blokova programskog rješenja. Glavni organizacijski blok „Main [OBI]“ dodan je automatski u svaki projekt. U ovom projektu koriste se dva organizacijska bloka.

7.1.1 Glavni organizacijski blok

Prvi blok je organizacijski blok „Main [OBI]“ koji je temeljni blok svakog PLC programa jer se u njemu pozivaju svi drugi blokovi. Blok „Main [OBI]“ izvršava se ciklički u PLC-u i redom poziva sve blokove koji se nalaze u njemu. Svi drugi blokovi koji nisu organizacijski ne mogu se samostalno izvesti, nego uvijek moraju biti pozvani od strane nekog organizacijskog bloka. Prva stvar koju blok „Main [OBI]“ odrađuje je inicijalizacijski korak koji se odvija samo kada je aktivna varijabla „First_Scan“, odnosno samo u prvom prolazu kroz kod. U inicijalizacijskom koraku u memoriju PLC uređaja zapisuju se osnovne varijable koje su PLC-u potrebne za početak rada kao što je broj zadnjeg programa koji se koristio kako bi korisnik mogao nastaviti gdje je stao prethodni put. Zatim se sve varijable koraka za izvođenje programa poput „ASC“ i „RSC“ inicijaliziraju na vrijednost 0 kako bi se osiguralo da se automatski ciklus i procedura za reset krenu izvršavati od početka. Također, upisuju se vrijednosti 0 u sve dnevne brojače

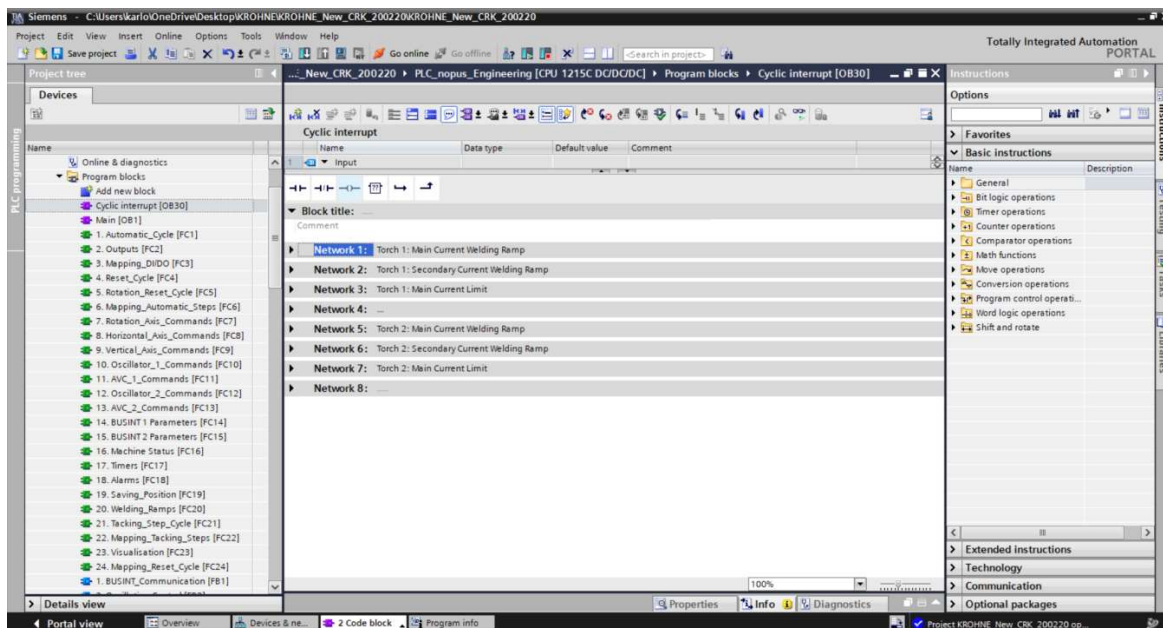
završenih automatskih ciklusa. Nadalje, u „Main[OB1]“ nalazi se 7 „SinaPos“ blokova koji komuniciraju s V90 servo upravljačkim sustavima. Kao zadnji ulazni parametar u „SinaPos“ naredbu upisuje se hardverski broj V90 servo upravljačkog sustava za kojeg je određeni „SinaPos“ blok zadužen. Time se osigurava da svaki „SinaPos“ blok upravlja samo jednim V90 servo upravljačkim sustavom, odnosno samo jednom osi. Slika 7.5 prikazuje pozivanje funkcija unutar glavnog organizacijskog bloka „Main [OB1]“.



Slika 7.5: Organizacijski blok Main

7.1.2 Ciklički prekid

Drugi blok koji se koristi je „Cyclic interrupt [OB30]“ i prikazan je na slici 7.6. On se ciklički poziva svakih 100 ms. Specifičnost ovog bloka je u tome što će se izvršiti bez obzira gdje se PLC nalazio u programskom kodu. Kada se javi zastavica (eng. *Flag*) koja signalizira da je proteklo 100 ms PLC stane s izvođenjem programskog koda gdje god da se nalazio unutar njega, izvrši organizacijski blok „Cyclic interrupt [OB30]“ te se vrati nazad u programski kod na isto mjesto u kojemu je prethodno stao i nastavi s izvršavanjem glavnog koda. Uloga cikličkog prekida u ovom programskom kodu koristi se za podizanje iznosa jakosti struje na aparatu za zavarivanje. Blok se ciklički izvrši svakih 100 ms i promijeni struju za iznos koji je definiran u funkciji „20. Welding_Ramps [FC20]“.



Slika 7.6: Organizacijski blok Cyclic interrupt [OB30]

7.2 Funkcije

Kao što je već objašnjeno, funkcije su blokovi bez predefiniране memorijske lokacije. Kada se poziva funkcija s njom se ne mora pozvati i podatkovni blok. To ne znači da se podatke iz funkcija ne može spremati u memoriju. Za spremanje podataka iz funkcija koriste se globalni podatkovni blokovi (eng. *Global Data Blocks*). Podatci zapisani u globalne podatkovne blokove pozivaju se direktno u programski kod pa nije potrebno definirati podatkovni blok kod pozivanja funkcije.

7.2.1 Automatski ciklus zavarivanja

Prva funkcija koja je definirana u programskom kodu je blok „*1. Automatic_Cycle [FC1]*“ u kojem se sekvencijski odvija automatski ciklus pozicioniranja stroja, rotacije radnog komada i zavarivanja. Blok „*1. Automatic Cycle [FC1]*“ sadržava 40 grana (eng. *Network*): 38 sekvencijskih grana koje izvedu automatski ciklus zavarivanja i 2 kontrolne grane. Sekvencijske grane kreću se izvršavati kada se na daljinskom upravljaču pritisne tipka „*START*“ dok je stroj u automatskom načinu rada. Grane od 1 do 38 izvršavaju se na način jedna nakon druge što se kontrolira pomoću kontrolne varijable „*ASC*“ koja je cjelobrojnog formata.

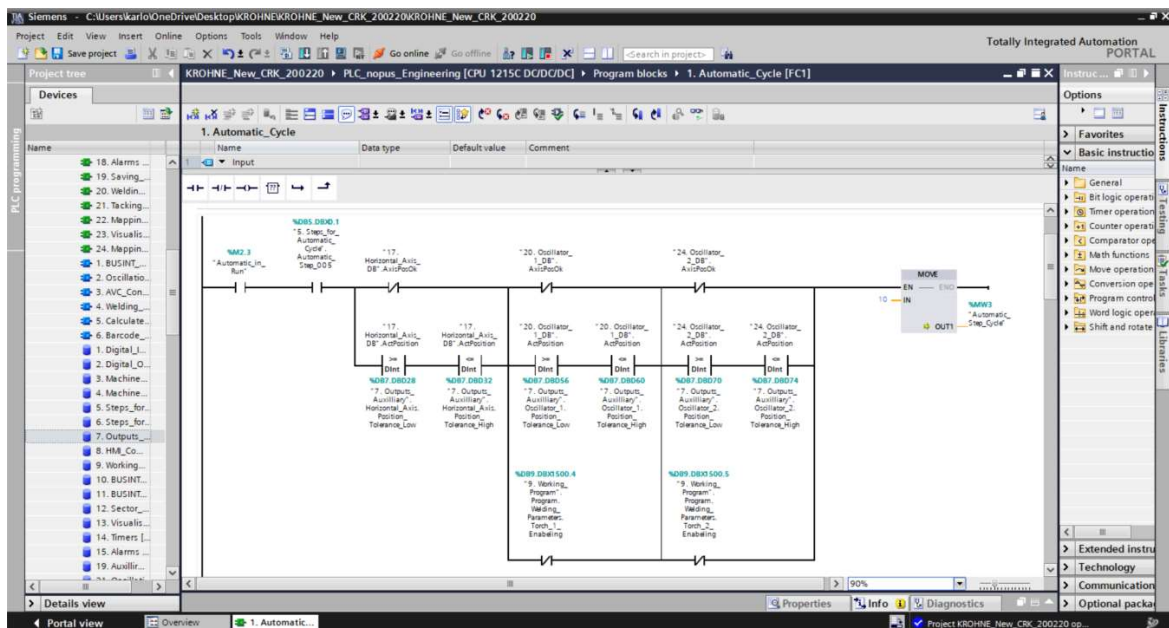
Svaka grana sastoji se od četiri vrste uvjeta:

- aktivacijskog uvjeta bloka
- ulaznog uvjeta
- srednjih uvjeta
- izlazna akcija.

Aktivacijski uvjet funkcije aktivira određenu funkciju stroja, u ovom slučaju automatski ciklus. On se kao zavojnica (eng. *Coil*) nalazi u bloku „16. *Machine_Status [FC16]*“. Kako bi se aktivacijski uvjet aktivirao potrebno je ispuniti određene preduvjete, što će se detaljnije opisati u poglavlju 7.2.8.

Ulazni uvjet uvijek je provjera vrijednosti kontrolne varijable. Ta ista kontrolna varijabla, ovisno o svojoj vrijednosti, aktivira funkcije stroja u drugim dijelovima programskog koda. Kako bi kontrolnu varijablu bilo lakše pozivati u programskom kodu, njezine vrijednosti mapirane su u bloku „6. *Mapping_Automatic_Steps [FC6]*“. Mapiranje je proces preslikavanja jedne varijable u drugu. Budući da je „*ASC*“ cjelobrojna varijabla, u programu bi se na svakom mjestu gdje se nešto želi aktivirati pomoću nje trebala obavljati provjera vrijednosti same varijable. Kako bi se izbjeglo prekomjerno provjeravanje vrijednosti varijable „*ASC*“, provjera se radi u funkcijskom bloku „6. *Mapping_Automatic_Steps [FC6]*“. Ako je provjera istinita, aktivira se varijabla logičkog tipa (eng. *Boolean*) pridružena vrijednosti „*ASC*“ koja se onda poziva u drugim blokovima programskog koda. Logička varijabla ponovno se provjerava na mjestima u kodu gdje je pozvana, no provjera logičke vrijednosti oduzima manje vremena od provjere cjelobrojne vrijednosti.

Funkcije stroja vraćaju povratnu informaciju nazad u „1. *Automatic Cycle [FC1]*“ preko srednjih uvjeta. Kada su ulazni uvjet i srednji uvjeti ispunjeni aktivira se izlazna akcija, odnosno povećanje kontrolne varijable. Povećanjem kontrolne varijable prestaje važiti ulazni uvjet u trenutnoj grani, a aktivira se ulazni uvjet u sljedećoj grani čime sekvenca prelazi na sljedeću granu. Također, povećanjem kontrolne varijable aktiviraju se funkcije stroja koje se trebaju aktivirati u tom koraku sekvence i vraćaju povratnu informaciju nazad u granu u obliku srednjih uvjeta. Ovakvim načinom programiranja omogućeno je kontrolirano izvođenje funkcija stroja i onemogućeno je da dvije grane budu aktivne u istodobno. Time se dobiva preciznost i redoslijed u izvođenju programskog koda.



Slika 7.8: Pozicioniranje horizontalnih osi u automatskom ciklusu zavarivanja

U grani NW2 prikazanoj na slici 7.8 odvija se pozicioniranje svih horizontalnih osi stroja. Prvi uvjet je aktivacijski uvjet bloka „Automatic_In_Run“ koji označava da je ciklus automatskog zavarivanja aktivan. Uvjet je logičkog tipa i aktivan je kroz cijeli automatski ciklus. Drugi uvjet je ulazni uvjet u granu NW2. Kada je varijabla „ASC“ jednaka 5, aktivan je i uvjet „5. ASC_DB“.005“ koji aktivira granu NW2. Nakon toga dolazi do grananja uvjeta.

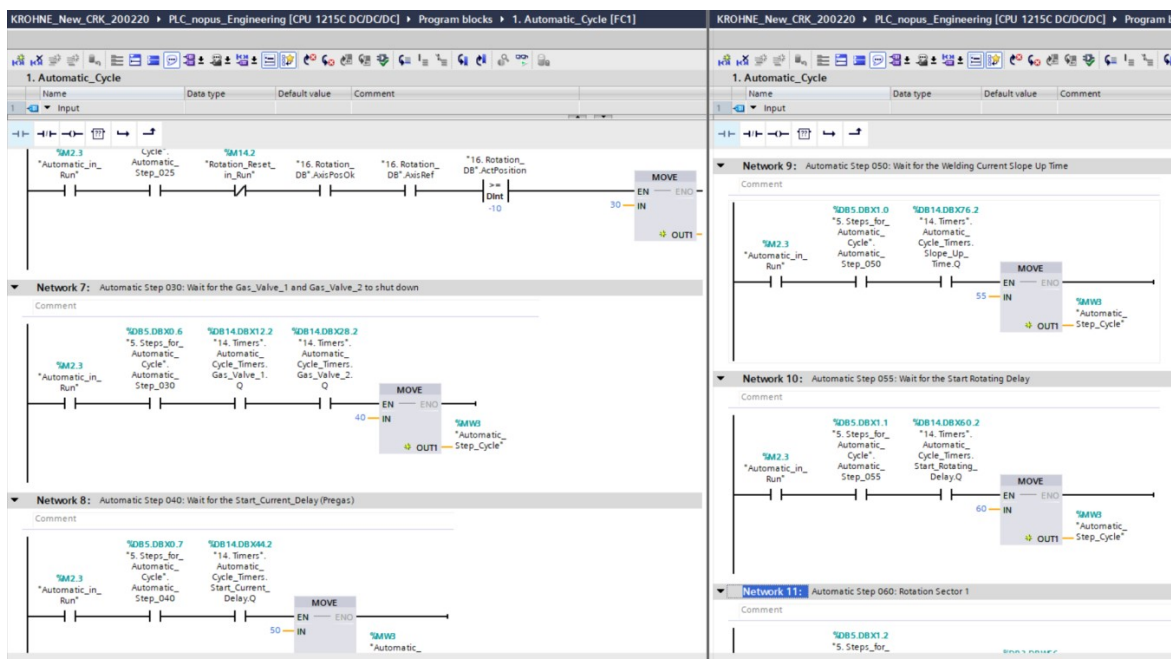
Prvo grananje odnosi se na glavnu horizontalnu os. Gornji uvjet „17. Horizontal_Axis_DB“.AxisPosOK“ provjerava je li os u korisnički zadanoj poziciji. Ako taj uvjet nije ispunjen, varijabla „ASC“ aktivira „SinaPos“ blok adekvatne osi, aktivira motor i procedura pozicioniranja osi na korisnički zadanu poziciju započne. Donji uvjet koji radi dvostruku provjeru varijable „17. Horizontal_Axis_DB“.ActPosition“ je tolerancijsko polje koje će propustiti signal dalje ako se aktualna pozicija osi nalazi unutar +/- 0.1 mm od korisnički zadane pozicije osi. Ovaj uvjet također propušta signal dalje u slučaju da se os već nalazi u korisnički zadanoj poziciji. Tolerancija od +/- 0.01 mm zanemariva je kod pozicioniranja stroja i smatra se da je stroj u poziciji ako se nalazi unutar tolerancijskih granica.

Drugi dio pozicioniranja odnosi se na osi oscilatora, to jest oscilatore. Njihovo pozicioniranje odvija se na isti način kao i kod glavne horizontalne osi.

Na slici 7.8 može se također vidjeti razlika u prikazu „ASC“ ulaznog uvjeta i uvjeta „17. Horizontal_Axis_DB“.AxisPosOK“ koji u granu vraća informaciju da je os

pozicionirana. Ulazni uvjet „ASC“ nalazi se u neoptimiziranom podatkovnom bloku pa je iznad imena uvjeta vidljiva njegova adresa memorijske lokacije. Uvjet „17. Horizontal_Axis_DB".AxisPosOK“ nalazi se u optimiziranom bloku i nema striktno dodijeljenu memorijsku lokacija pa stoga adresa te lokacije nije ni vidljiva iznad imena uvjeta.

Nakon što su se sve osi pozicionirale, na red dolaze brojači vremena (eng. *Timers*). Brojači vremena se pomoću vrijednosti varijable „ASC“ aktiviraju u funkcijskom bloku „17. Timers [FC17]“. Svi brojači koji se pozivaju u bloku „1. Automatic Cycle [FC1]“ su TON tipa što znači da aktiviraju izlaznu varijablu kada odbroje zadano vrijeme.



Slika 7.9: Izlazne varijable brojača vremena u automatskom ciklusu zavarivanja

Na slici 7.9 prikazani su izlazni uvjeti brojača vremena koji se nalaze u automatskom ciklusu zavarivanja.

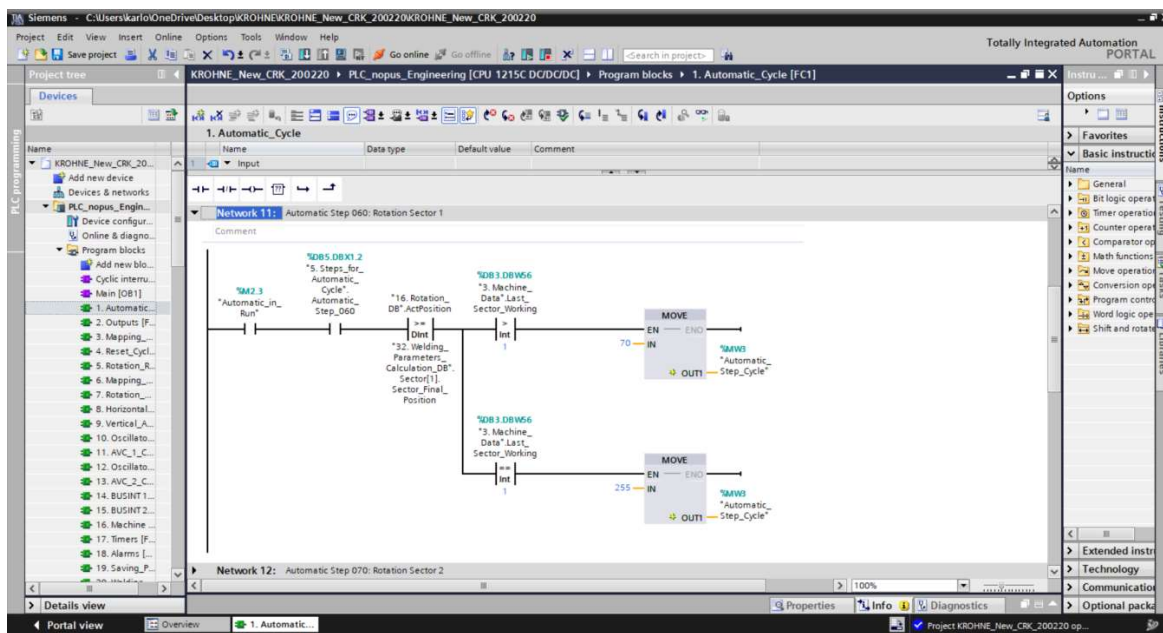
U radni komad koji se zavaruje ulazi zaštitni plin, a grana NW7 omogućava da proces zavarivanja ne započne dok ne isteku korisnički zadana vremena protoka plinova za oblikovanje. Kada započne ciklus automatski zavarivanja, PLC otvori glavni elektroventil i elektroventil za plin za oblikovanje 1. Kada istekne vrijeme protoka plina za oblikovanje 1, logička varijabla „14.Timers".AC.Forming_1.Q“ se aktivira, gasi se elektroventil za plin za oblikovanje 1, te se pali elektroventil za plin za oblikovanje 2. Kada istekne vrijeme protoka plina za oblikovanje 2 aktivira se logička varijabla „14.Timers".AC.Forming_2.Q“ Ako je vrijeme tečenja obadva plina za oblikovanje duže

od vremena pozicioniranja, stroj će čekati u radnoj poziciji dok vrijeme tečenja plinova za oblikovanje ne prođe. Ako je pak vrijeme pozicioniranja duže od vremena tečenja obadva plina za oblikovanje, čim se stroj postavi u radnu poziciju, započinje proces zavarivanja.

Proces započinje tečenjem početnog plina. To je zaštitni plin koji služi kao inertna zaštitna atmosfera koja sprječava da bilo kakve nečistoće iz zraka uđu u zavar. Kada vremena početnog plina („14.Timers“.AC.Pregas“) i odgode paljenja električnog luka („14.Timers“.AC.Start_Current.Q“) prođu, PLC aparatu za zavarivanje šalje signal za uspostavu električnog luka za zavarivanje. Iznos jakosti struje zavarivanja preko rampe definirane korisnički zadanim vremenom podiže se na radnu vrijednost. Aktivacijom izlaznog parametra brojača vremena „14.Timers“.AC.Slope_Up.Q“, iznos jakosti struje zavarivanja dosega je radnu vrijednost i PLC prelazi u novu sekvencu. U novoj sekvenci stroj zavaruje na mjestu za korisnički definirano vrijeme kako bi se stvorio bazen rastaljenog materijala. Taj bazen služi kao talina u kojoj se tali žica za zavarivanje. Bez rastaljenog bazena žica se ne bi talila jednoliko po obodu ranog komada i počele bi se stvarati kuglice rastaljenog materijala na žici što svakako želimo izbjeći. Kuglice ostaju na žici dok ne postignu dovoljnu masu i padnu na radni komad. To rezultira nepravilnim zavarom jer je radni komad mjestimično nezavaren, a na mjestima gdje padne kuglica nalazi se prevelika količina dodatnog materijala.

Kada se aktivira „14.Timers“.AC.Star_Rotation“, rotacija započinje i stroj ulazi u prvi od mogućih 20 sektora zavarivanja. Ulazni uvjet isti je kao i kod prethodno objašnjenih grana. Nakon njega dolazi srednji uvjet koji uspoređuje korisnički zadanu krajnju poziciju sektora u stupnjevima i stvarnu poziciju rotacijske osi. Kada stvarna vrijednost postane veća od korisnički zadane krajnje pozicije prelazi se na sljedeći sektor. Prije prelaska u idući sektor postavljen je uvjet koji provjerava broj posljednjeg aktiviranog sektora. Ako je uvjet istinit, u „ASC“ se upisuje vrijednost 255 koja preskače sve iduće sektore i odlazi na kraj zavarivačkog procesa, točnije u NW31 gdje započinje spuštanje iznosa jakosti struje zavarivanja. Ako bi se ovaj uvjet izostavio, a korisnik želi raditi samo sa sektorom broj 1, program bi i dalje prošao kroz sve ostale sektore do broja 20 jer bi u svakom sektoru vrijednost krajnje pozicije bila 0 što zadovoljava uvjet provjere pozicije. Ulaskom u svaki sektor, makar u samo jednom ciklusu izvršavanja koda, u program bi se učitali parametri za taj sektor. Kako je ciklus izvršavanja koda oko 30 ms, s ulaskom 19 neaktivnih sektora dobije se vrijeme od 570 ms, odnosno malo više od pola sekunde. U automatiziranim procesima ovih 570 ms mogu znatno utjecati na krajnji rezultat. U svakom od tih 19 sektora iznos struje zavarivanja je 0 ampera. To znači da bi proces zavarivanja tekao

iznosom jakosti struje zadanom u sektoru 1, zatim bi na 570 ms pao na 0 A, pa bi se ponovno digao na struju zadanu u sektoru 1 i započeo spuštati struju po rampi prema 0 ampera. Vrijeme od 570 ms prestanka električnog luka rezultiralo bi neispravnim zavarom i automatizirani proces zavarivanja gubi svoj smisao. Programski kod jednak je za svih 20 sektora. Grana NW11 sa sektorom 1 prikazana je na slici 7.10.



Slika 7.10 Uvjeti unutar sektora zavarivanja

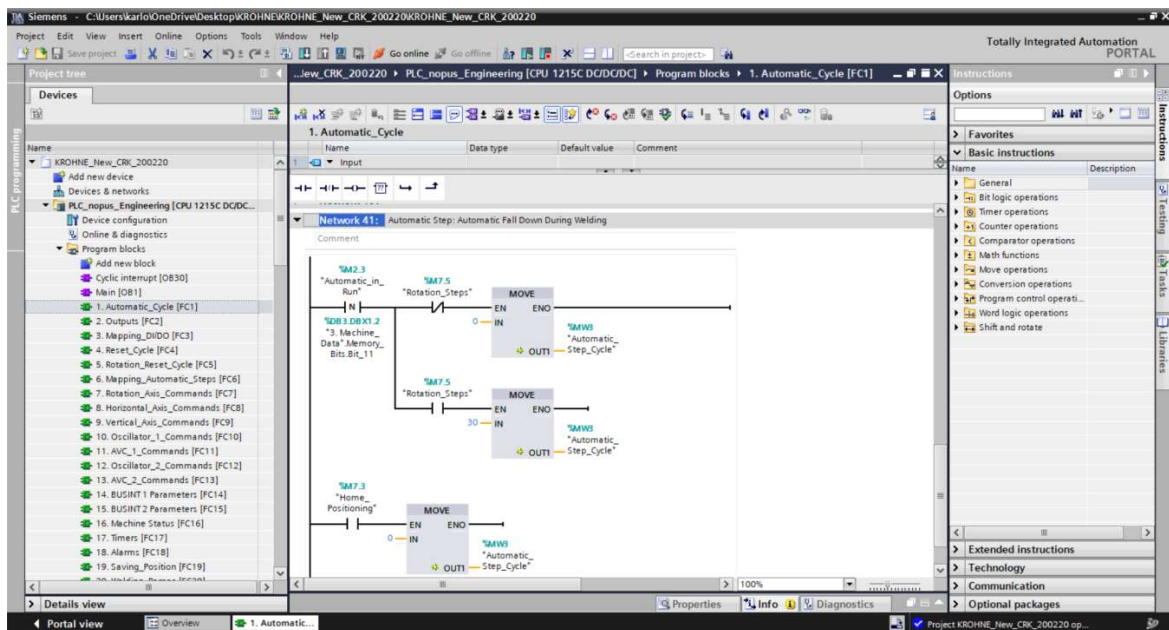
Nakon što rotacija dosegne krajnju poziciju zadnjeg aktivnog sektora na red dolazi spuštanje iznosa jakosti struje u NW31, koje je spomenuto u prethodnom odlomku. Spuštanje iznosa jakosti struje ne smije se odvijati dok je stroj u stacionarnoj poziciji jer to može imati negativne utjecaje za završni zavar poput poroznosti u zavaru. Zbog toga korisnik zadaje kut rotacije u stupnjevima tijekom kojega se smanjuje iznos jakosti struje. Vrijeme smanjivanja iznosa jakosti struje također određuje korisnik. Kada električni luk iščezne, zaštitni plin ostaje teći za korisnički zadano vrijeme kako bi se novonastali zavar i vruća elektroda zaštitili od negativnih utjecaja nečistoća iz zraka i okoline. Nakon što je vrijeme tečenja zaštitnog plina gotovo, stroj se pozicionira u parking poziciju. Prvo se sve vertikalne osi podignu na svoje pozicije. Kada se vertikalne osi pozicioniraju, horizontalne osi kreću u svoje pozicije.

Nakon što horizontalne osi dođu u svoje parking pozicije, PLC povećanjem kontrolne varijable ulazi u NW 37 u kojemu se označava kraj automatskog ciklusa. To se radi aktivacijom bita „*Condition_to_Stop_Automatic_Cycle*“. Ovaj bit nalazi se također u već

spomenutom bloku „16. Machine_Status [FC16]“ kao NC kontakt u grani koja aktivira aktivacijski uvjet „Automatic_in_Run“. NC kontakt ponaša se suprotno od NO, odnosno propušta signal kada bit nije aktivan i blokira signal kada je bit aktivan. Aktivacija bita „Condition_to_Stop_Automatic_Cycle“ prekinut će signal prema zavojnici „Automatic_in_Run“ u bloku „16. Machine_Status [FC16]“ i prekinuti automatski ciklus. Kako aktivacijski uvjet „Automatic_in_Run“ više nije aktivan, niti jedna grana koja sadržava ovaj uvjet ne može biti aktivna, pa tako ni NW 37. Ako grana NW 37 nije aktivna, onda ni zavojnica „Condition_to_Stop_Automatic_Cycle“ ne može biti aktivna. Ovakvim načinom programiranja aktivacija bita „Condition_to_Stop_Automatic_Cycle“ gasi automatski ciklus i samog sebe. Sljedeća grana NW 38 više ne sadržava aktivacijski uvjet „Automatic_in_Run“ budući da je automatski ciklus završio, nego provjerava je li bit „Condition_to_Stop_Automatic_Cycle“ deaktiviran. Također, kontrolna varijabla „ASC“ zadržava vrijednost 290 iz prethodne grane NW 37 kako bi se osiguralo da se grana NW 38 izvršava samo na izlasku iz automatskog ciklusa. Ako je uvjet „Condition_to_Stop_Automatic_Cycle“ deaktiviran, PLC u kontrolnu varijablu upisuje vrijednost 0 čime se automatski ciklus vraća u početni korak. Također, povećavaju se vrijednosti brojača ukupno završenih ciklusa i završenih ciklusa za prijavljenog operatera. Kada je ciklus završen, zelena „START“ tipka na daljinskom upravljačku prestaje svijetliti.

Ako se automatski ciklus zavarivanja prekine na bilo koji način (korisnik pritisne tipku „STOP“ ili sigurnosni prekidač, korisnik greškom prebaci stroj u ručni način na daljinskom upravljaču, dogodi se greška u PLC-u ili V90 servo upravljačkom sustavu) automatski ciklus se zaustavlja pomoću kontrolne grane NW41.

Ako se greška dogodi za vrijeme pozicioniranja stroja dok logička varijabla „Rotation_Steps“ ima negativnu vrijednost, stroj će se ponovnim pritiskom tipke „START“ krenuti pozicionirati ispočetka. Ako pak greška nastane tijekom zavarivanja dok logička varijabla „Rotation_Steps“ ima pozitivnu vrijednost, stroj će vratiti u korak 40 gdje započinje tečenje početnog plina. Nakon što prođe vrijeme tečenja početnog plina, vrijeme podizanja iznosa jakosti struje i vrijeme odgode početka rotacije, zavarivanje se praktički nastavlja u sektoru gdje je prethodno bilo i zaustavljeno. U stvarnosti PLC aktivira svaki prethodni sektor na jedan ciklus jer ne može samo preskočiti te sektore i krenuti od zadnjeg aktivnog sektora. Vrijeme koje prođe dok stroj ne dosegne zadnji aktivirani sektor zanemarivo je za zavarivački proces u ovom slučaju jer je zavarivanje aktivirano tako da nema nikakvih nedostataka u zavaru. Na slici 7.11 prikazani su uvjeti prekida automatskog zavarivačkog ciklusa.



Slika 7.11 Prekid automatskog ciklusa

7.2.2 Izlazi stroja

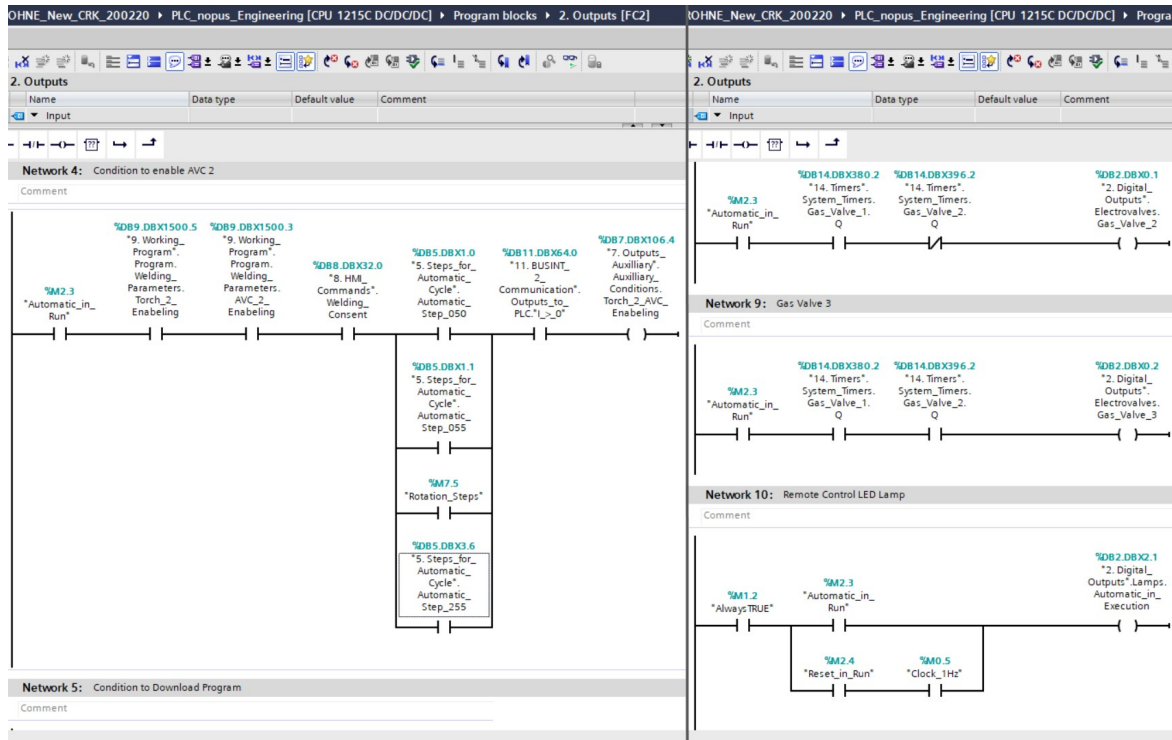
Blok „2. Outputs [FC2]“ je skup uvjeta koji rezultiraju novim izlaznim uvjetom, nešto poput sabirnice signala. Kao primjer navest će se NW3 u kojemu se ispituje uvjet za paljenje automatske kontrole električnog luka na Gorioniku 2.

Kao što je vidljivo iz slike 7.12 (lijevo), da bi se uključila automatska kontrola električnog luka ili AVC funkcija (eng. *Automatic Voltage Control*) trebaju se prvo ispuniti određeni uvjeti. Korisnik prvo na HMI uređaju mora aktivirati prekidač za gorionik s kojim želi raditi („9. Work_Prog“.Prog.Weld_Par.T2_Act“) i aktivirati prekidač AVC kontrole tog gorionika („9. Work_Prog“.Prog.Weld_Par.AVC2_Act“).

Zatim korisnik mora uključiti struju zavarivanja u automatskom načinu rada („8. HMI_CMD_DB“.Weld_Act“) i naposljetku uvjeti automatskog ciklusa moraju biti ispunjeni kako bi se osiguralo da će se AVC upaliti tek kada iznos jakosti struje zavarivanja dosegne radnu vrijednost. Zadnji uvjet je ujedno i najvažniji, a on PLC-u daje povratnu informaciju s aparata za zavarivanje da je električni luk zavarivanja uspostavljen. Bez ovog uvjeta nema ni AVC kontrole jer bez uspostavljenog električnog luka AVC kontrola nema radni napon prema kojem prilagođava svoju poziciju.

Na slici 7.12 također je vidljiv uvjet „Rotation_Steps“ u kojemu su povezani koraci u kojima radni komad rotira, odnosno svi sektori automatskog ciklusa zavarivanja. Kada se stroj nalazi u bilo kojem sektoru, ovaj uvjet je aktivan. Ovaj uvjet olakšava programiranje stroja u situacijama gdje je rotacija uvjet za određenu funkciju stroja. Bez uvjeta

„Rotation_Steps“ u NW3 nalazilo bi se po jedan uvjet za svaki sektor, odnosno 20 uvjeta sve skupa. Kada je bitno da komad rotira kao u ovom slučaju, PLC ne zanima koji je sektor trenutno aktivan, već samo da je aktivan jedan sektor i da se rotacija izvršava. Iz tog razloga svih 20 sektora može se povezati u samo jedan uvjet.



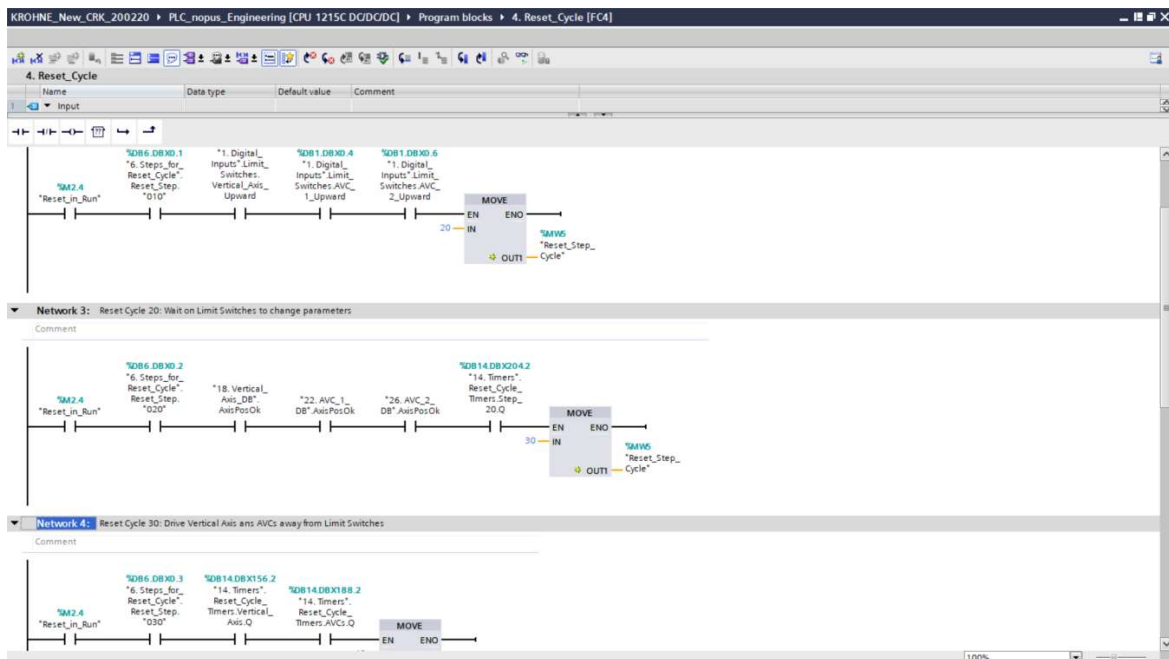
Slika 7.12 Funkcija 2. Outputs

Na slici 7.12 desno prikazani su uvjeti za paljenje plina za oblikovanje 2 u NW8, za paljenje plina za oblikovanje 3 (NW9) i za paljenje LED lampice na tipkalu „START“ na daljinskom upravljaču. Ako se stroj nalazi u automatskom ciklusu zavarivanja („Automatic_in_Run“), lampica konstantno svijetli zelenom bojom, a ako se stroj nalazi u proceduri za reset, LED lampica blica zeleno frekvencijom od 1 Hz.

7.2.3 Referenciranje

Blok „4. Reset_Cycle [FC4]“ je blok koji izvršava referenciranje, odnosno pronalaženje nultočaka. Nakon referenciranja stroj se pozicionira u parking poziciju.

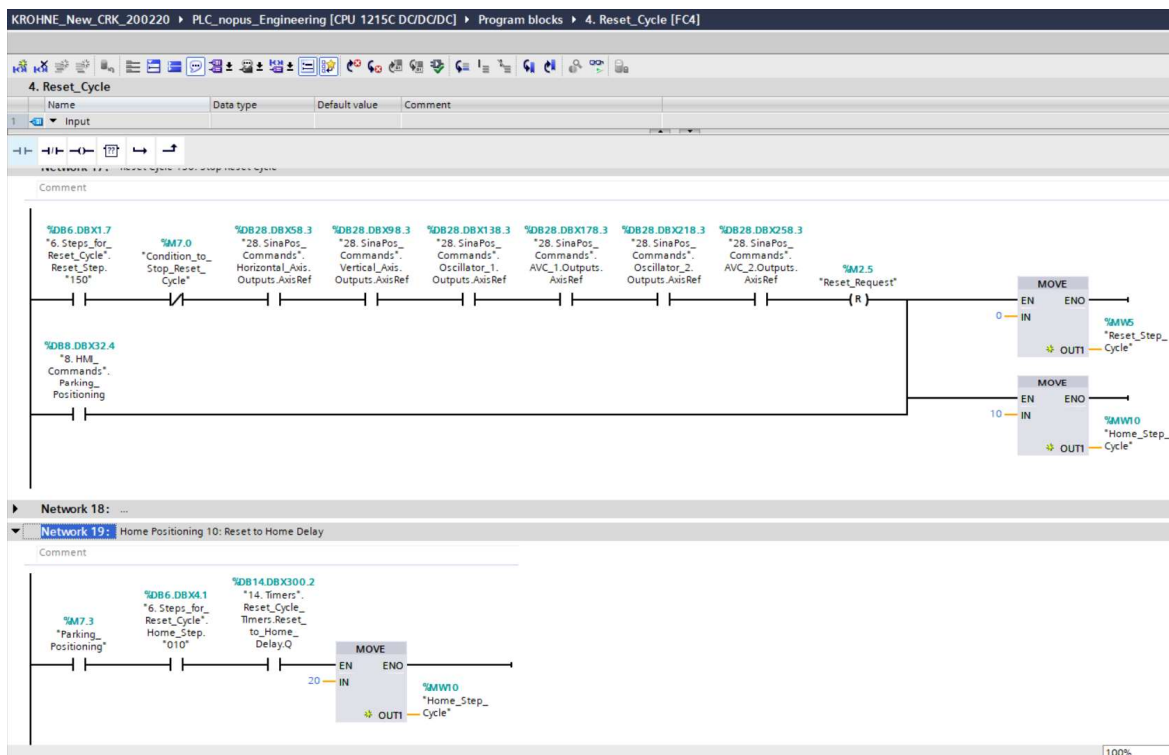
Pritiskom na tipku „RESET“ na početnoj stranici HMI uređaja, dok je stroj u automatskom načinu rada, započinje proceduri za reset. Prvo se referenciraju sve tri vertikalne osi stroja kako bi se izbjegla moguća kolizija s nepokretnim dijelovima stroja kao što je kućište STL-a. Na slici 7.13 prikazan je dio koda za referenciranje.



Slika 7.13: Traženje krajnjih prekidača

Vertikalne osi putuju prema gornjim krajnjim prekidačima kako bi pronašle njihove pozicije. Brzina traženja krajnjih prekidača je korisnički definirana i ograničena krajnjim vrijednostima. Brzina traženja referentnih pozicija mora se ograničiti kako operater ne bi unio nedozvoljenu vrijednost (npr. negativnu vrijednost brzine) ili preveliku vrijednost brzine. U NW2 nalaze se povratne veze sa sva tri gornja krajnja prekidača. Kod upravljanja svim osima, aktivacija krajnjeg prekidača onemogućava kretanje osi u smjeru krajnjeg prekidača. Ako jedna vertikalna os dođe na svoj gornji krajnji prekidač prije drugih, zaustavit će se i pričekati ostale osi da aktiviraju svoje gornje krajnje prekidače. Kada su sva tri gornja krajnja prekidača aktivirana, procedura za reset povećanjem kontrolne varijable „RSC“ odlazi na sljedeći korak. Grana NW3 u sebi ima povratnu vezu s brojača vremena „14 Timers.Reset_Cycle_Timers.Step_20.Q“ i povratnu vezu s V90 servo upravljačkog sustava „AxisPosOK“ koja javlja da je os pozicionirana u korisnički zadanu poziciju. Budući da se kod referenciranja sve osi voze u „Jog“ načinu rada koji ne koristi korisnički zadanu poziciju kao jedan od parametara, stacionarna pozicija uzima se kao korisnički zadana pozicija i aktivira bit „AxisPosOK“. Brojač vremena u NW3 postavljen je na 1.5 sekundu i unosi vremensku odgodu u proces referenciranja kako bi se osiguralo da sve osi budu u stacionarnom položaju i da se učitaju novi parametri za daljnje referenciranje. Nakon 1.5 sekunde osi krenu putovati 2 sekunde u smjeru suprotnom od krajnjih prekidača kako bi se krajnji prekidači oslobodili. Kada osi nakon 2 sekunde stanu u novoj poziciji gdje su krajnji prekidači otpušteni, stroj ponovno pričekava 1.5 sekundu.

Nakon toga osi ponovno krenu putovati u smjeru krajnjih prekidača, no ovaj put brzinom zadanom u samom V90 servo upravljačkom sustavu. Brzinu ponovnog traženja vertikalnih gornjih krajnjih prekidača ne može se korisnički definirati niti promijeniti kako bi se osiguralo da se stroj uvijek referencira u istoj poziciji. Ako bi se stroj referencirao s jednom brzinom i prema referentnim pozicijama korisnik napravi radne pozicije, ponovno referenciranje s drugom brzinom rezultiralo bi novim referentnim pozicijama i sukladno tome novim radnim pozicijama. Ako se svako referenciranje radi s drugom brzinom, tada i senzor okida na različitim pozicijama. Iako su senzori vrlo precizni i brzi, to su i dalje mehaničke komponente koje imaju određeno vrijeme odziva. Taj odziv traje par milisekundi koje u vrlo preciznim i brzim sustavima mogu imati veliku ulogu. Tako bi se u potpunosti izgubilo svojstvo ponovljivosti ciklusa i cijeli smisao automatizacije.



Slika 7.14: Kraj procedure za reset i ulazak u parking proceduru

Kada su sve tri vertikalne osi referencirane započinje referenciranje horizontalnih osi. Princip, to jest redoslijed referenciranja istovjetan je referenciranju vertikalnih osi, samo što se referenciranje odvija na lijevim krajnjim prekidačima kod glavne horizontalne osi i lijeve oscilatorske osi, te na desnom krajnjem prekidaču kod desne oscilatorske osi. Kada se stroj u potpunosti referencira, zelena LED lampica na daljinskom upravljaču prestaje bljeskati frekvencijom od 1 Hz i tako signalizira kraj procedure za reset.

Nakon što su sve osi referencirane pokreće se parking procedura u kojoj se sve osi pomiču u parking poziciju. Na slici 7.14 u gornjoj grani NW17 vidljiv je prelazak iz procedure za reset u parking proceduru. Ako je grana aktivna, u „RSC“ se upisuje vrijednost 0 koja gasi proceduru za reset i vraća ju u početni korak, a u „HSC“ upisuje se vrijednost 10 koja aktivira parking proceduru.

Parking procedura također se može pokrenuti pritiskom na tipku „PARKING POZICIJA“ na HMI uređaju, što je prikazano u donjoj grani NW17 na slici 7.14. Pritiskom tipke „PARKING POZICIJA“ iste vrijednosti upisat će se u „RSC“ i „HSC“. Jedan od uvjeta pokretanja parking procedure je da procedura za reset ne smije biti aktivna, pa upisivanje vrijednosti 0 u „RSC“ neće imati nikakvog utjecaja na stroj kada korisnik pritisne tipku „PARKING POZICIJA“ jer će varijabla „RSC“ već imati vrijednost 0. Pozicioniranje se odvija tako da se prvo pozicioniraju sve horizontalne osi te nakon njih sve vertikalne osi.

Parking procedura također se može pokrenuti pritiskom na tipku „PARKING POZICIJA“ na HMI uređaju, što je prikazano u donjoj grani NW17 na slici 7.14. Pritiskom tipke „PARKING POZICIJA“ iste vrijednosti upisat će se u „RSC“ i „HSC“. Jedan od uvjeta pokretanja parking procedure je da procedura za reset ne smije biti aktivna, pa upisivanje vrijednosti 0 u „RSC“ neće imati nikakvog utjecaja na stroj kada korisnik pritisne tipku „PARKING POZICIJA“ jer će varijabla „RSC“ već imati vrijednost 0. Pozicioniranje se odvija tako da se prvo pozicioniraju sve horizontalne osi te nakon njih sve vertikalne osi.

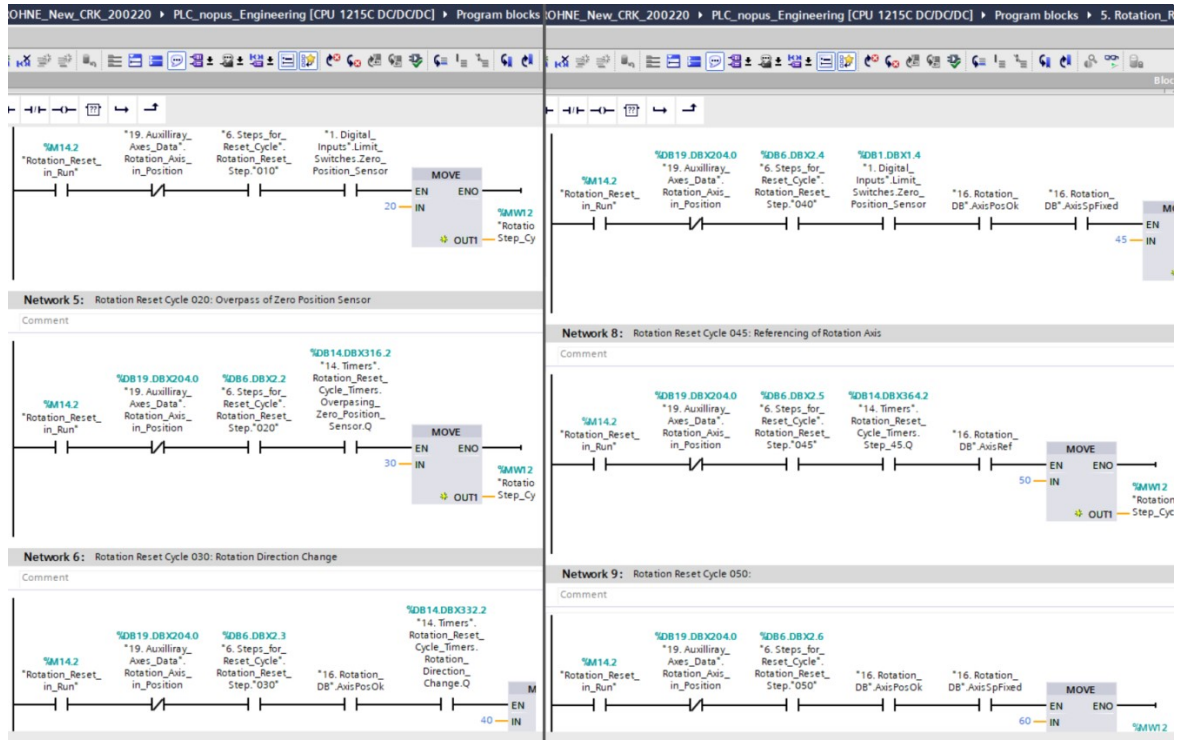
7.2.4 Referenciranje rotacijske osi

Blok „5. Rotation_Reset_Cycle [FC5]“ započinje zajedno s blokom „4. Reset_Cycle [FC4]“ nakon pritiska na tipku „RESET“ na HMI uređaju.

Prvi korak referenciranja je pronalazak referentne točke, odnosno senzora na STL-u korisnički zadanom brzinom. Kada rotacijska glava pronade senzor, NW2 zadaje naredbu da se rotacija nastavi u zadanom vremenu od 1,5 sekundi kako bi se referentni senzor otpustio. Zatim rotacija staje na 1,5 sekundu da rotacijska glava u potpunosti stane kako se zbog inercije rotacije ne bi oštetili zupčanici u zupčaničkom prijenosu STL-a.

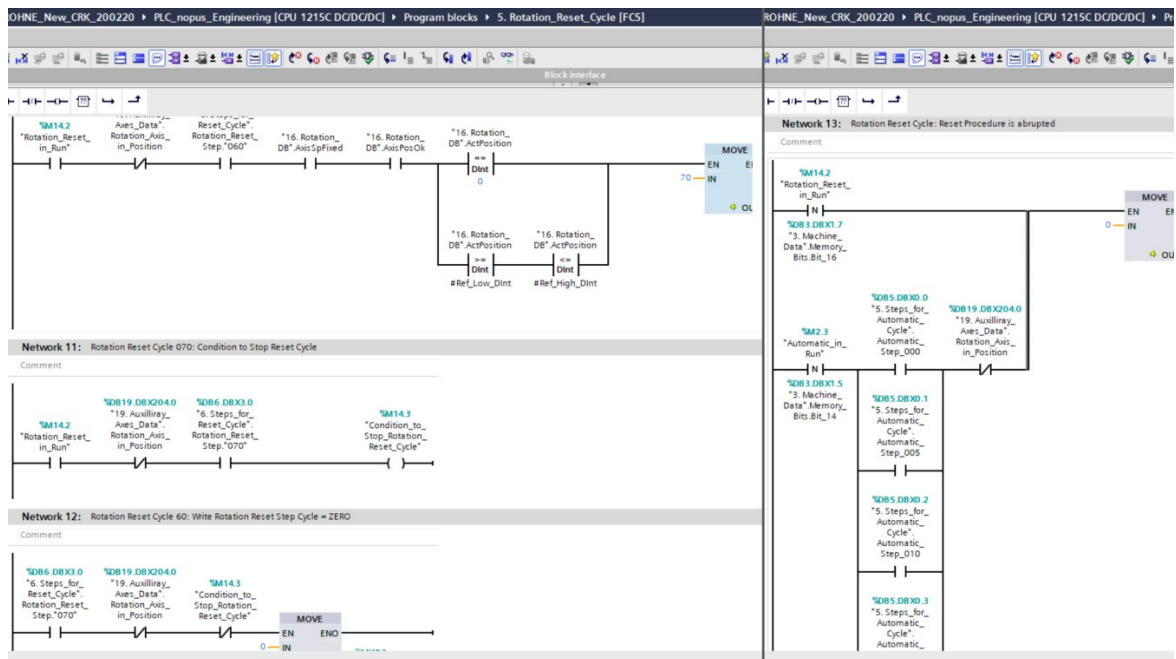
Također, u tih 1,5 sekundi se promijene parametri bitni za nastavak referenciranja. Nakon toga kreće ponovno trženje referentne točke u suprotnom smjeru, ali s predefiniranom brzinom zapisanom u V90 servo upravljačkom sustavu. Kao što je već

napomenuto u potpoglavlju 6.2.3, brzina traženje referentne točke mora uvijek biti odrađena istom brzinom kako bi senzor uvijek na istom mjestu uhvatio rotacijsku glavu i referencirao se.



Slika 7.15: Referenciranje rotacijske osi

Na slici 7.16 nalazi se uvjet koji se aktivira u slučaju prekida referenciranja rotacijske osi. Obadvije grane u grananju imaju isti rezultat, a to je upisivanje vrijednosti 0 u varijablu „*Rotation_Reset_Step_Cycle*“ ako se prekine izvršavanje bloka „*Rotation_Reset_Cycle [FC5]*“. Prva grana će na svaki negativni brid uvjeta „*Rotation_Reset_in_Run*“ izvršiti naredbu upisivanja 0 u „*Rotation_Reset_Step_Cycle*“. Negativni brid označava promjenu bita iz stanja istine ili 1 (eng. *True*) u stanje laž ili 0 (eng. *False*). Druga grana grananja će na svaki negativni brid uvjeta „*Automatic_Cycle_in_Run*“ izvršiti naredbu, ali uz dodatne uvjete. Naredba će se izvršiti samo ako se automatski ciklus nalazi u koracima pozicioniranja linearnih osi i ako se pozicija rotacijske osi ne nalazi unutar tolerancijskog polja krajnjeg prekidača. Naime, kada automatski ciklus završi, rotacijska os ostaje u krajnjoj poziciji.



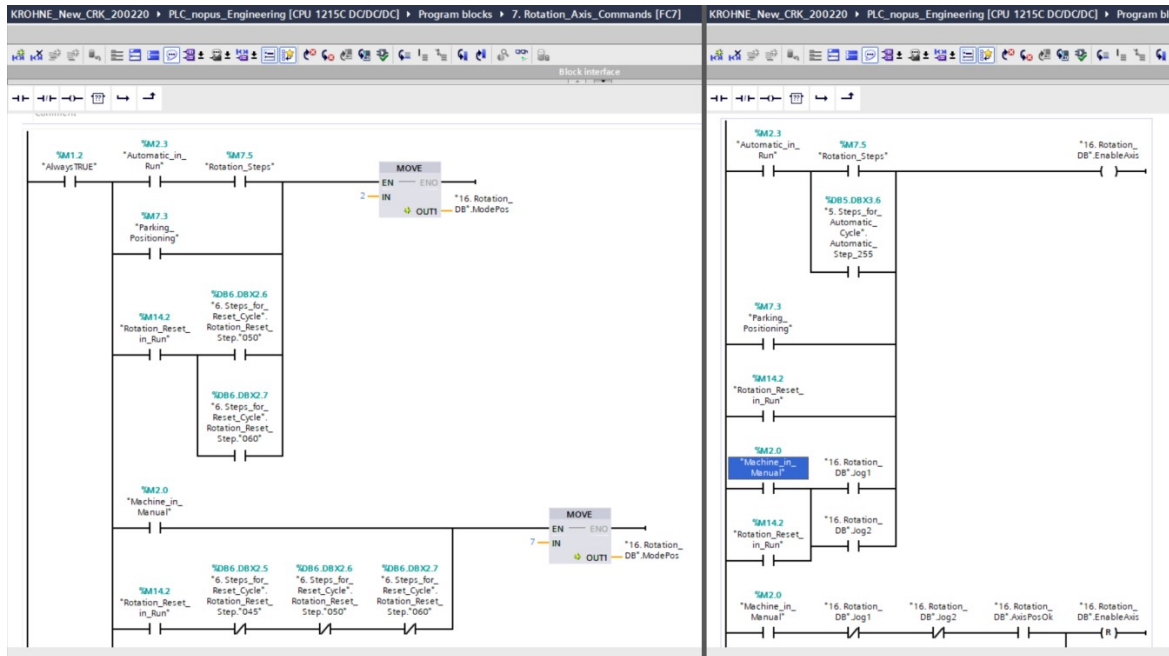
Slika 7.16: Kraj referenciranja rotacijske osi i sigurnosni uvjet u slučaju prekida

7.2.5 Rotacijska os

Blok „7. *Rotation_Axis* [FC7]“ je blok koji služi kao priprema uvjeta za izvršavanje rotacije rotacijske osi na STL-u. U glavnom bloku „*Main* [OB1]“ nalazi se „*SinaPos*“ naredba koja šalje informacije na V90 servo upravljački sustav i tako upravlja rotacijom motora, a blok „7. *Rotation_Axis* [FC7]“ služi za određivanje ulaznih parametara u „*SinaPos*“ naredbu. Ako bi se svi mogući ulazni parametri „*SinaPos*“ naredbe nalazili u bloku „*Main*[OB1]“, bilo bi se teško snaći u velikom broju parametara prilikom programiranja. Zbog toga je definiranje ulaznih parametara prebačeno u blok „7. *Rotation_Axis* [FC7]“ gdje je za svaki parametar „*SinaPos*“ naredbe koja se koristi predviđena nova grana. Na ovaj način olakšana je i dijagnostika i pronalaženje eventualnih grešaka u programskom kodu. Na slici 7.17 (lijevo) prikazan je dio koda za određivanje načina rada V90 servo upravljačkog sustava.

Prva stvar koja se definira za „*SinaPos*“ naredbu je bit „16. *Rotation_DB.ModePos*“ u NW2. To je parametar koji V90 servo upravljačkom ustavu govori u kojem načinu rada se nalazi. Ovo programsko rješenje koristi apsolutno pozicioniranje („*ModePos*“ = 2), postavljanje referentne točke („*ModePos*“ = 5) i „*Jog*“ način rada („*ModePos*“ = 7). Apsolutno pozicioniranje je vrsta pozicioniranja u kojemu se V90 servo upravljačkom sustavu daje točna pozicija u koju se treba pozicionirati. Prije nego što se pokrene apsolutno pozicioniranje, os je potrebno referencirati kako bi V90 servo upravljački sustav

znao koja mu je referentna točka od koje radi pozicioniranje. „Jog” način rada koristi se za ručno upravljanje strojem. U „Jog” načinu rada potrebno je definirati samo brzinu, bez krajnje pozicije, pa nije potrebno odraditi referenciranje prije početka korištenja „Jog” načina rada.



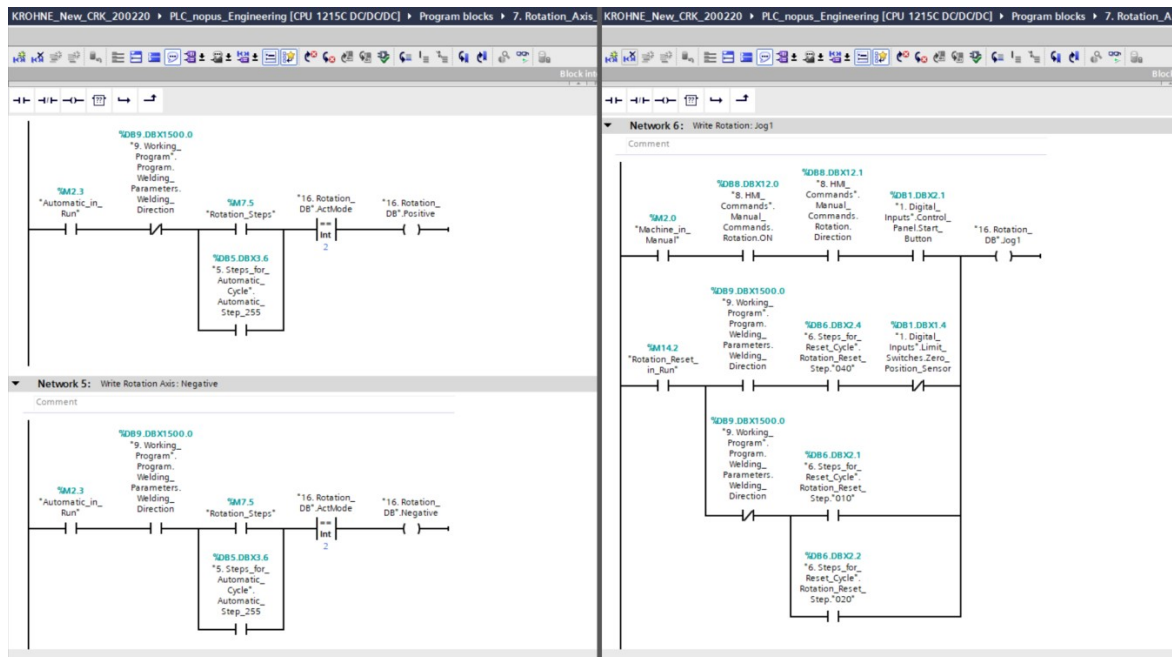
Slika 7.17: Određivanje načina rada i aktivacijski uvjet rotacijskog motora

Nakon određivanja „ModePos” načina rada, u NW3 aktivira se bit „16. Rotation_DB.EnableAxis“ koji aktivira V90 servo upravljački sustav i njegov motor. Ova aktivacija neće pokrenuti motor već će samo pustiti struju držanja na njega i držati ga na poziciji. Kada se stroj ugasi, napajanje na servo upravljačkim sustavima prestaje i sukladno time prestaje struja držanja motora. Za horizontalne osi to ne predstavlja problem jer je njihova sloboda gibanja horizontalan pravac, ali kod vertikalnih osi to bi moglo predstavljati problem jer je njihova sloboda gibanja vertikalni pravac i postoji opasnost od ispadanja. Taj problem riješen je postavljanjem pužnih mehaničkih prijenosa između motora i navojne šipke osi. Njihov prijenosni omjer dovoljno je velik kako bi se osi mehanički zaustavile u pozicijama u kojima su se našle u trenutku prestanka napajanja. Na slici 7.17 prikazano je kako je „16. Rotation_DB.EnableAxis“ aktivan dok god je aktivan stroj kako bi se osiguralo da je struja napajanja uvijek puštena na motor kako bi ga držala u poziciji.

Ako bi bit „16. Rotation_DB.EnableAxis“ bio ugašen dok je stroj u stacionarnom stanju, osi bi bile slobodne za pomicanje u slučaju vanjskih podražaja. Glavnu vertikalnu

os i sve pomoćne osi na mjestu bi držali pužni mehanički prijenosi, ali glavna horizontalna os koja se nalazi na zupčastoj letvi bila bi potpuno slobodna i korisnik bi ju teoretski mogao pomaknuti rukom.

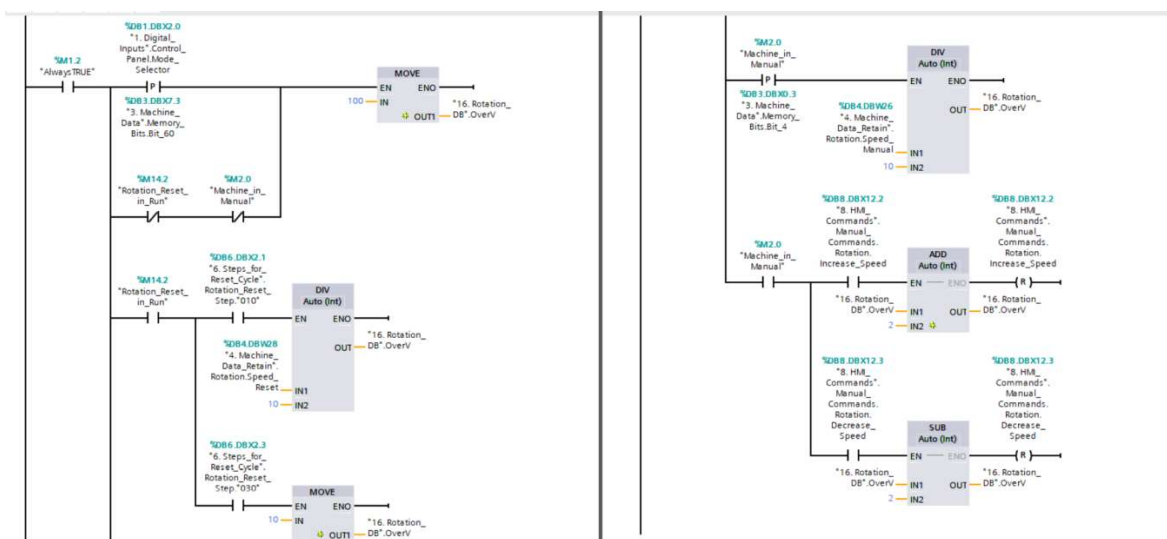
U automatskom pozicioniranju kod rotacijskih osi potrebno je odrediti smjer rotacije kako bi V90 servo upravljački sustav znao očitati vrijednosti s enkodera, tj. u kojem smjeru enkoder broji pozitivne, a u kojem negativne vrijednosti. Na slici 7.18 (lijevo) nalaze se grane NW4 i NW5 kojima se određuje smjer rotacije osi. Jedina razlika u ove dvije grane je u drugom uvjetu „9. Working_Program“.Program.Welding_Parameters.Welding_Direction“. Taj uvjet povezan je s tipkom na HMI uređaju pomoću koje se određuje smjer rotacije. Ako je tipka u položaju „0“ rotacija će ići u pozitivnom smjeru jer je uvjet u NW4 normalno zatvoren. Ako je tipka u „1“ položaju, onda rotacija ide u negativnom smjeru jer je uvjet u NW5 normalno otvoren. Ovo se odnosi i na smjer brojenja enkodera. To znači da će iznos pozicije rotacije nakon punog kruga iznositi 360° za pozitivan smjer rotacije i -360° za negativan smjer rotacije.



Slika 7.18: Odabir smjera rotacije i uvjeti za Jog1 način rada

Sljedeći parametri koji su definirani su „Jog1“ u NW6 i „Jog2“ u NW7. JOG način rada je način u kojemu motor vozi sve dok je aktivan jedan od „JogX“ bitova. Ako se „Jog1“ i „Jog2“ aktiviraju u isto vrijeme, V90 servo upravljački sustav zaustavit će motor i poslati grešku na PLC koji će ju prikazati na HMI uređaju. Naime, smjer rotacije je predefiniран za svaki „JogX“ parametar, pa aktivacija obadva parametra istodobno V90

servo upravljačkom sustavu govori da vozi istovremeno u pozitivnom i negativnom smjeru što je nemoguće. Zbog toga što je smjer rotacije predefiniiran za svaki „JogX“ parametar, nije ga potrebno definirati u granama NW4 i NW5 jer će V90 servo upravljački sustav te parametre zanemariti kada se nalazi u „Jog“ načinu rada. Smjer i brzina rotacije u „Jog“ načinu rada definiraju se u samome V90 servo upravljačkom sustavu. Ovakvo definiranje brzine značilo bi kako je u manualnom načinu rada motor moguće voziti samo konstantnom brzinom. Zbog toga V90 servo upravljački sustav ima parametar „OverV“ kojim mu se određuje postotna vrijednost brzine definirane u samom sustavu. Vrijednost „OverV“ varijable može se definirati od 0% do 199% brzine definirane u V90 servo upravljačkom sustavu. Ako je brzina definirana u V90 servo upravljačkom sustavu 1000 mm/min, a parametar „OverV“ 80, V90 servo upravljački sustav voziti će motor s 80% brzine, odnosno 800 mm/min . Kao što je vidljivo iz slike 7.18 (desno), „Jog“ način rada koristi se u manualnom i reset načinu rada. Manualno se rotacijska os može kontrolirati samo preko HMI uređaja na stranici „RUČNO UPRAVLJANJE“. Motor rotacijske osi počinje se rotirati tek kada korisnik odabere manualni način rada na daljinskom upravljaču, aktivira manualnu kontrolu osi i odabere smjer rotacije na stranici „RUČNO UPRAVLJANJE“, te pritisne tipku „START“ na daljinskom upravljaču.



Slika 7.19: Upisivanje vrijednosti parametra OverV

S „Jog“ načinom rada rotacijska os se vozi do krajnjeg prekidača i prelazi preko njega do pozicije u kojoj je krajnji prekidač otpušten. S te pozicije se onda vraća na krajnji prekidač, odnosno referentnu točku predefiniiranom brzinom u V90 servo upravljačkom sustavu. Kako se „Jog“ način rada koristi u više različitih funkcija stroja, u svakoj od tih

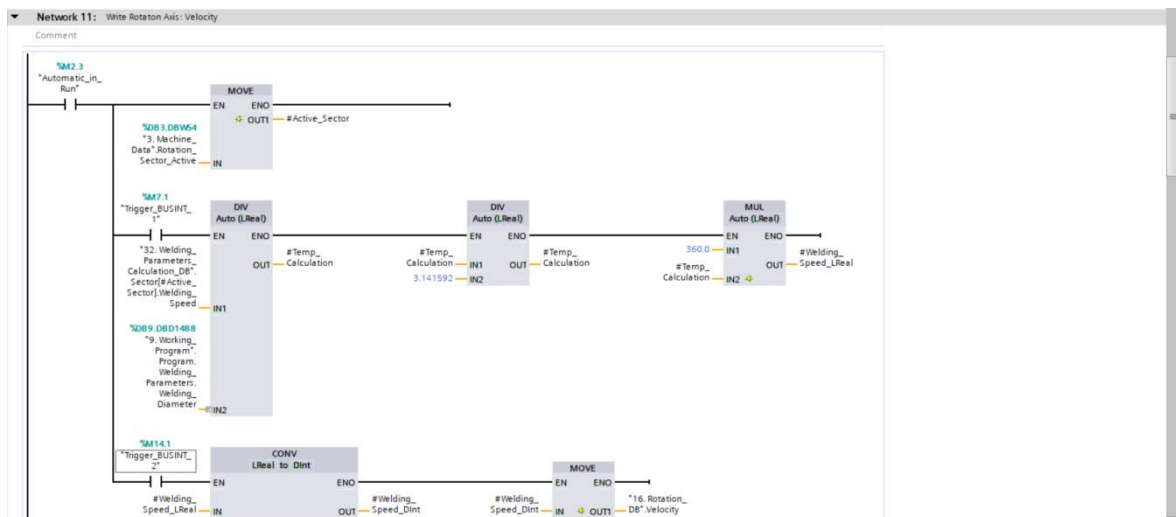
funkcija potrebno je definirati brzinu s kojom se ona odvija. Za to služi već spomenuta „*OverV*“ varijabla. Ovisno koja funkcija je aktivna, vrijednost njezine brzine upisuje se u *OverV* varijablu što je vidljivo na slici 7.19.

Upisivanje željene pozicije rotacije radi se u NW10 što je prikazano na slici 7.20. Ako se stroj nalazi u automatskom načinu rada, kao željena pozicija rotacije upisuje se vrijednost izračunata u funkcijskom bloku „5. *Calculate_Last_Sector_Working [FB5]*“. Način računanja te pozicije objasnit će se kasnije u potpoglavlju 6.3.4. Ako se stroj nalazi u parking proceduri u koraku 10 („6. *RSC_DB.PS."010"*“) ili proceduri za reset u koraku 50 („6. *RSC_DB.RRS."050"*“), veličina od 0° upisuje se u „*SinaPos*“ naredbu kako bi se rotacijska os vratila u početni položaj.



Slika 7.20: Upisivanje željene pozicije rotacijske osi

Upisivanje željene brzine u automatskom načinu rada radi se u NW11 što je prikazano na slici 7.21. Prvo grananje u NW11 služi za zapisivanje broja trenutnog sektora u kojemu se stroj nalazi „3. *Machine_DB.ROT_Sec_ACT*“ u privremenu varijablu (eng. *Temp*) imena „*SectorACT*“. Privremena varijabla „*SectorACT*“ mora se koristiti jer je varijabla „3. *Machine_DB.ROT_Sec_ACT*“ globalnog tipa i ne može ju se koristiti kao brojčanu vrijednost unutar druge varijable. U drugom grananju radi se pretvorba brzine rotacije radnog komada iz milimetara u minuti koju unosi korisnik u okretaje u minuti koji se šalju na V90 servo upravljački sustav. Za to su potrebni parametri željene brzina rotacije („32. *Weld_par_Calc_DB.Sector[#SectorACT].Welding_Speed*“) i promjer radnog komada („9. *Work_Prog_DB.Prog.Weld_Par.Weld_Diameter*“).

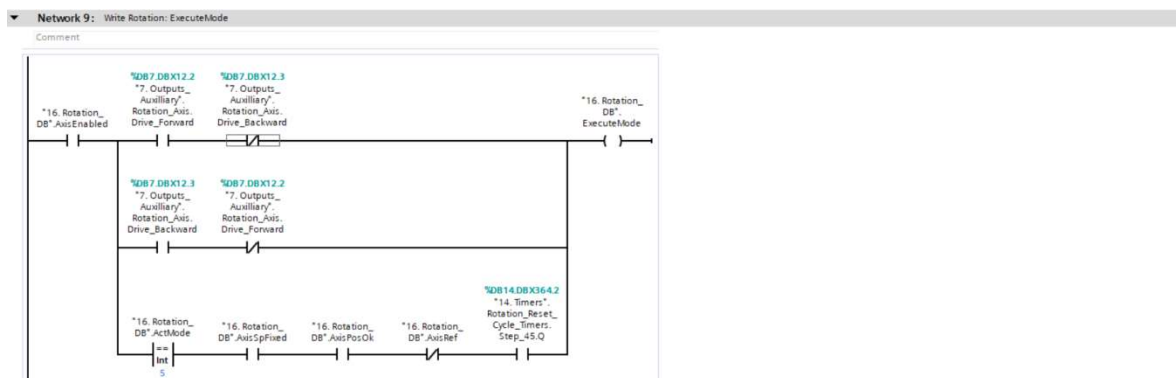


Slika 7.21: Računanje i zapisivanje željene brzine

Promjer radnog komada je konstantan broj u svakom programu i ne mijenja se kroz sektore, no brzina rotacije može varirati od sektora do sektora što znači da njezina vrijednost nikako ne može biti konstanta. Zato je potrebna privremena varijabla s brojem trenutnog sektora. Iz podatkovnog bloka „32. Weld_Par_Calc_DB [DB32]“ uz pomoć privremene varijable „SectorACT“ učitavaju se vrijednosti brzine rotacije radnog komada za trenutni sektor. U podatkovnom bloku „32. Weld_Par_Calc_DB [DB32]“ parametri sektora zapisani su u obliku niza od 20 podatkovnih tipova „9. Sector_Calculated“. Kako je niz sastavljen od 20 istovjetnih podatkovnih tipova, varijabla „32. Weld_Par_Calc_DB".Sector[].Welding_Speed“ nalazi se na istoj poziciji u svakom elementu niza. Zbog toga nakon naziva niza u uglate zagrade upisuje se privremena varijabla „SectorACT“ čija brojčana vrijednost poziva elemente niza u program. Ovisno koji broj je upisan u privremenu varijablu „SectorACT“, podatak brzine zapisan u podatkovni blok „32. Weld_Par_Calc_DB“ učitat će se u kod iz elementa niza „32. Weld_Par_Calc_DB".Sector[]“s istim rednim brojem. Ako bi, na primjer, vrijednost varijable „SectorACT“ iznosila 3, onda bi se u programski kod učitala vrijednost brzine iz trećeg sektora „32. Weld_Par_Calc_DB".Sector[3].Welding_Speed“. Prvi blok u drugoj grani dijeli vrijednost brzine zavarivanja s vrijednošću promjera zavarivanja kako bi se dobila vrijednost brzine rotacije radnog komada. U trećem grananju radi se pretvorba privremene varijable *Welding_Speed_LReal* u koju se sprema rezultat druge grane iz dugačkog realnog tipa podatka (eng. *Long Real*) u dupli cjelobrojni broj (eng. *Double Integer*) koji se zatim preko parametra „16. Rotation_DB".Velocity“ šalje u V90 servo upravljački sustav. Na ulazu u drugo i treće grananje nalaze se bitovi

„*Trigger_BUSINT_X*“ koji služe za aktiviranje grane. Operacije računanja s realnim brojevima i operacije konverzije realnih brojeva u druge tipove podataka zahtijevaju dosta vremena za izvršavanje, pa je preporučljivo koristiti ih samo kada je potrebno. Ako bi se ove operacije izvršavale svaki ciklus PLC uređaja, vrijeme ciklusa bi se znatno povećalo što bi moglo uzrokovati kašnjenja u izvršavanju drugih instrukcija i funkcija PLC uređaja. Bitovi „*Trigger_BUSINT_X*“ služe kako bi se ove operacije izračunale samo u ciklusu nakon što se promijeni broj sektora. Kada se broj sektora promijeni iz 1 u 2 „*Trigger_BUSINT_1*“ će se aktivirati na jedan ciklus i ugasiti sve do promjene u sektor 3. „*Trigger_BUSINT_2*“ aktivirat će se u ciklusu nakon „*Trigger_BUSINT_1*“. Na slici 7.21 vidljivo je kako se prva grana izvršava u svakom ciklusu, druga grana samo kada je aktivan bit „*Trigger_BUSINT_1*“, odnosno samo jedan ciklus nakon što se promjeni broj sektora, i treća grana kada je aktivan bit „*Trigger_BUSINT_2*“, odnosno samo jedan ciklus nakon ciklusa u kojemu je aktivan bit „*Trigger_BUSINT_2*“. Nakon što su svi prethodni parametri određeni, slijedi puštanje motora u pogon.

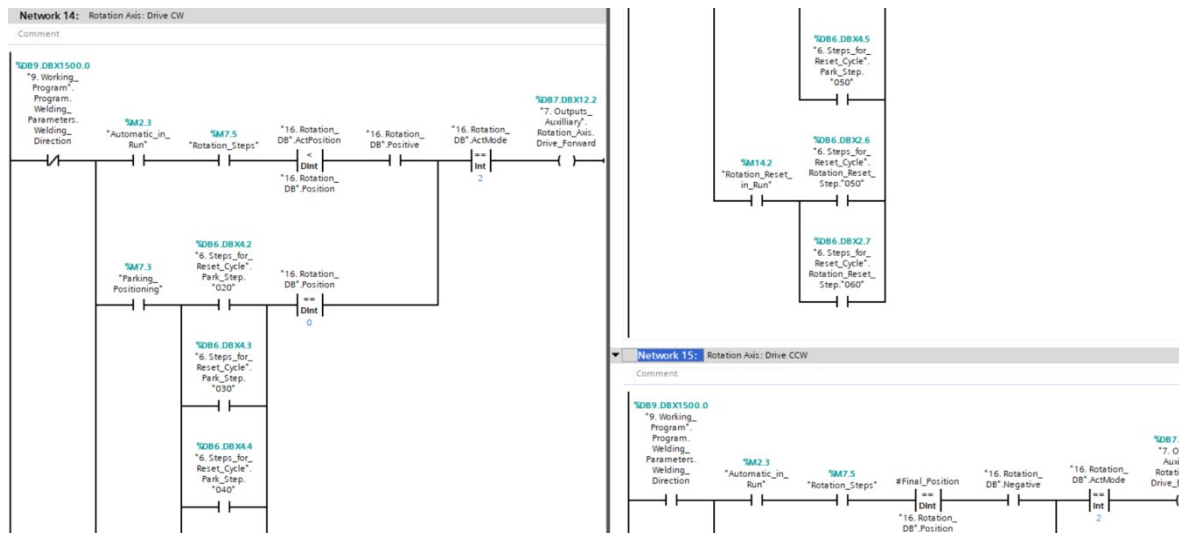
Na slici 7.22 prikazano je kako puštanju motora u pogon prethodi par uvjeta. Prvi od njih je bit „*AxisEnabled*“, odnosno povratna veza s V90 servo upravljačkog sustava koji javlja da je sustav pušten u pogon i da ja na motor puštena struja držanja. Drugim riječima ova povratna veza PLC-u govori kako su V90 servo upravljački sustav i motor spremni za rad.



Slika 7.22: Aktivacija akcije V90 servo upravljačkog sustava

Prve dvije grane nakon bita „*AxisEnabled*“ pokreću motor, dok treća grana V90 upravljačkom sustavu daje signal da trenutnu točku u kojoj se motor nalazi spremi u memoriju kao referentnu točku. Bitovi „*Drive_Forward*“, i „*Drive_Backward*“, su pomoćni sabirni uvjeti za pokretanje motora. U NW14 svi uvjeti koji određuju pokretanje motora spojeni su u bit „*Drive_Forward*“ a u NW15 u bit „*Drive_Backward*“. Bez grana NW14 i

NW15 veliki broj uvjeta koji pale motor nalazio bi se u NW9. Na ovaj način se taj velik broj uvjeta raspodijelio na dvije grane čiji su izlazni uvjeti paralelno spojeni u NW9. Kao dodatnu protekciju u svaku granu dodan je negirani kontakt drugoga uvjeta kako bi se osiguralo da ova dva uvjeta nikada ne budu aktivni u isto vrijeme.



Slika 7.23: Uvjet za pokretanje motora Drive_Forward

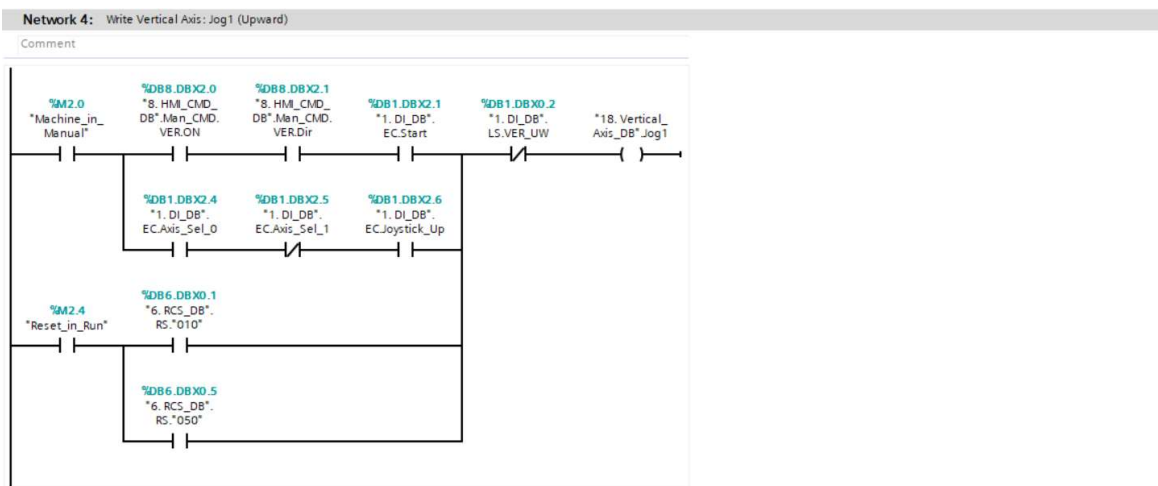
Kao primjer opisat će se uvjeti u NW14 na slici 7.23. Ulazni uvjet grane je negirani kontakt tipke za smjer rotacije na HMI uređaju jer tipka u negativnom položaju označava rotaciju u smjeru kazaljke na satu. Ovaj uvjet vrijedi za sve načine rada i zato se nalazi na ulazu u granu. Nakon njega NW14 se grana na tri uvjeta koji označavaju načine rada u kojima se stroj može nalaziti: automatski ciklus zavarivanja („*Automatic_in_Run*“), pozicioniranje u parking poziciju („*Parking_Positioning*“) i referenciranje („*Rotation_Reset_in_Run*“). Kada se stroj nalazi u automatskom ciklusu zavarivanja uvjet „*Rotation_Steps*“ koji označava korake automatskog ciklusa u kojima je rotacija aktivna mora biti aktivan i stvarna pozicija rotacijske osi mora biti manja od zadane pozicije („*16. Rotation_DB".ActPosition*“ < „*16. Rotation_DB".Position*“). Postavljanje ova dva uvjeta unosi redundantnost u sustav jer se pozicija već provjerava u „*Rotation_Steps*“, no radi sigurnosti i točnosti izvršavanja procesa ipak su stavljena obadva. Ako se stroj nalazi u procesu parkiranja, sekvencijalni funkcijski blok „*5 Rotation_Reset_Cycle [FC5]*“ mora se nalaziti u koraku 20 („*6.RSC_DB".PS."020*“), koraku 30 („*6.RSC_DB".PS."030*“), koraku 40 („*6.RSC_DB".PS."040*“), ili koraku 50 („*6.RSC_DB".PS."050*“) kako bi se signal propustio na izlazni uvjet. Zadnji uvjet u grani provjerava je li pozicija upisana u parametar pozicije V90 servo upravljačkog sustava jednaka 0 kako bi se osiguralo da je

parking pozicija upisana prije nego što se pošalje signal za pokretanje motora. Kada se blok nalazi u proceduri za reset u koracima 50 i 60 („6.RSC_DB".RRS."050" i „6.RSC_DB".RRS."060"), pozicija upisana u V90 servo upravljački sustav također mora biti 0 kako bi se signal propustio na izlazni uvjet za pokretanje motora.

7.2.6 Linearne osi

Blokovi linearnih osi (od [FC7] do [FC12]) služe kao priprema uvjeta za izvršavanje kretanja linearnih osi na SCAR-u. Kako su svi potrebni uvjeti za komunikaciju s V90 servo upravljačkim sustavom i motorom opisana u prethodnom potpoglavlju 6.2.6, u ovome poglavlju opisat će se samo razlike u načinu upravljanja linearnim osima. Budući da se sve linearne osi upravljaju na identičan način, neće se opisati svaka os zasebno već će se za primjer uzeti glavna vertikalna os.

Prva grana koja će se opisati u bloku „9. Vertical_Axis [FC9]” je NW4 u kojoj se aktivira „Jog1” način rada. Grana NW4 prikazana je na slici 7.24.

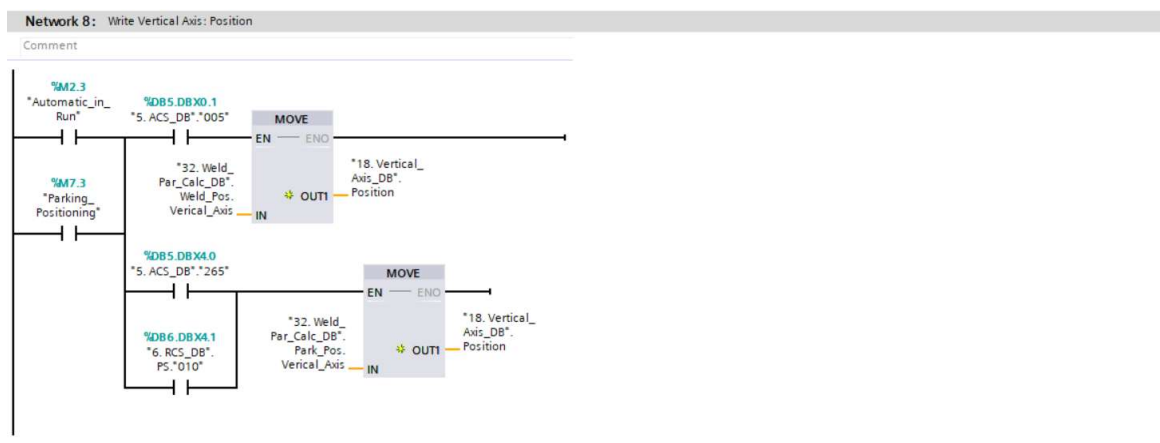


Slika 7.24: Aktivacija Jog1 funkcije vertikalne osi

Kod linearnih osi „JogX” funkcija koristi se u manualnom upravljanju i kod referenciranja. U manualnom načinu rada „Jog” funkcija može se aktivirati na dva načina, što je vidljivo iz grananja nakon uvjeta „Machine_in_Manual” na slici 7.24. Prva grana grananja aktivira se pomoću stranice „Ručno upravljanje” na korisničkom sučelju i tipke „START” na daljinskom upravljaču. Kako bi se aktivirao izlazni uvjet „18. Vertical_axis_DB".Jog1”, bit „8. HMI_CMD_DB".Man_CMD.VER.ON” za aktivaciju vertikalne osi i bit „8. HMI_CMD_DB".Man_CMD.VER.Dir” za odabir smjera kretanja

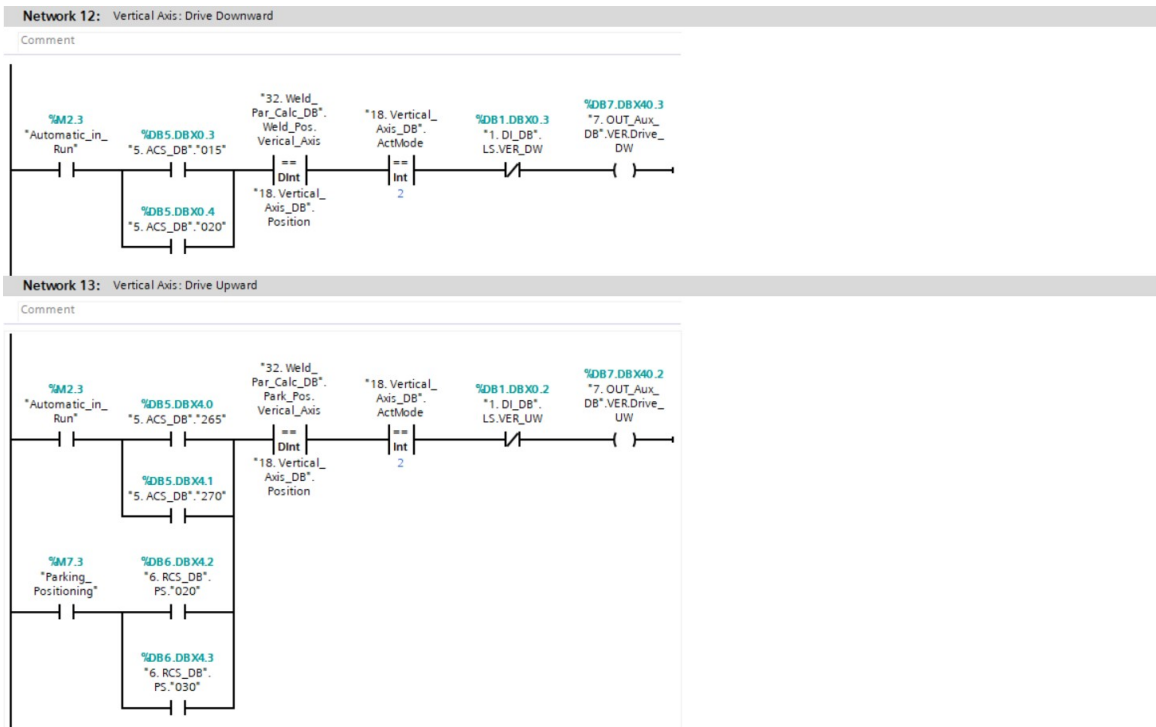
moraju biti aktivni te mora biti pritisnuta tipka „*START*” na daljinskom upravljaču. Druga grana aktivira se u slučaju da operater želi kontrolirati os pomoću komandne palice na daljinskom upravljaču. Kako bi grana bila aktivna, bit „*1.DI_DB".EC.Axis_Sel_0*” mora biti aktivan a bit „*1.DI_DB".EC.Axis_Sel_1*” neaktivan jer to znači da je na daljinskom upravljačku odabrano upravljanje sa glavnim osima stroja. Ako su prva dva uvjeta zadovoljena, operater aktivacijom bita „*1.DI_DB".EC.Joystick_Up*”, odnosno pomicanjem komandne palice u gornji položaj, aktivira izlazni uvjet „*18.Vertical_axis_DB".Jog1*”. U referenciranju, „*Jog1*” funkcija aktivna je samo u koracima 10 i 50, odnosno u koracima gdje se vertikalna os vozi prema referentnom krajnjem prekidaču.

Na slici 7.25 prikazana je grana u kojoj se određuju pozicije vertikalne osi u automatskom ciklusu zavarivanja i parking proceduri, odnosno kada se stroj nalazi u načinu rada apsolutnog pozicioniranja. Ako se stroj nalazi u automatskom ciklusu zavarivanja u koraku 5, radna pozicija trenutno odabranog zavarivačkog programa upisati će se kao pozicija vertikalne osi. Ako se pak stroj nalazi u automatskom ciklusu zavarivanja u koraku 265 ili u parking proceduri u koraku 10, parking pozicija trenutno odabranog zavarivačkog programa upisati će se kao pozicija vertikalne osi



Slika 7.25: Određivanje pozicije vertikalne osi kod apsolutnog pozicioniranja

Nakon zapisivanja pozicija mora se odrediti u kojim slučajevima se apsolutno pozicioniranje radi, što je prikazano na slici 7.26. Apsolutno pozicioniranje u grani NW12 može se aktivirati samo ako donji krajnji prekidač vertikalne osi nije aktivan i ako se stroj nalazi u načinu rada apsolutnog pozicioniranja.



Slika 7.26: Određivanje aktivacije apsolutnog pozicioniranja vertikalne osi

U automatskom ciklusu zavarivanja apsolutno pozicioniranje će se aktivirati u koracima 15 ili 20 ako je vrijednost radne pozicije trenutno odabranog zavarivačkog programa „32. *Weld_Par_Calc_DB*.*Weld_Pos.Vertical_Axis*” jednaka vrijednosti upisanoj u varijablu „18. *Vertical_Axis_DB*.*Position*”.

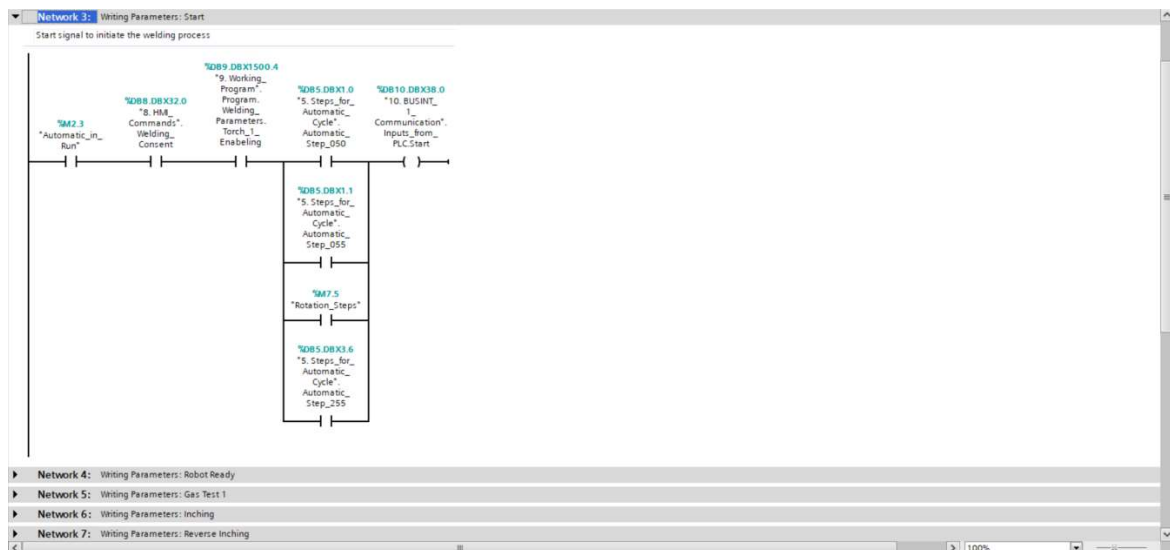
U grani NW13 apsolutno pozicioniranje može se aktivirati samo ako gornji krajnji prekidač vertikalne osi nije aktivan, ako se stroj nalazi u načinu rada apsolutnog pozicioniranja i ako je vrijednost parking pozicije trenutno odabranog zavarivačkog programa „32. *Weld_Par_Calc_DB*.*Park_Pos.Vertical_Axis*” jednaka vrijednosti upisanoj u varijablu „18. *Vertical_Axis_DB*.*Position*”. Kako bi se apsolutno pozicioniranje aktiviralo stroj se mora nalaziti u automatskom ciklusu zavarivanja u koracima 265 ili 270 ili u parking proceduri u koracima 20 i 30.

7.2.7 BUSINT

Blok „*BUSINT X [FC14 / FC15]*“ služi za određivanje uvjeta koji se šalju na BUSINT TETRIX X11 komunikacijski modul te s njega prosljeđuju na aparat za zavarivanje. Ovaj blok pali uvjete zavarivanja, izračunava sve potrebne vrijednosti i radi konverzije njihovih tipova podataka u oblik priklan za slanje na komunikacijski modul.

Na slici 7.27 prikazan je prvi parametar u NW3 koji se šalje na komunikacijski modul BUSINT TETRIX X11 i govori aparatu za zavarivanje da pokrene struju zavarivanja. Da bi uvjet „*10. BUSINT_1_DB*“.PLC_BUSINT.Start“ bio ispunjen stroj se mora nalaziti u automatskom ciklusu, korisnik na HMI uređaju mora aktivirati tipku za zavarivanje, rad s gorionikom mora biti aktiviran u učitanoj zavarivačkoj programu i sekvencijalni automatski ciklus mora se nalaziti između koraka 50 i 255 (koraci od 60 do 250 paralelno su povezani u uvjet „*Rotation_Steps*“). Tek kada su svi ti uvjeti ispunjeni, PLC komunikacijskom modulu šalje bit za pokretanje struje zavarivanja koji se zatim proslijedi dalje na aparat za zavarivanje.

U NW2 nalazi se bit „*10. BUSINT_1_DB*“.PLC_BUSINT.Robot_Ready“ koji aparatu za zavarivanje šalje signal da je stroj spreman za zavarivanje, odnosno da je stroju u radnoj poziciji. NW5 služi za manualnu kontrolu zaštitnog plina na gorioniku i koristi se za aktivaciju početnog i završnog plina. Manualna kontrola žice za zavarivanje kontrolira se u NW5 za izvlačenje i u NW6 za uvlačenje žice. Žicom se manualno upravlja samo kod podešavanja zavarivačkog procesa kako bi korisnik namjestio poziciju žice koja mu odgovara za radni komad.

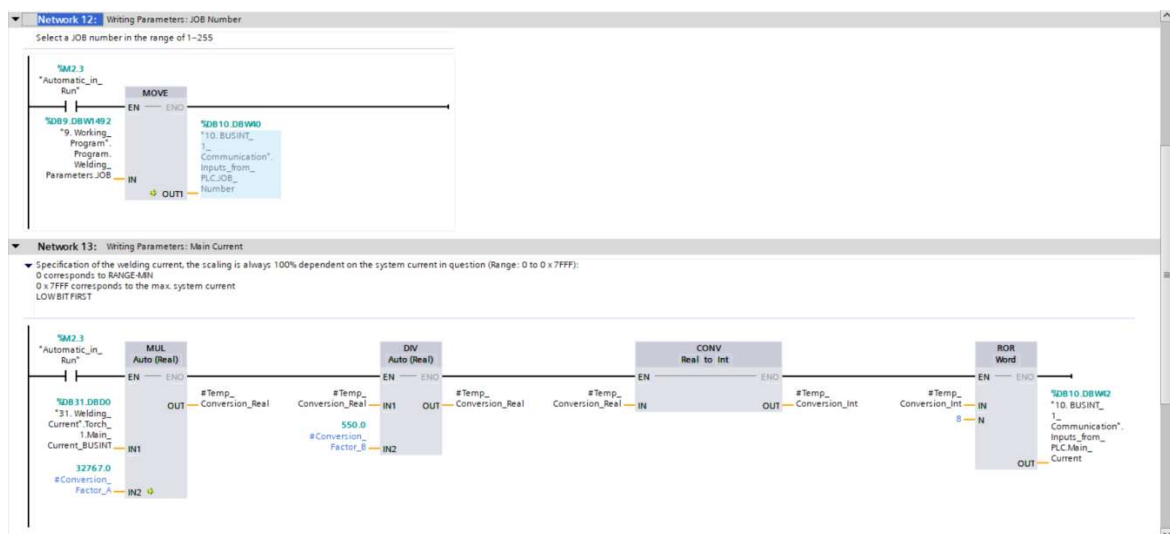


Slika 7.27: Uvjeti za aktivaciju zavarivanja na TETRIX aparatu za zavarivanje

Kako bi aparat znao koji zavarivački proces se koristi u automatiziranom zavarivanju, mora mu se poslati broj JOB-a tog zavarivačkog procesa, što je prikazano na slici 7.28. To je interna vrijednost proizvođača aparata za zavarivanje (EWM GmbH) koji sadržava vrijednosti bitne za zavarivački proces kao što su vrsta materijala, vrsta plina i vrsta zavara (npr. kutni, čeon) koji se koristi. JOB može poprimiti vrijednost od 1 do 256. Na stroju N1

koriste se JOB-ovi od 200 do 218 jer su oni predviđeni za automatizirano zavarivanje i većina parametara kontrolira se pomoću vanjskih signala, u ovom slučaju PLC uređaja.

Ostali JOB-ovi služe za ručno ili poluautomatsko zavarivanje. Oni imaju predefiniраних 16 programa s parametrima zavarivanja koji imaju istu funkciju kao i sektori na stroju N1. Samo JOB-ovima od 200 do 218 moguće je slati proizvoljne parametre poput jakosti struje. U svim ostalim JOB-ovima ti parametri su već definirani i zapisani u memoriju aparata za zavarivanje te se pozivaju brojem JOB-a i brojem programa unutar JOB-a. Tablica za brojevima JOB-ova, njihovim programima i svim parametrima JOB-a nalazi se u korisničkom priručniku svakog aparata za zavarivanje.

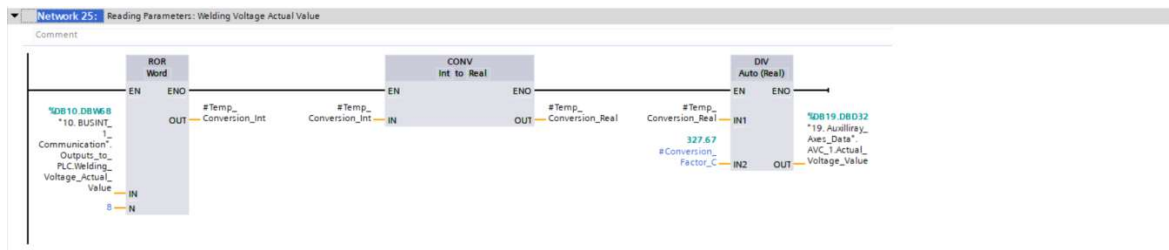


Slika 7.28: Zapisivanje broja JOB-a i konverzija vrijednosti jakosti struje zavarivanja

Slika 7.28 također prikazuje i jedan od glavnih parametara zavarivačkog procesa koji se šalju na komunikacijski modul, a to je vrijednost jakosti struje zavarivanja „31. Weld_Ramp_DB".T1.Main_BUSINT“. Korisnik preko HMI uređaja unosi vrijednost jakosti struje zavarivanja koja se sprema u memoriju HMI uređaja pod broj programa koji je učitao u PLC uređaj. Kada se pokrene automatski ciklus, blok „BUSINT X Parameters [FC1X]“ preračunava vrijednost upisanu u program u vrijednost adekvatnu ulazu u komunikacijski modul. Naime, komunikacijski modul prima vrijednosti od 0 do 32767 pa se korisnički upisana vrijednost mora linearno skalirati na vrijednost prihvatljivu komunikacijskom modulu. U korisničkom priručniku BUSINT TETRIX X11 komunikacijskog modula piše kako je vrijednost 32767 adekvatna maksimalnoj vrijednosti jakosti struje koju aparat može postići, odnosno 550 ampera prema korisničkom priručniku aparata za zavarivanje. Kako se tijekom izračuna vrijednosti ne bi zaokruživale na

cjelobrojne vrijednosti, korisnički upisana vrijednost u kalkulaciju ulazi kao realna vrijednost. Kada se vrijednost izračuna, pretvara se u cjelobrojnu vrijednost i funkcija ROR (eng. *Rotate on Right*) radi rotaciju na 8 bitu. Za vrijednost jakosti struje u komunikacijski modul mora se poslati informacija od jedne riječi, odnosno 16 bitova, u obliku niži bit prvi (eng. *low byte first*). Zbog toga se koristi funkcija ROR koja radi pomak pozicija bitova u riječi s povratom otpisanih bitova na kraj riječi. Varijabla N ($N = 8$) definira poziciju bita od kojeg se radi rotacija. Ako bi se funkcijom ROR željela rotirati riječ „12345678“ s pozicijom bita od kojeg se radi rotacija $N=4$, izlazni rezultat bio bi $OUT = „56781234“$.

Posljednja grana N25 u bloku „*BUSINT X Parameters [FC1X]*“ služi za čitanje parametara s komunikacijskog modula i prikazana je na slici 7.29. Moguće je čitati četiri parametra, no za ovo programsko rješenje jedini bitan parametar je napon ili razlika potencijala električnog luka koja se koristi za AVC kontrolu. Proces čitanja parametara započinje ROR funkcijom jer komunikacijski modul šalje vrijednosti u PLC u obliku riječi s nižim bitom prvim. Nakon rotacije riječi, vrijednost se konvertira iz cjelobrojne vrijednosti u realnu kako bi se dobila decimalna vrijednost očitano parametra. Posljednji korak je linearno skalirati vrijednost u oblik prihvatljiv korisniku. Proces čitanja parametara s komunikacijskog modula teče u suprotnom smjeru od procesa zapisivanja parametara na komunikacijski modul.



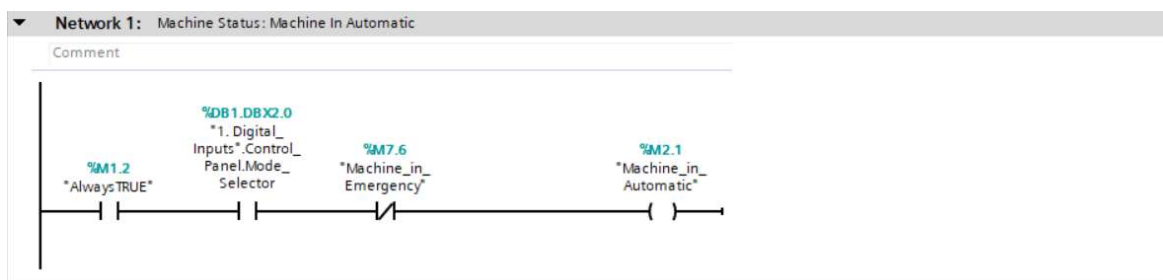
Slika 7.29: Očitavanje vrijednosti napona električnog luka

7.2.8 Status stroja

Blok „*16. Machine_Status [FC16]*“ je blok koji služi za aktiviranje bitova koji definiraju način rada u kojima se stroj N1 nalazi i odabrani način rada prikazuju na početnoj stranici pod varijablom naziva „*Status stroja*“.

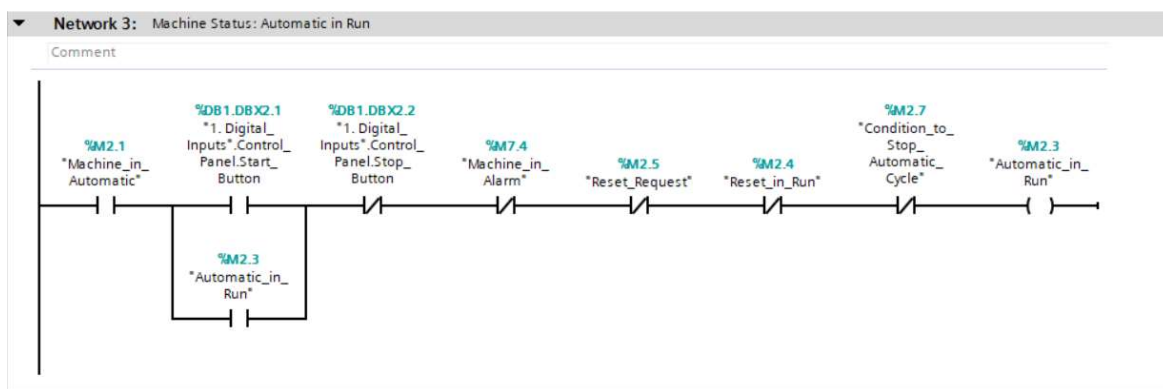
Na slici 7.30 prikazan je uvjet za automatski način rada. Kako bi stroj N1 bio u automatskom načinu rada, dvopoložajna sklopka za odabir načina rada mora biti u poziciji

TRUE i bit „*Machine_Emergency*“ koji se aktivira u slučaju pritisnute sigurnosne gljive ne smije biti aktivan.



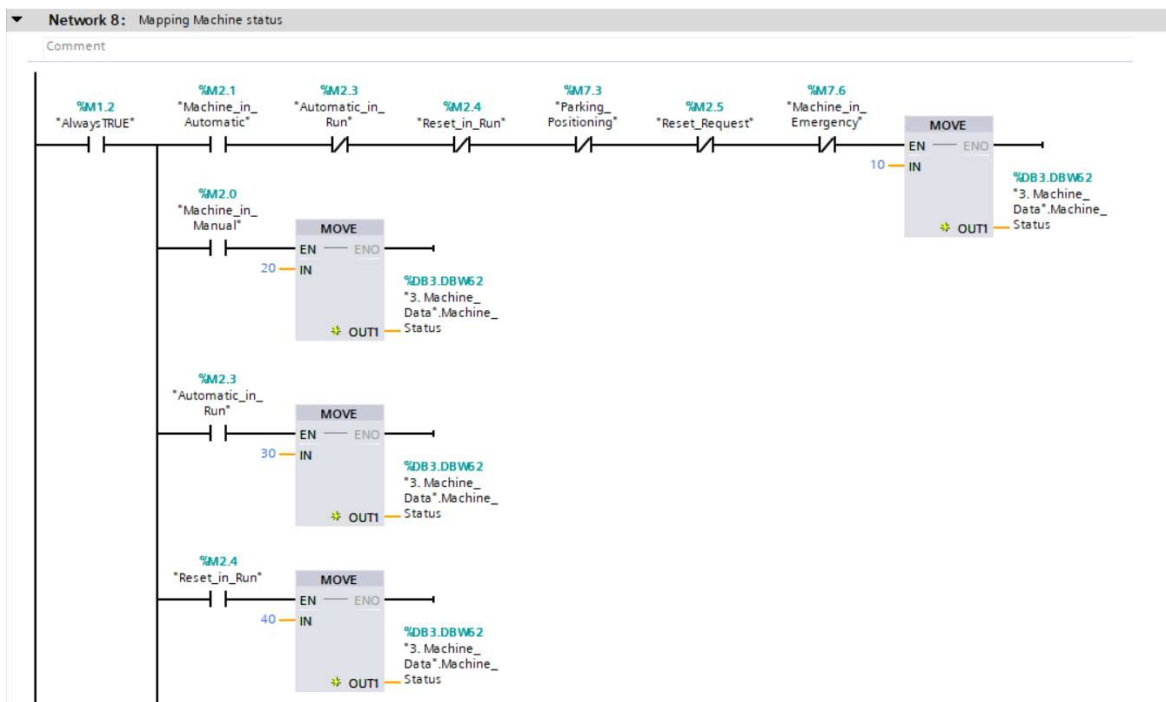
Slika 7.30: Odabir automatskog načina rada

Ako operater želi pokrenuti automatski ciklus zavarivanja, stroj se mora nalaziti u automatskom načinu rada. Automatski ciklus pokreće se pritiskom na tipku „*START*“ i to samo ako tipka „*STOP*“ nije pritisnuta, ako nije aktivan niti jedan alarm ili upozorenje, ako nije aktivan bit za referenciranje, ako procedura za reset nije aktivna i ako je uvjet za zaustavljanje automatskog ciklusa negativan. U paralelnu liniju s tipkom „*START*“ dodan je kontakt s bitom izlaznog uvjeta „*Automatic_in_Run*“ koji granu drži aktivnom i nakon što korisnik otpusti tipku start. Bez paralelne linije automatski ciklus bio bi aktivan samo dok korisnik drži tipku „*START*“ pritisnutom. Ovako jednim pritiskom na tipku start korisnik aktivira uvjet „*Automatic_in_Run*“ koji zatim cijelu liniju, odnosno samog sebe, drži aktivnim. U elektrotehnici, točnije u relejnoj tehnici ova funkcija naziva se samodržanje. Uvjet „*Automatic_in_Run*“ ostaje aktivan sve dok jedan od negiranih kontakata ne postane pozitivan. U tom slučaju prekida se prolaz signala, izlazni uvjet „*Automatic_in_Run*“ prestaje biti aktivan i zaustavlja se automatski ciklus. Uvjeti aktivacije automatskog ciklusa prikazani su na slici 7.31.



Slika 7.31: Aktivacija automatskog ciklusa

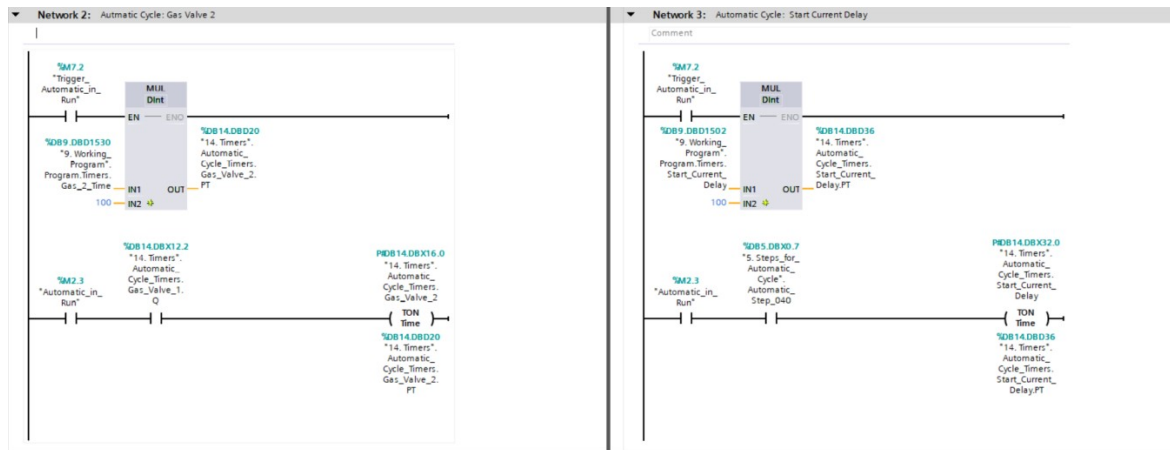
Kao što je već navedeno, blok „16.Machine_Status [FC16]“ također radi vizualizaciju na HMI uređaju. Ovisno koji način rada je aktivan, u varijablu „Machine_Status“ upisuje se brojevana vrijednost sukladna tom načinu rada, što je prikazano na slici 7.32. Brojevana vrijednost varijable „Machine_Status“ zatim na HMI uređaju pod „Status stroja“ ispisuje naziv aktivnog načina rada. Na primjer, ako je u varijablu „Machine_Status“ upisan broj 10, na HMI uređaju prikazat će se tekst „Stroj u automatskom načinu rada“, a ako je upisan broj 30 na HMI uređaju prikazat će se tekst „Automatski ciklus u radu“. Budući da se načini rada automatski ciklus, referenciranje stroja i parking pozicioniranje mogu aktivirati samo kada je stroj u automatskom načinu rada, u prvoj grani su svi ti uvjeti negirani. U suprotnome bi PLC u varijablu „Machine_Status“ naizmjenice upisivao vrijednosti 10 i 30 kada je automatski ciklus zavarivanja aktivan. To naizmjenično upisivanje različitih vrijednosti u „Machine_Status“ rezultiralo bi naizmjeničnim promjenama teksta na HMI uređaju što bi moglo zbuniti operatera. S negiranim kontaktima automatskog ciklusa, referenciranja stroja i parking pozicionirana u varijablu „Machine_Status“ broj 10 upisat će se samo kada je stroj u automatskom načinu rada i u stanju mirovanja. Čim se pokrene neki od prije navedenih načina rada, negirani kontakt tog načina rada postaje aktivan i deaktivira upisivanje vrijednosti 10 u varijablu „Machine_Status“.



Slika 7.32: Definiranje parametra Machine_Status za vizualizaciju na korisničkom sučelju

7.2.9 Brojači vremena

Blok „17. Timers [FC17]“ je blok koji u sebi sadržava brojače vremena i uvjete koji ih aktiviraju. Na slici 7.33 prikazana su dva primjera brojača vremena, prvi za aktiviranje drugog plina za oblikovanje, a drugi za odgodu paljenja električnog luka nakon što se stroj pozicionira.



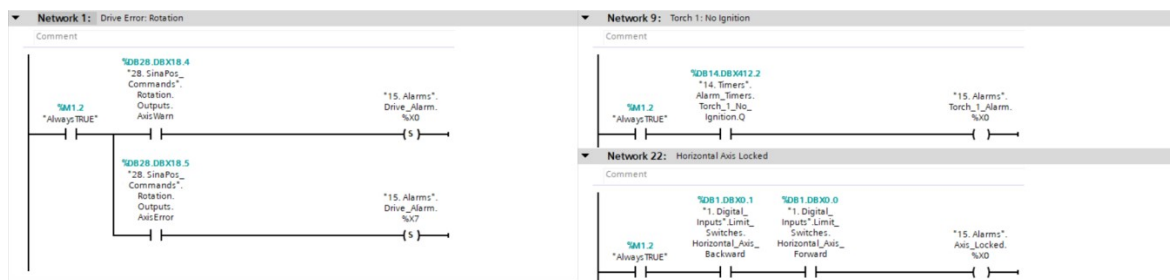
Slika 7.33: Primjeri brojača vremena

Prva grana u obadva primjera je zapisivanje korisnički zadanog vremena s HMI uređaja u memoriju brojača. Varijabla se preko HMI uređaja zapisuje u memoriju PLC uređaja kao dupla cjelobrojna varijabla pa ju dodatno treba pomnožiti sa 100 kako bi se dobila ekvivalentna vrijednost u vremenskoj varijabli (eng. *Time*). Korisnik na HMI uređaju ima opciju unosa vrijednosti s jednim decimalnim mjestom, ali to decimalno mjesto samo je vizualni efekt. U stvarnosti ono ne postoji jer, kao što je već navedeno, podatak se sprema u cjelobrojnu varijablu koja ne može sadržavati decimalna mjesta. To znači da kada korisnik unese vrijednost od 2.5 sekundi, u memoriju PLC uređaja spremi će se cjelobrojna vrijednost 25. Vremenska varijabla kao najmanju vrijednost unosa ima milisekundu, pa bi direktno zapisivanje vrijednosti 25 u vremensku varijablu rezultiralo vrijednošću od 25 ms. Zbog toga se prije zapisivanja vrijednosti u vremensku varijablu cjelobrojna vrijednost 25 pomnoži sa 100 kako bi se dobila vrijednost 2500. Ako se vrijednost 2500 spremi u vremensku varijablu, dobije se vrijeme od 2500 ms, odnosno 2.5 s što je korisnik u konačnici želio dobiti. U drugoj grani aktivira se vremenski brojač. Na lijevoj strani slike 6.30 prikazan je brojač za vrijeme tečenja drugog plina za oblikovanje. On će se aktivirati samo kada je automatski ciklus zavarivanja aktivan i nakon što odbroji brojač za prvi plin za oblikovanje. Bit „14. Timers_DB“.AC.Forming_1.Q“ je izlazni

uvjet brojača za prvi plin za zavarivanje koji se aktivira kada brojač odbroji korisnički zadano vrijeme. Izlazni uvjet iz brojača drugog plina za oblikovanje nalazi se u koraku 30 bloka „1. Automatic_Cycle [FC1]“ kao povratna veza s brojača. Kada brojač odbroji korisnički zadanu vrijednost, aktivira se bit „14. Timers_DB.AC.Forming_2.Q“ i program prijeđe na korak 40. Na slici 7.30 desno prikazan je vremenski brojač koji služi kao odgoda paljenja električnog luka nakon što se stroj pozicionira u radnu poziciju. Vremenski brojač aktivira se kada automatski ciklus zavarivanja uđe u korak 40. Kada vremenski brojač odbroji korisnički zadano vrijeme, aktivira se bit „14. Timers_DB.AC.Start_Current.Q“ u bloku „1. Automatic_Cycle [FC1]“ i program prijeđe na korak 50.

7.2.10 Alarmi

Blok „18. Alarms [FC18]“ je blok koji služi kao sabirnica svih sigurnosnih uvjeta PLC uređaja i vizualizira alarme na HMI uređaju. Na HMI uređaj se kao alarmi šalju varijable tipa riječ gdje svaki bit označava jedan alarm. Svi alarmi s V90 servo upravljačkih sustava nalaze se u riječi „15. Alarms_DB.Driver_Alarms“. Na slici 7.34 lijevo prikazani su alarmi s V90 servo upravljačkog sustava rotacijske osi. Upozorenje s V90 servo upravljačkog sustava rotacijske osi aktivira bit „15. Alarms_DB.Driver_Alarms.%X0“, a greška bit „15. Alarms_DB.Driver_Alarms.%X0“.



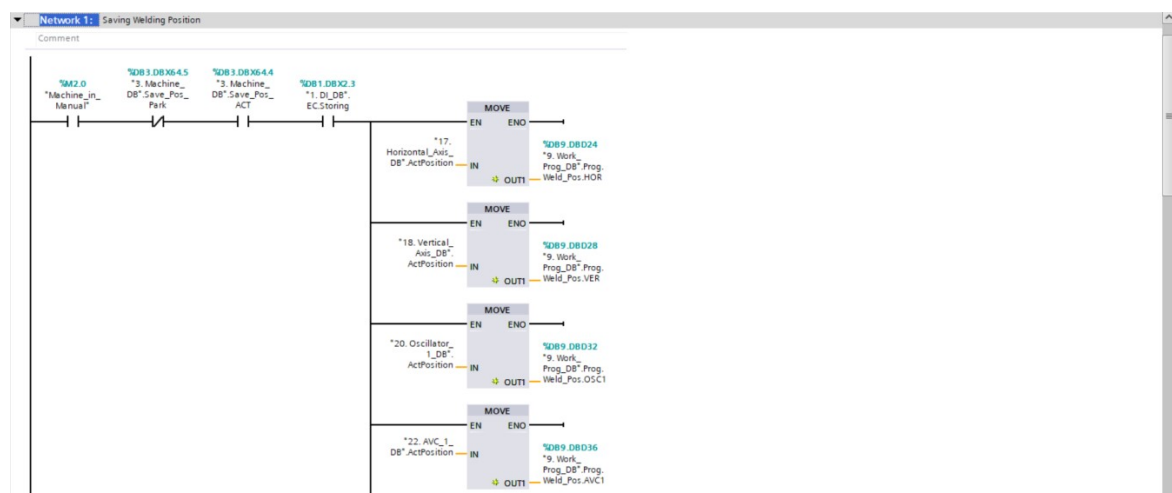
Slika 7.34: Primjeri aktivacije alarma

Na slici 7.34 gore desno nalazi se NW9 s greškom koja se aktivira ako se električni luk ne uspostavi unutar 5 sekundi nakon što je poslan signal za uspostavu električnog luka koju aktivira izlazni uvjet brojača vremena. Na komunikacijskom modulu postoji povratni signal koji PLC uređaju javlja da je električni luk uspostavljen. Kada se s PLC uređaja na komunikacijski modul pošalje signal za uspostavu električnog luka, aktivira se brojač vremena za aktivaciju alarma. Ako se s komunikacijskog modula na PLC uređaj vrati

signal kako je električni luk na gorioniku za zavarivanje uspostavljen, aktivirat će se bit „11. BUSINT_2_DB“.BUSINT_PLC."I>0" i njegov negirani kontakt blokirat će signal prema brojaču vremena te se alarm neće aktivirati. U suprotnome brojač vremena izbrojat će 5 sekundi i aktivirati alarm „Torch X: No Ignition“. Na slici 7.34 dolje nalazi se NW22 s greškom koja se aktivira ako su obadva krajnja prekidača horizontalne osi pritisnuta u isto vrijeme. Aktivacija obadva krajnja prekidača istodobno je nedozvoljeno stanje stroja stoga se javlja greška koja zaustavlja stroj jer je potrebna dodatna inspekcija kvara.

7.2.11 Spremanje pozicije

Blok „19. Saving_Position [FC19]“ je blok u kojemu se radi spremanje radne i parking pozicije, te procedura za osvježavanje tagova pozicija na HMI uređaju. Na slici 7.35 prikazan je dio bloka za spremanje pozicije.



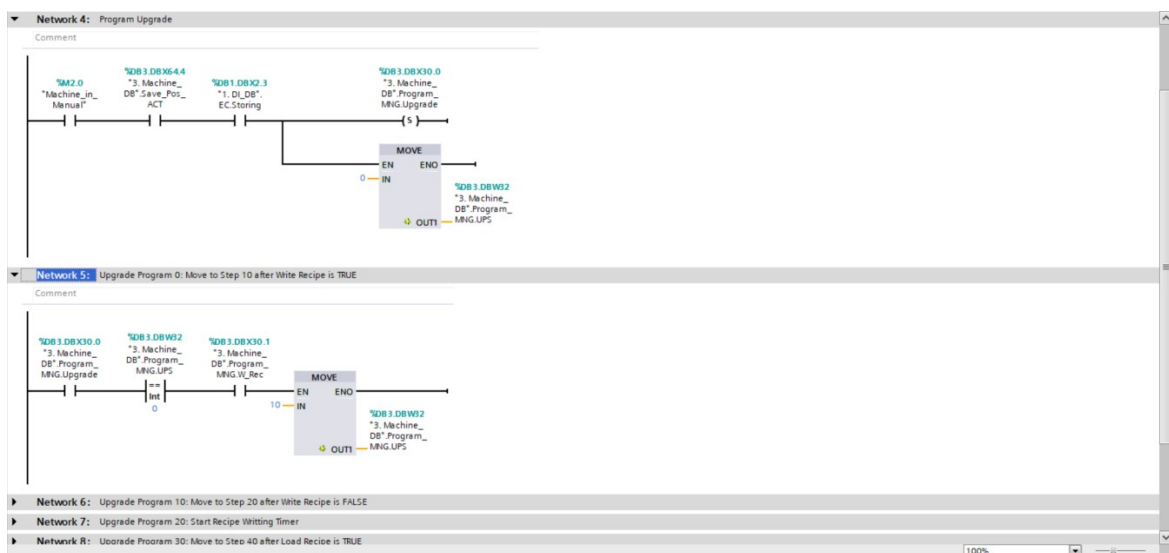
Slika 7.35: Spremanje pozicija

Prije spremanja pozicije korisnik mora aktivirati funkciju spremanja pozicije na HMI uređaju. Tipka za aktiviranje funkcije spremanja pozicije služi kao zaštita jer bez njene aktivacije tipka „STORING“ na daljinskom upravljaču nema funkciju. Time se onemogućava da slučajnim pritiskom tipke „STORING“ korisnik neku neželjenu poziciju spremi kao radnu ili parking poziciju. Korisnik spremanje i radne i parking pozicije radi pomoću tipke „STORING“ na daljinskom upravljaču. Kako bi PLC uređaj razlikovao koju poziciju korisnik želi spremi, na HMI uređaju se nalazi prekidač „POZICIJA“ pomoću kojeg korisnik odabire radnu ili parking poziciju.

Prva grana NW1 koristi se za spremanje radne pozicije. Radna pozicija spremi će se samo kada je stroj u manualnom načinu rada („*Machine_in_Manual*“), kada nije aktiviran bit za spremanje parking pozicije („*3. Machine_DB.Save_Pos_Park*“), kada je funkcija spremanja pozicije aktivirana („*3. Machine_DB.Save_ACT*“) i kada korisnik pritisne tipku „*STORING*“ („*DI_DB.EC.Storing*“) na daljinskom upravljaču.

Druga grana NW2 koristi se za spremanje parking pozicije. Jedina razlika između grana NW1 i NW2 je u drugom bitu „*3. Machine_DB.Save_Pos_Park*“ koji u grani NW2 treba biti aktivan kako bi se parking pozicija spremila.

U granama od NW4 do NW13 prikazanim na slici 7.36 izvršava se sekvencijalna procedura osvježavanja tagova parametara pozicija i njihovo zapisivanje u recepte na HMI uređaju. Ova procedura aktivira bitove u HMI uređaju koji će osvježiti stranicu „*PARAMETRI POZICIJA*“. Bez ove procedure korisnik bi morao izaći iz stranice i ponovno ući u nju kako bi mu se nove vrijednosti prikazale na ekranu. Procedura prvo čeka da se svi podatci zavarivačkog programa učitaju iz PLC uređaja u HMI uređaj i spremi u recept. Nakon toga novi recept šalje se nazad u PLC uređaj i osvježavaju se tagovi na HMI ekranu kako bi nove pozicije bile vidljive na stranici „*PARAMETRI POZICIJA*“. Više o receptima i HMI tagovima bit će objašnjeno u poglavlju 8.



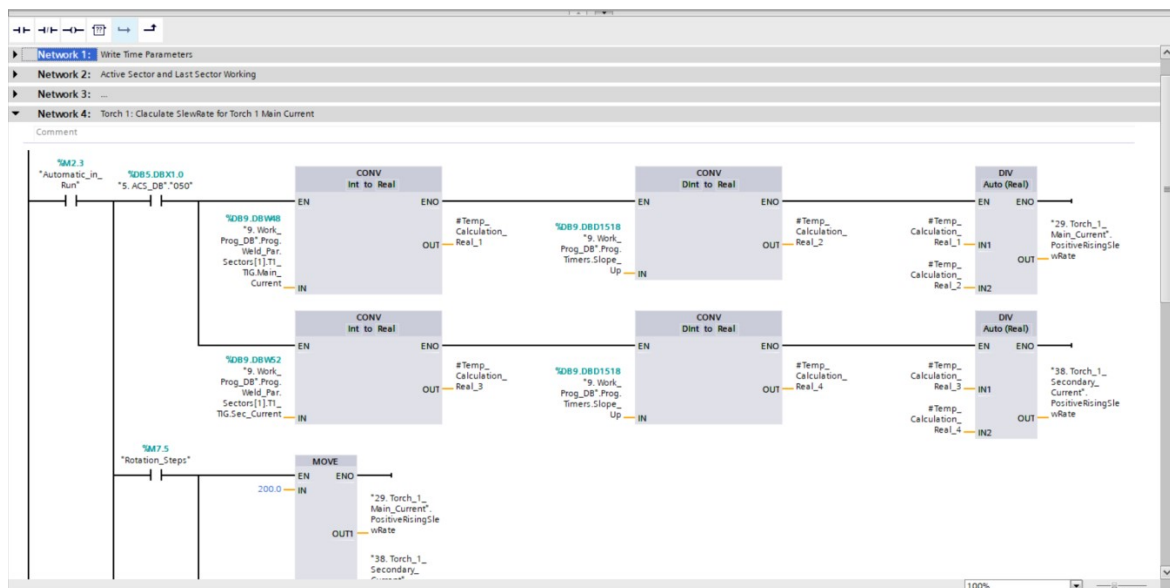
Slika 7.36: Procedura spremanja novog recepta i osvježavanja tagova

7.2.12 Rampe struje zavarivanja

Blok „*20. Welding_Ramps [FC20]*“ je blok koji radi kalkulaciju i pripremu podataka za uspješno izvođenje rampi struje zavarivanja u bloku cikličkog prekida „*Cyclic_interrupt*“

[OB30]“. U prve dvije grane radi se upis podataka i njihovih pretvorbi u oblike koji su potrebni za daljnji rad. Kod upisivanja vremena rampe u prvoj grani, osigurava se da vrijeme rampe ne bude 0 sekundi. Minimalna vrijednost koja će se upisati u vremensku vrijednost rampe je 100 ms.

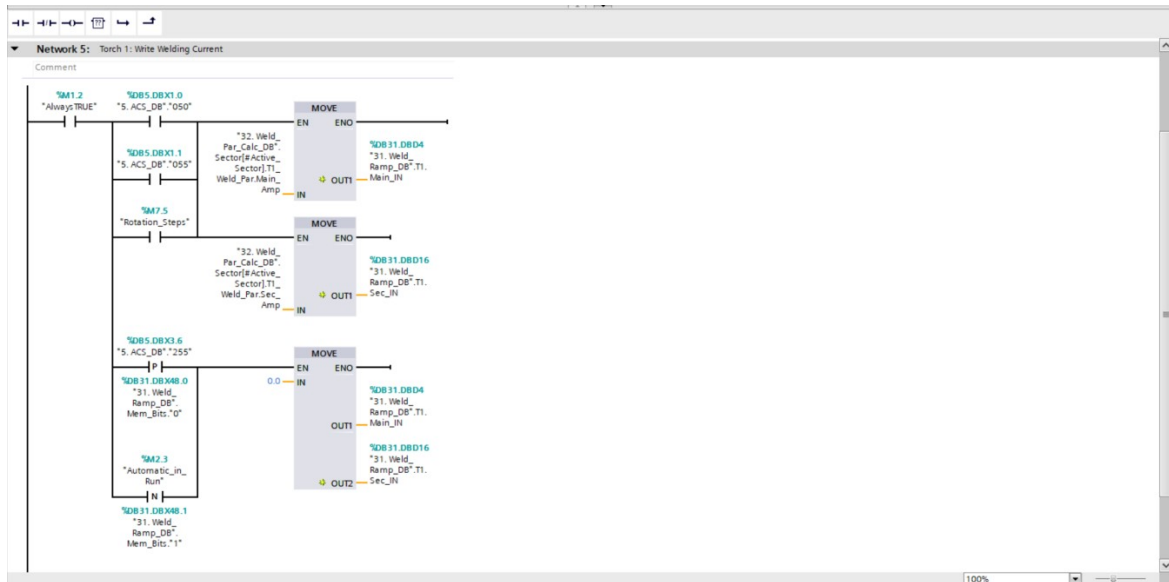
Na slici 7.37 prikazana je grana 3 u kojoj se računa korak (eng. *Slew rate*) rampe struje zavarivanja. U koraku 50 automatskog ciklusa zavarivanja vrijednost struje prvog sektora dijeli se s vremenom zadanim za rampu za dizanje struje zavarivanja. Rezultat je korak rampe struje zavarivanja, odnosno koliko ampera se rampa mora podići u sekundi kako bi dosegla zadanu vrijednost struje u zadanom vremenu trajanja rampe. Kalkulacija se radi za glavnu i pulsnu struju zavarivanja. Nakon toga, dok se stroj nalazi u sektorima kao korak rampe upisuje se vrijednost od 200 ampera po sekundi kako bi se prijelaz između struja zavarivanja svakog sektora dogodio što brže. Nakon što se zavarivanje u sektorima završi, stroj dolazi do mjesta prevara i započinje rampa spuštanja struje zavarivanja. Kalkulacija koraka rampe za spuštanje struje zavarivanja ista je kao i u slučaju računanja koraka rampe za dizanje struje zavarivanja.



Slika 7.37: Kalkulacija koraka rampe struje zavarivanja

Na slici 7.38 prikazana je grana NW5 u kojoj se zapisuju vrijednosti jakosti struje zavarivanja u glavnu varijablu jakosti struje zavarivanja koja se šalje u rampu struje zavarivanja. U koracima 50, 55 i koracima rotacije automatskog ciklusa u rampu struje zavarivanja upisuju se vrijednosti izračunate u bloku za kalkulaciju parametara zavarivanja. Ovisno koji sektor je trenutno aktiviran, podatak njegove jakosti struje

zavarivanja upisat će se u glavnu varijablu jakosti struje zavarivanja i poslati u rampu struje zavarivanja. Kada automatski ciklus dođe u korak 255 gdje započinje spuštanje jakosti struje zavarivanja ili ako se automatski ciklus prekine, u glavnu varijablu jakosti struje zavarivanja upisat će se vrijednost 0 i poslati u rampu struje zavarivanja.



Slika 7.38: Zapisivanje iznosa jakosti struje u rampe struje zavarivanja

7.3 Funkcijski blokovi

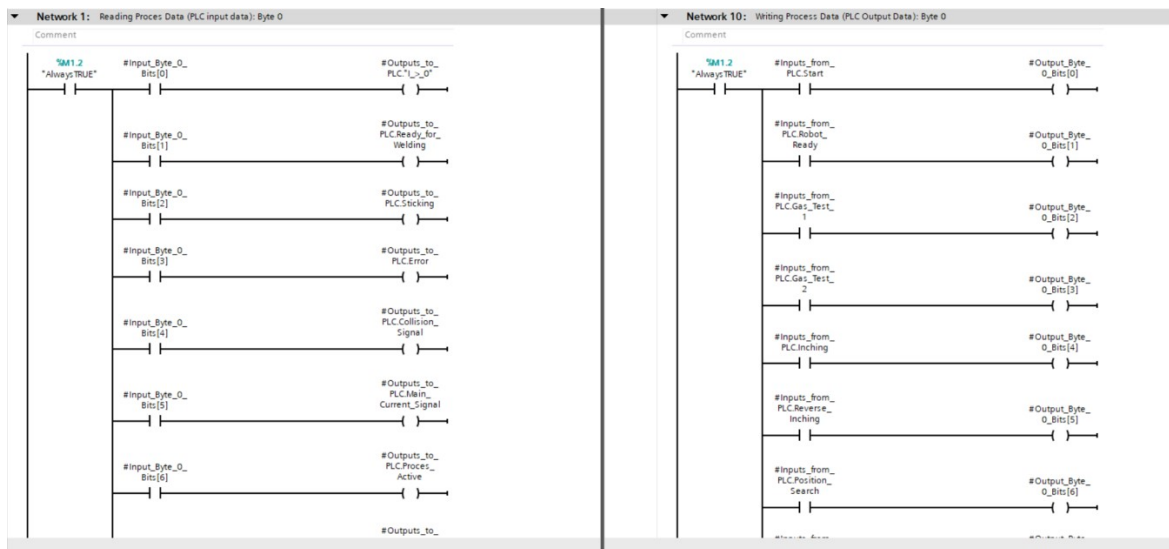
Funkcijski blokovi, kao što je već objašnjeno na početku poglavlja 5, su blokovi koji imaju predodređene memorijske lokacije za spremanje podataka. Za razliku od funkcija, pozivanje funkcijskog bloka u program zahtijeva i definiranje njegovog podatkovnog bloka u koji će se spremati podatci.

7.3.1 BUSINT TETRIX X11 komunikacija

Komunikacija s BUSINT TETRIX X11 modulima odvija se preko funkcijskog bloka „1. *BUSINT Communication[FBI]*“. Ovaj blok služi kao univerzalni blok za prijenos podataka iz podatkovnog bloka „IX. *BUSINT_IX_DB*“ na BUSINT TETRIX X11 komunikacijski modul i obratno. Funkcijski blok za komunikaciju s BUSINT X11 komunikacijskim modulom prikazan je na slici 7.39.

Prijenos podataka odvija se preko PROFINET komunikacije. Blok je napravljen kao univerzalni jer stroj N1 radi s dva aparata za zavarivanje, pa sukladno tome i s dva komunikacijska modula. Za slanje podataka na BUSINT X11 komunikacijski modul

napravljeni su ulazi u blok za lijeve strane bloka. Na desnoj strani bloka nalaze se izlazi pomoću kojih se podatci s BUSINT X11 modula šalju na EWM aparate za zavarivanje.



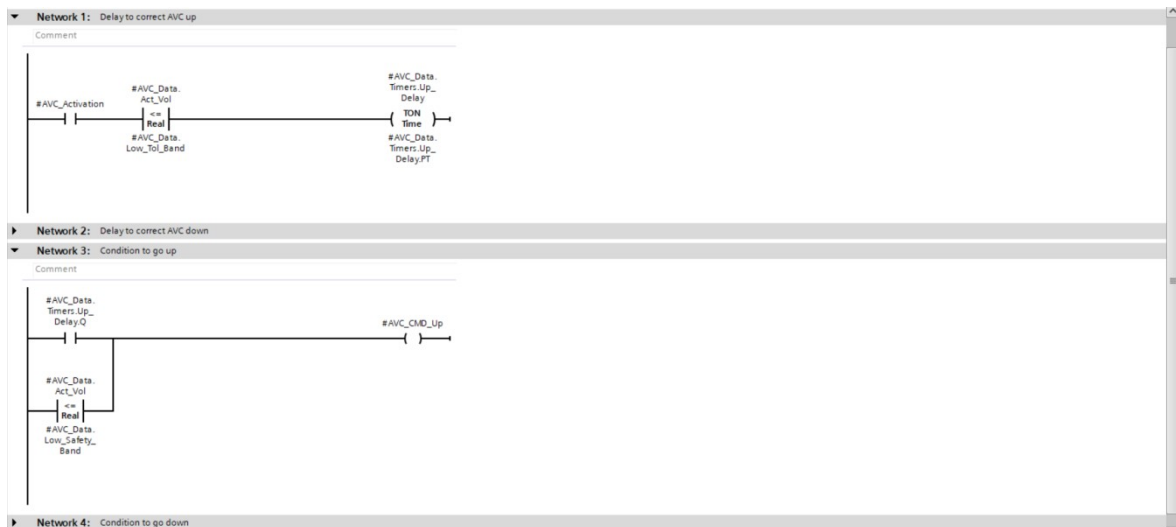
Slika 7.39: Slanje i primanje podataka s BUSINT X11 komunikacijskog modula

7.3.2 AVC kontrola

Funkcijski blok „*AVC_CTRL_FB [FB3]*“ je univerzalni blok koji služi za regulaciju automatske kontrole napona električnog luka. U suštini ovaj blok na temelju usporedbe stvarne vrijednosti napona električnog luka i korisnički postavljene vrijednosti upravlja pozicijom AVC osi. Blok ima mogućnost upravljanja samo s jednom osi. Iz tog razloga je napravljen kao univerzalan blok kako bi ga se pozvalo u programu i samo spojilo ulazne parametre s osi s kojom se želi upravljati.

Prve dvije grane NW1 i NW2 aktiviraju vremenske brojače za odgodu aktivacije AVC kontrole. Vremenski brojač za odgodu korekcije pozicije prema gore prikazan je na slici 7.40 (gore). Naime, električni napon neće uvijek biti konstantna vrijednost, nego njegova vrijednost varira oko neke točke. Zbog toga korisnik u postavkama sustava mora definirati tolerancijsko polje u kojemu se smatra da je korisnički napon postignut i vrijeme nakon kojega će se AVC kontrola aktivirati kada stvarna vrijednost električnog napona ispadne iz tolerancijskog polja. Ako korisnik postavi željenu vrijednost napona električnog luka na 12 volta, tolerancijsko polje na +/- 0.5 volta i vrijeme odgode aktivacije AVC kontrole na 1 sekundu, dok god se stvarna vrijednost napona električnog luka nalazi između 11.5 i 12.5 volta, os koja se kontrolira će mirovati. Ako stvarna vrijednost napona električnog luka ispadne iz tolerancijskog polja, na primjer na vrijednost 11 volta, aktivira se brojač

vremena koji je postavljen na 1 sekundu. Kada jedna sekunda protekne, a stvarna vrijednost napona električnog luka je i dalje ispod donje tolerancijske vrijednosti od 11.5 V, blok će motoru poslati signal da podigne os kako bi se stvarna vrijednost napona električnog luka povećala i vratila u tolerancijsko polje. Isti princip rada je i za spuštane stvarne vrijednosti napona električnog luka, smo što se gleda je li stvarna vrijednost napona električnog luka premašila gornju tolerancijsku vrijednost. U grani NW3 aktivira se signal koji motoru govori da podigne os, a u NW4 signal koji motoru govori da spusti os. U NW3 i NW4 također se provjerava je li stvarna vrijednost napona električnog luka ispala iz sigurnosnog polja. Sigurnosno polje je veće tolerancijsko polje koje služi kao dodatna kontrola stvarne vrijednosti napona električnog luka i sprječava naglo smanjivanje ili povećanje njezine vrijednosti. Kada stvarna vrijednost napona električnog luka ispadne iz sigurnosnog polja, stroj neće čekati vremensku odgodu kao kod tolerancijskog polja, nego odmah šalje signal za korekciju pozicije na motor. Signal za podizanje AVC osi prema gore prikazan je na slici 7.40 (dolje).



Slika 7.40: AVC funkcija

Funkcijski blok „*AVC_CTRL_FB [FB3]*“ pozvan je u funkcijama „*11. AVC_1 [FB11]*“ i „*13. AVC_2 [FB13]*“. Na primjeru „*11. AVC_1 [FB11]*“ objasnit će se spajanje ulaznih uvjeta u funkcijski blok „*AVC_CTRL_FB [FB3]*“. Prvi ulazni parametar je „*AVC_Activation*“ koji aktivira AVC kontrolu. Na taj ulazni parametar spojen je bit „*7. "OUT_Aux_DB".Aux_Cond.T1_AVC_Act*“. Ovaj bit nalazi se u bloku „*2. Outputs [FB2]*“ kao izlazna naredba grane NW3 koja je objašnjena u poglavlju 6.2.2. Drugi ulazni parametar je skup parametara definiran PLC podatkovnim tipom „*AVC_Data*“. U

podatkovnom bloku „23. *AVCI_CTRL_DB*“ funkcijskog bloka „3. *AVC_CTRL_FC*“ kao ulazni parametar definiran je PLC podatkovni tip „*AVC_Data*“ čiji parametri se pozivaju u programu. Zbog toga se na taj ulaz funkcijskog bloka „3. *AVC_CTRL_FC*“ mora spojiti kompatibilan tip podatka, odnosno također PLC podatkovni tip „*AVC_Data*“ koji je definiran u podatkovnom bloku „19. *Aux_Axes_DB [DB19]*“. U podatkovnom bloku nalaze se svi parametri potrebni za AVC kontrolu poput korisnički definirane vrijednosti napona električnog luka, vrijednosti tolerancijskog i sigurnosnog polja te ostali uvjeti i pomoćni uvjeti potrebni za pravilan rad AVC kontrole. Svi parametri računaju se samo u drugom ciklusu PLC uređaja nakon što se aktivira automatski ciklus zavarivanja.

7.3.3 Kalkulacija parametara za zavarivanje

Funkcijski blok „4. *Weld_Par_Calc_FB [FB4]*“ prvi je blok u kojemu se koristi SCL ili strukturirani kontrolni jezik (eng. *Structured Control Language*). SCL je visoki tekstualni programski jezik baziran na PASCAL programskom jeziku i definiran standardom IEC 1131-3 za standardne programske jezike kod programiranja PLC uređaja.

Funkcijskim blokom „4. *Weld_Par_Calc_FB [FB4]*“ računaju se svi parametri automatskog ciklusa zavarivanja i spremaju u „32. *Weld_Par_Calc_DB [DB32]*“. Funkcijski blok „4. *Weld_Par_Calc_FB [FB4]*“ poziva se nakon svakog učitavanja zavarivačkog programa u PLC uređaj. Kada se automatski ciklus zavarivanja pokrene, parametri izračunati u „32. *Weld_Par_Calc_DB [DB32]*“ samo se pozivaju u automatski ciklus. Ovakvo rješenje drastično smanjuje ciklus PLC uređaja u automatskom ciklusu jer se izbjegavaju sva nepotrebna računanja. Niti jedan parametar zavarivanja nije dinamički parametar pa ih nema ni potrebe izračunavati u svakom ciklusu PLC uređaja. Ako bi se ovi parametri ipak računali u svakom ciklusu PLC uređaja, unutar jednog sektora PLC uređaj bi kao rezultat kalkulacije dobivao iste vrijednosti što je totalno besmisleno. S druge strane kalkulacija parametara zavarivanja zahtijeva pretvorbu parametara iz cjelobrojnih u realne vrijednosti, kalkulacije s realnim vrijednostima, i ponovnu pretvorbu u cjelobrojne vrijednosti. Vrijeme izvođenje operacije vrijednostima s pomičnim zarezom, kao što su računanje i pretvaranje realnih brojeva, iznosi 2.3 μ s dok operacije s cjelobrojnim vrijednostima, kao što su pozivanje cjelobrojnih vrijednosti iz podatkovnih blokova, iznosi 1.7 μ s. Ako bi se u svakom ciklusu PLC uređaja izvodila operacija računanja parametara zavarivanja, ona bi iznosila 10.3 μ s (konverzija iz cjelobrojne u realnu vrijednost 2.3 μ s, računanje realne vrijednosti 2.3 μ s, konverzija iz realne u cjelobrojnu vrijednost 2.3 μ s,

konverzija iz cjelobrojne vrijednosti u riječ 1.7 μ s i rotacija riječi za 8 bitova 1.7 μ s). Ako se vrijednosti parametara zavarivanja izračunaju prije početka zavarivanja, jedina operacija koja se dešava u ciklusu PLC uređaja za vrijeme automatskog ciklusa zavarivanja jest premještanje podatka iz podatkovnog bloka u program koja iznosi 1.7 μ s. To je smanjenje trajanja vremena izvođenja operacije od skoro 10 puta. Na bazi jednog ciklusa računanje parametara iznosilo bi 153 μ s u svakom ciklusu, a ako se parametri izračunaju prije automatskog ciklusa zavarivanja 0 sekundi, osim u prvom ciklusu nakon promjene sektora, gdje bi zapisivanje novih parametara trajalo 44.2 μ s. Gledano iz ljudskog aspekta i poimanja vremena kao procesa 153 μ s su neznatjan broj, ali u automatizaciji gdje par mikrosekundi može biti ključno za uspješan rad procesa, ušteda od 153 μ s po ciklusu PLC uređaja ima veliki značaj. S jedne strane smanjio se vremenski ciklus PLC uređaja čime se automatski povećalo izvršavanje drugih funkcija, a s druge strane PLC uređaj se ne optereti s računanjem nepotrebnih realnih brojeva u svakom ciklusu.

```

1 // Home Position Calculation
2 #Park_Pos.Horizontal_Axis := "S. Work_Prog_DB".Prog.Home_Pos.HOB;
3 #Park_Pos.Vertical_Axis := "S. Work_Prog_DB".Prog.Home_Pos.VER;
4 #Park_Pos.Oscillator_1 := "S. Work_Prog_DB".Prog.Home_Pos.OSC1;
5 #Park_Pos.AVC_1 := "S. Work_Prog_DB".Prog.Home_Pos.AVC1;
6 #Park_Pos.Oscillator_2 := "S. Work_Prog_DB".Prog.Home_Pos.OSC2;
7 #Park_Pos.AVC_2 := "S. Work_Prog_DB".Prog.Home_Pos.AVC2;
8
9 // Welding Position Calculation
10 #Weld_Pos.Horizontal_Axis := "S. Work_Prog_DB".Prog.Weld_Pos.HOB;
11 #Weld_Pos.Vertical_Axis := "S. Work_Prog_DB".Prog.Weld_Pos.VER;
12 #Weld_Pos.Oscillator_1 := "S. Work_Prog_DB".Prog.Weld_Pos.OSC1;
13 #Weld_Pos.AVC_1 := "S. Work_Prog_DB".Prog.Weld_Pos.AVC1;
14 #Weld_Pos.Oscillator_2 := "S. Work_Prog_DB".Prog.Weld_Pos.OSC2;
15 #Weld_Pos.AVC_2 := "S. Work_Prog_DB".Prog.Weld_Pos.AVC2;
16
17 #Counter := 1;
18
19 //Sector Parameters Calculation
20 FOR #Counter := 1 TO 20 DO
21
22
23 IF "S. Work_Prog_DB".Prog.Weld_Par.Sectors[#Counter].Sec_Act = TRUE THEN
24
25 // Calculate Welding Speed
26 #Sector[#Counter].Weld_Speed := DINT_TO_REAL("S. Work_Prog_DB".Prog.Weld_Par.Sectors[#Counter].Weld_Speed);
27
28 // Calculate Sector Final Position
29 #Sector[#Counter].Sec_Final_Pos := "S. Work_Prog_DB".Prog.Weld_Par.Sectors[#Counter].Sec_Final_Pos * 100;
30
31 // Calculate Shielding Gas
32 #Temp_Calculation_Real_1 := INT_TO_REAL("S. Work_Prog_DB".Prog.Weld_Par.Shield_Gas);
33 #Temp_Calculation_Real_2 := (#Temp_Calculation_Real_1 * 163.835);
34 #Temp_Calculation_Int := REAL_TO_INT(IN := #Temp_Calculation_Real_2);
35 #Temp_Calculation_Word := INT_TO_WORD(#Temp_Calculation_Int);
36 #Sector[#Counter].T1_Weld_Par.Shield_Gas := BOR_WORD(IN := #Temp_Calculation_Word, N := 8);
37 #Sector[#Counter].T2_Weld_Par.Shield_Gas := BOR_WORD(IN := #Temp_Calculation_Word, N := 8);
38
39 // Torch 1 Calculation
40 IF "S. Work_Prog_DB".Prog.Weld_Par.T1_Act THEN
41

```

Variable Name	Data Type
"S. Work_Prog_DB".Prog.Home_Pos.HOB	VD89
"S. Work_Prog_DB".Prog.Home_Pos.VER	VD89
"S. Work_Prog_DB".Prog.Home_Pos.OSC1	VD89
"S. Work_Prog_DB".Prog.Home_Pos.AVC1	VD89
"S. Work_Prog_DB".Prog.Home_Pos.OSC2	VD89
"S. Work_Prog_DB".Prog.Home_Pos.AVC2	VD89
"S. Work_Prog_DB".Prog.Weld_Pos.HOB	VD89
"S. Work_Prog_DB".Prog.Weld_Pos.VER	VD89
"S. Work_Prog_DB".Prog.Weld_Pos.OSC1	VD89
"S. Work_Prog_DB".Prog.Weld_Pos.AVC1	VD89
"S. Work_Prog_DB".Prog.Weld_Pos.OSC2	VD89
"S. Work_Prog_DB".Prog.Weld_Pos.AVC2	VD89
"S. Work_Prog_DB".Prog.Weld_Par.Sectors[#Counter].Weld_Speed	VD89
"S. Work_Prog_DB".Prog.Weld_Par.Sectors[#Counter].Sec_Final_Pos	VD89
"S. Work_Prog_DB".Prog.Weld_Par.Shield_Gas	VD89
"S. Work_Prog_DB".Prog.Weld_Par.T1_Act	VD89

Slika 7.41: Zapisivanje parametara pozicija i kalkulacija parametara zavarivanja

Parametri pozicija računaju se na početku bloka što je prikazano na slici 7.41. Nakon toga na red dolazi FOR petlja. FOR petlja sve naredbe unutar petlje izvršava dok je vrijednost varijabla „#Counter“ manja od vrijednosti varijable 20. Kada vrijednost varijable „#Counter“ dosegne vrijednost varijable 20, FOR petlja izvršit će se posljednji put. Vrijednost varijable inkrementa može biti zanemarena ako joj je vrijednost 1 kao u ovom slučaju, za sve druge vrijednosti varijabla inkrementa mora se definirati.

FOR petlja u funkcijskom bloku „4. Weld_Par_Calc_FB [FB4]“ izvršit će se 20 puta. Prije FOR petlje brojač petlje „#Counter“ postavlja se na vrijednost 1 kako bi se osiguralo da petlja svaki puta krene od vrijednosti 1 kada se funkcijski blok „4. Weld_Par_Calc_FB [FB4]“ pozove. Varijabla „#Counter“ također se nalazi u svim parametrima zavarivanja unutar FOR petlje koji mogu biti različiti u svakom sektoru kao broj sektora iz kojeg se varijabla za računanje poziva i kao varijabla sektora u koju se izračunata vrijednost zapisuje. Drugim riječima, ako varijabla „#Counter“ ima vrijednost 3, kod računanja glavne struje zavarivanja pozvat će se vrijednost glavne struje zavarivanja iz sektora 3 i izračunata vrijednost zapisat će se u podatkovni blok „32. Weld_Par_Calc_DB [DB32]“ pod vrijednosti trećeg sektora. Kalkulacija parametara zavarivanja prikazana je na slici 7.42.

Lin.	OP.	FOR.	WHILE.	IF.	REGION
39				// Torch I Calculation	
40				IF "9. Work_Prog_DB".Prog.Weld_Par.TI_Act THEN	"9. Work_Prog_DB" NDB9
41					
42				// Calculate Main Current	
43				#Temp_Calculation_Reel_1 := INT_TO_REAL("9. Work_Prog_DB".Prog.Weld_Par.Sectors[#Counter].TI_IIG.Main_Current);	"9. Work_Prog_DB" NDB9
44				#Sector[#Counter].TI_Weld_Par.Main_Amp := #Temp_Calculation_Reel_1 / 10;	
45					
46				// Calculate Secondary Current	
47				#Temp_Calculation_Reel_1 := INT_TO_REAL("9. Work_Prog_DB".Prog.Weld_Par.Sectors[#Counter].TI_IIG.Sec_Current);	"9. Work_Prog_DB" NDB9
48				#Sector[#Counter].TI_Weld_Par.Sec_Amp := #Temp_Calculation_Reel_1 / 10;	
49					
50				// Calculate Main Current Duration	
51				#Temp_Calculation_Reel_1 := INT_TO_REAL("9. Work_Prog_DB".Prog.Weld_Par.Sectors[#Counter].TI_IIG.Main_Current_Dur);	"9. Work_Prog_DB" NDB9
52				#Temp_Calculation_Reel_2 := (#Temp_Calculation_Reel_1 * 327.67);	
53				#Temp_Calculation_Int := REAL_TO_INT(IN := #Temp_Calculation_Reel_2);	
54				#Temp_Calculation_Word := INT_TO_WORD(#Temp_Calculation_Int);	
55				#Sector[#Counter].TI_Weld_Par.Main_Amp_Dur := ROR_WORD(IN := #Temp_Calculation_Word, N := 8);	
56					
57				// Calculate Secondary Current Duration	
58				#Temp_Calculation_Reel_1 := INT_TO_REAL("9. Work_Prog_DB".Prog.Weld_Par.Sectors[#Counter].TI_IIG.Sec_Current_Dur);	"9. Work_Prog_DB" NDB9
59				#Temp_Calculation_Reel_2 := (#Temp_Calculation_Reel_1 * 327.67);	
60				#Temp_Calculation_Int := REAL_TO_INT(IN := #Temp_Calculation_Reel_2);	
61				#Temp_Calculation_Word := INT_TO_WORD(#Temp_Calculation_Int);	
62				#Sector[#Counter].TI_Weld_Par.Sec_Amp_Dur := ROR_WORD(IN := #Temp_Calculation_Word, N := 8);	
63					
64				// Calculate Wire Speed	
65				#Temp_Calculation_Reel_1 := INT_TO_REAL("9. Work_Prog_DB".Prog.Weld_Par.Sectors[#Counter].TI_IIG.Wire_Speed);	"9. Work_Prog_DB" NDB9
66				#Temp_Calculation_Reel_2 := (#Temp_Calculation_Reel_1 * 6.5534);	
67				#Temp_Calculation_Int := REAL_TO_INT(IN := #Temp_Calculation_Reel_2);	
68				#Temp_Calculation_Word := INT_TO_WORD(#Temp_Calculation_Int);	
69				#Sector[#Counter].TI_Weld_Par.Wire_Speed := ROR_WORD(IN := #Temp_Calculation_Word, N := 8);	
70					
71				// Calculate AVC Setpoint	
72				#Sector[#Counter].TI_Weld_Par.AVC_SF := "9. Work_Prog_DB".Prog.Weld_Par.Sectors[#Counter].TI_IIG.AVC_Setpoint;	"9. Work_Prog_DB" NDB9
73					
74				// Calculate Oscillator I Parameters	
75				#Sector[#Counter].TI_OSC_Par.Range_Left := "9. Work_Prog_DB".Prog.Weld_Par.Sectors[#Counter].OSCI.Range_Left * 100;	"9. Work_Prog_DB" NDB9
76				#Sector[#Counter].TI_OSC_Par.Range_Right := "9. Work_Prog_DB".Prog.Weld_Par.Sectors[#Counter].OSCI.Range_Right * 100;	"9. Work_Prog_DB" NDB9
77				#Sector[#Counter].TI_OSC_Par.Pause_Left := "9. Work_Prog_DB".Prog.Weld_Par.Sectors[#Counter].OSCI.Pause_Left / 10;	"9. Work_Prog_DB" NDB9
78				#Sector[#Counter].TI_OSC_Par.Pause_Right := "9. Work_Prog_DB".Prog.Weld_Par.Sectors[#Counter].OSCI.Pause_Right / 10;	"9. Work_Prog_DB" NDB9
79				#Sector[#Counter].TI_OSC_Par.Speed := "9. Work_Prog_DB".Prog.Weld_Par.Sectors[#Counter].OSCI.Speed;	"9. Work_Prog_DB" NDB9
80					

Slika 7.42: Kalkulacija parametara zavarivanja

7.3.4 Kalkulacija zadnjeg aktivnog sektora

Funkcijski blok „5. Calc_Last_Sec_FC [FC5]“ služi za računanje zadnjeg aktivnog sektora u zavarivačkom programu i njegove konačne pozicije, a napisan je u SCL programskom jeziku. Kalkulacija zadnjeg aktivnog sektora prikazana je na slici 7.40.

Kao i kod računanja parametara zavarivanja, u bloku se nalazi FOR petlja koja će se izvršiti 20 puta. Unutar petlje nalazi se IF naredba koja provjerava je li sektor s brojem varijable „#Counter“ iz FOR petlje aktiviran. Samo ako je uvjet IF petlje istinit, petlja će se izvršiti. Unutar petlje nalazi se i varijabla „Last_Sec_Work“ u koju se sprema vrijednost

varijable „#Counter“ ako je uvjet IF naredbe zadovoljen. Ako uvjet IF naredbe nije zadovoljen znači da sektor s brojem varijable „#Counter“ nije aktivan i u tome slučaju se izvršava dio naredbe nakon naredbe ELSE. U dijelu koda nakon ELSE nalazi se deaktivacija svih sektora nakon prvog deaktiviranog sektora. Kao što je već spomenuto, sektori moraju biti definirani redom od sektora 1 pa nadalje kako bi se osigurao ispravan rad automatskog ciklusa zavarivanja. Aktiviranje sektora napreskokce unosi grešku u zavarivački program i automatski ciklus zavarivanja neće korisniku pružiti željeni rezultat. Korisniku su na HMI uređaju na stranici „PARAMETRI SEKTORA“ uvijek prikazani svi aktivni sektori i prvi deaktivirani sektor. Postoji mogućnost da, primjerice, prilikom smanjivanja broja sektora s 10 sektora na 5, korisnik prvo deaktivira sektor 6 čime sektori 7, 8, 9 i 10 postaju nevidljivi na stranici „PARAMETRI SEKTORA“, ali ostaju i dalje aktivni što rezultira neispravnim zavarivačkim programom. Zbog toga je u FOR petlji osigurano da se pojavom prvog deaktiviranog sektora izvrši dio programskog koda nakon naredbe ELSE, odnosno deaktivacija sljedećeg sektora. Kada varijabla „#Counter“ dosegne vrijednost 6, IF uvjet neće biti zadovoljen i izvršit će se naredba ELSE, odnosno deaktivirat će se sektor 7. Nakon toga FOR petlja vraća se na početak i vrijednost varijable „#Counter“ povećava se na 7. Budući da je sektor 7 deaktiviran u prethodnom prolasku kroz FOR petlju, IF uvjet ponovno nije zadovoljen i ELSE naredba deaktivira sektor 8. FOR petlja kroz sve sljedeće prolaskes deaktivira ostale sektore sve do sektora 20.

```

1 #Counter := 1;
2
3 FOR #Counter := 1 TO 20 DO
4   // Check if sector is enabled
5   IF ("9. Work_Prog_DB".Prog.Weld_Par.Sectors[#Counter].Sec_Act = TRUE) THEN
6     // If sector is enabled, write sector number in Last Sector Working
7     #Last_Sec_Work := #Counter;
8   ELSE
9     "9. Work_Prog_DB".Prog.Weld_Par.Sectors[#Counter + 1].Sec_Act := FALSE;
10  END_IF;
11 END_FOR;
12
13 #Last_Sec_Pos := "32. Weld_Par_Calc_DB".Sector[#Last_Sec_Work].Sec_Final_Pos + "9. Work_Prog_DB".Prog.Weld_Par.Slope_Down_Degr;
14

```

Slika 7.40: Kalkulacija zadnjeg aktivnog sektora i njegove pozicije

7.4 PLC tagovi

PLC tagovi su procesne veličine koje služe za komunikaciju i razmjenu informacija PLC uređaja s okolinom, SCADA sustavima i HMI uređajima. Sve ulazne i izlazne signale, tj. signale sa senzora i signale za aktuatore, mora se definirati kao PLC tagove.

U pravilu, sve procesne veličine koje se žele prikazati korisniku na HMI uređaju moraju se definirati kao globalni tagovi u PLC uređaju. No, u TIA Portalu moguće je i procesne veličine iz podatkovnih blokova povezati s HMI uređajem. U ovom

programskom rješenju svi ulazi i izlazi PLC uređaja, kao i procesne veličine za komunikaciju s BUSINT TETRIX X11 komunikacijskim modulima i barkod čitačem definirani su kao globalni tagovi. Sve ulazne i izlazne procesne veličine moraju se definirati kao globalni PLC tagovi. Također sve procesne veličine koje služe za definiranje načina rada stroja N1 definirane su kao globalni PLC tagovi. Tagovi definirani kao ulazi nalaze se na lijevoj strani, a tagovi definirani kao izlazi na desnoj strani slike 7.41.

The screenshot displays two panels in the SIMATIC Manager interface. The left panel, titled '1. Digital_Inputs', lists 30 digital input tags. The right panel, titled '2. Digital_Outputs', lists 11 digital output tags. Each tag is defined with its name, data type (all Boolean), address, and various control flags like Retain, Access, Write, and Visibility.

1. Digital_Inputs							2. Digital_Outputs						
Name	Data type	Address	Retain	Access	Write	Visibl...	Name	Data type	Address	Retain	Access	Write	Visibl...
4. Barcode_COM_D...	Bool	%I0.0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	1. Gas_Valve_1_1	Bool	%Q0.0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
5. Barcode_DB [DB...	Bool	%I0.1		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	2. Gas_Valve_1_2	Bool	%Q0.1		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
6. TIC_DB [DB36]	Bool	%I0.2		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	3. Gas_Valve_1_3	Bool	%Q0.2		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
7. TIC_DB [DB37]	Bool	%I0.3		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	4. Gas_Valve_2_1	Bool	%Q0.3		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
8. L5_X1_Axis_Forward	Bool	%I0.4		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	5. Gas_Command	Bool	%Q0.4		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
9. L5_X1_Axis_Backward	Bool	%I0.5		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	6. Gas_Valve_2_2	Bool	%Q0.5		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
10. L5_Z1_Axis_Downward	Bool	%I0.6		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	7. Lamp_Machine_Emergency	Bool	%Q0.6		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
11. L5_Z1_Axis_Upward	Bool	%I0.7		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	8. Lamp_Automatic_Execution	Bool	%Q0.7		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
12. L5_Oscillator_1_Forward	Bool	%I0.8		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	9. Gas_Presence	Bool	%Q1.0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
13. L5_Oscillator_1_Backward	Bool	%I0.9		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	10. Gas_Valve_2_3	Bool	%Q1.1		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
14. L5_AVC_1_Upward	Bool	%I1.0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	11. +Add new...				<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
15. L5_AVC_1_Downward	Bool	%I1.1		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>							
16. L5_Oscillator_2_Forward	Bool	%I1.2		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>							
17. L5_Oscillator_2_Backward	Bool	%I1.3		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>							
18. L5_AVC_2_Upward	Bool	%I1.4		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>							
19. L5_AVC_2_Downward	Bool	%I1.5		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>							
20. Axis_Selector_Bit_0	Bool	%I8.0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>							
21. Axis_Selector_Bit_1	Bool	%I8.1		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>							
22. Increase_Speed	Bool	%I8.2		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>							
23. Decrease_Speed	Bool	%I8.3		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>							
24. Save_Welding_Position	Bool	%I8.4		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>							
25. Fast_Slow_Mode	Bool	%I8.5		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>							
26. Free_Input_8_6	Bool	%I8.6		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>							
27. Gas_Presence_Sensor	Bool	%I8.7		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>							
28. Selected_Axis_Upward_CMD	Bool	%I9.0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>							
29. Selected_Axis_Downward_CMD	Bool	%I9.1		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>							
30. Selected_Axis_Forward_CMD	Bool	%I9.2		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>							
31. Selected_Axis_Backward_CMD	Bool	%I9.3		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>							
32. Mode_Selector	Bool	%I9.4		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>							
33. Start_Button	Bool	%I9.5		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>							
34. Stop_Button	Bool	%I9.6		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>							
35. Machine_Emergency	Bool	%I9.7		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>							

7.41: Ulazni i izlazni PLC tagovi

7.5 PLC podatkovne strukture

Kao dodatnu pomoć za podatkovne blokove mogu se napraviti i PLC podatkovni tipovi (eng. *PLC Data Types*). To su korisnički predefimirane podatkovne strukture s nazivima i tipovima varijabli koji se mogu pozivati u podatkovnim blokovima. Drugim riječima, to su predefimirane strukture podataka koje se mogu pozvati u podatkovni blok i koje se u podatkovnom bloku ne mogu modificirati. Umjesto tipa varijable (npr. INT ili REAL), u podatkovni blok upiše se ime PLC podatkovnog tipa i sve njegove varijable automatski se upišu u podatkovni blok. U ovom kodu koristi se 9 PLC podatkovnih tipova, od kojih su najzastupljeniji „4. Barcode“ (300 puta) i „1. Sector“ (20 puta)

Kao što je vidljivo iz slike 7.42, PLC podatkovne tipove u podatkovnom bloku može se definirati i kao nizove podataka. U podatkovni blok uvezeno je 20 PLC podatkovnih tipova imena „1. Sector“, počevši od broja 1 i zaključno s brojem 20. Željeni PLC podatkovni tip u nizu se poziva dodavanjem rednog broja u uglatim zagradama nastavno

na ime varijable na sljedeći način: „Sector[1]“, „Sector[2]“, i tako dalje sve do „Sector[20]“. Također moguće je pozivati jednu vrstu PLC podatkovnog tipa u drugu, što je također vidljivo iz slike 7.42 jer je cijela tablica zapravo PLC podatkovni tip „2. Program“ koji definira jedan zavarivački programa koji se koristi u automatskom ciklusu zavarivanja.

2. Program								
	Name	Data type	Default value	Accessible f...	Writa...	Visible in ...	Setpoint	Comment
1	▶ Home_Position	Struct		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
2	▶ Welding_Position	Struct		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
3	▼ Welding_Parameters	Struct		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
4	▶ Sectors	Array[1..20] of *1. Sector*		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
5	Welding_Diameter	Real	0.0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
6	JOB	Int	0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
7	Shielding_Gas	Int	0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
8	Slope_Down_Degrees	DInt	0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
9	Welding_Direction	Bool	false	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
10	Hotwire_Enabeling	Bool	false	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
11	AVC_1_Enabeling	Bool	false	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
12	AVC_2_Enabeling	Bool	false	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
13	Torch_1_Enabeling	Bool	false	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
14	Torch_2_Enabeling	Bool	false	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
15	Timers	Struct		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
16	▶ General_Data	*3. General_Data*		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	

Slika 7.42: PLC podatkovne strukture

8. KORISNIČKO SUČELJE

Programiranje korisničkog sustava omogućeno je u TIA Portalu samo ako projekt sadržava barem jedan HMI uređaj.

Kada je uređaj dodan u projekt, prvo što se treba napraviti je definirati postavke Runtime-a (eng. *Runtime Settings*), odnosno parametre prikazivanja podataka na HMI uređaju. Na kartici „*Postavke Runtime-a*“ nalaze se parametri poput odabira početne stranice, dubine boja na ekranu, odabira zadanog predloška pozadine, postavke jezika i fontova i ostalih parametara Runtime-a.

Sljedeći korak je dodavanje ekrana u program. Prvi dodani ekran automatski će biti postavljen kao početni ekran projekta koji se aktivira kada se stroj upali. Na desnoj strani ekrana nalazi se kartica „*Alatna traka*“ (eng. *Toolbox*) na kojoj se nalaze svi elementi i objekti koje je moguće dodati na ekran.

8.1 Objekti korisničkog sučelja

Objekti korisničkog sučelja su elementi pomoću kojih korisnik čita i definira procesne veličine prikazane na HMI uređaju te pomoću kojih navigira kroz ekrane korisničkog sučelja.

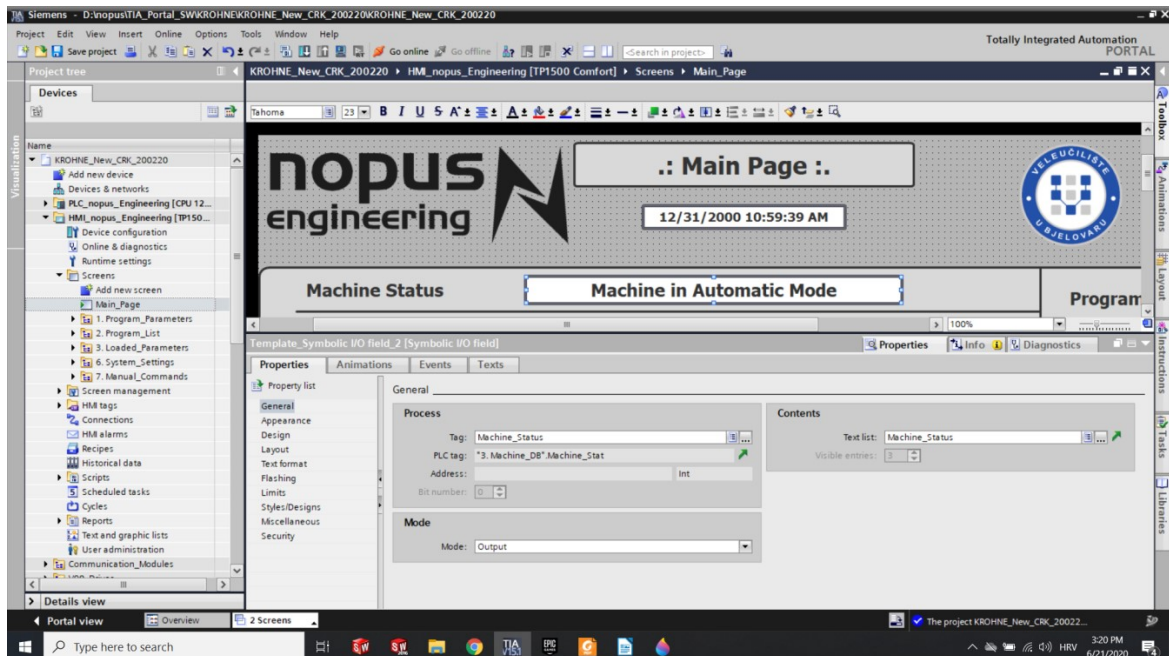
Najčešća vrsta objekata u ovom programskom rješenju korisničkog sučelja su elementi. Sljedeći elementi korišteni su u programskom rješenju korisničkog sučelja:

1. „*I/O polje*“ (eng. *I/O field*)
2. „*Tipka*“ (eng. *Button*)
3. „*Simbolično I/O polje*“ (eng. *Symbolic I/O field*)
4. „*Grafičko I/O polje*“ (eng. *Graphic I/O field*)
5. „*Prekidač*“ (eng. *Switch*)

I/O polje služi kao polje za unos korisnički definiranih podataka i njihovo čitanje, dok kod simboličnog i grafičkog I/O polja korisnik samo može odabrati jednu od ponuđenih opcija. Simboličko I/O polje mora biti definirano tekstualnom listom. Kada je polje aktivirano, iz njega se spusti padajući izbornik s podacima iz tekstualne liste. Primjer I/O polja je prikaz statusa stroja na početnom ekranu. To I/O polje je izlaznog tipa pa nema padajući izbornik, već samo prikazuje podatke iz tekstualne liste ovisno u kojem načinu rada se stroj N1 nalazi. Grafičko I/O polje mora biti definirano grafičkom listom. Grafičko I/O polje se ne koristi u ovome programskom rješenju.

8.1.1 Postavke

Kartica „*Postavke*“ služi za definiranje svih postavki odabranog objekta na HMI uređaju i prikazana je na slici 8.1.



Slika 8.1: Postavke objekata HMI uređaja

Na kartici „*Generalno*“ odabire se procesna veličina, odnosno tag HMI uređaja, koji će se pomoću objekta kontrolirati, te vrsta i format polja. U kućici „*Format*“ pod parametrom „*Uzorak formata*“ (eng. *Format pattern*) odabire se maksimalan broj znamenki koje korisnik može upisati, tj. koje će polje prihvatiti i prikazati. Ako korisnik unese veći broj znamenki nego što je dozvoljeno, polje će odbaciti novoupisanu vrijednost i vratiti prethodno upisanu vrijednost. Parametar „*Decimalna mjesta*“ (eng. *Decimal places*) određuje za koliko mjesta ulijevo će se pomaknuti decimalna točka. Primjerice, ako je vrijednost procesne veličine 123456, uzorak formata 99999 i broj decimalnih mjesta 2, polje na HMI uređaju prikazat će vrijednost 123.46 jer će se zbog uzorka formata od 5 znamenaka, zadnja znamenka izbaciti iz prikaza. Budući da je zadnja prikazana znamenka u polju 5, a iza nje se nalazi korisniku nevidljiva znamenka 6, HMI uređaj će automatski zadnju vidljivu znamenku zaokružiti na veću vrijednost, odnosno vrijednost 6.

Kartica „*Prikaz*“ služi za vizualni izgled polja. Na njoj se definiraju boje pozadine i teksta, te boje, vrste rubova i stilovi obruba polja.

Na kartici „*Karakteristike*“ moguće je odabrati funkciju „*Skriveni upis*“ (eng. *Hidden input*). Svaki upisani znak spremljen je normalno u pripadajuću procesnu veličinu, ali u polju su umjesto znakova prikazane zvjezdice. Jedino polje gdje se koristi skriveni unos u ovom programskom rješenju je polje za unos lozinke za prijavljivanje u sustav.

Kartica „*Raspored*“ služi za definiranje veličine polja i njegovog položaja na ekranu. Aktivacijom funkcije „*Prilagodi veličinu*“ (eng. *Fit to size*), onemogućuje se korisničko definiranje veličine polja već se veličina polja automatski prilagođava iznosu procesne veličine. Drugim riječima, polje će se raširiti taman toliko da je cijela procesna veličina prikazana u njemu. Ako je ova funkcija deaktivirana te se veličina polje definira manualno, postoji mogućnost da procesna veličina bude duža od veličine polja tako da će u polju biti vidljiv samo dio procesne veličine.

Na kartici „*Format teksta*“ definira se vrsta i veličina fonta, te položaj fonta polja.

Na kartici „*Blicanje*“ moguće je aktivirati blicanje polja u slučaju da procesna veličina izađe iz krajnjih vrijednosti definiranih u postavkama procesna veličine.

Kartica „*Ograničenja*“ služi za postavljanje boje pozadine polja ako procesna vrijednost izađe iz krajnjih vrijednosti definiranim u postavkama procesne veličine.

8.1.2 Animacije

Kartica „*Animacije*“ (eng. *Animations*) služe za postavljanje izgleda objekata i prikazana je na slici 8.2. Kartica „*Animacije*“ dodatno je podjeljena u tri potkartice.

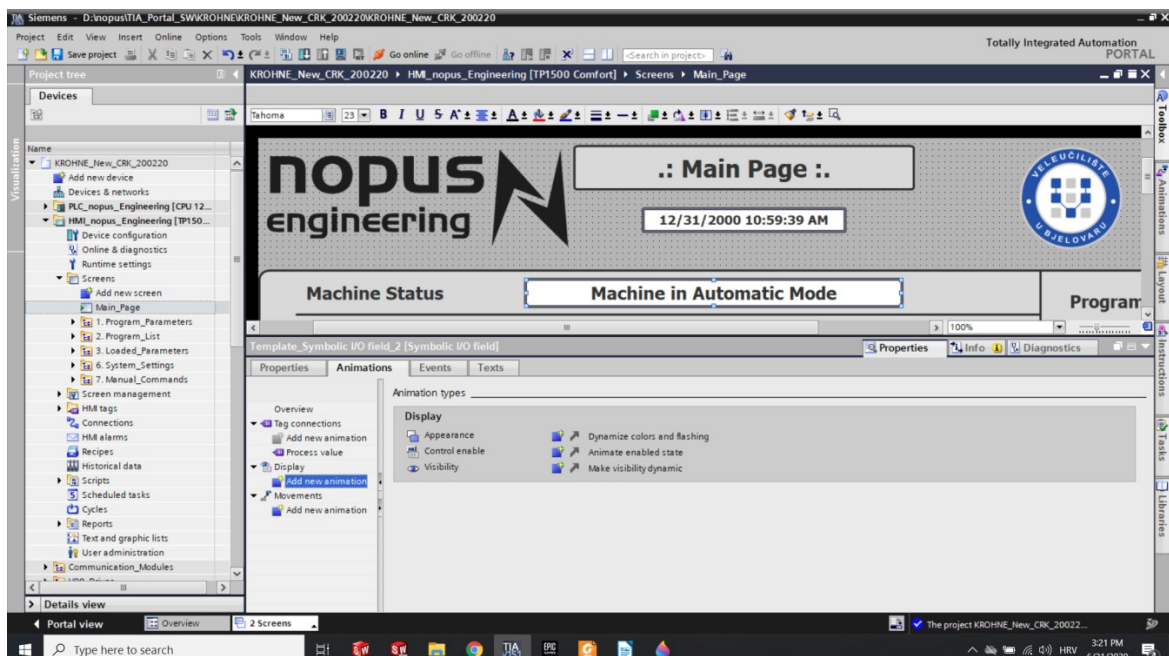
Prva potkartica „*Tag konekcije*“ (eng. *Tag connections*) služi za povezivanje objekta s HMI tagom. Kao prvi primjer uzeti će se prekidač. Kada se prekidač poveže s tagom, aktivacija prekidača aktivira i njemu pridruženi HMI tag. Ako se stanje HMI taga promijeni unutar PLC uređaja, tj. promijeni se stanje njemu povezanog PLC taga; tada će i prekidač promijeniti svoje stanje. U suštini, prekidač i HMI tag su povezani i sinkronizirani, te jedan može kontrolirati stanje drugoga.

Druga potkartica „*Prikaz*“ služi za odabir prikaza i kontrole objekta. Funkcije objekta ovise o stanju i vrsti HMI taga koji je povezan u postavkama prikaza objekta. Prikaz se dijeli u tri skupine:

1. „*Izgled*“ (eng. *Appearance*) objekta može se mijenjati ovisno o stanju i vrsti pridruženog HMI taga. Ako je pridruženi HMI tag logičkog oblika, tada prikaz pozadine objekta može poprimiti dvije različite boje. S druge strane, ako je HMI tag cjelobrojnog oblika, tada se objektu mogu odrediti rasponi boja

pozadina. Ovisno o vrijednosti HMI taga, objekt će promijeniti boju pozadine sukladno definiciji u kartici „Izgled“. Ova funkcija koristi se kod definiranja bitnih parametara zavarivačkog procesa. Ako je vrijednost 0 upisana u polje parametra, tada polje poprima crvenu boju pozadine kako bi se korisnika upozorilo na nedopuštenu vrijednost.

2. „Omogućavanje kontrole“ (eng. *Control enable*) služi za omogućavanje, odnosno onemogućavanje upravljanja objektom. Kod HMI taga logičkog oblika, kontrola objekta ovisi o stanju HMI taga. Ako je HMI tag cjelobrojnog oblika, tada se kontrola omogućava, odnosno onemogućava za određeni raspon vrijednosti HMI taga. Ova vrsta se ne koristi u ovome programskom rješenju.
3. „Vidljivost“ (eng. *Visibility*) služi za prikazivanje, odnosno sakrivanje objekta na ekranu. Ako je HMI tag logičkog oblika, njegovo stanje može sakriti ili prikazati objekt na ekranu. Kod HMI tagova cjelobrojnog oblika, prikaz i sakrivanje objekta definira se za određeni raspon vrijednosti HMI taga. Ova funkcija koristi se kod definiranja parametara sektora zavarivačkog programa. Tek kada se određeni sektor aktivira, parametri sljedećeg sektora prikažu se na ekranu.

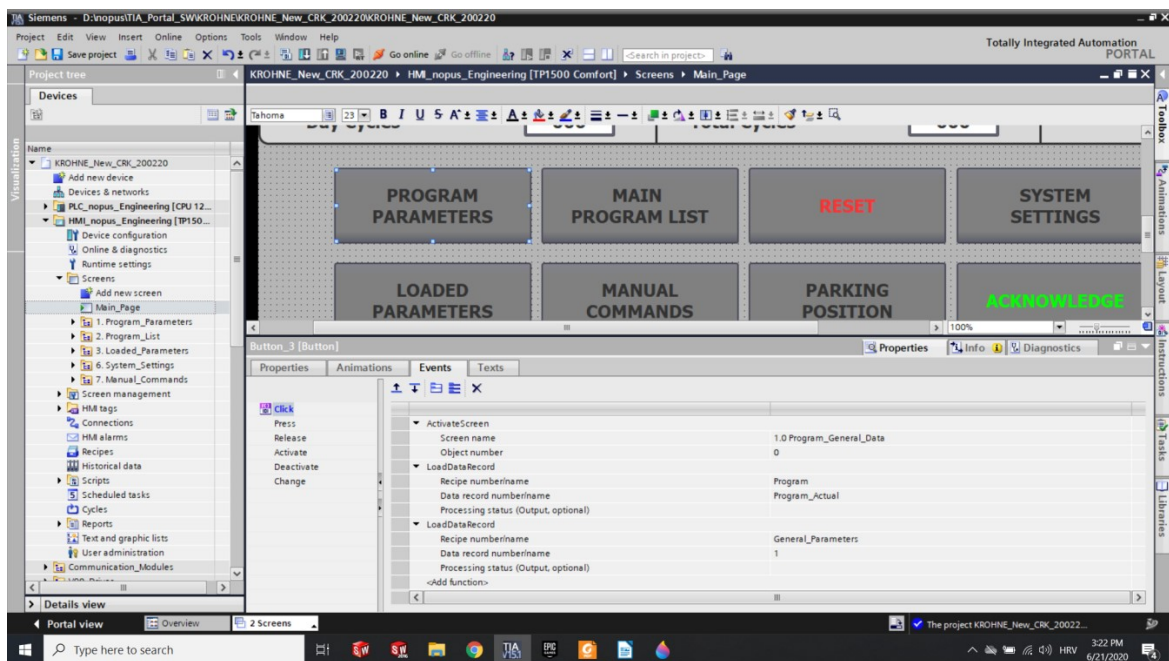


Slika 8.2: Animacije objekta HMI uređaja

Treća potkartica „*Kretnje*“ (eng. *Movements*) služi za prikaz animacije kretanja objekta na ekranu. Budući da se ova funkcija objekta ne koristi u ovom programskom rješenju, neće se niti opisivati njezine mogućnosti.

8.1.3 Događaji

Kartica „*Događaji*“ (eng. *Events*) služe za definiranje događaja koji će se dogoditi u HMI uređaju kada operater aktivira neku opcija. Kao primjer uzet će se tipka „*Parametri programa*“. Kada operater klikne na tipku „*Parametri programa*“ prvo će se aktivirati stranica „*Parametri programa*“. Nakon toga će se iz recepta „*Program*“ učitati parametri zavarivačkog programa koji je unesen u I/O polje unos programa. Kao zadnja opcija recept „*Generalni podatci*“ upisat će se u polja glavne liste programa. Sva tri događaja tipke „*Parametri programa*“ prikazani su na slici 8.3.

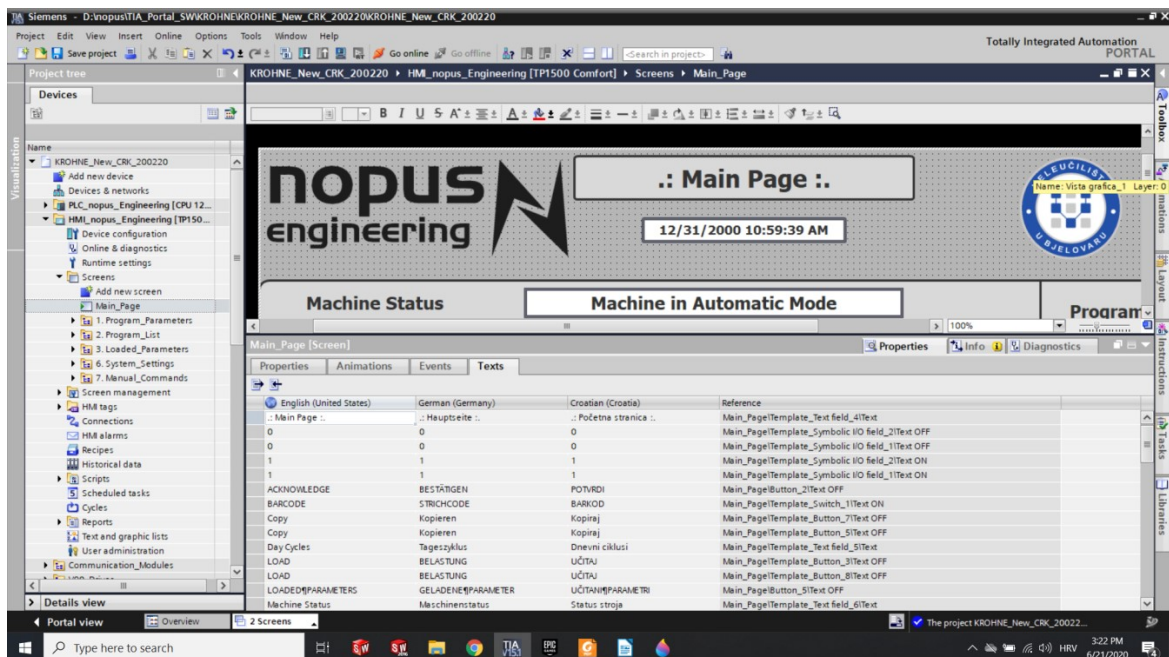


Slika 8.3: Događaji objekta HMI uređaja

8.1.4 Tekstovi

Kartica „*Tekstovi*“ (eng. *Text*) služi za pregled i prevođenje tekstova. Ako se odabere specifični objekt, na kartici „*Tekstovi*“ prikazat će se samo tekstovi tog objekta. Ako se odabere stranica, u kartici „*Tekstovi*“ bit će prikazani svi tekstovi koji se nalaze na toj stranici. U ovom programskom rješenju odabrana su tri jezika u postavkama HMI uređaja:

hrvatski, engleski i njemački, pa se sva tri jezika pojavljuju u kartici „Tekstovi“. Kartica „Tekstovi“ prikazana je na slici 8.4.



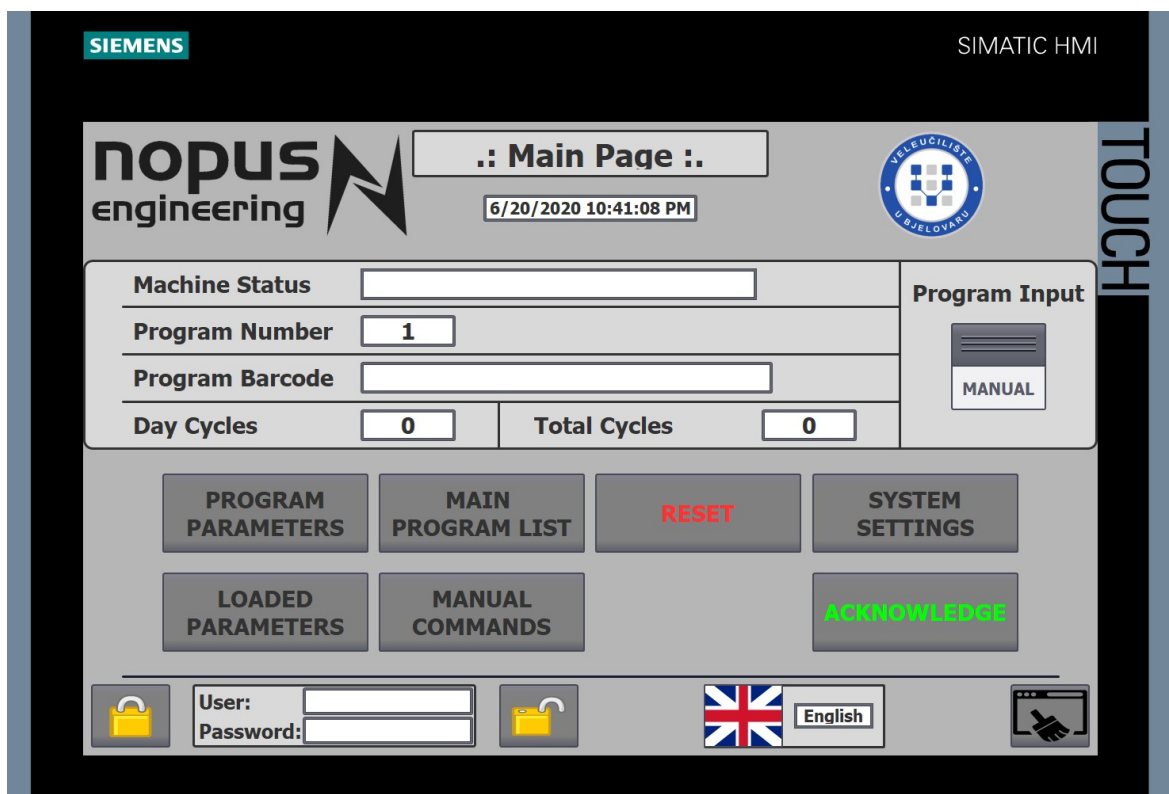
Slika 8.4: Tekstovi objekta HMI uređaja

8.2 Ekran

Kartica „Ekran“ (eng. *Screens*) koristi se za postavljanje ekrana za prikaz procesnih veličina i informacija o sustavu na HMI uređaju te za navigaciju kroz iste. Prvi ekran automatski je generiran u projekt i dodijeljen mu je atribut početnog ekrana, odnosno ekrana koji se prikaže na HMI uređaju nakon podizanja sustava. Na početnom ekranu zatim se definiraju tipke koje korisnika vode i navigiraju na ostale ekrane sustava. Kao prvi primjer detaljnije će se opisati početna stranica (eng. *Home screen*) sustava.

Na početnom ekranu nalaze se ulazno-izlazna polja i simbolična ulazno-izlazna polja, obadvoje izlaznog tipa. Operater u polja izlaznog tipa ne može upisivati vrijednosti, već samo iz njih pročitati vrijednosti procesnih veličina, tj. vrijednosti u njima povezane HMI tagove. Ona operateru pokazuju neke osnovne informacije stroja poput statusa stroja koji je trenutno aktivan, broja trenutno učitano zavarivačkog programa ili broj sveukupno odrađenih automatskih ciklusa zavarivanja na stroju. Na srednjem dijelu stranice nalaze se tipke za navigiranje kroz funkcije sustava. Svaki gumb u svojim postavkama ima definiranu stranicu koju će aktivirati ako korisnik pritisne tu tipku. Primjerice, pritiskom na tipku „Parametri programa“ sustav će prikazati stranicu za upis parametara zavarivačkog

programa. Kako bi korisnik pristupio ovoj stranici prethodno se mora prijaviti u sustav s korisničkim imenom i lozinkom koji imaju pristup stranici. Korisnički pristup funkcijama stroja detaljnije je objašnjen u poglavlju 7.10. Kako bi se korisnik prijavio u sustav, na dnu stranice ima dva ulazna polja za unos korisničkog imena i pripadajuće lozinke. Nakon što ih upiše, korisnik mora pritisnuti tipku s ikonom otključanog lokota koja prijavljuje korisnika u sustav i njegovo korisničko ime šalje u HMI tag koji se koristi za prikaz korisničkog imena na drugim stranicama sustava (u gornjem desnom kutu ekrana). Korisničko ime mora biti vidljivo na drugim stranicama sustava iz sigurnosnih razloga. Primjerice, ako se operater iz prve smjene zaboravi odjaviti, a operater iz druge smjene zaboravi prijaviti, operater iz druge smjene nastaviti će raditi pod korisničkim imenom korisnika iz prve smjene. S druge strane, korisnik koji ima pristup svim funkcijama stroja može se zaboraviti odjaviti i ostaviti stroj potpuno otvoren za korisnika koji nema pristup svim funkcijama stroja. Ako je korisničko ime vidljivo na drugim stranicama sustava, veća je šansa da će se prvi korisnik odjaviti ili u drugom slučaju da će drugi korisnik primijetiti tuđe korisničko ime i samostalno ga odjaviti iz sustava. Na slici 8.5 prikazana je početna stranica na engleskom jeziku. Na njoj se mogu vidjeti spomenuta ulazno-izlazna polja i simbolička ulazno-izlazna polja te prostor za prijavu korisnika u sustav.



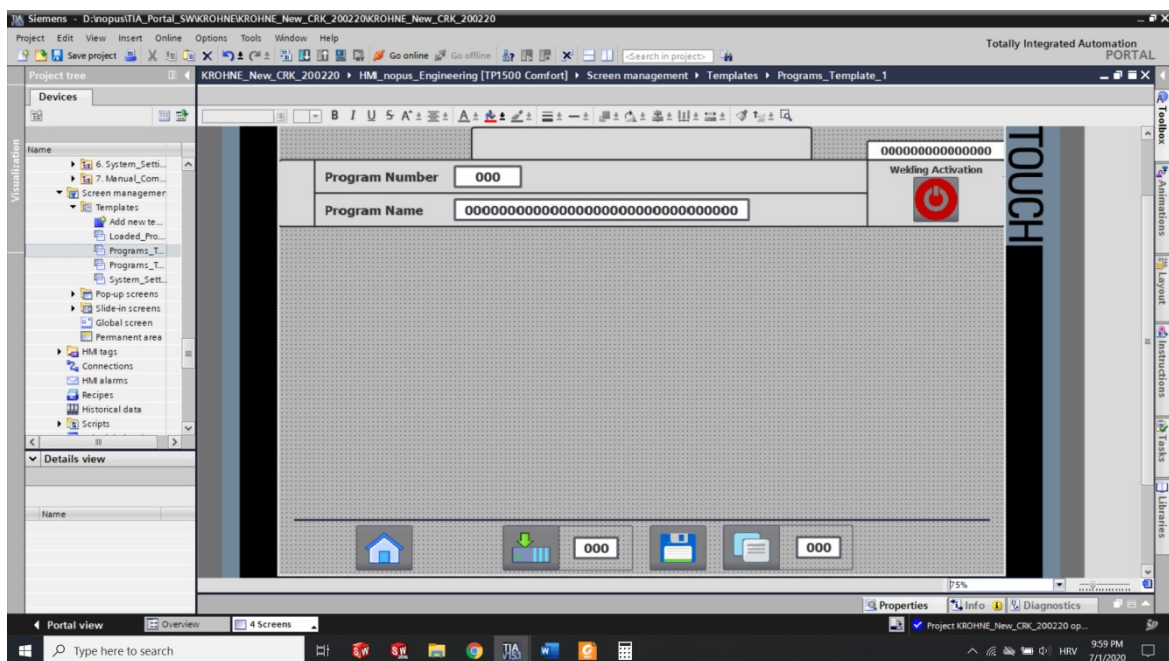
Slika 8.5: Primjer početne stranice na engleskom jeziku

Kao drugi primjer navest će se stranica za definiranje parametara zavarivanja. Na lijevoj stranici prikazani su generalni parametri zavarivačkog programa. Budući da sustav ima mogućnost definiranja 300 različitih zavarivačkih programa, to bi značilo da je u sustavu također potrebno definirati 300 stranica koje prikazuju parametre programa. Izrada 300 stranica bila bi mukotrpan i dugotrajan proces. Zbog toga se kod parametara programa koriste funkcije multipleksiranja (eng. *Multiplexing*) i recepata (eng. *Recipes*).

U TIA-Portalu proces multipleksiranja podrazumijeva spajanje više ulaza na jedan izlaz koji se zatim prikazuje na ekranu. Proces multipleksiranja detaljnije je opisan u poglavlju 8.4. Recepti bi se najjednostavnije mogli opisati kao nizovi multipleksiranih podatkovnih struktura. Više o receptima i njihovim funkcijama opisano je u poglavlju 8.7.

8.3 Upravljanje ekranima

Na kartici „Upravljanje ekranima“ mogu se kreirati predlošci ekrana, te skočni i klizni ekrani. U ovome programskom rješenju koriste se samo predlošci ekrana pa će se samo njih detaljnije objasniti.



Slika 8.6: Predložak ekrana Programs_Template_1

Predlošci ekrana služe kako bi se definirali konstantni objekti pojedinih programa. Ako se na određenim ekranima pojedini objekti ponavljaju i nalaze na istim pozicijama, za te ekrane napravi se predložak koji sadržava te objekte. Zatim se predložak metodom povuci i

pusti (eng. *drag and drop*) dodaju na željene ekrane. U suprotnome bi se svaki objekt morao definirati na svakoj stranici. Kao primjer na slici 8.6 prikazan je predložak „*Programs_Template_1*“ koji se koristi na stranicama za definiranje parametara zavarivanja.

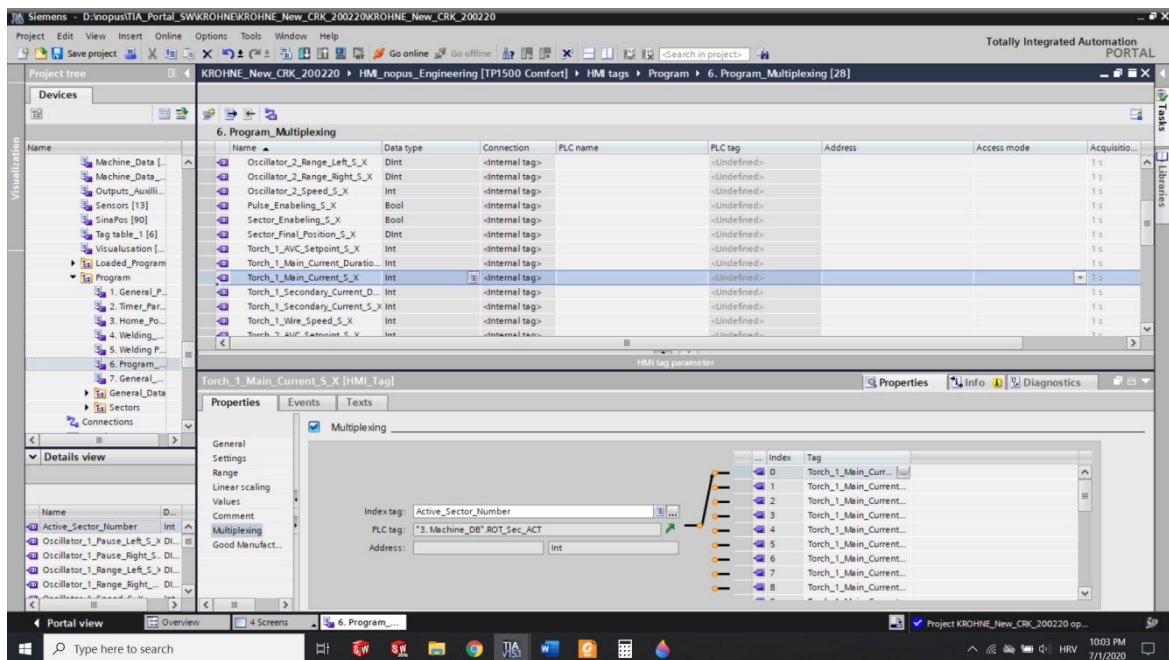
Na gornjem dijelu stranice nalaze se ulazno polje za unos zavarivačkog programa, izlazno polje za prikaz barkoda programa, izlazno polje za prikaz trenutno prijavljenog korisnika i prekidač za aktivaciju zavarivanja u zavarivačkom programu. Na donjem dijelu stranice nalaze se tipke za povratak na početnu stranicu, tipka „*Učitaj*“ za kopiranje trenutno učitano podatkovnog zapisa recepta u memoriju PLC uređaja, tipka „*Snimi*“ za snimanje parametara prikazanih na ekranu u podatkovni zapis recepta učitano u sustav i tipka „*Kopiraj*“ za kopiranje parametara prikazanih na ekranu u podatkovni zapis recepta upisanog u ulazno polje pokraj tipke za kopiranje. Bez mogućnosti definiranja predložaka ekrana, svi ovi objekti morali bi se definirati na svakoj stranici za definiranje zavarivačkih parametara.

8.4 HMI tagovi

HMI tagovi definiraju se pod karticom „*HMI tags*“ u lijevom izborniku projekta. Svaki tag povezan s PLC uređajem može prikazivati samo jednu procesnu veličinu. Kako svaki zavarivački program ima mogućnost definiranja 20 sektora, to bi značilo da u projekt za svaki sektor treba staviti posebnu stranicu koja će prikazivati parametre tog sektora. Izrada 20 stranica s istim setovima parametara nepotreban je posao za programera i nepotrebno se zauzima memorija HMI uređaja. Kako bi se to izbjeglo, u ovom rješenju korisničkog sučelja koristi se ranije spomenuto multipleksiranje koje je prikazano na slici 8.7.

To je postupak ranžiranja više ulaznih parametara na jedan izlazni parametar, ovisno o kontrolnom parametru. Za primjer će se uzeti jakost glavne struje zavarivanja gorionika 1. Parametri jakosti glavne struje zavarivanja gorionika 1 iz svih 20 sektora povezani su u jedan tag naziva „*Torch_1_Main_Current_S_X*“ koji se nalazi na stranici „*Parametri zavarivanja*“. U gornjem desnom uglu stranice nalazi se I/O polje s brojem trenutno aktivnog sektora koje je povezano s tagom „*Active_Sector_Number*“. Ovisno koji broj je upisan u I/O polje broja sektora, odnosno u tag „*Active_Sector_Number*“, multipleksiranje će na ekranu prikazati tag koji je upisan pod indeks s istom vrijednošću. Uz pomoć multipleksiranja, svih 20 sektora prikazano je na jednoj stranici. Ovakvo rješenje također je puno elegantnije i za kontrolu zavarivačkog procesa. Na stranici „*UČITANI PARAMETRI*

”korisnik može provjeriti sve parametre koji su definirani za zavarivački program. Kada provjeri sve parametre, operater se prebacuje na stranicu „UČITANI PARAMETRI ZAVARIVANJA“ kako bi uživo mogao pratiti sve parametre zavarivanja. Kada dođe do promjene sektora, na stranici se adekvatno tome moraju prikazati parametri novog sektora. Bez korištenja multipleksiranja, promjena parametara kod promjene sektora morala bi se riješiti na puno kompliciraniji način.



Slika 8.7: Postupak multipleksiranja HMI tagova

Svaki HMI tag definiran je pripadajućim PLC tagom. Nemoguće je, na primjer, da je PLC tag cjelobrojnog, a HMI tag realnog oblika. HMI tagu moraju se definirati vrsta i ciklus osvježavanja (eng. *Acquisition mode and cycle*). Svi HMI tagovi definirani u ovom projektu kao vrstu osvježavanja imaju definirano čitanje, dok je stranica s tagom aktivna (eng. *Cyclic in operation*) i to svaku sekundu. Kao način osvježavanja podataka mogu se još odabrati kontinuirano čitanje i čitanje na zahtjev. Kontinuirano čitanje znači da će HMI osvježavati sve tagove u svakom ciklusu, što može dovesti do zagušenja ciklusa jer HMI uređaj nema ni približnu brzinu ciklusa i procesorsku sposobnost kao PLC uređaj. Osvježavanje na zahtjev potrebno je definirati ako se na jednom ekranu vrijednosti ne osvježavaju u istome trenutku, pa se svakom tagu, na primjer, osvježavanje radi kada se vrijednost taga promijeni. Ovakav način osvježavanja ponajviše se koristi kod kontrolnih procesa i procesa gdje se određeni parametri moraju odmah prikazati na ekranu. Na stroju N1 sasvim je dovoljno osvježavanje tagova dok je stranica aktivna. Jedina stranica koja

zahtijeva dinamiku promjene parametara jest „UČITANI PARAMETRI ZAVARIVANJA“, ali ni tamo nisu potrebni neki visoki zahtjevi za dinamikom, tako da je osvježavanje svaku sekundu i više nego dovoljno.

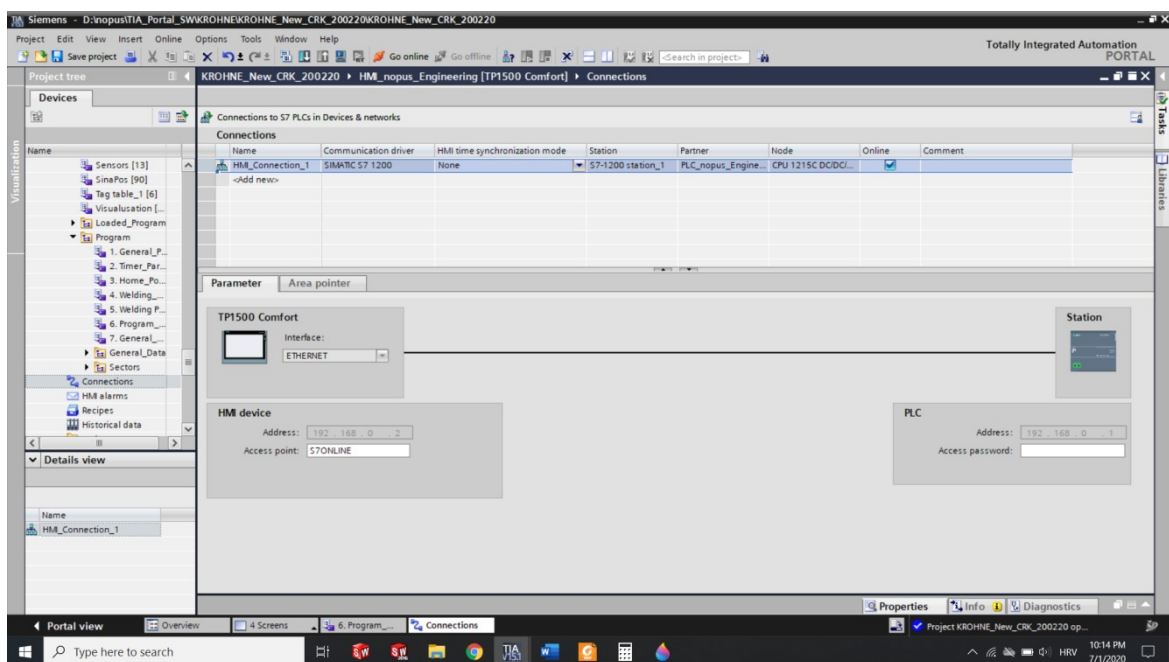
Nadalje, svakom HMI tagu može se definirati raspon i linearno skaliranje. Kod raspona HMI taga definiraju se dvije gornje i dvije donje vrijednosti. Raspon se koristi za dvije stvari. Ako je tag definiran kao kontrolna, odnosno izlazna vrijednost, onda I/O polje promijeni boju ako tag ispadne iz vrijednosti raspona. U postavkama I/O polja definiraju se boje za granice polja. Ako pripadajući HMI tag ispadne iz manjeg raspona, tada I/O polje poprimi boju definiranu za prvu granicu, obično žutu, kako bi se korisnika upozorilo da je vrijednost parametra iznad neke granične vrijednosti. Ako pak vrijednost HMI taga ispadne iz većeg raspona, tada I/O polje poprimi boju definiranu za drugu granicu, obično crvenu, kako bi se korisnika upozorilo da je vrijednost parametra prešla kritičnu vrijednost. Budući da kod stroja N1 kontrolnih parametara tog tipa nema, raspon se koristi za ograničavanje mogućnosti unosa vrijednosti parametara preko korisničkog sučelja. Naime, ako se raspon koristi kod ulaznih polja, veći raspon definira minimalnu i maksimalnu vrijednost koja se može unijeti u I/O polje s pripadajućim tagom. Kao primjer uzet ćemo jakost glavne struje zavarivanja gorionika 1 u prvom sektoru. Kao minimalna vrijednost definirano je 50, a kao maksimalna 5000. To su zapravo vrijednosti od 5 i 500 ampera, ali kao što je već prije spomenuto, za spremanje parametara zavarivačkog programa koriste se cjelobrojne vrijednosti kako bi se uštedjelo na memoriji. U I/O polju jakosti glavne struje zavarivanja gorionika 1 definiran je pomak decimalne točke za jedno mjesto, tako da će se vrijednost 50 zapravo prikazati kao 5,0. EWM Tetrix 552 Synergic RC HW aparat za zavarivanje može raditi s vrijednostima jakosti električne struje od 5 do 550 ampera. Zbog toga je donja granica koju korisnik može upisati u zavarivački program 5. Gornja granica stavljena je na 500 kako aparat nikada ne bi radio s maksimalnom vrijednošću. Radni komadi koje stroj N1 zavaruje ne zahtijevaju jakost električne struje zavarivanja veću od 150 ampera. Ipak, u budućnosti korisnik može dodati nove serije na stroj za koje je potrebna veća jakost električne struje zavarivanja, pa je s toga maksimalna vrijednost jakosti električne struje zavarivanja ograničena na 500 ampera.

Linearno skaliranje HMI tagova ne koristi se u ovom programskom rješenju. Iako se HMI tagovi skaliraju, što je vidljivo na prethodno spomenutom primjeru skaliranja vrijednosti jakosti glavne struje zavarivanja gorionika 1, taj postupak odrađuje se u PLC uređaju. Linearno skaliranje HMI tagova u HMI uređaju odvijalo bi se u svakom ciklusu osvježavanja, a kao što je već rečeno, niti jedan parametar nema takve zahtjeve za

dinamikom. Svi parametri koji se skaliraju su parametri zavarivanja, koji se u funkcijskom bloku Kalkulacija parametara zavarivanja izračunaju na početku zavarivačkog procesa i zapišu u podatkovni blok iz kojega se zatim pozivaju ovisno o sektoru koji je aktivan. Dakle, linearno skaliranje parametara potrebno je samo jednom u zavarivačkom procesu, pa nema potrebe da se taj dio odvija u svakom ciklusu osvježavanja HMI tagova. S druge strane, budući da se kalkulacija parametara odvija na početku zavarivačkog procesa, taj jedan ciklus u kojemu se oni računaju neće utjecati na brzinu izvršavanja glavnog dijela upravljačkog koda.

8.5 Konekcija

Prije nego što se bilo što krene programirati na HMI uređaju, potrebno je uspostaviti konekciju između HMI uređaja i PLC uređaja prikazanu na slici 8.8.



Slika 8.8: Konekcija HMI uređaja s PLC uređajem

Prvi dio konekcije odrađuje se povezivanjem ETHERNET portova u kartici „Uređaji i mreže“ pod izbornikom PLC uređaja. Nakon toga pod izbornikom HMI uređaja potrebno je u kartici „Konekcije“ (eng. *Connections*) dodatno definirati parametre konekcije HMI uređaja s PLC uređajem. Konekcija definirana u kartici „Konekcije“ nazvana je „HMI_Connection_1“ i koristi se kod povezivanja HMI tagova s PLC tagovima. Konekcija između HMI i PLC uređaja prikazana je na slici 8.8

8.6 HMI alarmi

Na kartici „HMI alarmi“ (eng. *HMI alarms*) definiraju se procesne veličine koji će aktivirati alarme HMI uređaja i prikazati ih na skočnom prozoru korisničkog sučelja. Svi alarmi definirani su kao diskretni alarmi (eng. *Discrete alarms*) jer se aktiviraju pomoću bitova. Na slici 8.9 prikazan je popis HMI alarmova.

ID	Name	Alarm text	Alarm class	Trigger tag	Trigge...	Trigger address	HMI acknowl...	HMI a...	Report
1	Torch 1: No Ignition	Ignition of welding arc was not estal	Errors	Torch_1_Alarm	0	*15. Alarms_...	<-No tag>	0	
2	Torch 1: Wire Stic...	Wire is sticking to the workpiece or t	Errors	Torch_1_Alarm	1	*15. Alarms_...	<-No tag>	0	
3	Torch 1: Error	General error message from the pro	Errors	Torch_1_Alarm	2	*15. Alarms_...	<-No tag>	0	
4	Torch 1: Collision	Collision protection has been trigge	Errors	Torch_1_Alarm	3	*15. Alarms_...	<-No tag>	0	
5	Torch 2: No Ignition	Ignition of welding arc was not estal	Errors	Torch_2_Alarm	0	*15. Alarms_...	<-No tag>	0	
6	Torch 2: Wire Stic...	Wire is sticking to the workpiece or t	Errors	Torch_2_Alarm	1	*15. Alarms_...	<-No tag>	0	
7	Torch 2: Error	General error message from the pro	Errors	Torch_2_Alarm	2	*15. Alarms_...	<-No tag>	0	
8	Torch 2: Collision	Collision protection has been trigge	Errors	Torch_2_Alarm	3	*15. Alarms_...	<-No tag>	0	
9	Rotation Drive Wa...	Rotation Drive Warning	Warnings	Driver_Alarm	0	*15. Alarms_...	<-No tag>	0	
10	Horizontal Axis Driv...	Horizontal Axis Warning	Warnings	Driver_Alarm	1	*15. Alarms_...	<-No tag>	0	
11	Vertical Axis Drive ...	Vertical Axis Drive Warning	Warnings	Driver_Alarm	2	*15. Alarms_...	<-No tag>	0	
12	Oscillator 1 Drive W...	Oscillator 1 Drive Warning	Warnings	Driver_Alarm	3	*15. Alarms_...	<-No tag>	0	
13	AVC 1 Drive Warning	AVC 1 V90 Drive Warning	Warnings	Driver_Alarm	4	*15. Alarms_...	<-No tag>	0	
14	Oscillator 2 Drive W...	Oscillator 2 Drive Warning	Warnings	Driver_Alarm	5	*15. Alarms_...	<-No tag>	0	
15	AVC 2 Drive Warning	AVC 2 Drive Warning	Warnings	Driver_Alarm	6	*15. Alarms_...	<-No tag>	0	
16	Rotation Drive Fault	Rotation Drive Fault	Errors	Driver_Alarm	7	*15. Alarms_...	<-No tag>	0	
17	Horizontal Axis Driv...	Horizontal Axis Drive Fault	Errors	Driver_Alarm	8	*15. Alarms_...	<-No tag>	0	
18	Vertical Axis Drive F...	Vertical Axis Drive Fault	Errors	Driver_Alarm	9	*15. Alarms_...	<-No tag>	0	
19	Oscillator 1 Drive F...	Oscillator 1 Drive Fault	Errors	Driver_Alarm	10	*15. Alarms_...	<-No tag>	0	
20	AVC 1 Drive Fault	AVC 1 V90 Drive Fault	Errors	Driver_Alarm	11	*15. Alarms_...	<-No tag>	0	
21	Oscillator 2 Drive F...	Oscillator 2 Drive Fault	Errors	Driver_Alarm	12	*15. Alarms_...	<-No tag>	0	
22	AVC 2 Drive Fault	AVC 2 Drive Fault	Errors	Driver_Alarm	13	*15. Alarms_...	<-No tag>	0	
23	Torch 1: Welding R...	Error occurred on Torch 1 during W	Errors	Torch_1_Alarm	4	*15. Alarms_...	<-No tag>	0	
24	Torch 2: Welding R...	Error occurred on Torch 2 during W	Errors	Torch_2_Alarm	4	*15. Alarms_...	<-No tag>	0	
25	Horizontal_Axis_Lo...	Both horizontal LS are active	Errors	Axis_Locked	0	*15. Alarms_...	<-No tag>	0	
26	Vertical_Axis_Locke...	Both vertical LS are active	Errors	Axis_Locked	1	*15. Alarms_...	<-No tag>	0	
27	Oscillator_1_Locke...	Both Oscillator 1 LS are active	Errors	Axis_Locked	2	*15. Alarms_...	<-No tag>	0	
28	AVC_1_Locke...	Both AVC 1 LS are active	Errors	Axis_Locked	3	*15. Alarms_...	<-No tag>	0	
29	Oscillator_2_Locke...	Both Oscillator 2 LS are active	Errors	Axis_Locked	4	*15. Alarms_...	<-No tag>	0	
30	AVC_2_Locke...	Both AVC 2 LS are active	Errors	Axis_Locked	5	*15. Alarms_...	<-No tag>	0	

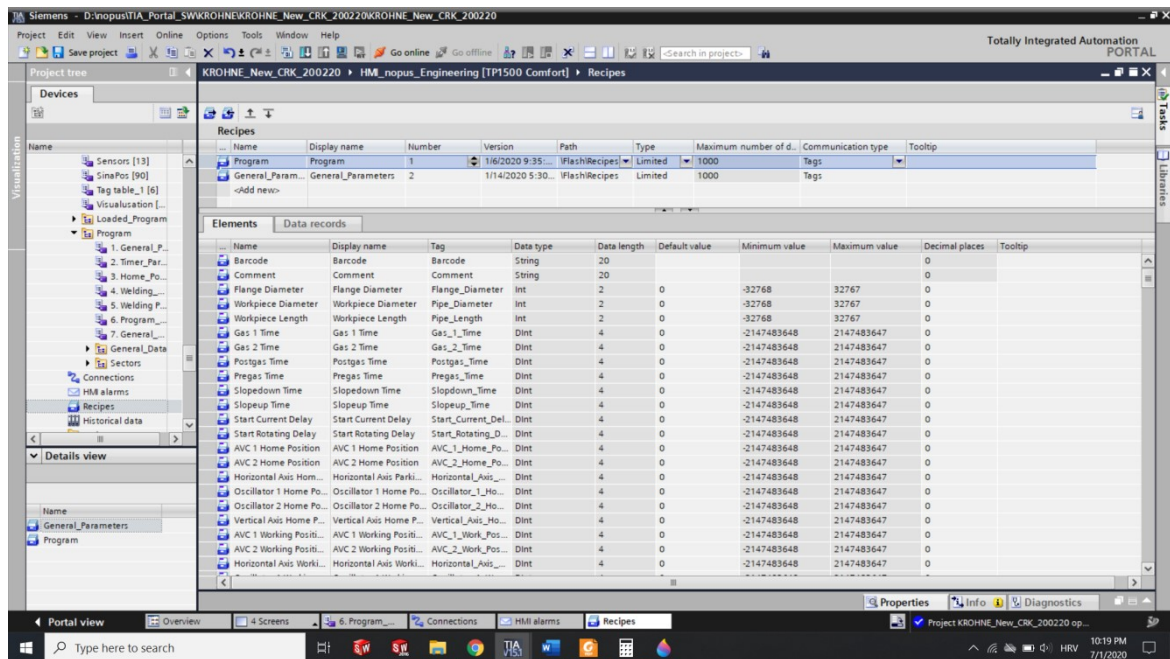
Slika 8.9: HMI alarmi

Alarmi još mogu biti definirani kao analogni, tj. alarmi koji se aktiviraju ako procesna veličina prijeđe definiranu granicu, no takvih alarma u ovom programskom kodu nema pa ih se neće detaljnije objašnjavati. Kontrola procesnih veličina alarma odvija se u PLC uređaju. Svi alarmi spremljeni su u memoriju PLC uređaja u obliku riječi, gdje svaki bit označava jedan alarm. HMI tagovi alarma sukladno tome isto su riječi. U kartici „HMI alarmi“ HMI tagovi alarma razbijaju se na bitove i svakom bitu se dodjeljuje ime i vrsta, odnosno opis alarma koji je aktiviran u PLC uređaju. Drugim riječima, u kartici „HMI alarmi“ alarmu se dodaje tekst koji će se ispisati na ekranu kada se alarm aktivira. Također, uz ime alarma prikazat će se i vrsta alarma, tj. je li alarm upozorenje ili greška. Svi alarmi deaktiviraju se pritiskom na tipku „POTVRDI“ na početnoj stranici. Ako alarm ostane aktivan nakon što ga operater potvrdi, potrebna je fizička intervencija kako bi se kvar ispravio. Na primjer, ako se aktivira alarm „Obadva krajnja prekidača horizontalne osi su aktivna“ koji kaže da su obadva krajnja prekidača na glavnoj horizontalnoj osi pritisnuta,

korisnik mora intervenirati i vidjeti što je uzrokovalo simultano aktiviranje obadva krajnja prekidača. Ako korisnik potvrdi ovu grešku, krajnji prekidači i dalje će ostati pritisnuti jer je greška fizičke prirode i alarm će i dalje ostati aktivan. Alarm će biti aktivan sve dok se jedan krajnji prekidač ne otpusti i nakon toga potvrdi alarm.

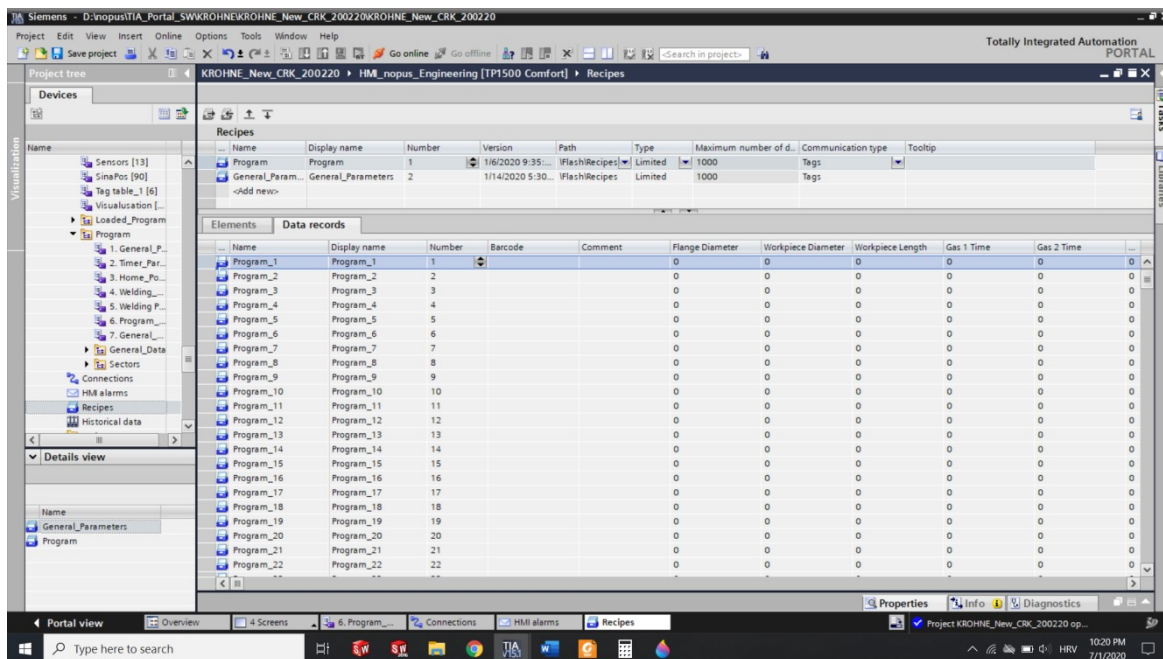
8.7 Recepti

Kartica „*Recepti*“ se koristi za definirane memorijskih lokacija za istovjetne skupove parametara u HMI uređaju. U ovom programskom rješenju recepti su iskorišteni za spremanje parametara svih 300 zavarivačkih programa u memoriju HMI uređaja. Prvo što se treba napraviti jest definirati ime recepta, u ovom slučaju „*Program*“. Nakon toga se definiraju „*Elementi*“ (eng. *Elements*) recepta prikazani na slici 8.10.



Slika 8.10: Elementi recepta Program

Svi parametri zavarivačkog programa definirani su kao HMI tagovi i povezani su s elementima recepta. HMI tagovi koji služe kao izlazne varijable funkcije multipleksiranja ne mogu se definirati unutar recepta jer svaki element recepta mora imati točno određenu memorijsku lokaciju. Elementi se mogu upisivati ručno ili metodom „povuci i pusti“ povući iz HMI tagova. Kada su svi elementi definirani, zadnji korak je definirati „*Podatkovne zapise*“ (eng. *Data records*), odnosno zavarivačke programe, prikazane na slici 8.11.



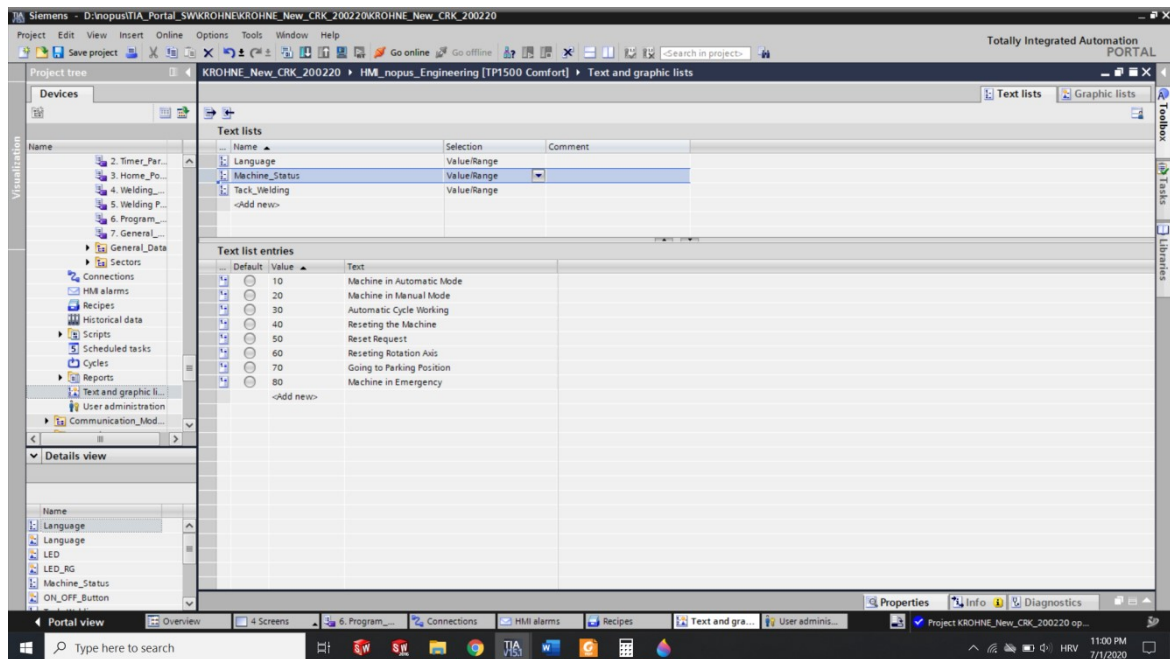
8.11: Podatkovni zapis recepta Program

Svaki podatkovni zapis sastoji se od elementa recepta. Recepti HMI uređaja vrlo su slična struktura PLC podatkovnim strukturama, jer se ovdje isto definira skup različitih vrsta podataka koji se zatim više puta poziva u programski kod. Kod recepata se također koristi princip multipleksiranja. U memoriji HMI uređaja spremljeno je 300 zavarivačkih programa sa svim parametrima koji se zatim, ovisno o broju zavarivačkih programa, prebacuju u HMI tagove definirane u elementima recepta. U suštini svi zavarivački programi spremljeni su u memoriji HMI uređaja čime se oslobađa poprilična memorija PLC uređaja. Pritiskom na tipku „Učitaj“ na stranici „Parametri programa“ ili direktnim očitavanjem barkoda radnog komada, samo recept odabranog zavarivačkog programa učitava se u pripadajuće HMI tagove i njihove tagove PLC uređaja. Recepti se također mogu pritiskom na tipku „Uvezi“ učitati iz USB uređaja ili pritiskom na tipku „Izvezi“ na stranici „Postavke sustava“ spremiti na isti u „Backup.csv“ datoteku. Datoteke tipa .csv mogu se otvoriti programom MS Excel. To korisniku omogućava da sve zavarivačke programe otvori na svome računalu, gdje ih, na primjer, može prepraviti brže nego na stroju i prepravljene parametre učitati nazad u stroj. S druge strane, ako je korisnik zadovoljan sa strojem i želi investirati u još jedan identičan ili sličan stroj, parametre definirane u stroju N1 može pomoću USB uređaja prebaciti na novi stroj. Jedini uvjet je da recepti budu identično definirani u obadva stroja. Još jedna pogodnost funkcija „Izvezi“ i „Uvezi“ jest stvaranje pohrane zavarivačkih programa. Kada korisnik napiše novi zavarivački program, korištenjem funkcije „Izvezi“ lako može pohraniti novi program u

novu ili postojeću datoteku „Backup.csv“. Ako se u bilo kojem od zavarivačkih programa dogodi greška, pomoću funkcije „Uvezi“ korisnik vrlo jednostavno može u memoriju stroja učitati sve zavarivačke programe iz „Backup.csv“ datoteke spremljene na USB uređaju.

8.8 Tekstualne i grafičke liste

Kartica „Tekstualne i grafičke liste“ (eng. *Text and graphic lists*) koriste se za prikazivanje različitih tekstova i grafika u istim I/O poljima. Kao primjer navest ćemo I/O polje „Status stroja“ na početnoj stranici korisničkog sučelja. Ovisno u kojem načinu rada se stroj nalazi, primjerice automatskom, manualnom ili proceduri za reset, pripadajući tekst pojavit će se u I/O polju „Status stroja“. Grafička lista „Statusa stroja“ prikazana je na slici 8.12.



Slika 8.13: Prikaz grafičke liste Status stroja

U poglavlju 7.2.8 objašnjeno je na koji način se određuje način rada u kojemu se stroj nalazi. Sukladno odabranom načinu rada, predefimirani broj tog načina rada zapisuje se u procesnu veličinu „3. Machine_Data.Machine_Status“. Ta procesna veličina povezana je s HMI tagom „Machine_Status“ koji je definiran kao procesna veličina I/O polja „Status stroja“. U I/O polje kao tekstualna lista dodana je tekstualna lista „Machine_Status“. Ovisno koji broj se upiše u PLC tag „3. Machine_Data.Machine_Status“, odnosno HMI tag „Machine_Status“, tekst iz tekstualne liste „Machine_Status“ s istom vrijednošću

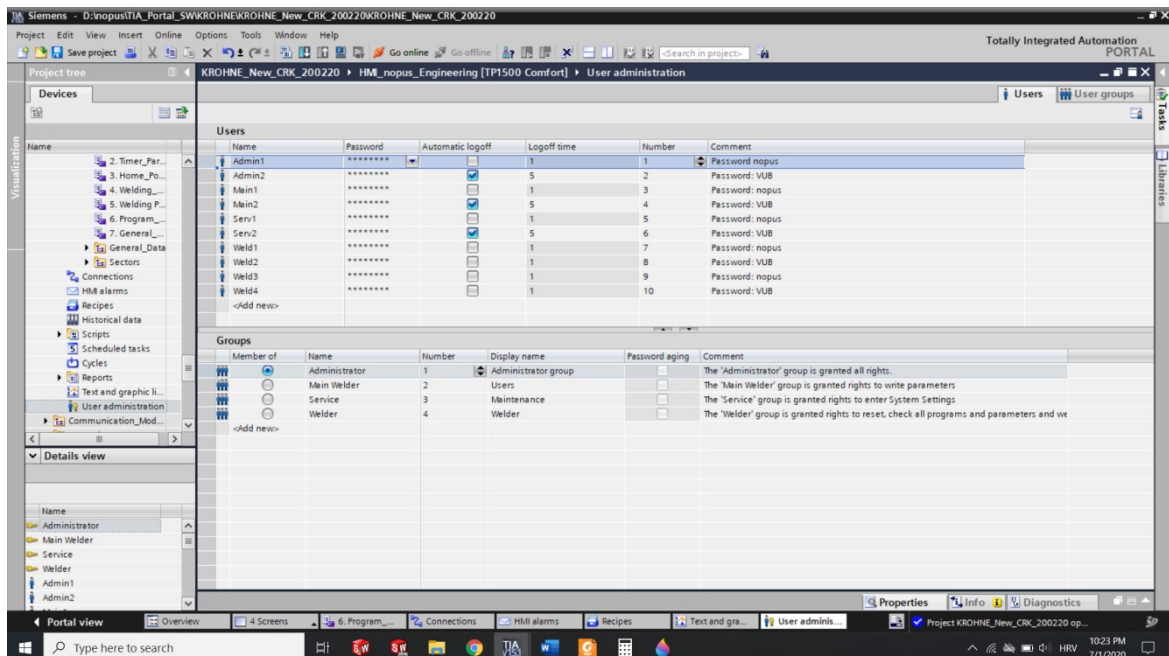
prikazat će se u I/O polju „*Status stroja*“. Isti se princip koristi se i za grafička polja. Primjer grafičkog polja je ikona zastave odabranog jezika na dnu početne stranice HMI uređaja. Ovisno koji jezik je odabran, hrvatski, engleski ili njemački, pripadajuća ikona zastave prikazat će se na dnu početne stranice korisničkog sučelja.

8.9 Korisnička administracija

Korisnička administracija podijeljena je u dvije kartice : „*Korisnici*“ (eng. *Users*) i „*Korisničke grupe*“ (eng. *User groups*). Stranica korisničke administracije prikazana je na slici 8.13.

Prva stvar koja se mora definirati jest autorizacija na kartici „*Korisničke grupe*“. Njima se definira stupanj dopuštenja pristupa funkcijama stroja za svakog korisnika koji ima mogućnost prijave u sustav. Stroj N1 ima četiri stupnja autorizacije:

- „*Administration*“
- „*Main Welder*“
- „*Service*“
- „*Welder*“



Slika 8.13: Stranica korisničke administracije

Autorizacijski stupanj „*Welder*“ ima najmanji stupanj autorizacije i može pristupiti samo određenim funkcijama stroja kao što su „*GLAVNA LISTA PROGRAMA*“ i „*UČITANI*

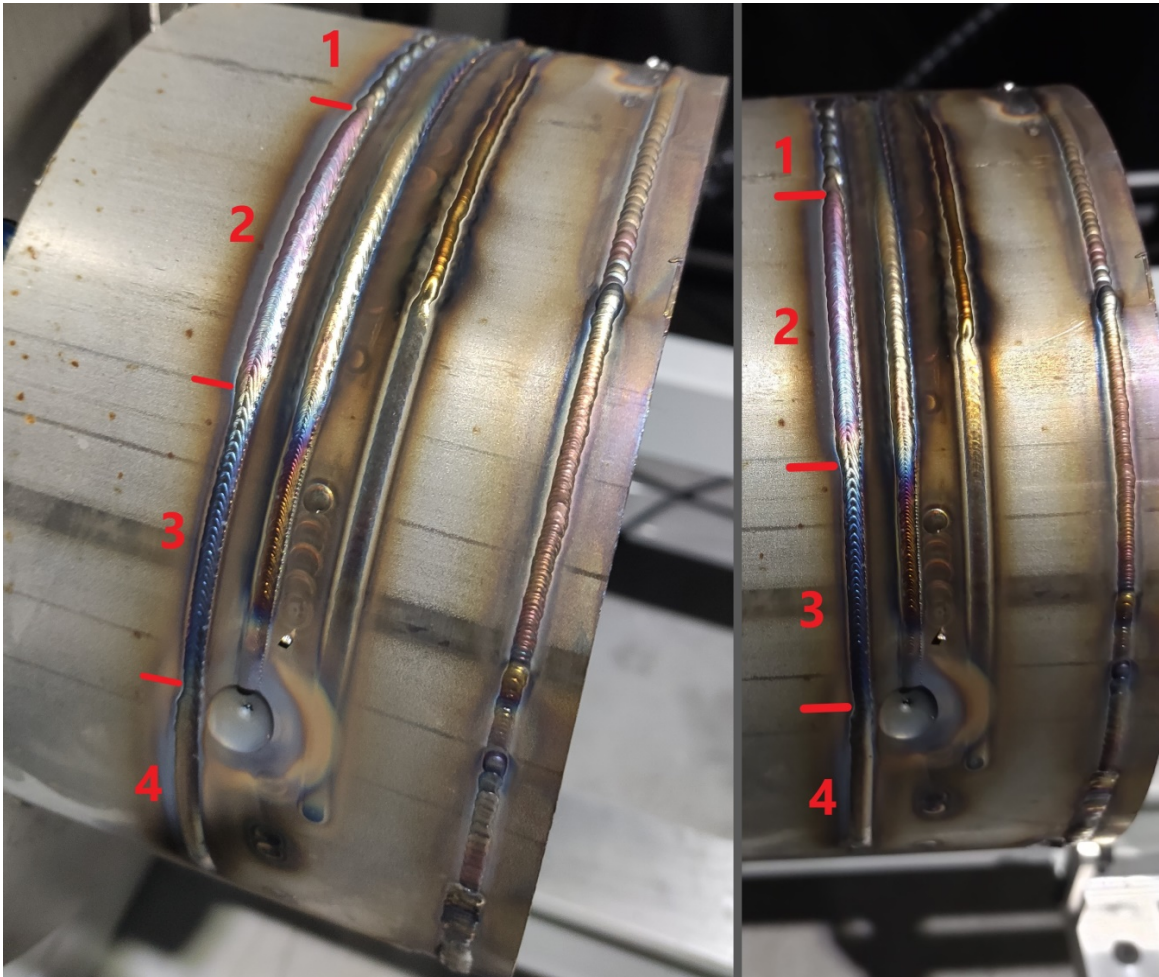
PARAMETRI“. Najmanji autorizacijski stupanj može pristupiti samo funkcijama stroja za koje je definiran, dok svaki sljedeći stupanj autorizacije ima pravo pristupa funkcijama stroja za koje je definiran i svim funkcijama koje su definirane za sve njemu niže stupnjeve autorizacije. Sukladno tome, autorizacijski stupanj „*Administrator*“ ima pristup svim funkcijama stroja N1.

Nakon definiranja stupnjeva autorizacije potrebno je odrediti korisničke grupe. Korisničke grupe stroja N1 imaju iste nazive kao i njihovi pripadajući stupnjevi autorizacije.

Zadnji korak definiranja korisničke administracije je određivanje korisnika koji imaju pravo pristupa funkcijama stroja. Svakom korisniku pripisuje se stupanj autorizacije i vrijeme nakon kojega će se stroj automatski odjaviti iz sustava. Korisnici „*AdminX*“, „*MainX*“ i „*ServX*“ imaju dvije vrste pristupa. Svako navedeno korisničko ime s brojem 1 ima pravo neograničenog pristupa stroju, tj. korisnik je prijavljen u sustav dok god je stroj upaljen ili dok se sam ne odjavi iz sustava. Svako navedeno korisničko ime s brojem 2 automatski će se odjaviti iz sustava nakon 5 minuta, bio korisnik aktivan na stroju ili ne. Kada se korisnika automatski odjavi iz sustava, sustav će ostati na stranici na kojoj se korisnik nalazio u trenutku odjave. Kao primjer navesti ćemo tipku postavke sustava koja zahtijeva autorizacijski stupanj „*Service*“. Ako se u trenutku odjave korisnik nalazi na stranici „*Postavke sustava*“, on će i dalje moći nesmetano raditi i listati sve stranice postavki sustava. Onoga trenutka kada se korisnik vrati na početnu stranicu, više neće moći pristupiti stranici „*Postavke sustava*“ jer se pritiskom na tipku „*Postavke sustava*“ od korisnika zahtijeva autorizacija. Drugim riječima, ako je korisnik prešao autorizacijski prag, u trenutku odjave iz sustava korisnik i dalje može koristiti sve funkcije koje se nalaze na stranicama nakon autorizacijskog praga. Onoga trenutka kada se korisnik vrati iza praga autorizacije, sve njegove autorizacijske ovlasti prestaju i potrebno se ponovno prijaviti u sustav ako bi želio pristupiti postavkama sustava.

9. PREZENTACIJA RADA STROJA N1

Na slici 9.1 prikazana su četiri testa zavarivačkog programa broj 1 kao konačni rezultat programskog rješenja opisanog u ovom završnom radu.



Slika 9.1: Četiri testa zavarivačkog programa

Prvi zavar s desna na lijevo je prvi zavar napravljen na stroju N1 i, kao što se vidi na slici, poslužio je za pronalazak parametara zavarivanja. Zavar estetski nije prihvatljiv jer nije rađen u jednom prolazu nego na više puta. Svaki put kada je uočena greška u programskom kodu, zavarivanje je zaustavljeno, greška ispravljena, te zavarivanje nastavljeno s mjesta gdje je prethodno bilo zaustavljeno.

Drugi zavar sastojao se od dva sektora. Na slici 9.1 vidljivo je kako je prvi sektor puno viši od drugoga. Razlog je to što se u prvom sektoru koristila žica za zavarivanje kao dodatni materijal, a drugi sektor je rađen bez dodatnog materijala samo sa električnim lukom.

Treći zavar je primjer krivo definiranih parametara zavarivanja. Prva tri sektora definirana su dobro, ali u četvrtom sektoru pojavila se greška. Naime četvrti sektor ostao je definiran s vrijednošću brzine zavarivanja od 0 mm/min. Svi ostali parametri zavarivačkog procesa bili su definirani dobro. U četvrtom sektoru jasno je vidljiv krater nastao nedostatkom rotacije u sektoru jer je zavarivanje radilo na jednom mjestu, te se bazen rastaljenog materijala proširio po površini komada. Radni komad ima debljinu mesa 4 mm pa se toplina unesena u njega brzo proširila po komadu što se vidi u promjeni strukture materijala oko nastalog kratera. Budući da je sektor četiri definiran s žicom za zavarivanje kao dodatnim materijalom, sam dodatni materijal i dio osnovnog materijala radnog komada propali su kroz radni komad na donju unutrašnju stranu cijevi.

Četvrti zavar je prvi zavar na stroju N1 koji je odrađen bez ikakvih grešaka u definiranju sektora ili grešaka u programskom kodu.

10. ZAKLJUČAK

Stroj Nopus N1 je potpuno automatizirana naprava za orbitalno zavarivanje. U električnom ormaru lijevo od stroja nalazi se kompletno hardversko rješenje upravljačkog sustava sa PLC uređajem kao glavnom upravljačkom jedinicom. PLC uređaj povezan sa korisničkim sučeljem na HMI uređaju koji se nalazi na gornjoj vanjskoj strani ormara. Preko korisničkog sučelja operater ima mogućnost definiranja parametara stroja i definiranja do 300 različitih zavarivačkih programa. Stroj se pogoni za 7 V90 servo upravljačkih sustava i 7 SIMOTICS S-1FL6 motora. Jedan motor pogoni rotacijsku os sa steznom čeljusnom glavom za prihvat radnog komada, a ostalih 6 motora služi za pozicioniranje linearnih osi za pozicioniranje gorionika za zavarivanje. Preko BUSINT X11 komunikacijskih modula PLC uređaj ima mogućnost upravljanja s 2 EWM Tetrix 552 Synergic RC HW aparata za TIG zavarivanje.

Strojem mogu upravljati samo korisnici kojima je dozvoljen pristup u sustav. Nakon što se korisnik uspješno prijavi u sustav, ovisno o stupnju autorizacije za svoje korisničko ime, određene funkcije stroja postaju mu dostupne. Prije puštanja stroja u rad, korisnik obavezno mora unesti parametre stroja poput brzina svih osi i njihovih tolerancija

Nakon definiranja parametara stroja korisnik smije definirati parametre zavarivačkih programa. Jedan zavarivački program sastoji se od jedne radne i jedne parking pozicije, te 20 sektora definiranih u stupnjevima u kojima korisnik može definirati različite parametre zavarivačkog procesa. Prednost rada s 20 sektora s različitim parametrima zavarivačkog procesa je u tome što se nakon jednog prolaza stroj ne treba zaustaviti kako bi se parametri zavarivačkog procesa izmijenili. Stroj će radni komad rotirati dok god ne dosegne korisnički definiranu vrijednost stupnjeva rotacije i pritom kod promjene u novi sektor automatski očitati parametre zavarivačkog procesa definirane u tom sektoru. Kada korisnik definira prvi zavarivački program mora ga pomoću tipke „*Snimi*” na dnu korisničkog sučelja spremiti u memoriju HMI uređaja kako bi ga kasnije mogao pozivati iz memorije i koristiti na stroju. Korisnik program može pozvati na dva načina: upisom broja programa u korisničko sučelje i zatim pritiskom tipke „*Skini*” na dnu korisničkog sučelja pozvati program ili direktno očitavanjem barkoda s radnog komada pomoću barkod čitača.. U simulacijskom načinu rada (kada je isključena tipka za aktivaciju zavarivanja na početnom ekranu) može testirati pozicije programa i očitati vrijednosti zavarivačkog procesa sa EWM Tetrix 552 Synergic RC HW aparata za TIG zavarivanje. Ako je korisnik zadovoljan

s testom zavarivačkog programa u simulacijskom načinu rada, aktivacijom tipke za aktivaciju zavarivanja i pritiskom tipke „*START*” na daljinskom upravljaču može pokrenuti automatski ciklus zavarivanja. Kada je automatski ciklus zavarivanja završen, stroj će se automatski vratiti u parking poziciju zavarivačkog programa kako bi se radni komad mogao izvaditi iz stroja. Nakon toga stroj je spreman za nastavak rada s istim ili učitavanje novog zavarivačkog programa.

Nakon svakog uspješno završenog automatskog ciklusa zavarivanja, stroj ažurira vrijednosti brojača ciklusa stroja. Korisnik u svakom trenutku može pogledati koliko automatskih ciklusa je napravljeno na stroju od kada je stroj pušten u pogon i koliko ciklusa je napravio svaki korisnik koji se može prijaviti u sustav.

11. LITERATURA

[1] Polysoude. Orbital and automated welding: Automated welding [Online]. 2020.

Dostupno na:

<https://www.polysoude.com/orbital-and-automated-welding/automated-welding/>(06.07.2020.)

[2] Schnelldorfer Maschinenbau. Circumferential Seam Welding Machine [Online]. 2020.

Dostupno na:

<https://www.schnelldorfer-maschinenbau.de/en/welding-machines/circumferential-seam-welding-machines.html>(06.07.2020.)

[3] SIEMENS. Industry Mall, Product Details: 6ES7215-1AG40-0XB0 [Online]. 2020.

Dostupno na:

<https://mall.industry.siemens.com/mall/en/WW/Catalog/Product/6ES7215-1AG40-0XB0>
(06.07.2020.)

[4] SIEMENS. SINAMICS S; G, V: Function blocks to control the SINAMICS with SIMATIC S7 in TIA-Portal [Online]. Dostupno na:

<https://support.industry.siemens.com/cs/document/109475044/sinamics-communication-blocks-drivelib-for-activation-intia-portal?dti=0&lc=en-WW> (06.07.2020.)

[5] SIEMENS. SINAMICS: SINAMICS V90 SINAMICS V-ASSISTANT Online Help, Operating Manual [PDF], SIEMENS, 2014. Dostupno na:

https://cache.industry.siemens.com/dl/files/527/93997527/att_79355/v1/OnlineHelp_V90_en_US_en-US.pdf (06.07.2020.)

[6] SIEMENS. SINAMICS/SIMOTICS: SINAMICS V90 SIMOTICS S-1FL6, Operating Instructions [PDF]. SIEMENS, 2014. Dostupno na:

https://cache.industry.siemens.com/dl/files/218/93005218/att_74528/v1/SINAMICS_V90_OPI_en-US_en-US.pdf (06.07.2020.)

[7] SIEMENS. SIMATIC: STEP 7 and WinCC Engineering V15.1, System Manual [PDF], SIEMENS, 2018. Dostupno na:

<https://support.industry.siemens.com/cs/document/109755202/simatic-step-7-basic-professional-v15-1-and-simatic-wincc-v15-1?dti=0&lc=en-KZ>

12. OZNAKE I KRATICE

eng. – engleski

mm – milimetar

mm/min – milimetara u minuti

ms – milisekunda

npr. – na primjer

s – sekunda

tj. – to jest

13. SAŽETAK

Naslov: Automatizacija uređaja za zavarivanje

Tema ovog završnog rada je automatizacija zavarivačkog procesa pomoću stroja N1. Rad pobliže opisuje zahtjeve zavarivačkog procesa koje stroj mora zadovoljiti i sve funkcije dostupne na stroju. Osnovne upute za rukovanje strojem opisane su na početku rada. Navedena je kompletna oprema koja je korištena u izradi automatizacije, te se detaljno opisuju funkcije i konfiguracija V90 servo upravljačkih sustava. Objasnjava se i opisuje kompletno programsko rješenje automatiziranog sustava u vidu koda upravljačke logike na PLC uređaju i vizualizacije na HMI uređaju. Rezultat završnog rada je potpuno automatizirani stroj za orbitalno zavarivanje s mogućnošću definiranja 300 različitih zavarivačkih programa od strane korisnika.

Ključne riječi: zavarivanje, automatizacija, PLC, HMI, V90 servo upravljački sustav

14. ABSTRACT

Title: Welding Machine Automation

Topic of this final paper is automatization of welding process with N1 machine. Paper closely describes requests of welding processes that machine has to fulfill and all functions available on the machine. Basic user manual for machine handling are described on the beginning of the paper. Complete equipment used for automatization is stated and detailed description and configuration of V90 servo converters is given. Complete programming solution of automatization system is described, both control logic code on PLC device and visualisation code on HMI device. Result of this final paper is completely automated orbital welding machine with ability of defining 300 different welding programs by the user.

Keywords: welding, automatization, PLC, HMI, V90 servo converter

IZJAVA O AUTORSTVU ZAVRŠNOG RADA

Pod punom odgovornošću izjavljujem da sam ovaj rad izradio/la samostalno, poštujući načela akademske čestitosti, pravila struke te pravila i norme standardnog hrvatskog jezika. Rad je moje autorsko djelo i svi su preuzeti citati i parafraze u njemu primjerenom označeni.

Mjesto i datum	Ime i prezime studenta/ice	Popis studenta/ice
U Bjelovaru, 26. 07. 2020.	KARLO TURKVIĆ	Karlo Turković

Prema Odluci Veleučilišta u Bjelovaru, a u skladu sa Zakonom o znanstvenoj djelatnosti i visokom obrazovanju, elektroničke inačice završnih radova studenata Veleučilišta u Bjelovaru bit će pohranjene i javno dostupne u internetskoj bazi Nacionalne i sveučilišne knjižnice u Zagrebu. Ukoliko ste suglasni da tekst Vašeg završnog rada u cijelosti bude javno objavljen, molimo Vas da to potvrdite potpisom.

Suglasnost za objavljivanje elektroničke inačice završnog rada u javno dostupnom nacionalnom repozitoriju

KARLO TURKVIĆ

ime i prezime studenta/ice

Dajem suglasnost da se radi promicanja otvorenog i slobodnog pristupa znanju i informacijama cjeloviti tekst mog završnog rada pohrani u repozitorij Nacionalne i sveučilišne knjižnice u Zagrebu i time učini javno dostupnim.

Svojim potpisom potvrđujem istovremest tiskane i elektroničke inačice završnog rada.

U Bjelovaru, 05.06.2022.

Karlo Turkvić

potpis studenta/ice