

# Upravljanje i nadzor automatskim sustavom za kontrolu hidraulične preše za prešanje metalnog otpada u tri osi

---

Vlahović, Ivan

Master's thesis / Diplomski rad

2023

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **University of Rijeka, Faculty of Engineering / Sveučilište u Rijeci, Tehnički fakultet**

Permanent link / Trajna poveznica: <https://um.nsk.hr/um:nbn:hr:190:373681>

Rights / Prava: [Attribution 4.0 International](#) / [Imenovanje 4.0 međunarodna](#)

Download date / Datum preuzimanja: **2024-12-30**



Repository / Repozitorij:

[Repository of the University of Rijeka, Faculty of Engineering](#)



SVEUČILIŠTE U RIJECI

TEHNIČKI FAKULTET

Diplomski sveučilišni studij elektrotehnike

Diplomski rad

**Upravljanje i nadzor automatskim sustavom za kontrolu hidraulične preše za  
prešanje metalnog otpada u tri osi**

Rijeka, studeni 2023.

Ivan Vlahović

0069083778

SVEUČILIŠTE U RIJECI

TEHNIČKI FAKULTET

Diplomski sveučilišni studij elektrotehnike

Diplomski rad

**Upravljanje i nadzor automatskim sustavom za kontrolu hidraulične preše za  
prešanje metalnog otpada u tri osi**

Mentor: Prof.dr.sc. Dario Matika

Rijeka, studeni 2023.

Ivan Vlahović

0069083778

Rijeka, 21. travnja 2023.

Zavod: **Zavod za automatiku i elektroniku**  
Predmet: **Automatizacija postrojenja i procesa**  
Grana: **2.03.06 automatizacija i robotika**

## ZADATAK ZA DIPLOMSKI RAD

Pristupnik: **Ivan Vlahović (0069083778)**  
Studij: Sveučilišni diplomski studij elektrotehnike  
Modul: **Automatika**

Zadatak: **Upravljanje i nadzor automatskim sustavom za kontrolu hidrauličke preše za prešanje metalnog otpada u tri osi/ Control and monitoring of the automatic control system of the hydraulic press for pressing metal waste in three axes**

### Opis zadatka:

Potrebno je projektirati i izraditi programsku aplikaciju za upravljanje i nadzor hidrauličke preše za prešanje metalnog otpada u tri osi. Projektiranje uključuje detaljan opis sustava, razvoj programske aplikacije upravljanja u programskom okruženju Siemens TIA portal, razvoj programske aplikacije vizualizacije u programskom okruženju AVEVA InTouch te funkcionalno ispitivanje sustava korištenjem aplikacije za simulaciju na programabilnom logičkom kontroleru Siemens Tia Portal S7-1500. Tijekom funkcionalnog ispitivanja sustava potrebno je provesti usporedbu različitih modova kontrole automatskog pozicioniranja hidrauličkih cilindara.

Rad mora biti napisan prema Uputama za pisanje diplomskih / završnih radova koje su objavljene na mrežnim stranicama studija.

  
Zadatak uručen pristupniku: 20. ožujka 2023.

Mentor:

  
Prof. dr. sc. Dario Matika

  
Dr. sc. Dominik Cikač (komentor)

Predsjednik povjerenstva za  
diplomski ispit:

  
Prof. dr. sc. Dubravko Franković

# IZJAVA

Izjavljujem da sam ovaj diplomski rad izradio potpuno samostalno, uvažavajući načela akademske čestitosti.

Rijeka, studeni 2023.

---

Ivan Vlahović

# ZAHVALA

Zahvaljujem se mentoru prof. dr. sc. Dariu Matiki i komentoru dr.sc. Dominiku Cikaču na savjetima i izdvojenom vremenu tijekom izrade ovog diplomskog rada.

Zahvaljujem se i tvrtki Danieli Systec na danoj mogućnosti korištenja njihove opreme, te njezinim djelatnicima na brojnim savjetima tijekom izrade ovoga diplomskog rada.

# 1. Sadržaj

1. Sadržaj.....	1
2. Uvod.....	3
3. Programibilni logički kontroler - PLC .....	4
3.1. Dijelovi PLC-a .....	6
3.1.1 Centralna procesorska jedinica.....	7
3.1.2 Memorija .....	9
3.1.3 Napajanje.....	10
3.1.4. Ulazno/izlazni moduli .....	11
3.1.5. Ostali moduli.....	13
3.2. Princip rada PLC-a .....	14
3.3. SIMATIC S7-1500.....	15
4. Hidraulična preša.....	16
4.1. Hidraulična jedinica .....	19
4.2. Tehnička specifikacija.....	20
4.3. Senzori.....	21
5. Programiranje softvera za upravljanje i nadzor hidraulične preše .....	22
5.1. Automatska sekvenca .....	23
5.2. „Home“ sekvenca .....	36
5.3. Funkcija automatske i „home“ sekvence .....	37
5.4. Funkcija upravljanja cilindrom .....	41
5.5. Funkcija machine selftest .....	48
5.6. Funkcija hidraulične jedinice .....	50
5.7. Funkcija simulacije rada.....	52
5.8. Usporedba različitih modova kontrole automatskog pozicioniranja cilindara.....	56
6. Radio daljinska jedinica .....	58

7.	Vizualizacija.....	65
7.1.	LANDEF .....	66
7.2.	Izrada stranica vizualizacije .....	67
7.3.	Izrađene stranice vizualizacije .....	68
8.	Zaključak.....	74
9.	Literatura .....	75
10.	Sažetak i ključne riječi .....	76
11.	Summary and key words .....	77
12.	Popis slika i tablica.....	78
	Dodatak A. Automatska sekvenca .....	81
	Dodatak B. Funkcija machine selftest za pomak cilindra C1 unaprijed .....	83
	Dodatak C. LANDEF <i>sheet</i> statusnih varijabli .....	85



## 2. Uvod

Automatski sustav hidraulične preše za prešanje metalnog otpada koristi se u procesu recikliranja starog metala. Ovaj sustav može obraditi različite vrste i dimenzije metalnih materijala, pa tako primjerice od komada ili kompletnih vozila, bačvi do šipki, žica, limenih profila i slično. Unutar same preše je potrebno postaviti željeni metalni materijal, te ručnim komandama isprešati metal u kompaktnu balu nizom pokreta triju hidrauličnih cilindara. Ova tri cilindra vrše prešanje u sve tri osi. Osim toga postoji odabir automatske sekvence, što znači da je potrebno samo postaviti metal unutar preše i pokrenuti sekvencu. U nekim primjerima unutar preše postoji senzore težine, koji omogućuje automatsko pokretanje sekvence kada se postigne određena težina metala.

Postoje različite vrste hidrauličnih preša kao što su PTC (engl. *Press triple compression*), PBC (engl. *Press briquetting compression*) te PSC i PDC (engl. *Press single/double compression*) hidraulične preše, koje se razlikuju u svojoj konfiguraciji i karakteristikama. To uključuje dimenzije bale, snagu, silu kompresije, broj potrebnih pumpi i slično. Također, neke izvedbe imaju prednosti kod obrade određenih metalnih materijala. Sama primjena hidrauličnih preša se našla radi transporta metalnog otpada, gdje je ne isprešani metalni otpad skup za skladištenje i transport.

Ukoliko se uz navedena tri cilindra doda dodatni koji bi imao oštricu za rezanje metalnih bala na manje komade, ta se mašina u industriji često naziva "hidraulične škare". Princip je sličan, izvršavanje kompresije metala u tri osi uz dodatno rezanje nakon prešanja.

Sama hidraulična preša dakle ispreša metalni otpad u balu odgovarajuće dimenzije koja se dalje transportira. Primjerice, bale se postavljaju u *shredder* koji ih melje na sitne komade. Često se prije *shredder*-a koristi i *preshredder* koji veće komade usitnjuje u manje prije nego što idu na dodatno usitnjavanje unutar *shredder*-a. Nakon toga, slijedi niz pokretnih traka, magneta, tresilica te separatora kako bi se pravilno odvojio različiti metalni otpad. Nakon čega se pojedini metali mogu taliti i koristiti za proizvodnju novih metalnih proizvoda. Tu završava proces recikliranja metalnog otpada.

### 3. Programibilni logički kontroler - PLC

Programibilni logički kontroler, poznat i kao PLC (*engl. programmable logic controller*), predstavlja vrstu računala specijaliziranu za industrijske primjene u obliku kontrolnih aplikacija za automatskog upravljanje i optimizaciju rada različitih procesa. Primjena PLC-a u različitim industrijskim procesima rezultira povećanom učinkovitošću, pouzdanošću i smanjenjem troškova. Osnovni razlozi za korištenje PLC-a u industriji leži u njihovoj visokoj pouzdanosti, sposobnosti da izdrže nepovoljne atmosferske uvjete, široke temperaturne raspone, vibracije i elektromagnetske smetnje. Hardver i softver koji se u njima koriste se znatno razlikuju od onih koji se koriste u standardnim uredskim računalima. Koja su optimizirana za zadatke izračuna i prikaza, dok je PLC za zadatke upravljanja i industrijsko okruženje.[1]

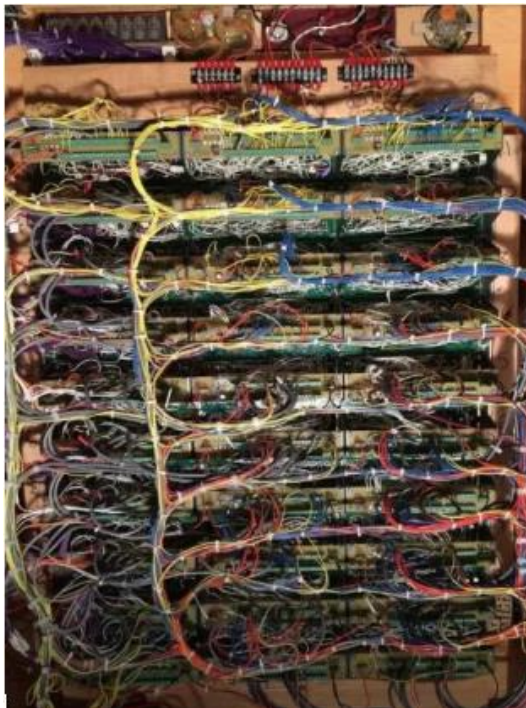
PLC posjeduje programibilnu memoriju u koju se unose naredbe i instrukcije koje definiraju logiku i akcije PLC-a. Drugim riječima, oni analiziraju ulazne varijable, primjenjuju programiranu logiku na njih i donose odluke koje rezultiraju promjenama u izlaznim varijablama. Na taj način automatizira se rad industrijskih procesa.

Primjena PLC-a je izuzetno raznolika, a ovaj rad će se konkretno baviti primjerom primjene u upravljanju hidrauličnom prešom. Korišten je SIEMENS-ov PLC SIMATIC S7 – 150 CPU 1518F -4 PN/DP V2.6 prikazan na slici 3.1.



Slika 3.1. PLC [2]

Razvojem PLC-a zamjenjuju prijašnje sustave automatizacije koja su se temeljila na relejima (Slika 3.2.). U slučaju kompleksnoga procesa upravljanje je zahtijevalo primjenu velikog broja releja te žičanih veza sa ostatkom sustava.



*Slika 3.2. Relejni ormar [3]*

Releji su elektromehanički uređaji koji svoj rad temelje na elektromagnetskom principu. Imaju ograničeni "mehanički životni vijek", odnosno ograničen broj operacija sklapanja/preklapanja. U današnje vrijeme taj broj često prelazi milijun operacija, no u prošlosti je bio znatno manji. Kvar jednog releja u kompleksnom sustavu može stvarati probleme u dijagnostici i otklanjanju kvara. Također, problemi se javljaju prilikom promjene logike upravljanja sustavom ili dodavanja nove komponente unutar sustav. To zahtijeva prespajanje postojećih žičanih veza i dodavanje novih releja s dodatnim žicama. Glavni razlog ovih problema leži u potrebnom vremenu, ali često su i sheme bile neispravne ili nepotpune. Iz tih razloga, sustavi temeljeni na PLC-u zamjenjuju ove elektromehaničke releje, rješavajući navedene probleme, smanjuju potrebu za znatnim brojem žica, a promjene u upravljanju ili dodavanje komponenti u sustav postižu se promjenom programskog koda. Prelaskom na novu tehnologiju automatizacije, zahtijevalo se da izgled programa PLC-a bude razumljiv i koristan tadašnjim inženjerima i električarima na održavanju. Pa je tako jedan od osnovnih načina programiranja PLC-a koncipiran na relejnoj logici u vidu ljestvičastog dijagrama koji se zatim pretvara u računalni strojni jezik i zapisuje u memoriju.

### 3.1. Dijelovi PLC-a

PLC se sastoji od centralne procesorske jedinice (*engl. Central processing unit*) CPU, ulazno i izlaznih (I/O) modula, memorije te izvora napajanja. Navedene komponente mogu se smatrati arhitekturom, no arhitekturom se može smatrati osim hardverskog dijela i onaj softverski. Postoji razlika između sistema otvorene arhitekture u kojoj su komponente proizvedene po određenim standardima, pa je u tom slučaju moguće povezivanje komponenti različitih proizvođača. Dok s druge strane zatvorenu arhitekturu štite proizvođači komponentata pa je povezivanje komponenti različitih proizvođača teško ostvarivo. [4]

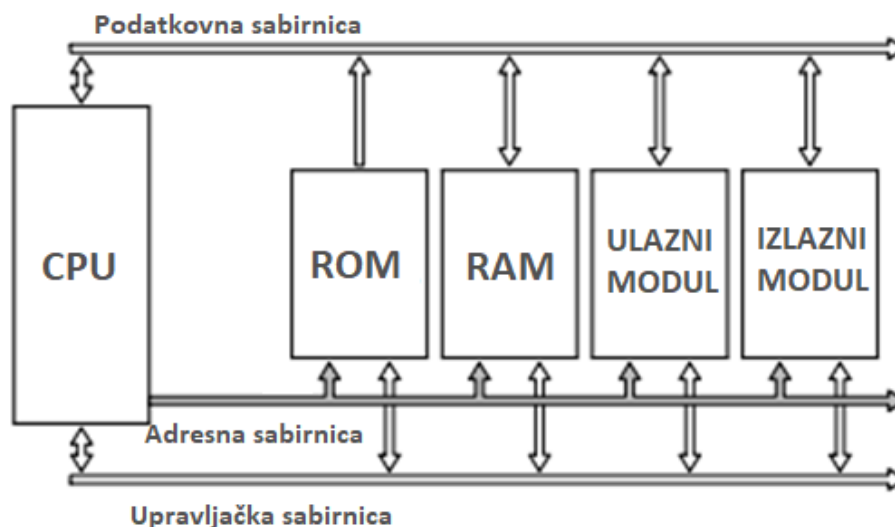
Također, postoji podjela i u samoj konfiguraciji PLC-a. Ukoliko su CPU, napajanje, memorija, ulazi, izlazi te komunikacijsko sučelje između njih integrirano unutar zajedničkog kućišta, to predstavlja kompaktni PLC. Oni se proizvodi s ograničenim brojem ulaza i izlaza koji se ne mogu naknadno nadograditi što značajno ograničena fleksibilnost upotrebe. No s druge strane to smanjuje samu cijenu PLC-a. Primjena se pronalazi u najjednostavnijim sustavima automatizacije sa vrlo malim brojem ulaznih i izlaznih jedinica. U drugoj vrsti konfiguracije CPU i ulazno/izlazni moduli su odvojeni no posjeduju posebne konektore za njihovo međusobno povezivanje čime se stvara sabirnička komunikacija. Ta sabirnica se naziva stražnja ploča (*engl. backplane*).

Modularna konfiguracija omogućuje dodavanje velikog broja modularnih komponenti na zajedničku baznu jedinicu (*engl. rack*) koja služi kao sabirnica za međusobnu komunikaciju svih modularnih komponenti smještenih na njoj. Najčešće su prva dva mjesta s lijeve strane rezervirana za modul napajanja i CPU, dok se ostali moduli postavljaju u nastavku, a oni mogu biti funkcijski, komunikacijski ili signali moduli. Ovo značajno povećava funkcionalnost upotrebe, no i u slučaju kvara primjerice unutar neke komponente kompaktnog PLC-a, cijeli uređaj mora biti zamijenjen, dok kod modularnih PLC-ova potrebno je zamijeniti samo neispravnu komponentu. Moderne verzije modularne konfiguracije čak omogućavaju zamjenu modularnih komponenti i kada je PLC pod napajanje i u radu (*engl. hot swapping*). [5]

### 3.1.1 Centralna procesorska jedinica

CPU je središnji dio PLC-a te upravlja svim njegovim procesima donoseći odluke prema unaprijed definiranoj logici od strane programera. CPU se sastoji od memorije i procesora koji može biti sastavljen od više jezgri. Mikroprocesor obavlja donošenje odluka ažurirajući izlazne varijable na osnovu ulaznih varijabli i to prema upravljačkom algoritmu izrađenog od strane programera, koji se zapisuje unutar programibilne memorije. Također nadzire ostale dijelove PLC-a te njihovu međusobnu interakciju. Obavlja logičke operacije (I,ILI i slično), aritmetičke operacije poput zbrajanja, oduzimanja, dijeljenja te dodatnih operacija poput rada brojača, tajmera, okidača i komparatora. Jedinica za aritmetičke i logičke operacije se naziva ALU (*engl. arithmetic and logic unit*).

Tijekom svog rada CPU kontinuirano komunicira s ostatkom sustava (ulazima, izlazima, memorijom) putem adresne i podatkovne sabirnice. Adresna sabirnica se koristi za prijenos adresa memorijskih lokacija, pa se preko adresne sabirnice odgovarajuća adresna lokacija identificira čime je ona spremna za slanje/primanje podataka preko podatkovne sabirnice koja služi upravo za prijenos podataka unutar PLC-a koji se koriste u obradi koju provodi CPU. Osim njih prisutna je i upravljača sabirnica koja prenosi vremenske upravljačke signale do svih potrebnih jedinica kako bi se funkcije procesora mogle odvijati sinkronizirano, te aktivira CPU ili I/O module/memoriju kao pošiljatelja ili primatelja podatka. Dok sustavna sabirnica stvara komunikaciju između ulaznih/izlaznih priključaka i ulazne/izlazne jedinice. Na slici 3.3 je prikazana arhitektura CPU-a.



Slika 3.3. Arhitektura CPU-a [6]

CPU je, dakle najbitniji dio PLC-a, toliko da se čak različite verzije PLC-a razlikuju upravo po njemu. Na primjer, Siemens-ovi PLC-ovi se dijele na S7 – 400, S7 – 1200, S7 – 1500, a svaka od tih verzija se daljnje dijeli na različite modele kao što su CPU 412-1, CPU 414-1, CPU 414-2 kod verzije S7 – 400, ili pak CPU 1211C, CPU 1212C, CPU 1215C kod S7 – 1200, dok kod S7 – 1500 CPU 1511-1 PN, CPU 1513-1, CPU 1518-4 PN.

Siemens CPU-ovi također se razlikuju u nekoliko verzija. Osim standardne postoje verzije sa integriranim sigurnosnim funkcijama (*engl. failsafe*, oznaka F uz CPU) za sprječavanje ili ublažavanje nesigurnih posljedica kvara sustava, kao što su ozljede ili oštećenja opreme. Verzija SIPLUS CPU se primjenjuje u ekstremnim uvjetima okoline gdje je prisutna visoka temperatura, vlaga i slično. Tehnološka verzija MFP CPU (*engl. multifunctional platform CPU*) omogućuju izvođenje programa napisanih C/C++ programskim jezicima pa se mogu primijeniti za složene i napredne aplikacije koje zahtijevaju specifične algoritme i prilagodbe. Redundantna verzija dizajnirana za primjenu kada je ključna konstantna dostupnost samog kontrolera, odnosno u kritičnim sustavima gdje se ne smije dogoditi prekid u kontroli procesa.

Kod S7-1200 standardnih verzija CPU se razlikuje i po konfiguracijama napajanja. Kombinacije su DC/DC/DC, DC/DC/RLY ili AC/DC/RLY. Prva oznaka označava tip napajanja koju CPU zahtjeva sa svoj rad, DC(24V) ili AC(230V), druga oznaka označava tip digitalnih ulaznih kanala, dok treća oznaka za tip digitalnih izlaza(24VDC kod DC, ili relejni izlaz sa AC/DC snagom kod RLY tipa).

### 3.1.2 Memorija

Memorija PLC-a služi za pohranu operacijskog sustava, svih korisničkih programa, instrukcija te podataka kako bi im se moglo pristupiti i obraditi tijekom rada. Razlikuju se nekoliko vrsti memorija primijenjenih unutar PLC sustava.

ROM memorija (*engl. Read Only Memory*) omogućava samo čitanje sadržaja te se koristi za pohranu operacijskog sustava, programa i podataka koji definiraju mogućnosti PLC-a. Operacijski sustav je zapisan u ROM memoriju od strane proizvođača. Ova memorija je trajna i ne gubi svoj sadržaj prilikom prekida napajanja, čime se osigurava očuvanje podataka.

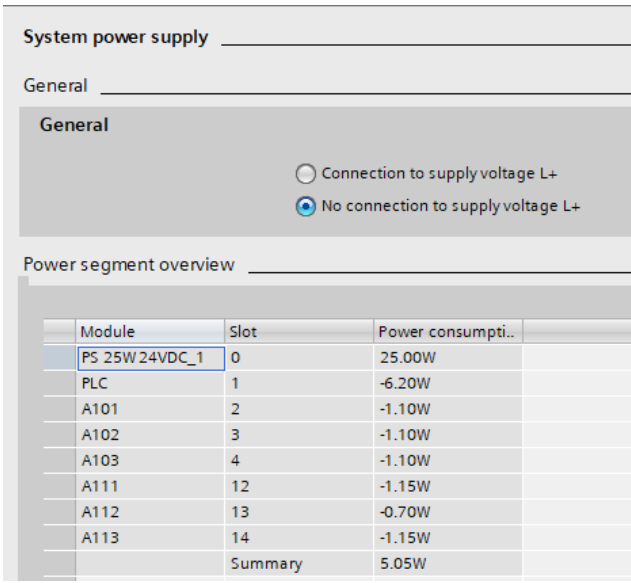
RAM (*engl. Random Acces Memory*) ili memorija s izravnim pristupom omogućuje čitanje i pisanje podataka unutar nje. Koristi se za pohranu korisničkih programa, instrukcija i podataka poput statusa ulaznih uređaja, vrijednosti unutarnje logike PLC-a. Prilikom definiranja nove memorijske varijable unutar korisničkog programa, ona se pohranjuje unutar RAM memorije koja se prepisuje prilikom svake promjene istog. To je netrajna memorija koja gubi pohranjene podatke u slučaju prekida napajanja. Da bi se spriječio gubitak programa koristi se pomoćna baterija. Neki sustavi posjeduju i kondenzatore u slučaju da baterija nije umetnuta.

Nakon što je korisnički program razvijen i testiran unutar RAM memorije kako bih on ostao stalan se često prepisuje unutar EPROM/EEPROM memorije. Kod EPROM (*engl. Erasable Programmable Read Only Memory*) memorije podatci se mogu izbrisati jedino primjenom ultraljubičastog UV zračenja. Sam proces brisanja takve memorije je često prezahtjevan jer zahtjeva posebni uređaj za brisanje nakon čega slijedi period hlađenja, pa je njihovu primjenu uvelike smanjilo primjena EEPROM memorije (*engl. Electrically Erasable Programmable Read Only Memory*). Takvu memorije je moguće prebrisati električnim naponom.

Memorija SIEMENS-ovih PLC-ova se dijeli na *load*, radnu, sistemsku te *retentive* memoriju. *Load* memorija (ROM) ili memorija učitavanja je stalna memorija unutar koje se pohranjuje korisnički program, podatci te konfiguracije. Prilikom preuzimanja programa na CPU prvo se pohranjuje unutar nje. Moguće ju je proširiti sa memorijskom karticom, koja je kod nekih tipova PLC-ova nužna za rad. Radna memorija (RAM) je nestalna memorija u koju se pohranjuje kod i podatkovni blokovi. Integrirana je u CPU i ne može se proširiti te se koristi samo tijekom rada CPU-a. *Retentive* (EEPROM) ili zadržavajuća memorija je odgovorna za pohranu ograničene vrijednosti radne memorije, odnosno odabranih korisničkih memorijskih lokacija. Sistemska memorija može pohraniti sve podatke adresiranja korisničkog programa (ulaze, izlaze, brojače).[5]

### 3.1.3 Napajanje

Izvor napajanja osigurava potrebni napon za kompletan PLC sustav, uključujući i CPU i ulazno/izlazne module. Kao jedna od njegovih ključnih uloga je održavanje stabilnosti napona kako bi se izbjegle vršne vrijednosti koje bi mogle negativno djelovati na proces kojim PLC upravlja ili oštetiti same komponente PLC-a. Izvor napajanja mora pretvoriti ulaznih 230VAC u odgovarajuću istosmjernu vrijednost, koja je obično 24VDC. Proces započinje transformatorom koji pretvara 230VAC na 24VAC, nakon čega se koristi ispravljač, primjerice Graetzov spoj, za daljnje ispravljanje napona u istosmjernu vrijednost. Nakon toga, kondenzatori se koriste za zaglađivanje napona, pri čemu također služe i kao spremnici energije u slučaju prekida napajanja sa ulazne strane izvora napajanja. Slijedi regulator koji osigurava da izlazna vrijednost bude što bliža referentnoj vrijednosti napona za rad ostatka sustava. Sam izvor napajanja se obično montira na zajednički stalak na prvoj lijevoj poziciji, zatim slijede CPU i ulazno/izlazni module, te ostali potrebni moduli poput komunikacijskog i mjerača energije. Na ovaj način izvor napajanja može pružiti potreban napon putem zajedničke sabirnice na stalku, ili u protivnom je potrebno svaki modul posebno ožičiti. U kritičnim sustavima gdje je nužna kontinuirana dostupnost napajanja se koristi redundantno napajanje. Odnosno prilikom ispada jednog od izvora napajanja, drugi preuzima tako kako bi se izbjegao prekid u radu sustava. Također se kao zaštita od prekida napajanja koriste baterije ili superkondenzatori.[10] Na slici 3.3. je prikazan dijagnostika snage PLC sustava unutar TIA portala za primjer izvora napajanja od 25W, CPU-at te nekoliko I/O modula.



Module	Slot	Power consumpti..
PS 25W 24VDC_1	0	25.00W
PLC	1	-6.20W
A101	2	-1.10W
A102	3	-1.10W
A103	4	-1.10W
A111	12	-1.15W
A112	13	-0.70W
A113	14	-1.15W
Summary		5.05W

Slika 3.4. Pregled segmenata snage



### 3.1.4. Ulazno/izlazni moduli

Ulazno/izlazni moduli omogućavaju vezu između PLC-a i različitih uređaja unutar procesa upravljanja. Odnosno opskrbljivanju PLC sa svim potrebnim podacima o stanju sustava, te omogućuju upravljanje sustavom slanjem upravljačkih signala natrag u proces. Modul može sadržavati samo ulaze, samo izlaze ili kombinaciju oba. Razlikuju se digitalni (diskretni) i analogni moduli te moduli specifične primjene.

Kako je već spomenuto kod kompaktnih izvedbi PLC je ograničen na relativno mali broj ulazno/izlaznih priključaka pa se takvi koriste u jednostavnim procesima, dok kod modularnih omogućeno je povećanje broja priključaka dodavanjem dodatnih ulazno/izlaznih modula, čime se može konfigurirati PLC sustav za najspecifičnije primjene.

Ulazni moduli sadržavaju priključne stezaljke na koju se spajaju različiti uređaji u „polju“ poput senzora, pretvarača, tipkala, prekidača i sličnih ulaznih uređaja procesa kojim PLC upravlja. Moraju pouzdano detektirati signal i njegovu vrijednost, podesiti napon na vrijednost primjerenu za rad CPU-a te zaštititi PLC-a od prevelikih vanjskih napona. Kao zaštita se koristi kombinacija LED diode i fototranzistora (poznato i pod nazivom *engl. optocoupler*) čime je ostvarena električna izolacija vanjskog i unutarnjeg kruga.

Kod digitalnih ulaznih modula signali mogu biti različitih vrijednosti i to istosmjerne ili izmjenične vrijednosti. Primjerice 24, 110 ili 230 V DC ili AC. Takvi ulazni moduli moraju primjenom ispravljača pretvoriti izmjeničnu u istosmjernu komponentu, zatim vrijednost spustiti na potrebnu primjenom otpornika te zatim stabilizirati kondenzatorima, nakon čega signal prolazi kroz LED diodu koja u slučaju visoke vrijednosti signala emitira infracrveno zračenje koje aktivira fototranzistor. Kod digitalnih signala se dakle razlikuju dva stanja 0 ili 1 (ON/OFF, *True/False*). Kao najčešća naponska razina za digitalne ulaze se koristi ona od 24VDC, gdje se visokom (logička 1) vrijednošću smatra 15-24V, dok niskom 0-5V (logička 0). Ovo je bilo objašnjeno za primjer *sourcing* ulaza koji opskrbljiva PLC sa visokom vrijednošću (24VDC) kada je primjerice senzor aktivan: Dok druga primjena nazvana *sinking* opskrbljiva PLC u tom slučaju sa referentom vrijednošću (0V). Sličan princip *sourcing* ili *sinking* vrijedi i kod izlaznih modula.[7]

Dok ako je riječ o analognim ulaznim modulima razlikuju se naponske i strujne ulazne vrijednosti te njihovi rasponi vrijednosti. Strujni signali mogu biti u rasponu od 0 do 20 mA ili od 4 do 20 mA. Preporučeno je koristiti raspon od 4 do 20 mA jer je dobro rješenje za brzo

dijagnosticiranje prekida voda. U tom slučaju bi vrijednost signala bila 0 mA što bi se odmah moglo registrirati kao kvar. Za naponske razine također se razlikuju različiti rasponi primjerice 0-10 VDC ili od -10 do +10 VDC za registriranje negativnih vrijednosti. Sve te vrijednosti je potrebno pretvoriti u vrijednosti koje su značajne, to se radi skaliranjem unutar softvera. Unutar softvera se neće vidjeti skala od 4-20mA nego njihova digitalna reprezentacija koju je potrebno skalirati prema specifičnoj primjeni što taj signal znači, primjerice brzina vrtnje, napunjenost spremnika, otvorenost ventila i slično.

Izlazni moduli sadržavaju priključne stezaljke na koju se spajaju signali namijenjeni izvršnim uređajima procesa, odnosno elektromotorima, signalnim lampicama, elektromagnetnim ventilima, razvodnicima, relejima i slično. Izlazni modul od CPU-a prima i obrađuje signale niskog napona, pa je potrebno naponsku razinu prilagoditi upravljačkom naponu. Također je potrebno pojačanje snage za aktiviranje vanjskih uređaja. Slično objašnjenje je kao kod ulaznih modula ali naravno uz očito razliku da PLC šalje upravljačke signale prema uređajima.

Digitalni izlazni moduli mogu bit izvedeni kao relejni, tranzistorski ili pomoću trijaka. Kod relejnog tipa unutar izlaznog modula postoje zavojnica i kontakti releja koji mijenjaju svoj položaj ovisno o naponu na zavoju releja koji dolazi sa CPU-a. Kontakti su spojeni na vanjsko napajanje za uključivanje ili isključivanje uređaja. Releji osiguravaju električnu izolaciju između opterećenja i PLC-a, oni su sporijeg djelovanja nego tranzistorski. Tranzistorski se koriste samo za sklapanje istosmjernih tereta ali su brži su od relejnih. Izvedba pomoću trijaka je slična tranzistorskoj po brzini sklapanja ali se za razliku od tranzistorski koriste samo za sklapanje izmjeničnih tereta. [1]

Analogni izlazni signali primaju podatke od CPU-a u digitalnoj formi koje transformira u odgovarajuću vrijednost istosmjernog napona ili struje. Kao bitna stavka kod analognih ulazni/izlaznih modula je njihova rezolucija, koja može biti primjerice 12-bitna ili 16-bitna. Jasno što je rezolucija veća da će preciznost vrijednosti analognog modula biti veća, ukoliko se koristi 4-20 mA i 16-bitna rezolucija, svaka promjena od 0,000244140625 mA će biti registrirana.

Ukoliko je riječ o SIEMENS-ovim modulima, digitalni moduli za svaki I/O kanal posjeduju posebnu LED žaruljicu koja svojom zelenom bojom prikazuje status pojedinog signala kanala. Kod analognih vrijedi slično samo prikazuje da je kanal konfiguriran i aktivan. Također ulazni i izlazni moduli posjeduju i LED žaruljicu za dijagnostiku (DIAG) koja svijetli zelenom bojom kad je modul operativan, dok crvenom kada je neispravan. [8]

### 3.1.5. Ostali moduli

Ukoliko PLC sustavu nedostaje nekoliko ulaznih ili izlaznih kanala, umjesto da se dodaje novi modul moguće je na CPU dodati signalnu pločicu. Koja će unesti dodatne potrebne kanale, signalna pločica obično posjeduju od 1 (analognog) do 4 (digitalna) kanala.

Moduli mjerača energije omogućuje upravljanje energijom putem bilježenja nekoliko stotina različitih električnih mjerenja i energetske vrijednosti. Ovo uključuje ulazne vrijednosti za cijeli proces/postrojenje, distribuciju pojedinačne linije te krajnja električna opterećenja strojeva i uređaja. Modul bilježi mjerenja na temelju kojima se određuje potrošnja energije te potrebe za snagom, prema čemu se mogu odrediti predviđanja za buduću potrošnju te donositi mjere za smanjenje iste. Također bitno djeluje na planiranje planskih zastoja procesa zbog održavanja. [8]

Komunikacijski moduli omogućuju komunikaciju između PLC-a i ostalih uređaja, računala i mreža. Sam CPU posjeduje nekom komunikacijskih portova, ovisno o vrsti CPU, obično od 1 do 4 port-a. Svaki od njih podržava određeni komunikacijski protokol kao što su PROFINET, PROFIBUS, Ethernet i serijski komunikacijski protokol.

PROFINET je otvoreni industrijski Ethernet protokol za izmjenu podataka između PLC-a i ostalih uređaja. PROFIBUS je digitalni komunikacijski protokol temeljen na serijskoj izmjeni podataka.

Postoje posebni komunikacijski moduli koji se dodaju na zajednički stalak s CPU-om te i omogućuju CPU-u da djeluje kao "*master*" pram drugim uređajima. Također postoje moduli koji omogućuju CPU-u ulogu "*slave*"-a unutar komunikacijskog protokola. Osim toga, postoje komunikacijski moduli koji podržavaju ASi (*engl. Actuator Sensor interface*) protokol. To je protokol koji omogućuje jednostavno povezivanje senzora i aktuatora upotrebom samo dvije žice. Moduli poput komunikacijskog procesora, PLC sustav pretvaraju u RTU (*engl. Remote Terminal Unit*), koji komunicira sa udaljenom centralnom stanicom. Takav modul se kod SIEMENS-a naziva *TeleControl* i omogućuje komunikaciju sa centralnom stanicom pomoću radio ili internetske veze. [9]

Memorijski moduli omogućuju proširenje kapaciteta memorije PLC-a kako bi se pohranili programi i podatci.

Sigurnosni moduli koji osiguravaju siguran rad PLC-a u smislu zaštite od ozljeda i oštećenja. Ovi moduli su prepoznatljivi po žutim oznaka kod SIEMENS opreme.

### 3.2. Princip rada PLC-a

PLC kontinuirano prati ulazne vrijednosti, te prema njima i prema korisničkoj logici ciklički ažurira vrijednosti na svom izlazu unutar beskonačne petlje sve dok je PLC u RUN-u. To je repetitivan proces koji se često naziva ciklus.

PLC obrađuje signale s ulaznih modula i zapisuje njihove vrijednosti u područje interne memorije koje se naziva "slika ulaza". Na taj način je osigurava se konzistencija vrijednosti tijekom izvršavanja korisničkog programa. U suprotnom neka od vrijednosti mogla bi se promijeniti tijekom izvođenja programa što bi rezultiralo stvaranjem netočnih izlaznih vrijednosti.

Nakon toga, CPU obrađuje korisnički program koji se sastoji od niza blokova. U okviru TIA Portala, glavni organizacijski blok je ciklički OB1 blok ,unutar kojeg se obično pozivaju ostali blokovi i funkcije, te se izvršava u petlji sve dok je PLC u radu ili se ne pojavi zahtjev za prekid njegova izvođenja. Osim glavnog bloka, neki od korisnih blokova uključuju OB100 koji se izvršava isključivo samo jednom, prilikom prelaska CPU-a iz stanja STOP u RUN, blok za prekidanje izvođenja programa (*engl. interrupt*) s vremenskom odgodom OB20, ciklički "*interrupt*" OB30, hardverski "*interrupt*" OB40 itd.

Tijekom obrade korisničkog programa, vrijednosti memorijskih i izlaznih adresa se mijenjaju. No samo posljednja vrijednost unutar ciklusa će biti ona koja će se koristiti u obradi tijekom idućeg ciklusa. Također, kako se vrijednosti izlaza mijenjaju tijekom izvođenja programa, one se prvo zapisuju u "sliku izlaza", a zatim posljednja vrijednost nakon završetka ciklusa prenosi na pojedini izlaz. Time se sprječava "treperenje" vrijednosti fizičkih izlaza. Ciklus se zatim ponavlja istim tokom sve dok je PLC u stanju RUN.

Osim tih koraka, CPU obrađuje i internu dijagnostiku i sve komunikacijske zadatke. Vrijeme potrebno za obradu jednog ciklusa ovisi o samom CPU, korisničkom programu, broju ulaza i izlaza te broju potrebnih komunikacijskih zadataka. No vrijeme izvođenja ciklusa je obično reda veličine nekoliko milisekundi, što znači da PLC ima vrlo brzi odziv.

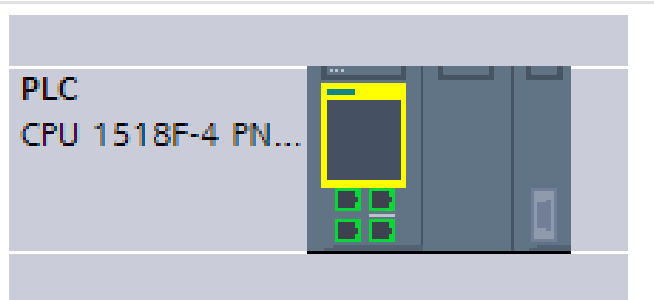
### 3.3. SIMATIC S7-1500

PLC korišten u ovom radu dolazi s CPU-om 1518F -4 PN/DP i pripada seriji SIEMENS SIMATIC S7-1500 PLC-ova, koja predstavlja seriju najkvalitetnijih SIEMENS-ovih PLC-a. Sam PLC je prikazan na slici 3.1 te slici 3.5 u *Topology View*-u unutar TIA Portala.

PLC sadrži *Fail-safe* CPU, odnosno kako je prethodno spomenuto, sigurnosne funkcije upravljanja su integrirane. Kod takvih CPU-ova se dodaje oznaka "F" koja označava *Fail-safe*. Također, to se može prepoznati i po žutoj oznaci na prikazu PLC-a. PLC posjeduje zaslon koji je koristan za dijagnostiku i prilikom puštanja u pogon. Posjeduje 4 komunikacijska sučelja, što se također može prepoznati po oznaci broja 4 uz naziv CPU-a. To su 2 PROFINET IRT porta (2 porta s odvojenom IP adresom se mogu koristiti za razdvajanje mreže) , PROFINET RT, PROFINET Gigabit (omogućuje prijenos podataka i do 1 Gbps) te PROFIBUS DP. DP je kratica za decentraliziranu periferiju (engl. *Decentralized Periphery*) koja koristi decentraliziranu serijsku komunikaciju ulaznih/izlaznih jedinica velike brzine sa PLC-om.

CPU posjeduje radnu memoriju od 6MB za korisnički program te 60MB za podatke. *Load* memoriju se omogućuje SIMATIC memorijskom karticom.

Ovaj PLC je korišten iz razloga što je bio dostupan tijekom izrade ovog rada unutar prostorija tvrtke Danieli Systec.

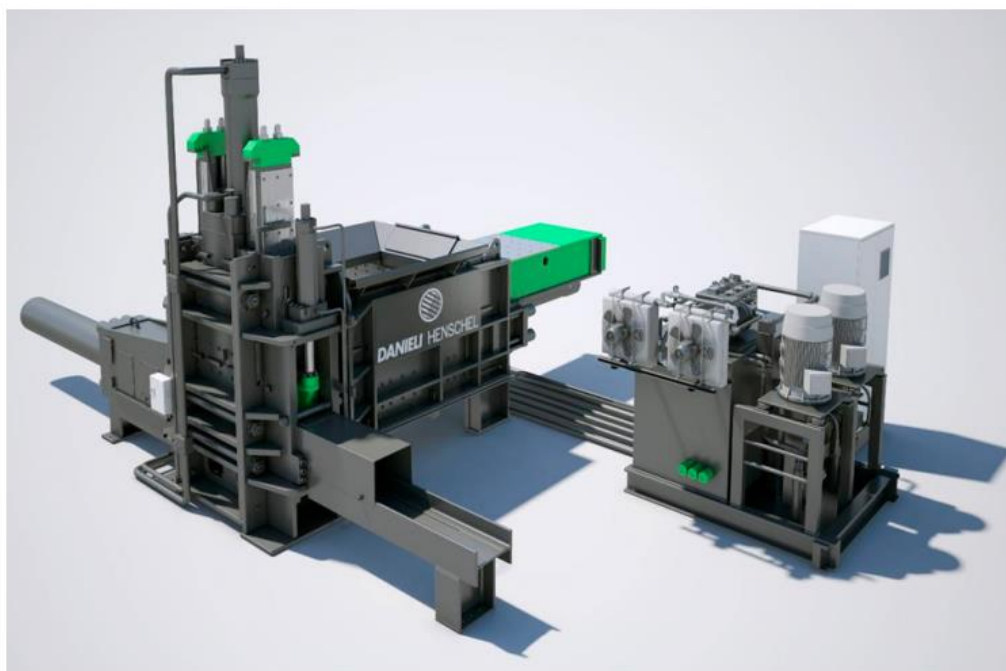


Slika 3.5. Prikaz PLC-a CPU 1518F -4 PN/DP

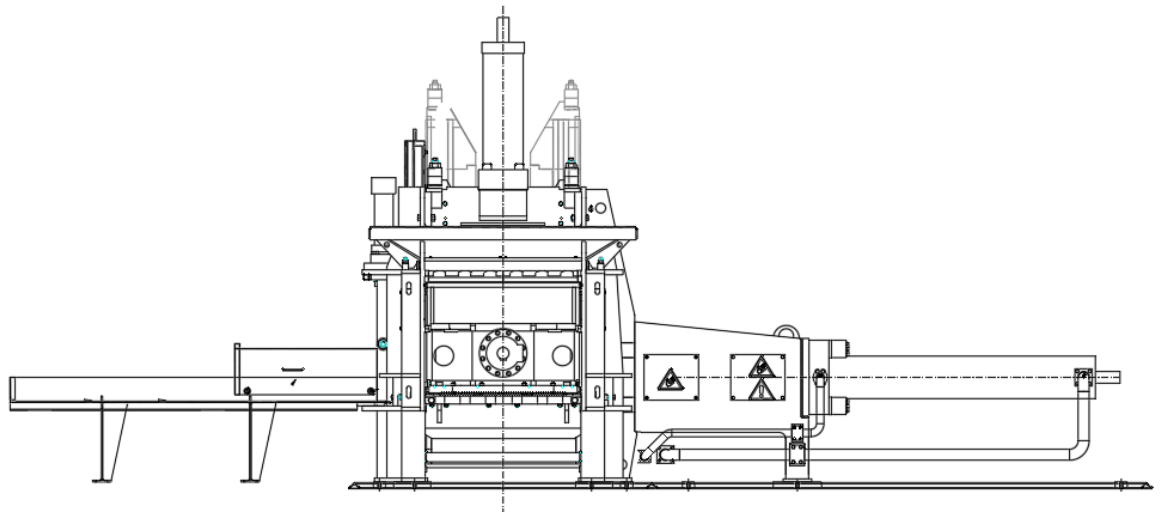
## 4. Hidraulična preša

Kako je spomenuto u uvodu ovoga rada, hidraulična preša se koristi za prešanje metalnog otpada. U okviru ovoga rada hidraulična preša je izvedbe PTC, koja će biti detaljnije opisana. Osim PTC izvedbe, postoje na primjer izvedbe poput PBC, koja se naziva i prešom za briketiranje, PCV koja prešanje izvodi poklopcem, te PDC koja je slična PTC-u, uz manjak mogućnosti prešanja u jednoj osi.

PTC hidraulična preša izvodi prešanje metalnog otpada u tri osi, kako bi se smanjile dimenzije istog, te bile pogodne za transport. Transport isprešanih bala metalnog otpada se može odvijati unutar istog industrijskog postrojenja, gdje se metalni otpad reciklira, odnosno koristi kao sirovina za izradu novih proizvoda. Najčešća primjena hidrauličnih preša je na odlagalištima otpada, gdje se iz njih transportira dalje na obradu. PTC preša može služiti za unutarnju primjenu, gdje se ista snabdijeva metalnim otpadom putem pokretnih traka preko ulaznog lijevka preše, odnosno kao dio automatizirane linije, ali i za vanjsku primjenu, gdje se obično za postavljanje materijala unutar nje koristi dizalica. Slika 4.2. prikazuje osnovni shematski prikaz, a slika 4.1. prikazuje fizički izgled preše gdje je također prikazana hidraulična jedinica sastavljena od spremnika ulja, pumpi, ventilatora te pripadajućeg elektro ormara.



*Slika 4.1. PTC hidraulična preša s hidrauličnom jedinicom*



*Slika 4.2. PTC hidraulična preša*

PTC preša je prikladna za obradu metalnog otpada poput željeza, aluminija i nehrđajućeg čelika, koji zahtijeva visoku razinu kompresije i visok proizvodni kapacitet. Same bale koje su vidljive na slici 4.3, formiraju se pokretanjem tri hidraulična kompresijska cilindra, dva vodoravna i jednog okomitog. Nakon toga posebni okomiti hidraulični cilindar otvara izlazni otvor za izbacivanje bale izvan preše, odnosno na izlazni žlijeb.




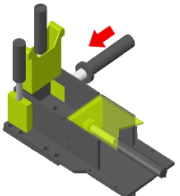
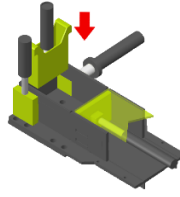
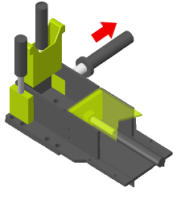
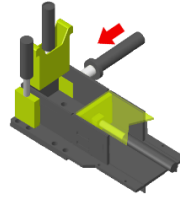
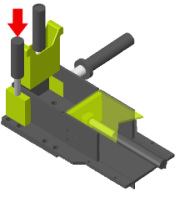
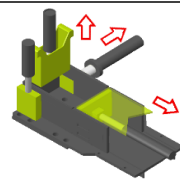
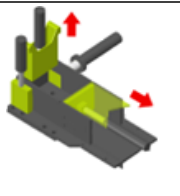
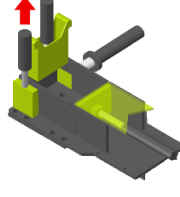
*Slika 4.3. Bala sprešanog metalnog otpada*

Izrađeni upravljački algoritam hidraulične preše podijeljen je na nekoliko logičkih cjelina:

- Ručno upravljanje - pomoću komandi za pomicanje svakog od hidrauličnih cilindara, omogućeno preko HMI-a ili preko radio upravljačke jedinice
- Poluautomatsko upravljanje – ručni početak ciklusa nakon čega slijedi prebacivanje u automatsko upravljanje
- Automatsko upravljanje – kontinuirani ciklus prema određenom slijedu

Automatska sekvenca se sastoji od niza pokreta hidrauličnih cilindara za optimalno prešanje umetnutog metalnog otpada koji su prikazani tablicom 4.1. Unutar tablice su pozicije svih cilindara pretpostavljane njihove početne, ukoliko to nije slučaj vrši se repositioniranje prije automatske sekvence, osim u slučaju prebacivanja sa ručnih komandi.

Tablica 4.1. Slijed automatske sekvence

 <p>1. Pomak cilindra C1 naprijed</p>	 <p>6. Izbacivanje bale cilindrom C3.</p>
 <p>2. Spuštanje cilindra C2</p>	 <p>7. Povratak C3 u početnu poziciju kako bi se omogućilo zatvaranje vrata</p>
 <p>3. Pomak cilindra C3 naprijed.</p>	 <p>8. Zatvaranje vrata.</p>
 <p>4. Mali pomak cilindara kako bi se vrata oslobodila pritiska</p>	 <p>9. Povratak C2 zatim C1 u njihove početne pozicije</p>
 <p>5. Otvaranje vrata</p>	<p>(Kraj sekvence)</p>

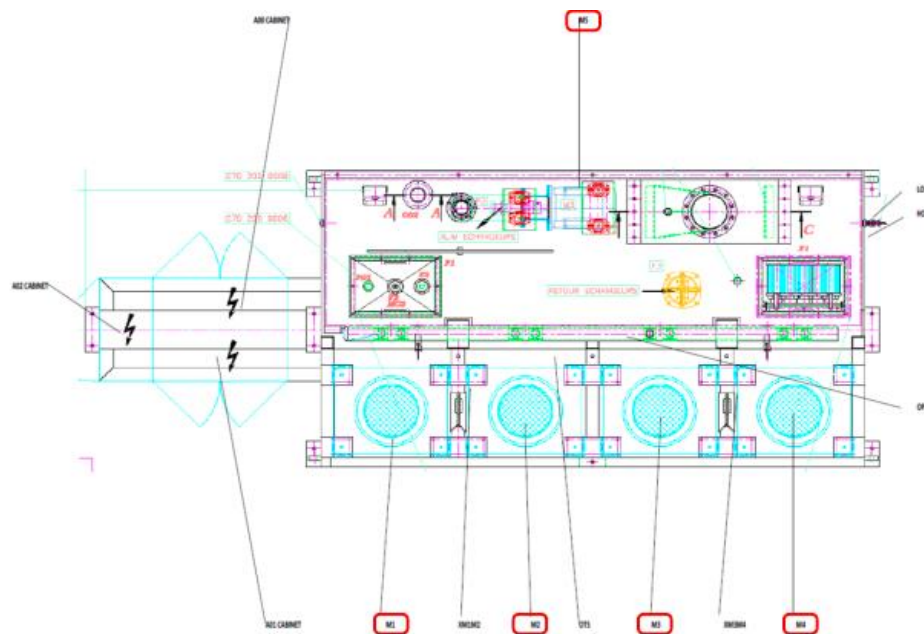


## 4.1. Hidraulična jedinica

Hidraulična jedinica preše omogućuje njezin rad na način na nizom pumpi osigurava dobavu ulja cilindrima koji obavljaju rad odnosno pomak. Sama jedinica je realizirana od 4 pumpe, podijeljene unutar 2 grupe, 2 pumpe unutar svake grupe. Svaka grupa je povezana sa 2 cilindra, cilindri C2 i C3 su unutar jedne, dok cilindri C1 i C4 unutar druge grupe. Na taj način se ostvaruje veća sila preša, dva cilindra unutar iste grupe se ne mogu istovremeno pomicati.

Hidraulična pogonska jedinica se sastoji od :

- Glavnog spremnika ulja s pokazateljem razine ulja
- 4 elektromotora
- 4 visokotlačnih pumpi podešenih na visoki nazivni tlak za maksimalno radno opterećenje
- Grijača i senzora temperature ulja



Slika 4.4. Hidraulična jedinica

## 4.2. Tehnička specifikacija

Unutar tablice 4.2. su vidljivi neki od tehničkih podataka za hidrauličnu prešu.

Tablica 4.2. Tehnička specifikacija hidraulične preše

<i>PTC</i>	<i>SI sustav jedinica</i>
<i>Sila cilindara</i>	
<i>Prva sila kompresije C1</i>	<i>100t</i>
<i>Druga sila kompresije C2</i>	<i>157t</i>
<i>Treća sila kompresije C3</i>	<i>157t</i>
<i>Sila cilindra vrata</i>	<i>20t</i>
<i>Pritisak na balu</i>	<i>≤174 daN/cm<sup>2</sup></i>
<i>Krajnje pozicije cilindara cilindra</i>	
<i>C1</i>	<i>4740 mm</i>
<i>C2</i>	<i>678 mm</i>
<i>C3</i>	<i>2500 mm</i>
<i>Vrata</i>	<i>750 mm</i>
<i>Kompresijska komora</i>	
<i>Dužina</i>	<i>2550 mm</i>
<i>Širina</i>	<i>1250 mm</i>
<i>Visina</i>	<i>900 mm</i>
<i>Volumen</i>	<i>2869 dm<sup>3</sup></i>
<i>Izlazna bala</i>	
<i>Duljina</i>	<i>Varijabilna</i>
<i>Širina</i>	<i>300 mm</i>
<i>Visina</i>	<i>300 mm</i>
<i>Volumen</i>	<i>Varijabilan</i>
<i>Dimenzije preše bez hidraulične jedinice</i>	
<i>Dužina</i>	<i>7250 mm</i>
<i>Širina</i>	<i>6100 mm</i>
<i>Visina</i>	<i>3800 mm</i>
<i>Masa</i>	
<i>Masa preše</i>	<i>39 t</i>

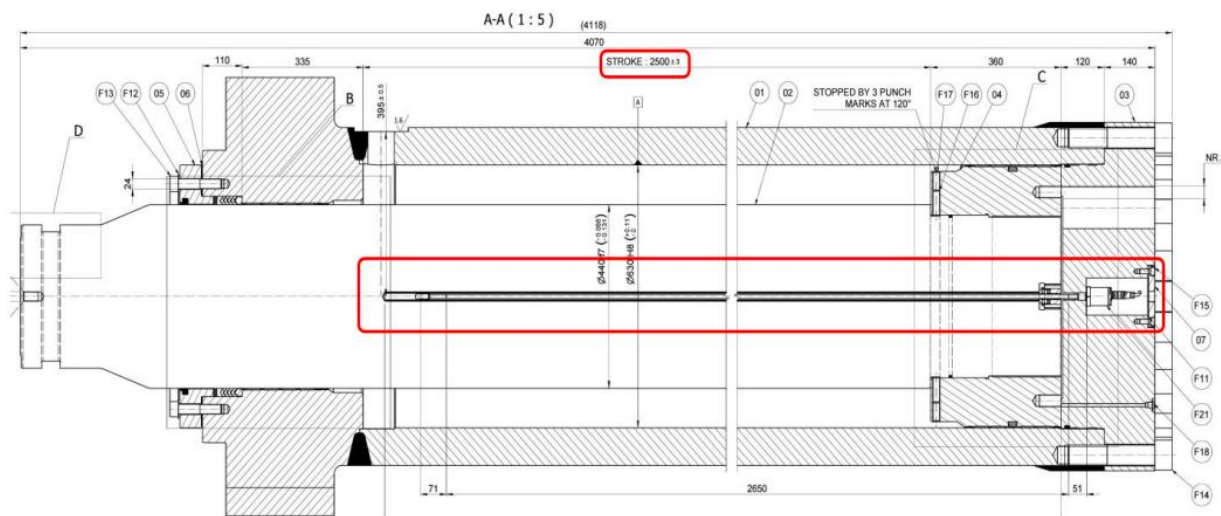
### 4.3.Senzori

Senzori korišteni za hidrauličnu jedinicu uključuju:

- Detektor razine ulja (visoka i niska razina)
- Detektor temperature ulja
- Senzor začepljenja filtera ulja
- Senzor radnog tlaka ulja
- Senzor temperature stroja

Za precizno određivanje pozicije pokretnih dijelova koriste se magnetnostriktivni senzori linearnog položaja (*engl. temposonics*), koji su ugrađeni u svaki cilindar. *Temposonics* senzori se sastoje od magneta koji je smješten unutar cilindra u pokretu te stvara magnetsko polje na svom mjestu unutar valovoda. Dok se u kućištu senzora, koje je smješteno u nultu poziciju cilindra, se generira magnetsko polje pomoću impulsnih struja, koje se širi duž senzora odnosno valovoda. Kada se ta dva magnetska polja susretnu, stvara se torzijsko naprezanje na valovodu, što oslobađa torzijski impuls (ultrazvučni val) koji se širi valovodom. Kada ultrazvučni val dosegne kućište senzora, izmjeri se vrijeme potrebno za refleksiju vala od slanja impulsa struje. Brzina ultrazvučnog vala u valovodu je poznata, što omogućuje precizno određivanje linearne pozicije magneta, odnosno cilindra preše. [11]

Slika 4.5. je prikazuje *temposonic* senzor ugrađen unutar cilindra C3.

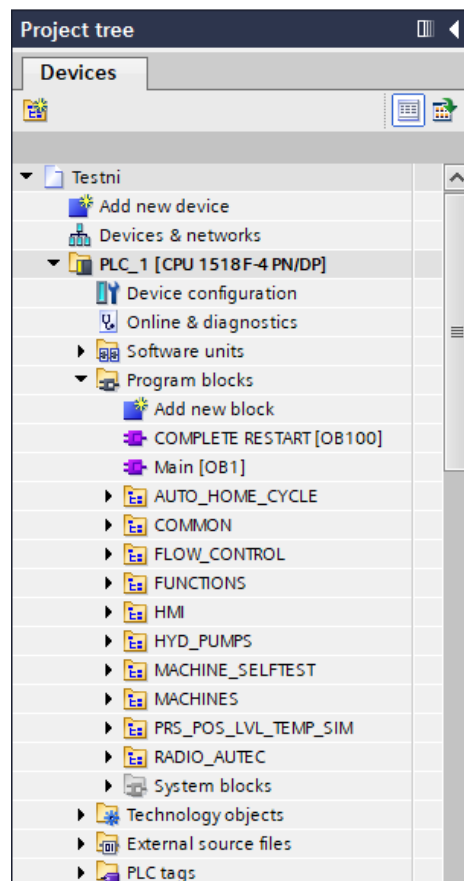


Slika 4.5. *Temposonic* senzor unutar cilindra C3

## 5. Programiranje softvera za upravljanje i nadzor hidraulične preše

Programiranje softvera za upravljanje i nadzor hidraulične obavljeno je unutar TIA Portal-a. Kako je spomenuto, u ovom radu korišten je PLC iz serije SIEMENS S7-1500, točnije CPU 1518F – 4 PN/DP.

Slika 5.1. prikazuje „*Project tree*“ unutar TIA Portal-a, gdje su organizirane mape za programiranje PLC-a. Unutar mape „programskih blokova“ (*engl. program blocks*) se nalazi niz podmapa koje čine pojedine dijelove upravljačkog algoritma i nadzora hidraulične preše. Unutar svake od mapa se nalaze funkcije i funkcijski blokovi s pripadajućim podatkovnim blokovima. Funkcije i blokovi su organizirani unutar mapa kako bi logika bio preglednija. Sve funkcije se pozivaju unutar glavnog organizacijskog bloka OB1, a neke od njih biti će opisane u nastavku ovog poglavlja.



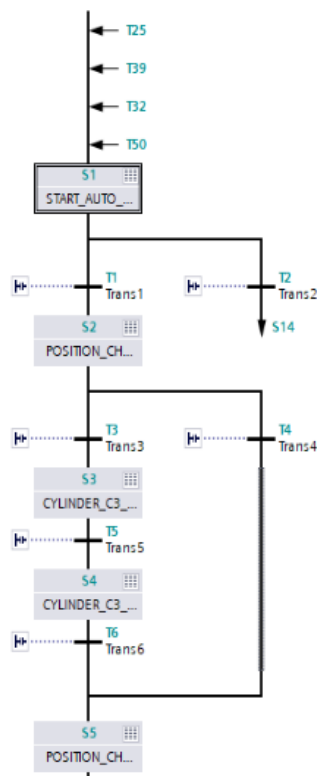
Slika 5.1. *Project tree* unutar TIA Portal-a

## 5.1. Automatska sekvenca

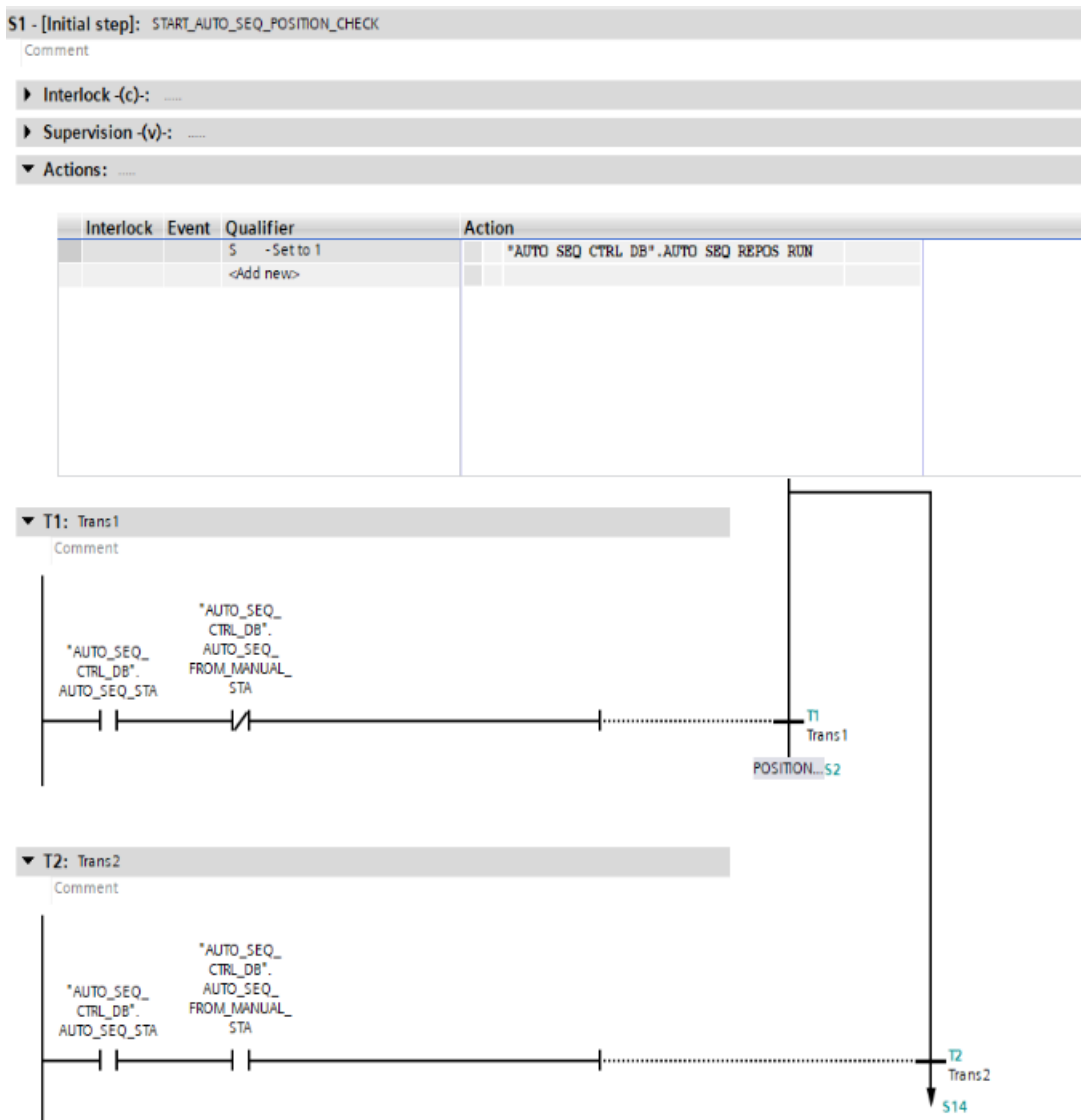
Automatska sekvenca predstavlja niz pokreta cilindara preše koji su predefimirani sa ciljem optimalnog prešanja metalnog otpada umetnutog unutar nje. Sekvenca slijedi korake opisane u tablici 4.1.

Prvi korak u sekvenci je pokretanje same sekvence, što se može postići na dva načina, a oba uključuju interakciju sa operaterom. Prvi način pokretanja sekvence je od njenog početnog stanja otvorenosti, a ostvaruje se putem HMI-a, gdje se setira varijabla „AUTO\_SEQ\_STA“. Nakon toga, svi koraci u ostatku sekvence se izvršavaju (ukoliko je prešanje bilo uspješno, koraci vezani za neuspješno prešanje se ne izvršavaju). Ovo pokretanje sekvence rezultira repozicioniranjem svih cilindara preše u njihove početne (nulte) pozicije. Ovaj način rada preše se odabire ukoliko operater nije ručno pokušao isprešati metalni otpad prije pokretanje sekvence. Međutim ukoliko je, to predstavlja drugi način pokretanja sekvence gdje se preskače upravo to početno repozicioniranje cilindara. No postoje još određeni uvjeti za prelazak iz ručnog u automatski način rada preše, što će kasnije biti objašnjeno i prikazano.

Prvi korak predstavlja početno pozicioniranje unutar sekvence prema operaterovim željama, te je prikazan na slici 5.3., dok je unutar same sekvence sa idućim koracima prikazan na slici 5.2.



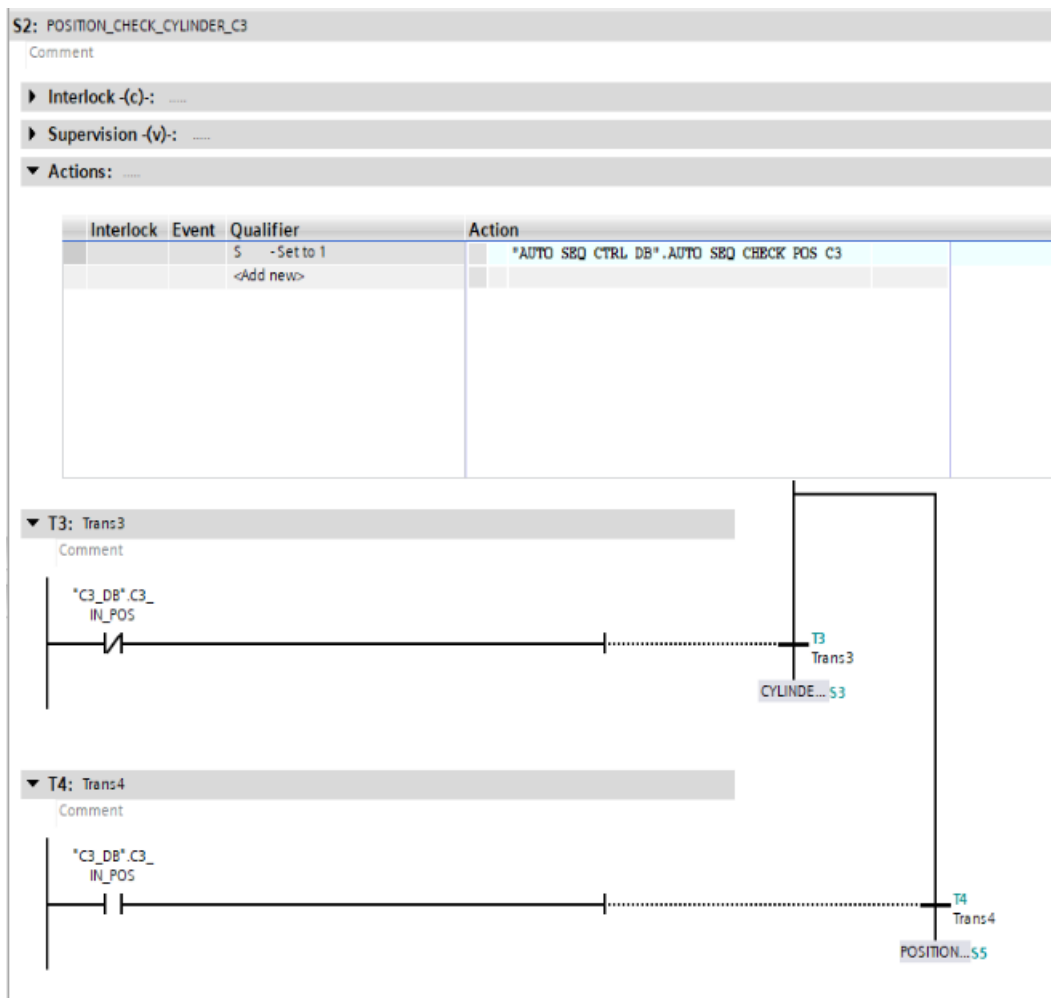
Slika 5.2. Automatska sekvenca prvi dio



Slika 5.3. Prvi korak automatske sekvence

Programiranje same sekvence izvedeno je pomoću GRAPH-a odnosno grafičkog načina programiranja PLC-a koji pruža TIA Portal v16. Unutar GRAPH-a dodaju se koraci (*engl. steps*) te definiraju prijelazi (*engl. transitions*) između njih. Svaki korak ima određenu akciju koja se definira unutar mape „Actions“, dok prijelaz u druge korake se definira na temelju određenih uvjeta koji moraju biti *bool* vrijednosti.

U prvom koraku postoje dva prijelaza na sljedeće korake. Ukoliko nisu korištene ručne komande prije pokretanje sekvence, tada se prelazi na korak 2 putem prijelaza 1. Unutar koraka 2 prikazanog na slici 5.4, provjerava se pozicija cilindra C3. Ako cilindar nije u svojoj početnoj (nultoj) poziciji, prelazi se na korak 3 gdje se izvršava pozicioniranje u tu poziciju. Međutim, ako je cilindar već u toj poziciji, tada se taj dio sekvence preskače prijelazom 4 i prelazi na korak 5 gdje se izvršava ista provjera pozicije za cilindar vrata.



Slika 5.4. Drugi korak automatske sekvence

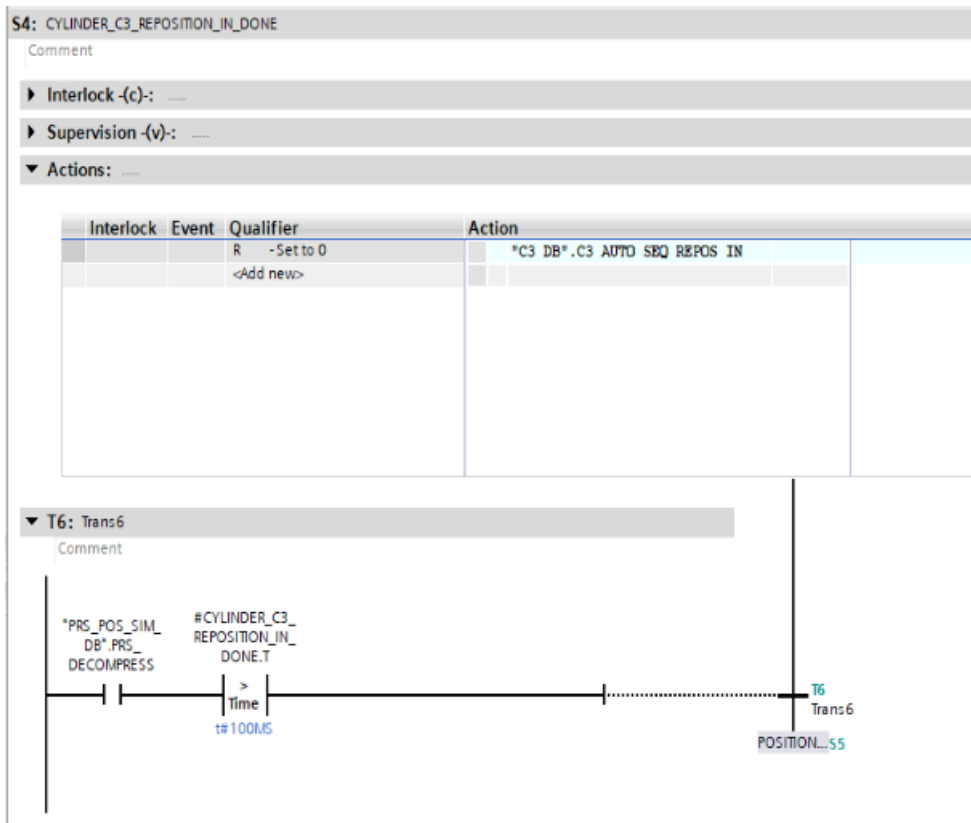
Ako je repozicioniranje cilindra C3 potrebno, izvodi se u koraku 3, prikazanim na slici 5.5. Setira se varijabla „C3\_AUTO\_SEQ\_REPOS\_IN“, koja služi kao jedan od zahtjeva (*engl. request*) za pomicanje cilindra C3 unatrag (*backward movement*). Cilindar se pomiče prema natrag dok se ne postigne nulta pozicija („C3\_IN\_POS“), nakon čega se prelazi na idući korak 4 prikazan na slici 5.6.

U koraku 4 čeka se izvršavanje dekompresije nakon povratnog pomaka cilindra C3, uz vremensko zatezanje od 0.1 sekunde se prelazi u idući korak 5. Unutar automatske sekvence, nakon svakog pomaka bilo kojeg od cilindara, čeka se dekompresija pritiska sustava ispod određene razine kako bi se mogao izvršiti sljedeći pomak nekog od cilindara.

U koraku 5 provjerava se pozicija vrata, a daljnji dio sekvence je sličan prethodno objašnjenom dijelu gdje prvo izvodi repozicioniranje cilindra vrata, a zatim cilindra C2 te cilindra C1, čime se izvršava početno repozicioniranje. Taj dio sekvence prikazan je na slici 5.7.

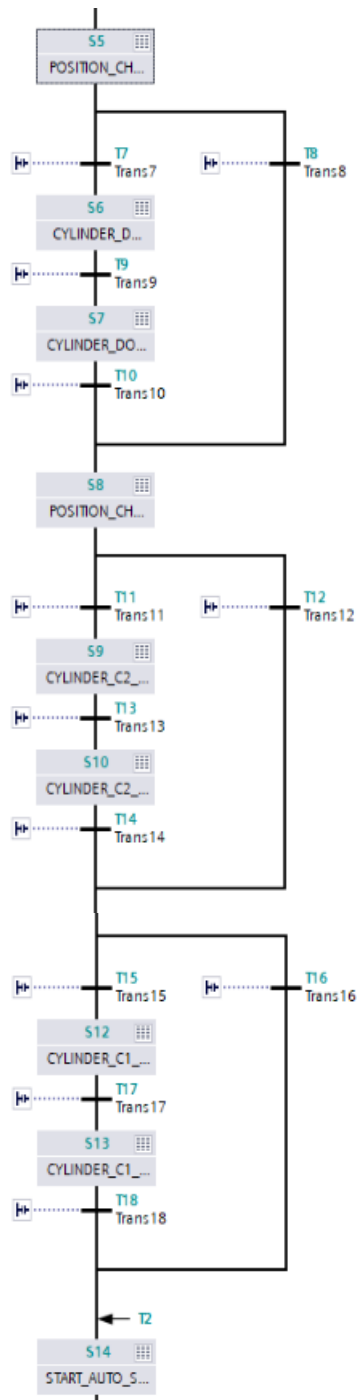


Slika 5.5. Treći korak automatske sekvence – repositioniranje cilindra C3



Slika 5.6. Četvrti korak automatske sekvence – završeno repositioniranje cilindra C3



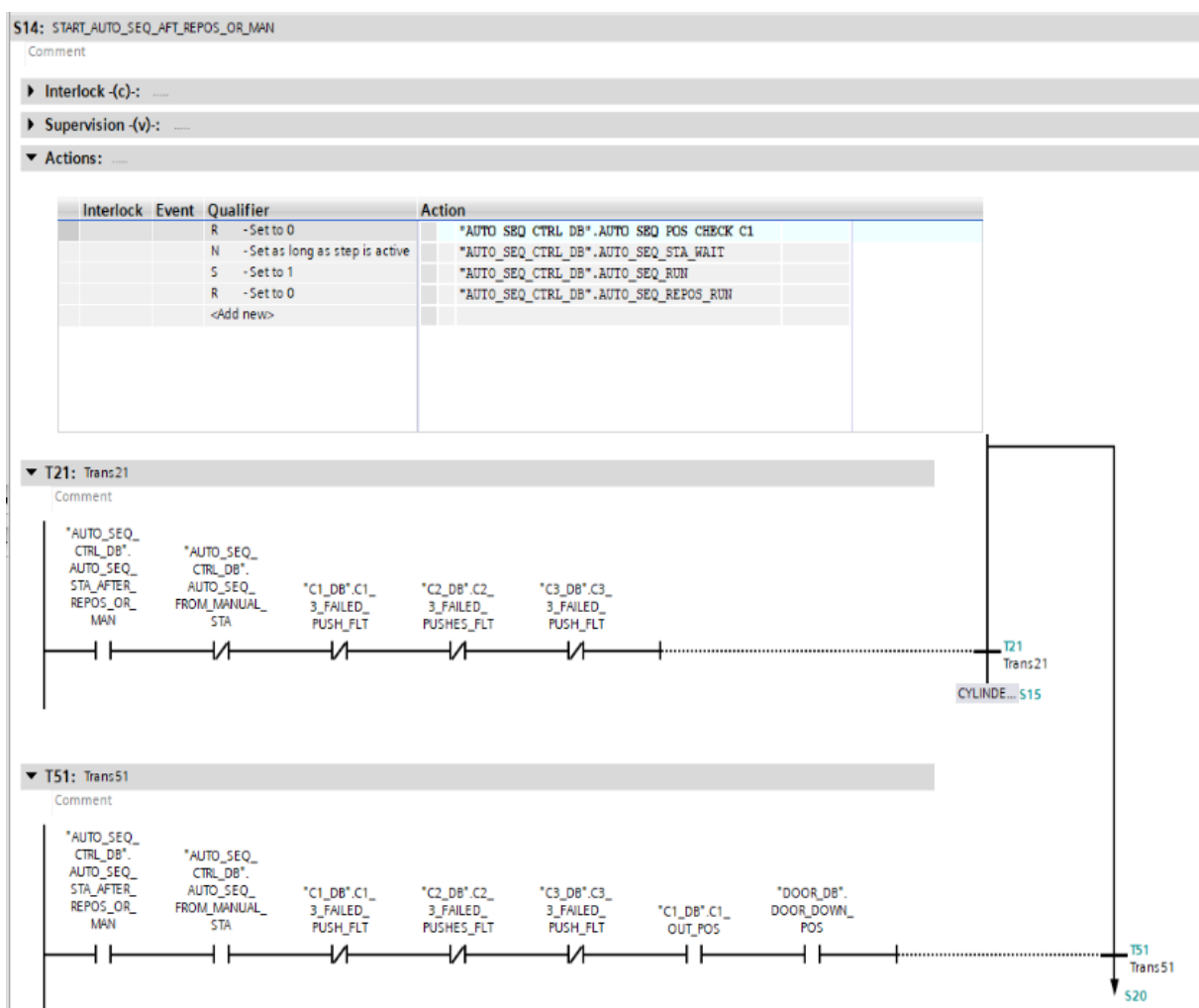


Slika 5.7. Automatska sekvenca drugi dio

Nakon početnog repozicioniranja ili ako je sekvenca pokrenuta nakon ručnog upravljanja, sekvenca dolazi na korak 14 prikazan na slici 5.8. Unutar ovog koraka postoje dva prijelaza. Prijelaz 21 se ostvaruje nakon repozicioniranja, te ako je prethodna sekvenca bila uspješna, odnosno ako niti jedna od varijabla „3\_FAILED\_PUSH\_FLT“ nije aktivna. Ove varijable postaju aktivne nakon tri neuspješna pokušaja prešanja pojedinog cilindra. Pokretanje ostatka sekvence nakon koraka 14 također se ostvaruje pomoću HMI-a gdje se setira varijabla, „AUTO\_SEQ\_STA\_AFTER\_REPOS\_OR\_MAN“.

Ovaj pristup je implementiran kako bi se omogućilo zaustavljanje sekvence nakon „otvaranja“ preše, kako bi se omogućilo postavljanje metalnog otpada unutar nje. Nakon što operater postavi materijal, automatski ciklus se nastavlja, tj. sekvenca se pomiče u korak 15(ovaj dio je dodan iz razloga što obično preša posjeduje senzor težine kojim se nakon postizanje određene težine omogućava nastavak rada automatske sekvence preše).

Ako je operater ručno prešao materijal, dakle početno repositioniranje nije bilo potrebno, pokretanje ostatka sekvence (od koraka 20, jer nije potrebno prešanje od cilindra C1) ostvaruje se uz prethodno objašnjene uvjete, uz dodatno da cilindar C1 mora biti u konačnoj (*out*) poziciji, cilindar vrata mora biti u donjoj poziciji. Ovi uvjeti su implementirani iz sigurnosnih razloga kako bi se spriječilo oštećenje preše u slučaju nepravilnog prelaska iz ručnog u automatski način rada.



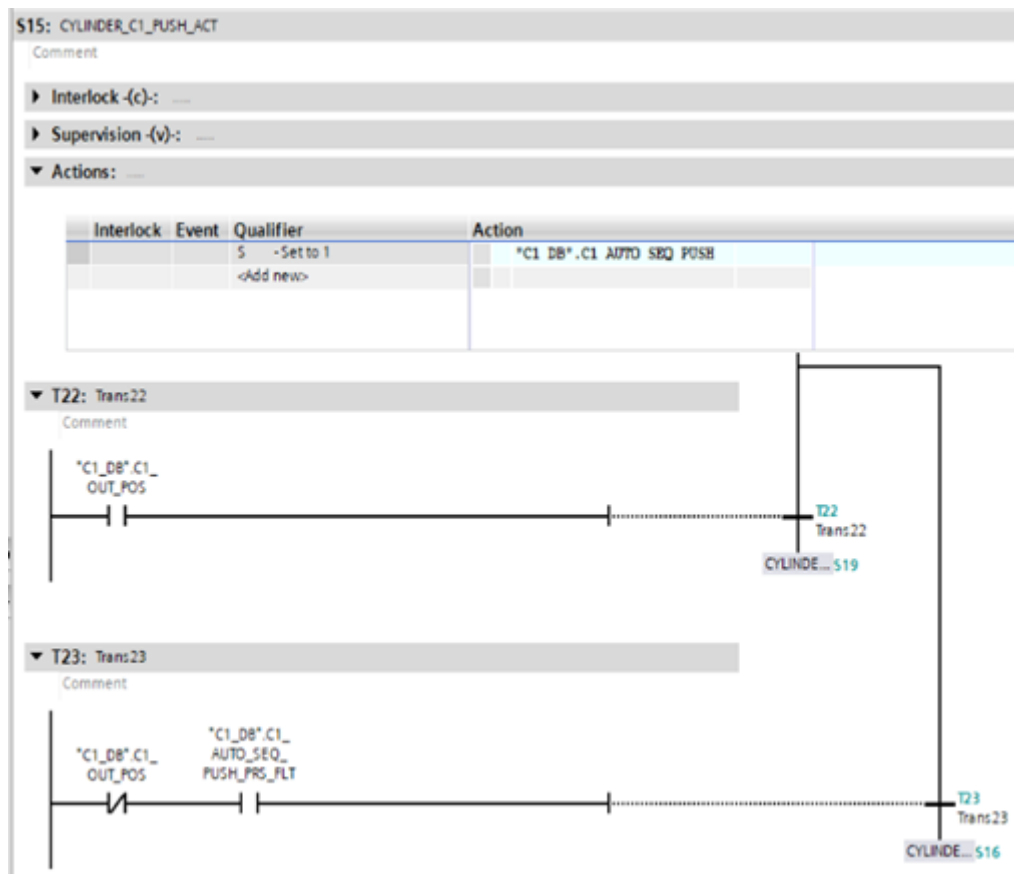
Slika 5.8. Četrnaesti korak automatske sekvence – start ostatka sekvence nakon repositioniranja ili ručnog rada

Sljedeći dio sekvence nakon gore opisanog djela je prikazan na slici 5.9.



Slika 5.9. Automatska sekvenca treći dio

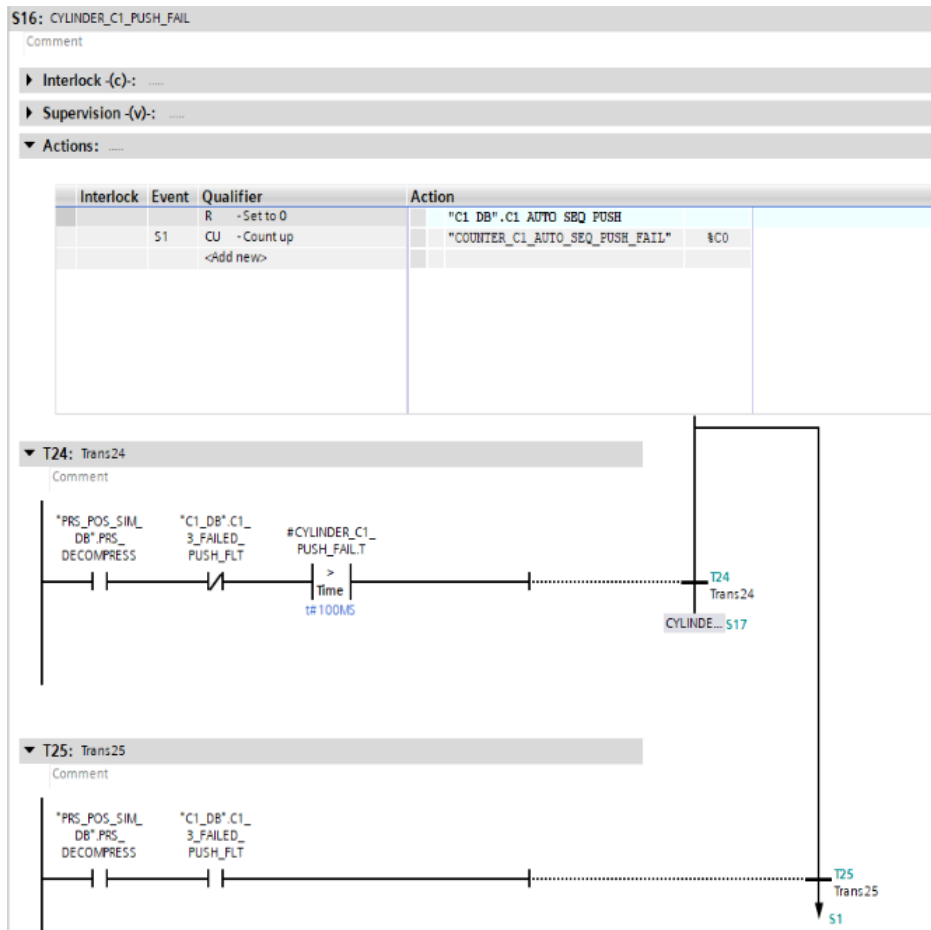
Nakon što se dio sekvence pokrene u koraku 14, cilindar C1 pomiče se prema naprijed (*forward movement*) kako bi se materijal isprešao u toj osi. Proces prešanja traje do krajnje pozicije (*out*), gdje se postiže željeni stupanj prešanja cilindra C1. Ova radnja se izvodi korakom 15 prikazanim na slici 5.10.



Slika 5.10. Petnaesti korak automatske sekvence – prešanje cilindrom C1

Ako je prvo prešanje uspješno izvršeno, odnosno postignuta je krajnja pozicija cilindra C1, sekvenca prijelazom 22 prelazi u korak 19, unutar kojeg se čeka dekompresija tlaka ulja sustava. Međutim, ako se tlak ulja sustava povećao iznad određene granice, (što se simulira ručnim postavljanjem vrijednosti tlaka na visoku razinu) a krajnja pozicija nije postignuta, to ukazuje na neuspješno prešanje. U tom slučaju, sekvenca se odvodi prijelazom 23 na „odvojenu granu“ u korak 16. Neuspješno prešanje se događa u slučaju kada je previše metalnog otpada postavljeno za prešanje ili je materijal previše tvrd. Tada se „odvojenom granom“ (korakom 17), cilindar C1 se vraća u početnu poziciju, a zatim se pokušava ponovno izvršiti prešanje. Korak 16 prikazan na slici 5.11. se ostvaruje nakon neuspješnog prešanja cilindra C1, inkrementira se brojač C0 (koji broji neuspješne pokušaje prešanja cilindra C1). Ako taj broj dosegne vrijednost 3, aktivira se varijabla „C1\_3\_FAILED\_PUSH\_FLT“. To rezultira pojavom alarma na HMI-u te prijelazom 25 na odbacivanje ostatka sekvence (*reject*), tj. sekvenca se vraća na početno repositioniranje kako bi se preša „otvorila“ te omogućila operateru uvid u stanje unutar nje.

Međutim, ako broj neuspješnih prešanja nije dostignuo vrijednost 3, sljedeći korak sekvence je korak 17 vidljiv na slici 5.12, gdje se *setira* zahtjev za pomak cilindra u početnu poziciju, a zatim nakon dekompresije, sekvenca se vraća na korak 15, odnosno na novi pokušaj prešanja.



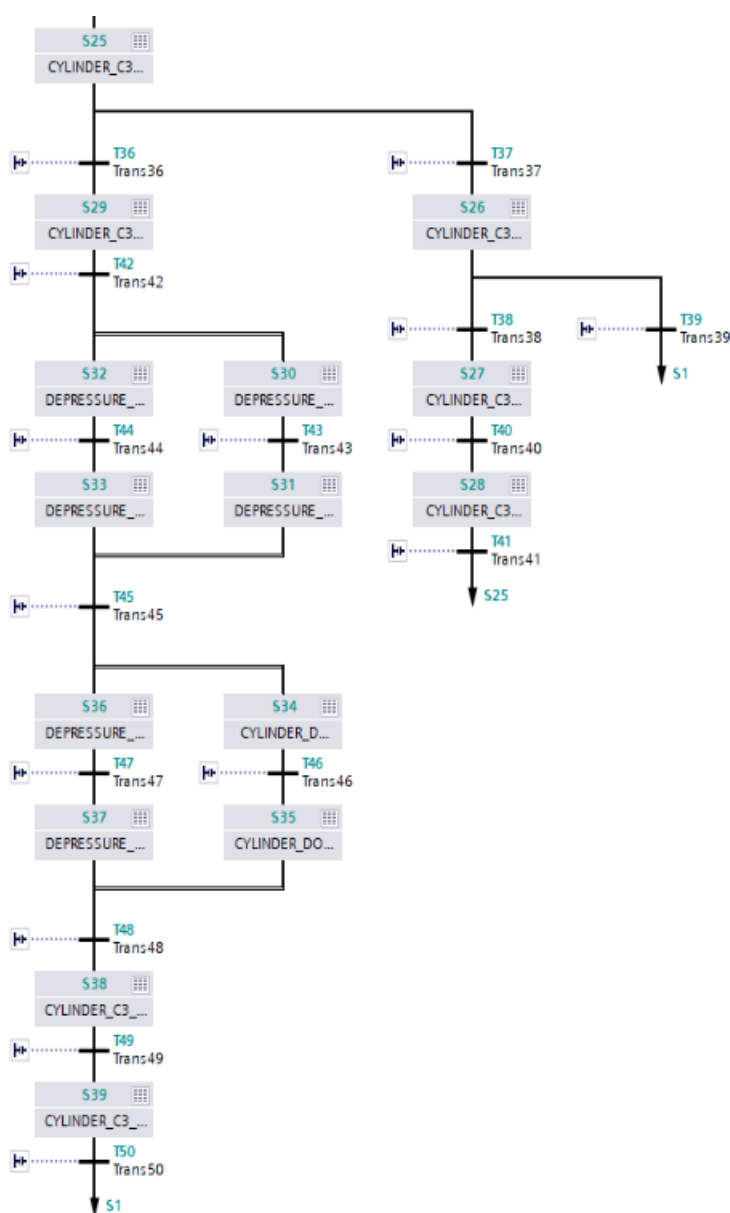
Slika 5.11. Šesnaesti korak automatske sekvence – neuspješno prešanje cilindra C1



Slika 5.12. Sedamnaesti korak automatske sekvence – povratak cilindra C1 nakon neuspješnog prešanja

Nakon što je prešanje cilindrom C1 uspješno izvršeno, slijedi prešanje drugog cilindra C2, u drugoj osi u koraku 20. Također kao i kod cilindra C1, u korak 24 se prelazi nakon uspješnog prešanja, dok ako ono nije uspješno izvršeno, sekvenca prelazi se u korake 21, 22 te 23. Tijekom ovih koraka, cilindar se vraća u početnu poziciju kako bi omogućio ponovni pokušaj prešanja. Također 3 neuspješna prešanja rezultira *reject*-om sekvence, odnosno prijelazom 32 se sekvenca vraća na početno repozicioniranje.

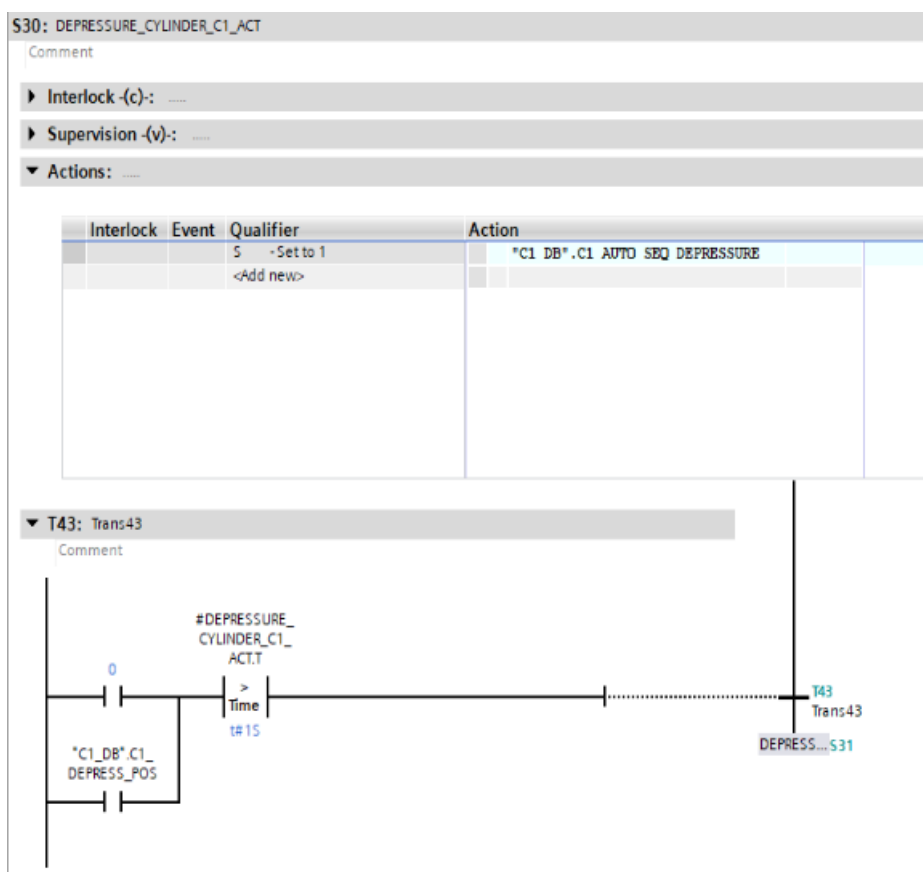
Poslije cilindra C2, cilindar C3 izvodi prešanje u preostaloj trećoj osi, a logika sekvence ostaje ista kao kod prethodna dva cilindra. Jedina razlika je da se prešanje smatra završenim kada se postigne određena vrijednost tlaka ulja, a taj dio sekvence je prikazan slikom 5.13.



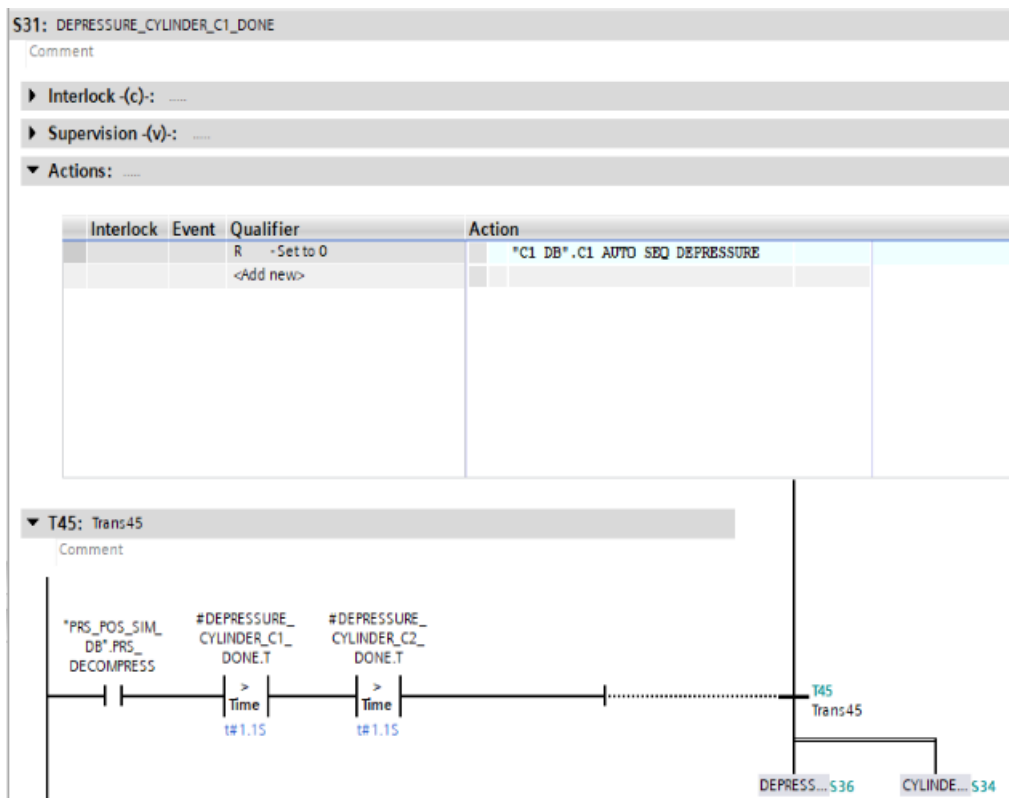
Slika 5.13. Automatska sekvenca četvrti dio

Nakon uspješnog prešanja u sve tri osi, sekvenca prelazi u korake 30 i 32, unutar kojih se provodi *depressure* cilindara C1 i C2. To znači vraćanje cilindara unatrag za mali pomak, čime se bala metala odnosno vrata oslobađaju pritiska ta dva cilindra. *Depressure* cilindra C1 korakom 30 prikazanim na slici 5.14 setira se zahtjev za pomicanje cilindra unatrag do pozicije „C1\_DEPRESS\_POS“, koja je za 3% manja od krajnje pozicije. Također, postoji opcija da se taj pomak ostvaruje tijekom jedne sekunde.

Sličan postupak *depressure*-a se izvodi i za cilindar C2 paralelno s cilindrom C1, budući da pripadaju različitim hidrauličnim grupama. Unutar koraka 31 (slika 5.15) te 33 vrši se čekanje dekompresije nakon tog kratkog pomaka. Nakon što je *depressure* za oba cilindra ostvaren, s vremenskim zatezanjem od 1.1 sekunde se prelazi u sljedeći dio sekvence.

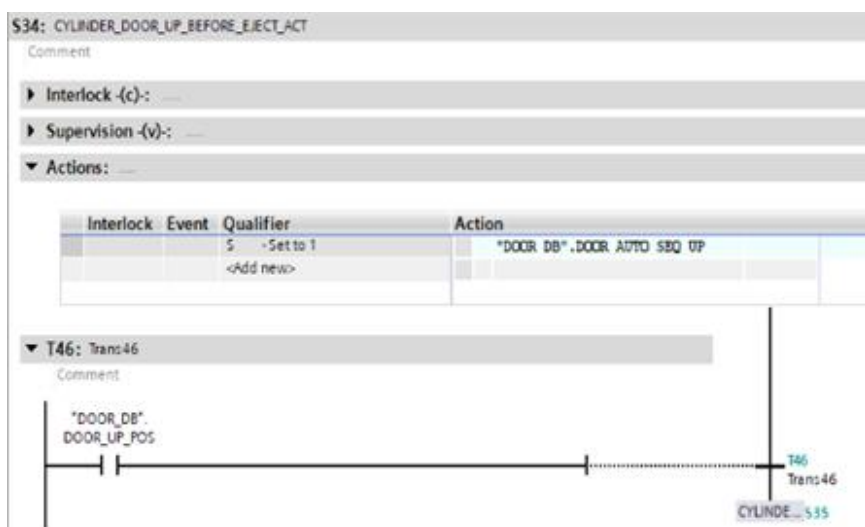


Slika 5.14. Trideseti korak automatske sekvence – *depressure* cilindra C1



Slika 5.15. Tridesetprvi korak automatske sekvence – izvršen depressure C1

Završetkom *depressure*-a cilindara C1 i C2 slijede koraci 34 i 36, koji ostvaruju pomak vrata prema gore, odnosno otvaranje izlaznog otvora, dok istovremeno se provodi *depressure* cilindra C3. Korak 34 prikazan na slici 5.16. uključuje setiranje zahtjeva za pomak vrata prema gore. Korak 35 je sličan koraku 31 uz vremensko zatezanje od 0.1 sekunde.

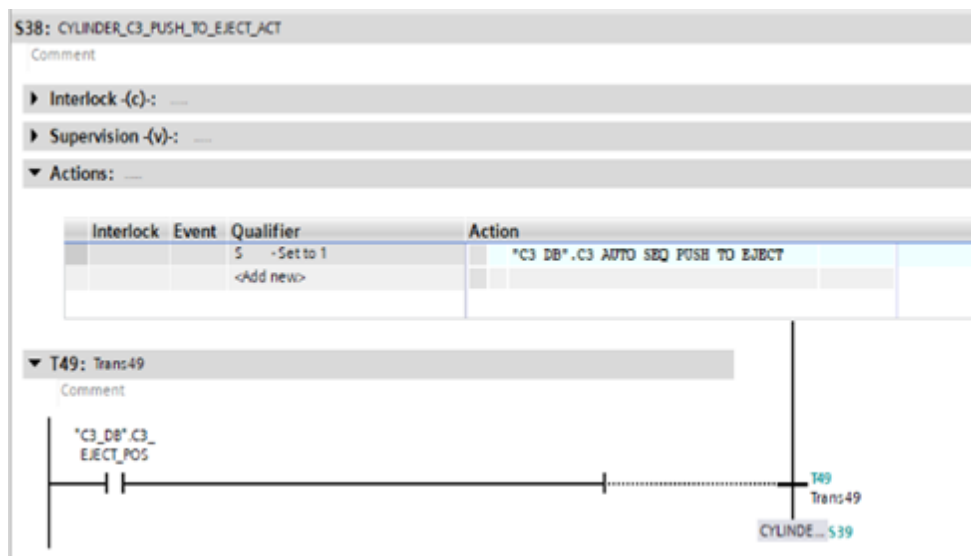


Slika 5.16. Tridesetčetvrti korak automatske sekvence – pomak vrata gore



Nakon što su *depressure* cilindara C1, C2 te C3 izvršeni, a vrata su dosegla gornju poziciju te su otvorila izlazni otvor, cilindar C3 je spreman za gurnuti balu metala izvan preše. To je ostvareno korakom 38, prikazanim na slici 5.17, u kojem se setira varijabla „C3\_AUTO\_SEQ\_PUSH\_TO\_EJECT“ kojom se cilindar C3 pomiče do pozicije za izbacivanje (*eject*). Nakon postizanja te pozicije, čeka se dekompresija tog pomaka, a zatim se sekvenca vraća na korak jedan, odnosno na repositioniranje, počevši redom cilindar C3, zatvaranjem vrata, cilindrom C2 te C1. Na taj način sekvenca se vraća na početak, a preša je spremna za prešanje novog metalnog otpada.

U Prilogu 1 je prikazana kompletna automatska sekvenca radi lakšeg razumijevanja iste.

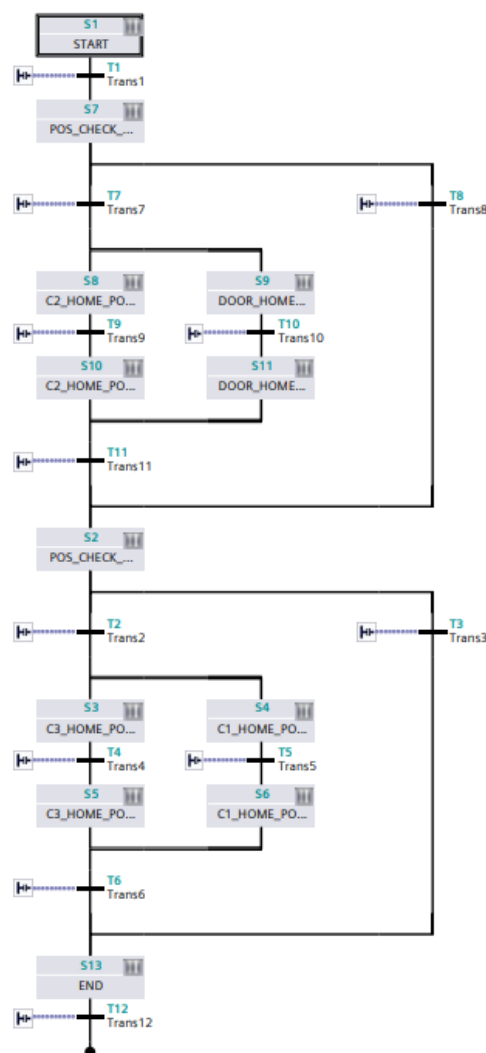


Slika 5.17. Tridesetosmi korak automatske sekvence – pomak C3 to eject pozicije

## 5.2. „Home“ sekvenca

„Home“ sekvenca predstavlja niz pokreta cilindara preše kako bi postigla svoje neutralno, otvoreno odnosno „home“ stanje. Sama sekvenca je slična procesu repositioniranja unutar automatske sekvence što se tiče cilindara C1, C2 i C3, dok cilindar vrata se pozicionira u gornju poziciju.

Na slici 5.18. prikazana je „home“ sekvenca, koja se poziva putem HMI-a kada se želi preša postaviti u otvoreno stanje, primjerice kod prvog pokretanja u danu, kod održavanja i slično. Cilindri C2 i vrata se pomiču paralelno, isto kao i C3 i C1, budući da pripadaju različitim hidrauličnim grupama. Na kraju sekvence se nalazi korak „END“ kojim se izlazi iz sekvence. Ovaj korak nije prisutan kod automatske, jer je kod nje potrebno kontinuirano izvršavanje iste sekvence.



Slika 5.18. „Home“ sekvenca

### 5.3. Funkcija automatske i „home“ sekvence

Obje sekvence su programirane unutar funkcijskog bloka, budući kada se programira unutar GRAPH-a pripadajući blok podataka se automatski generira. Unutar kojeg se nalaze statusi svih koraka i prijelaza između njih, osim toga i ulazno i izlazne varijable kako je prikazano na slici 5.19. Pomoću ulaznih varijabli je moguće pokretanje i zaustavljanje sekvence, potvrđivanje alarma ukoliko su programirani, dok kod izlaznih je moguće utvrditi da li je sekvenca aktivna, unutar kojeg koraka se sekvenca nalazi i slično.

AUTO_SEQ_DB										
	Name	Data type	Start value	Retain	Accessible f...	Writa...	Visible in ...	Setpoint	Supervis...	Comment
1	Input									
2	OFF_SQ	Bool	false	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Turn sequence off
3	INIT_SQ	Bool	false	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Set sequence to initial state
4	ACK_EF	Bool	false	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Acknowledge all errors and faults
5	S_PREV	Bool	false	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Output previous step in parameter S_NO
6	S_NEXT	Bool	false	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Indicate next step in parameter S_NO
7	SW_AUTO	Bool	false	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Automatic mode
8	SW_TAP	Bool	false	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Semiautomatic/switch with transition
9	SW_TOP	Bool	false	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Semiautomatic/ignore transition
10	SW_MAN	Bool	false	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Manual mode
11	S_SEL	Int	0	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Select step to be output to S_NO
12	S_ON	Bool	false	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Activate step indicated in S_NO
13	S_OFF	Bool	false	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Deactivate step indicated S_NO
14	T_PUSH	Bool	false	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Enable transition to switch in semi automatic r
15	Output									
16	S_NO	Int	0	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Step number
17	S_MORE	Bool	false	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	More steps are available and can be shown in
18	S_ACTIVE	Bool	false	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Step indicated in S_NO is active
19	ERR_FLT	Bool	false	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Interlock or supervision group error
20	AUTO_ON	Bool	false	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Automatic mode is active
21	TAP_ON	Bool	false	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Semiautomatic mode/step with transition enal
22	TOP_ON	Bool	false	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Semiautomatic mode/ignore transition enable
23	MAN_ON	Bool	false	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Manual mode is active

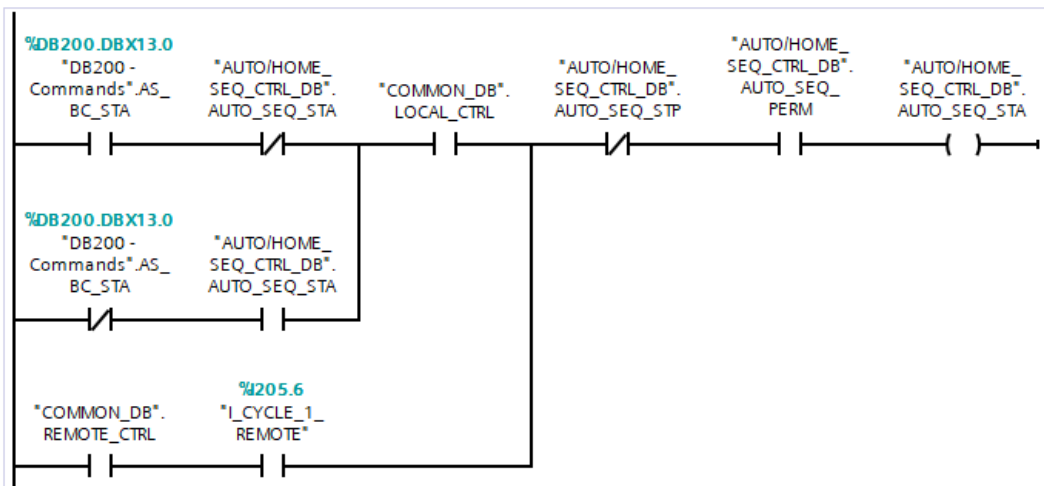
Slika 5.19. Ulazno/izlazne varijable DB-a automatske sekvence

Kako bi se ostvarila sva upravljačka logika vezana za obje sekvence ali izvan samih sekvenci, poput pokretanja i zaustavljanja istih, definiranja alarmnih stanja, izračun vremena trajanja sekvence, broj napravljenih bala, duljina istih te mapiranja varijabli unutar blokova podataka za komunikaciju sa HMI-om, koristi se posebna funkcija s pripadajućim blokom podataka.

Pokretanje automatske sekvence se ostvaruje putem HMI-a ukoliko su odabrane lokalne komande te preko radio jedinice u slučaju odabranog udaljenog upravljanja, kao što je prikazano na slici 5.20. Unutar mreže 6 se setira bit za pokretanje sekvence, dok se unutar mreže 17 setira bit za pokretanje sekvence nakon upotrebe ručnih komandi. Ako su korištene ručne komande za prešanje metalnog otpada, i ako se pokuša pokrenuti automatska sekvenca 60 sekundi nakon toga, to je moguće uz uvjete opisane u poglavlju automatske sekvence. Ukoliko se pokuša pokrenuti automatska sekvenca nakon tih 60 sekundi, izvršava se repozicioniranje svih cilindara.

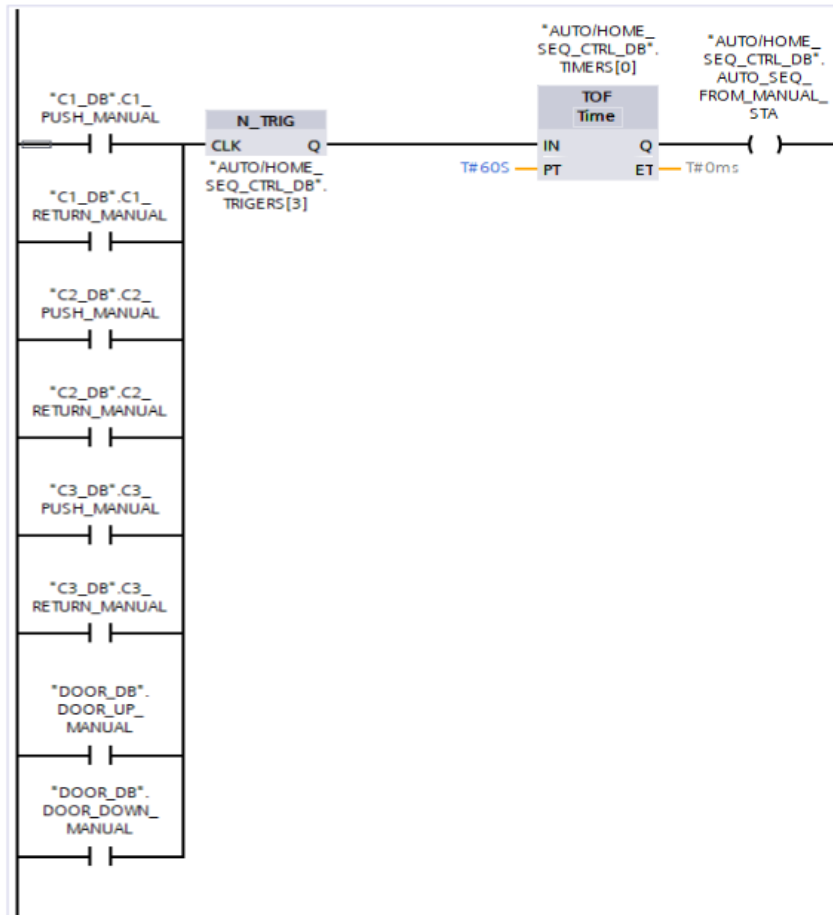
Network 7: AUTO SEQUENCE START

SET BIT FROM HM



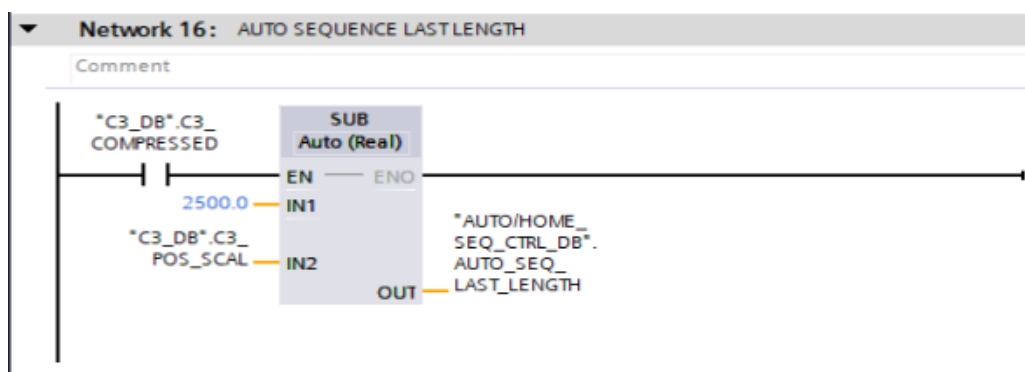
Network 17: START AUTO SEQUENCE AFTER MANUAL COMMANDS

Comment



Slika 5.20. Mogućnosti pokretanje automatske sekvence

Za izračun duljine isprešane bale (prikazano slikom 5.21.), od konačne pozicije cilindra C3, koja je 2500 mm, oduzima se pozicija koju cilindar postigne kada je tlak narastao iznad određene vrijednosti. U tom slučaju smatra se da je prešanje uspješno izvršeno, te varijabla „C3\_COMPRESSED“ postaje aktivna. Međutim, ukoliko tijekom prešanja pozicija cilindra ne dosegne određenu vrijednost koja odgovara 65% konačne vrijednosti (ta vrijednost se može prilagoditi prema potrebama), to se smatra neuspješnim prešanjem, te se cilindar povlači i pokušava ponovno prešanje.



Slika 5.21. Izračun duljine bale

Nakon završetka sekvence, odnosno nakon što je cilindar C3 postignuo *eject* poziciju, inkrementira se broj izrađenih bala metala za jedan.

Sve važne informacije, uključujući statuse i procesne varijable iz različitih funkcija unutar softvera, prepisuju se u posebne blokove podataka kako bi se omogućila komunikacija podacima između PLC-a i vizualizacijskog sustava, u ovom slučaju AVEVA InTouch-a. Stoga, unutar svake funkcije se izvodi prepisivanje (mapiranje) podataka te funkcije unutar blokova podataka za komunikaciju sa HMI-om. Na slici 5.22 prikazano je mapiranje varijabli automatske sekvence unutar blokova podataka DB215 i DB220.

Unutar mape HMI se nalazi nekoliko blokova podataka i to:

- DB200 – komande sa HMI-a
- DB210 – alarmni statusi prema HMI-u
- DB215 – statusi prema HMI-u
- DB220 – procesne varijable prema HMI-u
- DB99 – pomoćni blok podataka za alarme

Network 1: HMI MAPPING		
Comment		
1	//	AUTO SEQ
2	//	STATUS
3		
4	A	"AUTO/HOME_SEQ_CTRL_DB".AUTO_SEQ_REPOS_RUN
5	=	"DB215 - Status".AS_BS_REPOS_RUN
6	A	"AUTO/HOME_SEQ_CTRL_DB".AUTO_SEQ_RUN
7	=	"DB215 - Status".AS_BS_RUN
8	A	"AUTO/HOME_SEQ_CTRL_DB".AUTO_SEQ_REPOS_C1
9	=	"DB215 - Status".AS_BS_REPOS_C1
10	A	"AUTO/HOME_SEQ_CTRL_DB".AUTO_SEQ_REPOS_C2
11	=	"DB215 - Status".AS_BS_REPOS_C2
12	A	"AUTO/HOME_SEQ_CTRL_DB".AUTO_SEQ_REPOS_C3
13	=	"DB215 - Status".AS_BS_REPOS_C3
14	A	"AUTO/HOME_SEQ_CTRL_DB".AUTO_SEQ_REPOS_DOOR
15	=	"DB215 - Status".AS_BS_REPOS_DOOR
16	A	"AUTO/HOME_SEQ_CTRL_DB".AUTO_SEQ_END
17	=	"DB215 - Status".AS_BS_END
18	A	"AUTO/HOME_SEQ_CTRL_DB".AUTO_SEQ_STA_WAIT
19	=	"DB215 - Status".AS_BS_STA_WAIT
20	A	"AUTO/HOME_SEQ_CTRL_DB".AUTO_SEQ_FLT
21	=	"DB215 - Status".AS_BS_FLT
22	A	"AUTO/HOME_SEQ_CTRL_DB".AUTO_SEQ_PRS_FLT
23	=	"DB215 - Status".AS_BS_PRS_FLT
24	A	"AUTO/HOME_SEQ_CTRL_DB".AUTO_SEQ_ENAB
25	=	"DB215 - Status".AS_BS_ENAB
26		
27	//	PROCESS VARIABLES
28		
29	L	"AUTO/HOME_SEQ_CTRL_DB".AUTO_SEQ_RUN_TIME
30	T	"DB220 - Process variables".AS_WS_RUN_TIME
31	L	"AUTO/HOME_SEQ_CTRL_DB".AUTO_SEQ_BALES_MADE
32	T	"DB220 - Process variables".AS_WS_BLS_MADE
33	L	"AUTO/HOME_SEQ_CTRL_DB".AUTO_SEQ_LAST_LENGTH
34	T	"DB220 - Process variables".AS_WS_LST_LEN

Slika 5.22. Mapiranje varijabli automatske sekvence unutar blokova za komunikaciju sa HMI-om

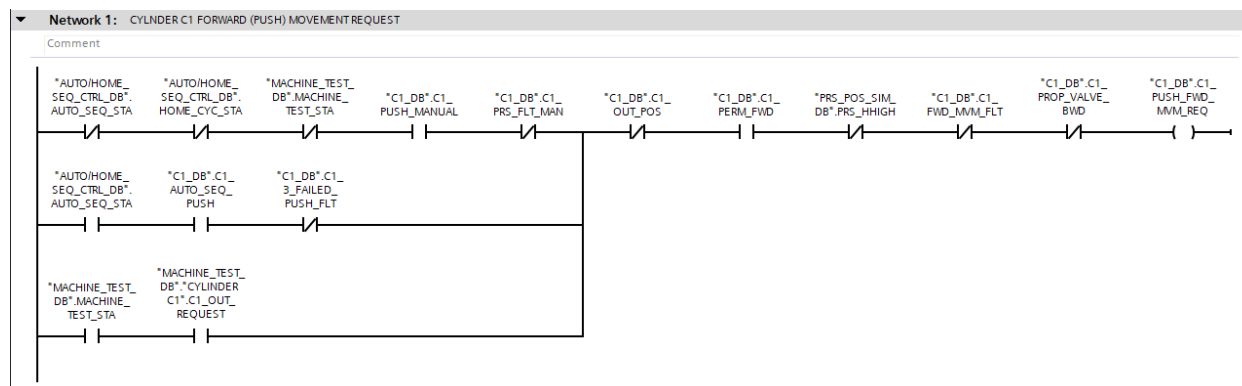
Ukoliko se tijekom automatske sekvence pojavi bilo koje alarmno stanje, alarm će biti prikazan na HMI-u, a status automatske sekvence će preći u „FAULT“. Događaji koji mogu uzorkovati ovo stanje uključuju 3 neuspješna pokušaja prešanja bilo kojeg od cilindara, neuspješno pokretanje bilo kojeg od cilindara ili „*timeout fault*“ za bilo koji pokret nekog od cilindara. „*Timeout fault*“ se javlja ukoliko trenutni pokret nekog od cilindara nije izvršen u definiranom vremenskom okviru koji je postavljen, a da u slučaju normalnog i sigurnog pomicanja cilindra bi se postigao.

Osim toga, alarmno stanje unutar automatske sekvence se aktivira kada tlak sustava tijekom pokreta premaši određenu razinu i ostane iznad te razine tijekom određenog vremena, tj. kada prešanje nije uspješno izvršeno.

## 5.4. Funkcija upravljanja cilindrom

Unutar mape „MACHINES“ nalaze se četiri funkcije za upravljanje svakim od cilindara, pri čemu svaka funkcija posjeduje posebni blok podataka karakterističan za taj cilindar. Funkcije cilindara su međusobno slične, s nekoliko pojedinačnih razlika. U nastavku će biti opisan dio funkcije za cilindar C1, pri čemu vrijedi slična logika za ostale cilindre.

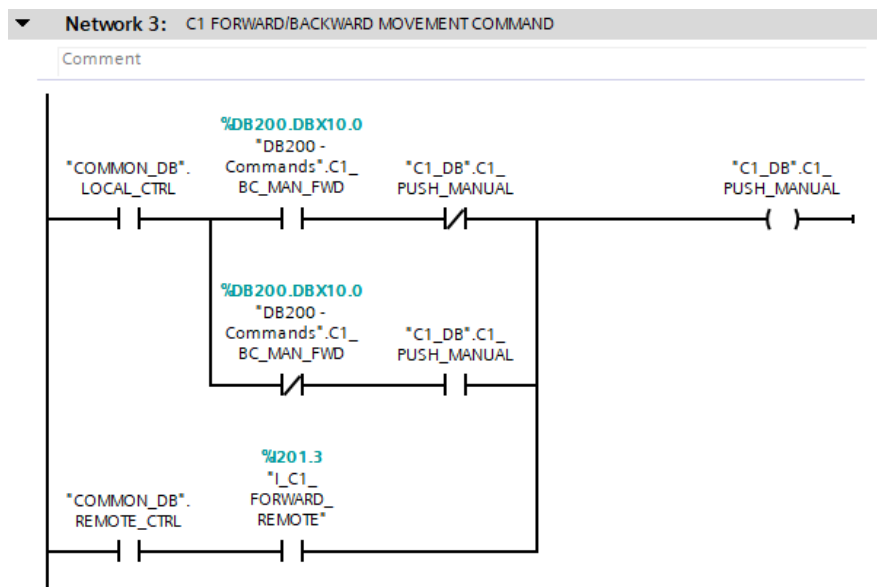
Na slici 5.23 prikazana je mreža 1 unutar funkcije upravljanja cilindrom C1. Unutar te mreže setira se zahtjev (*request*) za pomak cilindra prema naprijed. Pokretanje cilindra unaprijed se može inicirati kroz nekoliko načina. I to ručnim komandama, ukoliko nijedna od sekvenci nije aktivna i *machine selftest* nije aktivan, koji će kasnije biti opisan. Putem komande iz automatske sekvence ili komandom iz *machine selftest*-a. Nakon aktivacije neke od komandi, pomak cilindra postaje moguć uz ispunjenje nekoliko uvjeta. A to su, da sam cilindar nije u svojoj krajnjoj poziciji, da postoje dopuštenja za taj pomak (kasnije će se prikazati koja su to dopuštenja), da tlak u sustavu nije previsok, da nema greške pomaka cilindra te da prethodno nije aktivan zahtjev za pomak cilindra unatrag.



Slika 5.23. Zahtjev za pomak cilindra C1 unaprijed

Zahtjev za pomak unatrag se aktivira na sličan način, uz to što postoji i komanda za pomak cilindra u svoju nultu poziciju iz „home“ sekvence.

Ručnu komandu za pomak cilindra moguće je aktivirati na dva načina, kao što je prikazano slikom 5.24. Prvi način omogućuje komande putem HMI-a , što predstavlja lokalno upravljanje prešom. Dok drugi način odgovara komandi dobivene putem radio jedinice, čime se prešom upravlja iz udaljene pozicije (*remote*).

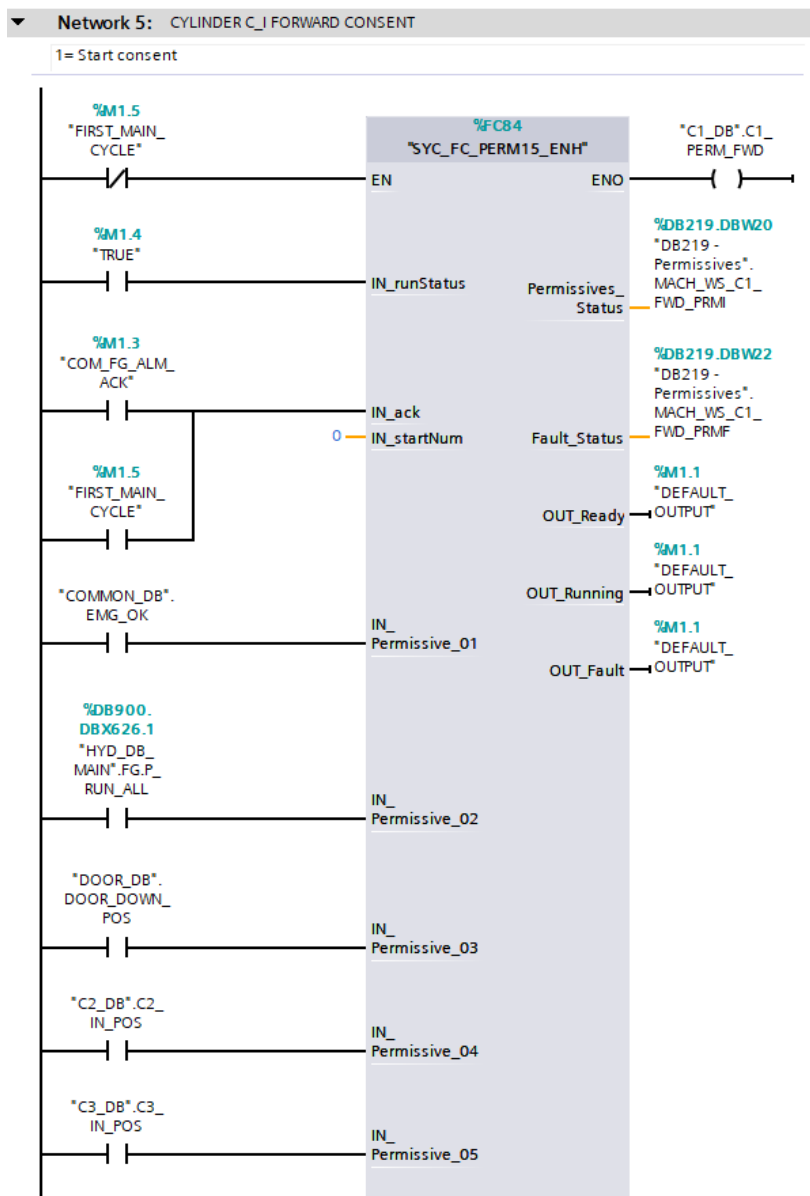


Slika 5.24. Ručna komanda za pomak cilindra

Slika 5.25 prikazuje koji uvjeti (*permissives*) moraju biti zadovoljeni kako bi se omogućilo pokretanje cilindra C1 prema naprijed. Uvjeti uključuju da nije aktivno izvanredno (*emergency*) stanje , da je hidraulična jedinica u radu, da je cilindar vrata u donjoj poziciji, da su cilindri C2 i C3 u svojim nultim pozicijama. Uvjeti se postavljaju na ulaz funkcije „SYS\_FC\_PERM15\_ENH“, koja je na upotrebu dobivena od tvrtke Danieli Systec. Funkcija pregledava stanja svih ulaznih uvjeta te prema njima određuje izlazne varijable „C1\_FWD\_PRMI“ i „C1\_FWD\_PRMF“. Ako su uvjeti zadovoljeni, varijabla PRMI će biti vrijednosti -1 što omogućuje pokretanje cilindra. No ukoliko uvjeti nisu zadovoljeni, to će biti vidljivo nekom drugom vrijednošću varijable PRMI.

Također unutar funkcije je moguće odrediti dvije različite vrste uvjeta, uvjete za start i uvjete za rad, jer primjerice kod nekih komponenti potrebno je da su neki od uvjeta prilikom starta zadovoljeni, dok se istim uvjetima može promijeniti status dok je ta komponenta u radu. Korištenje ove funkcije korisno je i radi vizualizacije jer omogućuje jednostavan pregled stanja uvjeta.

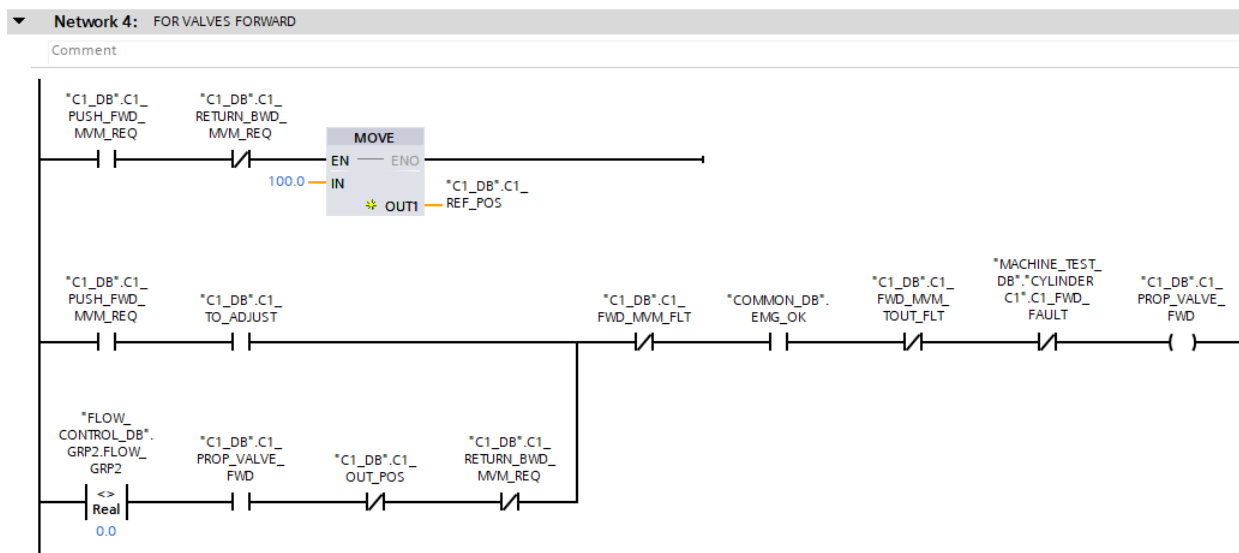




Slika 5.25. Uvjeti za pomak cilindra 1 naprijed

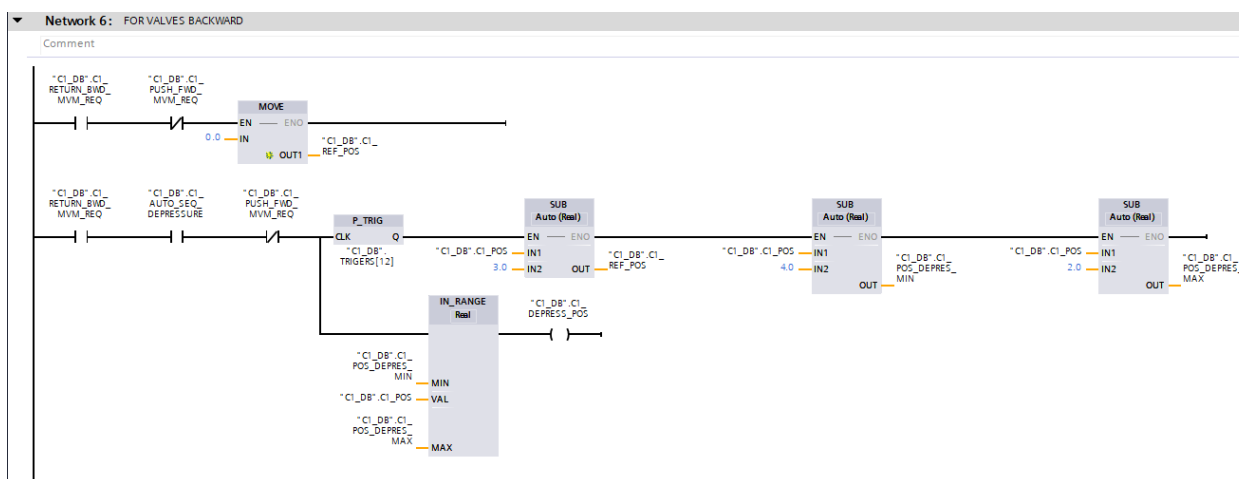
Slika 5.26 prikazuje referenciranje pozicije cilindra C1 na 100% pozicije kada je aktivan zahtjev za pomak unaprijed. Sama referenca pozicije koristi se kao referenca otvorenosti ventila za pomak tog cilindra unutar simulacije. Varijabla „C1\_TO\_ADJUST“ postaje aktivna kada pozicija cilindra nije jednaka referentnoj, te zahtjev za pomak koji je prethodno opisan aktivira varijablu za aktivaciju proporcionalnog ventila za pomak cilindra unaprijed. Zahtjev pomaka može prestati biti aktivan ali varijabla prema ventilu ostaje aktivna radi samodržanja dok god je protok ulja kroz grupu različit od nule. U slučaju nestanka zahtjeva za pomak postavlja se drugačija referenca. Cilindri C2 i C3 pripadaju unutar hidraulične grupe 1, cilindri C1 i vrata unutar hidraulične grupe 2.

Protok se kao i pozicija i pritisak simuliraju unutar posebnih funkcija.



Slika 5.26. Referenca pozicije i otvorenosti ventila za pomak C1 unaprijed

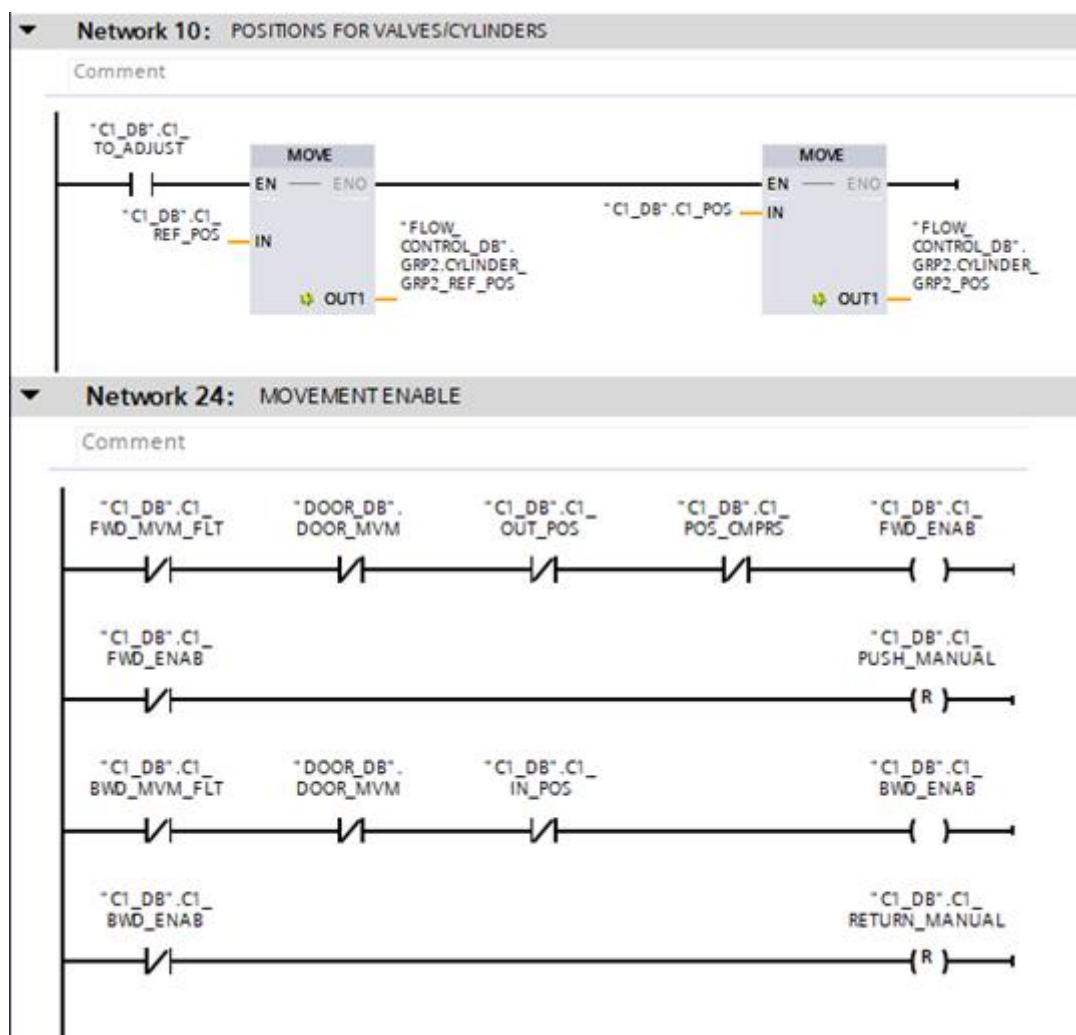
Referenciranje pozicije za pomak cilindra unatrag slično je kao u prethodnom slučaju. Postavlja se referenca pozicije i otvorenosti ventila na 0% pozicije cilindra kada je aktivan zahtjev za taj pomak. Dodatno, kada se aktivira zahtjev za „depressure“ unutar automatske sekvence, referenca pozicije se smanji za 3% kako bi se bala oslobodila pritiska tog cilindra, odnosno kako bi se cilindar vrata mogao podignuti bez poteškoća uzorkovanim pritiskom bale na njih.



Slika 5.27. Referenca pozicije i otvorenosti ventila za pomak C1 unatrag

Kada zahtjev za pomak cilindra prestaje biti aktivan, primjerice promjeni se ručna komanda, referenca pozicije se postavlja na aktualnu poziciju cilindra, dok se protok ulja kroz cilindar smanjuje i cilindar usporava. Na ovaj način se izbjegavaju nagli mehanički udarci koji bi se dogodili kada bi se cilindar naglo zaustavio. Cilindar usporava prema simulaciji, što će kasnije biti prikazano.

Na slici 5.27 su prikazane dvije mreže, u mreži 10 šalju se referentna i stvarna pozicija cilindra C1 prema hidrauličnoj grupi 2 kada je aktivna varijabla „C1\_TO\_ADJUST“. Dok unutar mreže 24 su prikazani dodatni uvjeti za pomak cilindra C1, vezanih oko cilindra vrata koji pripada istog grupi. Naime ta dva cilindra se na ovaj način ne mogu pomicati zajedno, a ako su aktivni zahtjevi za pomak oba cilindra, prvo će se izvršavati onaj koji je prvi aktiviran.

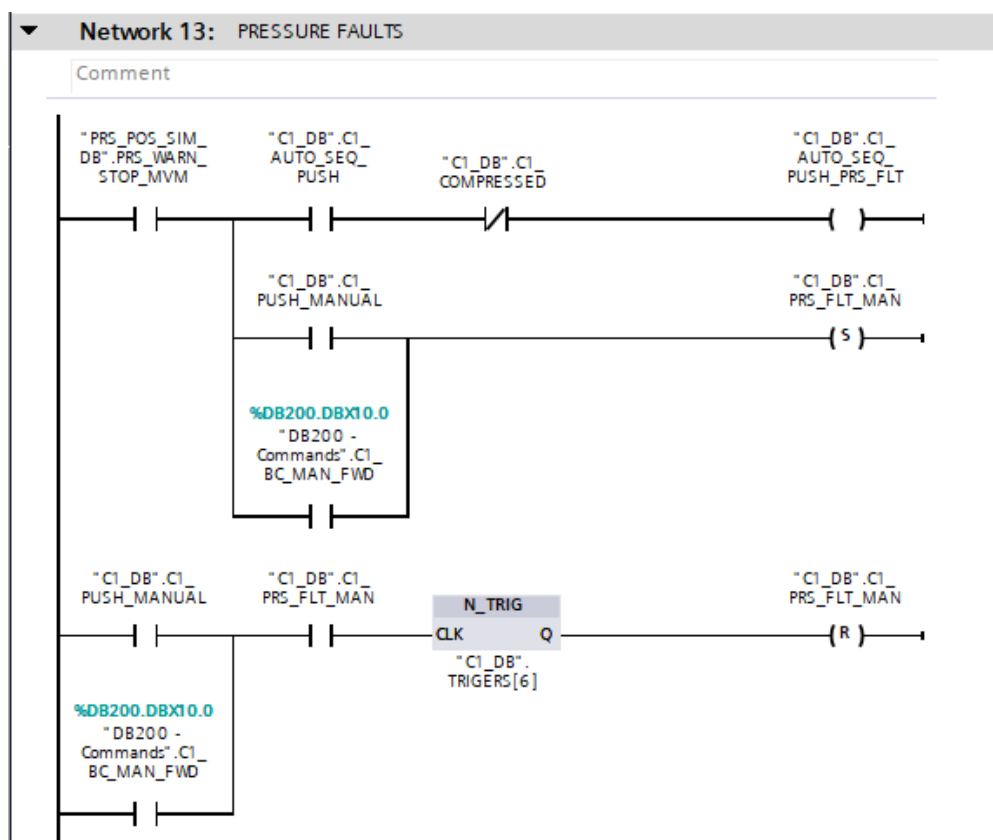


Slika 5.28. Referenciranje i aktiviranje hidraulične grupe dva za pomak cilindra C1

U slučaju da tijekom pomicanja (prešanja) cilindra C1 tlak u sustavu nadmaši određenu vrijednost, aktivira se varijabla „PRS\_WARN\_STOP\_MVM“, te se setira odgovarajuća greška koja se dalje koristi za prekid tog pomaka, kao i za alarm i status prema HMI-u (slika 5.29). Varijabla „PRS\_WARN\_STOP\_MVM“ se aktivira ovisno o razini pritiska i vremenu koji je isti prisutan. Jer primjerice pritisak sustav može kratkotrajno porasti na visoku vrijednost kada se materijal ne može odmah sprešati, tada se cilindar polako usporava kako bi se izbjeglo preveliko opterećenje. Međutim, ako se materijal ipak uspije isprešati unutar određenog vremena tlak sustava će pasti, a cilindar nastavlja gibanje.

Osim greške razine pritiska unutar sustava, greške koje se mogu aktivirati su greške uspješnog pokretanja cilindra, „*timeout fault*“ greška ta tri neuspješna pokušaja prešanja do željene pozicije unutar automatske sekvence. Kada se aktivira neka od grešaka istu je moguće resetirati pomoću HMI-a.

Na kraju ove funkcije slijedi mapiranje varijabli iz pripadajućeg bloka podataka unutar odgovarajućega bloka podataka za komunikaciju s HMI-om.



Slika 5.29. Greške visoke razine pritiska

Na slici 5.30 prikazan je blok podataka „C1\_DB“, podataka vezanih za cilindar C1.

C1_DB										
	Name	Data type	Start value	Retain	Accessible f...	Writa...	Visible in ...	Setpoint	Supervis...	Comment
1	Static									
2	C1_POS	Real	0.0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			Cylinder C1 position
3	C1_START_POS	Real	0.0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			Cylinder C1 position when movement started
4	C1_POS_SCAL	Real	0.0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			Cylinder C1 scaled position
5	C1_REF_POS	Real	0.0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			Cylinder C1 referent position
6	C1_IN_POS	Bool	false		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			Cylinder C1 position in (start)
7	C1_OUT_POS	Bool	false		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			Cylinder C1 position out (end)
8	C1_DEPRESS_POS	Bool	false		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			Cylinder C1 depressure position
9	C1_AUTO_SEQ_REPOS...	Bool	false		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			Cylinder C1 return request to in position from auto se...
10	C1_HOME_SEQ_REPO...	Bool	false		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			Cylinder C1 return request to in position from home s...
11	C1_AUTO_SEQ_PUSH	Bool	false		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			Cylinder C1 push request from auto sequence
12	C1_AUTO_SEQ_RETUR...	Bool	false		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			Cylinder C1 push from auto sequence failed, return re...
13	C1_AUTO_SEQ_DEPRE...	Bool	false		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			Cylinder C1 depressure request from auto sequence
14	C1_PUSH_FWD_MVM...	Bool	false		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			Cylinder C1 push request (forward movement)
15	C1_RETURN_BWD_MV...	Bool	false		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			Cylinder C1 return request (backward movement)
16	C1_PUSH_MANUAL	Bool	false		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			Cylinder C1 manual request to push
17	C1_PROP_VALVE_FWD	Bool	false		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			Cylinder C1 proportional valve open for forward move...
18	C1_PROP_VALVE_BWD	Bool	false		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			Cylinder C1 proportional valve open for backward mo...
19	C1_MVM	Bool	false		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			Cylinder C1 movement
20	C1_RETURN_MANUAL	Bool	false		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			Cylinder C1 manual request to return
21	C1_TO_ADJUST	Bool	false		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			Cylinder C1 need to adjust position
22	C1_ADJUSTED	Bool	false		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			Cylinder C1 adjusted position
23	C1_PRS_FLT_MAN	Bool	false		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			Cylinder C1 pressure fault in manual
24	C1_AUTO_SEQ_PUSH...	Bool	false		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			Cylinder C1 pressure fault from push in auto sequence
25	C1_PRS_FLT	Bool	false		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			Cylinder C1 pressure fault
26	C1_AUTO_SEQ_PUSH...	Bool	false		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			Cylinder C1 push from auto sequence failed
27	C1_AUTO_SEQ_PUSH...	Bool	false		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			Cylinder C1 push done from auto sequence
28	C1_PUSH_FAIL_NUMB...	Int	0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			Cylinder C1 number of failed pushes in auto sequence
29	C1_PUSH_FAIL_NUMB...	Int	0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			Cylinder C1 number of failed pushes all time
30	C1_3_FAILED_PUSH_F...	Bool	false		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			Cylinder C1 fault after 3 failed pushes
31	C1_MVM_REQ	Bool	false		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			Cylinder C1 movement request
32	C1_MVM_FLT	Bool	false		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			Cylinder C1 movement fault
33	C1_HMI_RESET_FLT	Bool	false		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			Cylinder C1 reset fault from HMI
34	C1_FWD_MVM_TOUT...	Bool	false		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			Cylinder C1 forward movement timeout fault
35	C1_BWD_MVM_TOUT...	Bool	false		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			Cylinder C1 backward movement timeout fault
36	C1_FWD_STA_FAIL	Bool	false		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			Cylinder C1 forward movement start fail
37	C1_BWD_STA_FAIL	Bool	false		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			Cylinder C1 backward movement start fail
38	C1_STA_FAIL	Bool	false		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			Cylinder C1 start fail
39	C1_FWD_MVM_FLT	Bool	false		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			Cylinder C1 forward movement fault
40	C1_BWD_MVM_FLT	Bool	false		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			Cylinder C1 backward movement fault
41	C1_MVM_DIR	Real	0.0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			Cylinder C1 movement direction
42	C1_COMPRESS_DONE	Bool	false		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			Cylinder C1 compress done
43	C1_COMPRESSED	Bool	false		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			Cylinder C1 compressed
44	C1_POS_DEPRES_MIN	Real	0.0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			Cylinder C1 aux depress position min
45	C1_POS_DEPRES_MAX	Real	0.0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			Cylinder C1 aux depress position max
46	C1_POS_CMPRS	Bool	false		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			Cylinder C1 compress position
47	C1_FWD_ENAB	Bool	false		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			Cylinder C1 forward movement enable
48	C1_BWD_ENAB	Bool	false		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			Cylinder C1 backward movement enable
49	C1_OVR_SIZE_BALE	Bool	false		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			Cylinder C1 oversize bale
50	C1_RST_MVM_FLT	Bool	false		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			Cylinder C1 reset movement fault
51	C1_PERM_FWD	Bool	false		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			Cylinder C1 forward movement permissive
52	C1_PERM_BWD	Bool	false		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			Cylinder C1 backward movement permissive
53	TRIGERS	Array[0..20] of Bool			<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			Cylinder C1 triggers
54	TIMERS	Array[0..10] of IEC...			<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			Cylinder C1 timers

Slika 5.30. Blok podataka cilindra C1

## 5.5.Funkcija machine selftest

„*Machine selftest*“ odnosno testiranje pokreta određenog cilindra provodi se na sljedeći način, unutar preše nije postavljan metalni materijal, cilindar se postavlja u nultu poziciju, potom se pomiče prema naprijed, nakon čega slijedi *decompression* nakon što dosegne krajnju poziciju, te povratak natrag u početnu poziciju. Tijekom ovih pokreta, mjeri se vrijeme potrebno za svaki pojedinačni pokret, kao i brzine za svaki od njih. Određuju se prosječna i trenutna brzina za *forward i backward movement*.

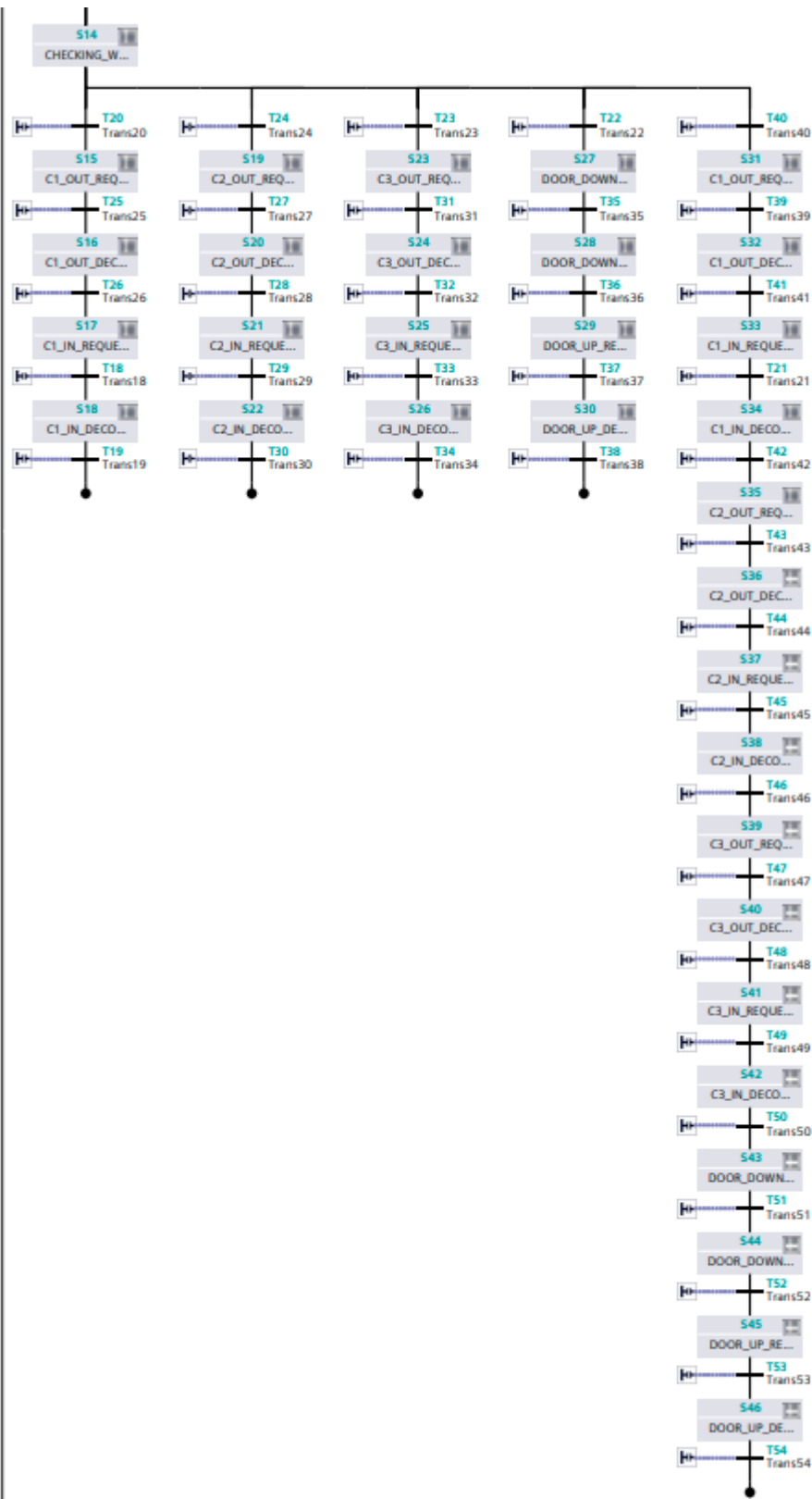
Postoje dvije mogućnosti testiranja pokreta cilindara. Prva opcija je testiranje samo jednog određenog odabranog cilindra, dok druga opcija je kompletni *selftest* unutar kojega se testiraju pokreti svakog od cilindara, jedan za drugim.

U prilogu dva nalazi se mreža unutar funkcije „*machine selftest*“, gdje se mjere vremena i brzine za pomak cilindra C1 prema naprijed. Svaki pokret svakoga od cilindara posjeduje sličnu mrežu unutar koje se mjere vrijednosti za taj cilindar.

Slika 5.31. prikazuje sekvencu „*machine selftesta*“, početni dio sekvence uključuje početno repositioniranje svih cilindara i ne nalazi se na slici. Nakon toga slijedi ulazak u granu ovisno o odabranom načinu testiranja, odabir se vrši putem HMI-a.

Sam „*machine selftest*“ provodi se nakon puštanja u pogon hidraulične preše. Mjere se potrebna vremena i brzine, a rezultati se bilježe unutar određene baze podataka. Tijekom rada preše, ukoliko se primijeti da se neki od pokreta preše usporio, moguće je provesti test te dobiti informacije o tome koliko se pokret usporio. To su značajni podaci za izradu izvještaja o stanju preše te dobar pokazatelj gdje unutar sustava može biti prisutna pogreška.

Ako se tijekom testiranja javi „*timeout fault*“ ili neuspješno pokretanje bilo kojeg od pokreta cilindara, to predstavlja alarmno stanje. Test se zaustavlja, a potrebna je hitna intervencija popravka preše.



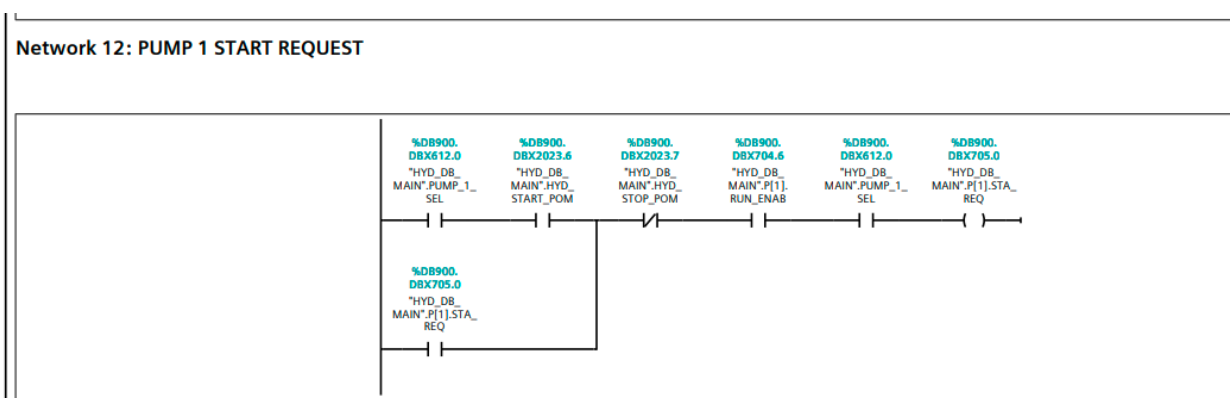
Slika 5.31. Sekvenca machine selftest-a

## 5.6. Funkcija hidraulične jedinice

Funkcija hidraulične jedinice podijeljena je na dva dijela. Jedna funkcija služi za pokretanje pojedinačnih pumpi, dok druga funkcija upravlja hidrauličnom jedinicom koja se u ovom radu sastoji od 4 pumpe i grijača ulja unutar spremnika. Pokretanje pumpi ovisi o odabranim pumpama za rad, što se ostvaruje pomoću HMI-a sučelja. Prva po redu odabrana pumpa se pokreće odmah, dok se preostale odabrane pumpe pokreću s vremenskom zadržkom od 2 sekunde nakon što se prethodna pumpa pokrenula, što se može vidjeti uvidom u slike 5.32. i 5.33.

Za pokretanje odabranih pumpi koristi se zajednički zahtjev, naravno ukoliko su one spremne za rad, tj. ukoliko su sva dopuštenja (*permissivies*) zadovoljena. Dopuštenja uključuju zahtjeve kao što su da je odabrana barem jedna od pumpi, odsustvo vanrednih stanja, te da su temperatura i razina ulja unutar spremnika između minimalnih i maksimalnih vrijednost. Ukoliko je temperatura ulja ispod minimalne aktivira se zahtjev za uključenje u rad grijača koji se sastoji od grijačeg otpornika. Nakon aktivacije varijable za pokretanja pumpe, ukoliko u određenom vremenskom periodu pumpa se ne pokrene, što bi se očitovalo statusom *sofstater*-a, zaustavljaju se sve pumpe. Za svaku od pumpi se mjeri vrijeme u radu, te se zajedno sa ostalim bitnim informacija prikazuje na HMI-u što će kasnije biti prikazano.

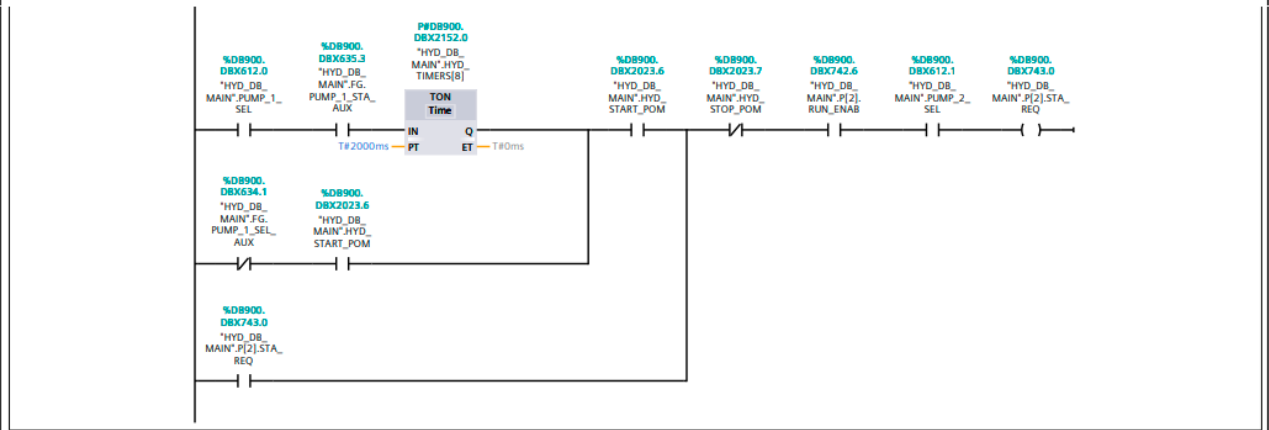
Nakon pokrenute hidraulične jedinice odnosno svih odabranih pumpi moguće je ostvariti pomake cilindra.



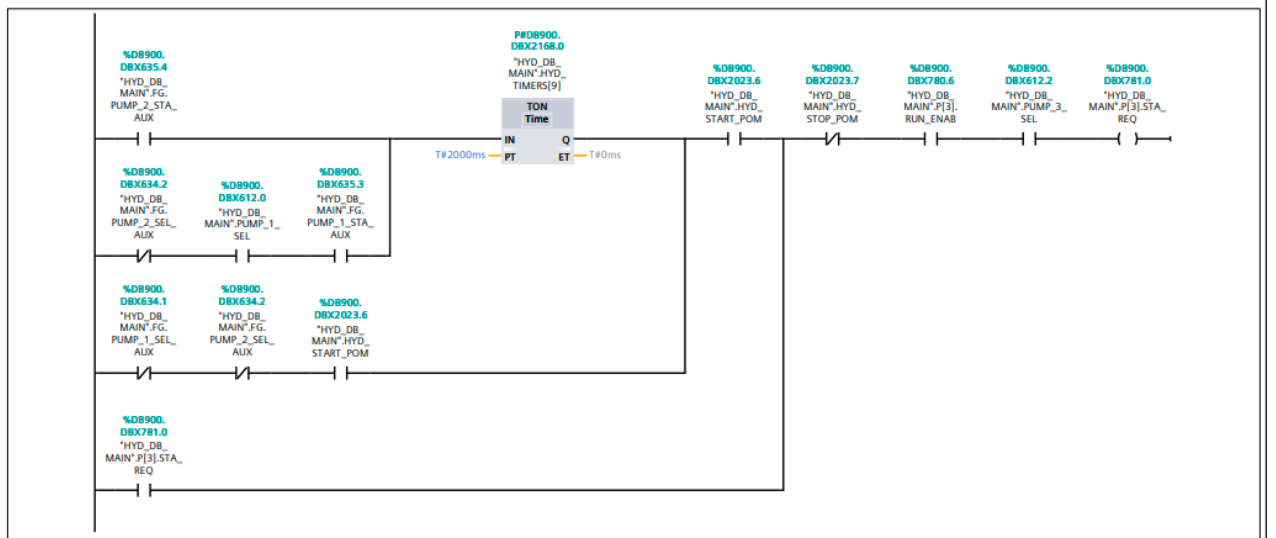
Slika 5.32. Logika pokretanje pumpi 1 dio



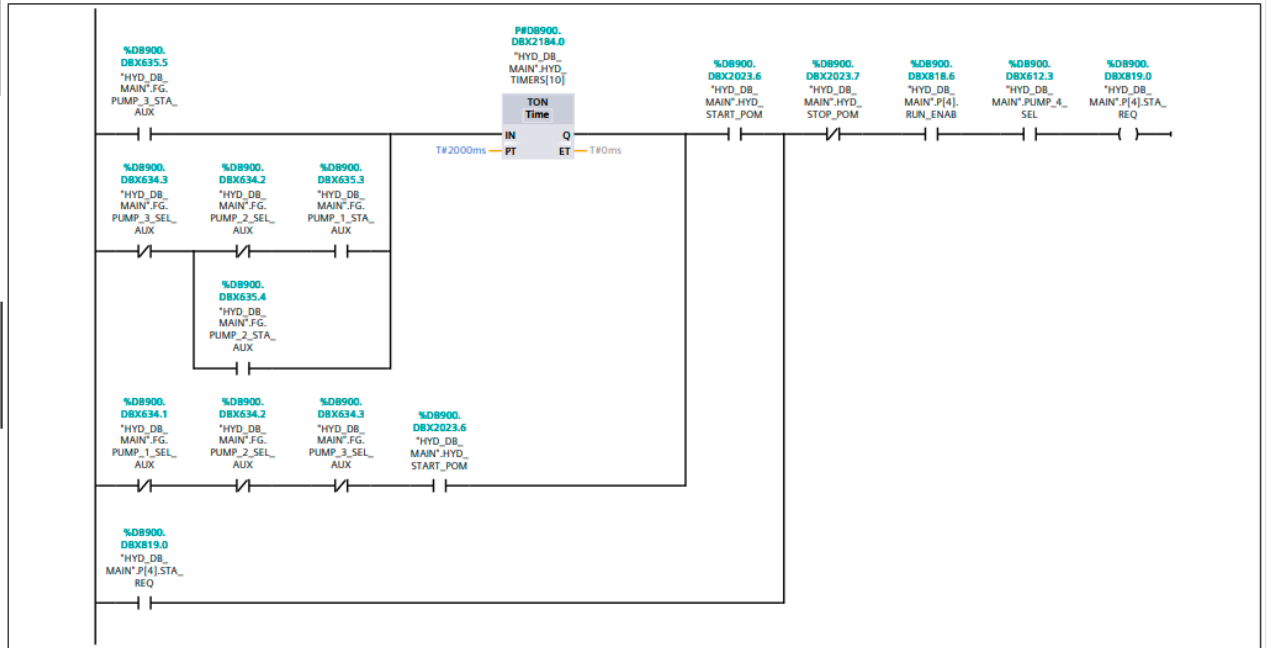
**Network 13: PUMP 2 START REQUEST**



**Network 14: PUMP 3 START REQUEST**



**Network 15: PUMP 4 START REQUEST**



Slika 5.33. Logika pokretanje pumpi 2 dio

## 5.7. Funkcija simulacije rada

Unutar funkcije simulacije simuliraju se protok ulja (*flow*), otvorenost proporcionalnih ventila, pozicije cilindara i pritisak ulja unutar sustava.

Jedan od osnovnih ciljeva rada bio je provesti referenciranje otvorenosti proporcionalnih ventila, što se unutar rada postiže simulacijom. U radu su korištene iste vrijednosti za protok ulja i otvorenosti ventila, pošto su to komplementarne varijable. S vrijednošću otvorenosti ventila tj. protoka ulja određuju se pomaci pozicija cilindara unutar simulacije. U stvarnosti bi određeni ventil s određenim postotkom bio otvoren te dopustio protok ulja prema cilindru, koji bi uzorkovao pomak cilindra u željenom smjeru. Za svaki cilindar bi u tom slučaju bila potrebna najmanje dva ventila, svaki za jedan od dva smjera kretanja. No u stvarnosti, za hidraulične mašine općenito, pa tako i za prešu, često se koriste i dodatni ventili koji omogućavaju brže kretanje ili veći moment.

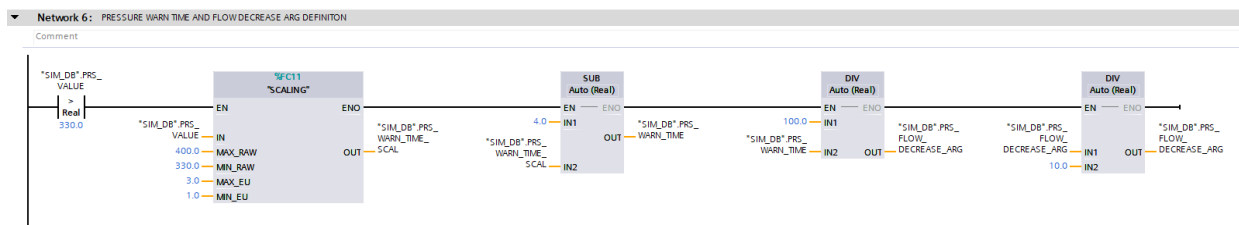
Željeno kretanje cilindara pri radu preše je da se cilindar ubrzava po rampi do maksimalne brzine, zatim kretanje maksimalnom brzinom dok se ne javi potreba za usporavanjem. Usporavanje se izvodi također preko rampe, ali razlikuju se dvije rampe s različitim nagibima.

U slučaju *selftest*-a poznata je željena pozicija koju cilindar želi postići, a to je njegova krajnja ili početna pozicija. Usporavanje se inicira na poziciji nešto prije željene, a usporava se jednim nagibom do polovične maksimalne brzine, nakon čega slijedi veći nagib rampe usporavanja kako se cilindar približava željenoj poziciji. Upotrebom proporcionalnih ventila, odnosno referenciranjem njihove otvorenosti postiže se željeno ubrzavanje ili usporavanje. Na taj način se minimiziraju hidraulični udarci i naprezanja cijelog hidrauličnog sustava. Jer primjerice kod ON/OFF ventila koji imaju samo dvije moguće pozicije, hidraulični udarci su značajni, što smanjuje životni vijek hidraulične jedinice, ventila i samih cilindara preše.

Kod ručnih komandi, ubrzavanje je isto kao u prethodnom slučaju, a u slučaju ako se ručna komadna drži aktivnom sve do krajnje pozicije, tada je i usporavanje identično. No, u slučaju da se ručnim komandama želi postaviti cilindar u neku poziciju između dviju krajnjih, potrebno je usporiti cilindar što brže kako bi se cilindar zaustavio što bliže poziciji gdje je deaktivirana komanda za pomak cilindra. Pošto se za pomak cilindra ručna komanda mora držati aktivnom tijekom pomicanja, te kada se ista makne, cilindar se mora što prije zaustaviti, pa u tom slučaju usporavanje se izvršava po strmijem nagibu sa maksimalne brzine na nultu vrijednost.

Tijekom kretanja cilindra u svrhu prešanja materijala, pritisak ulja unutar sustava raste. Što je veći pritisak, to veći otpor materijal pruža cilindru, te u slučaju previsokog pritiska potrebno je zaustaviti taj cilindar kako pritisak unutar sustava ne bi premašio određene granice, iznad kojih prijeti oštećenje hidraulične jedinice. Tijekom prešanja pritisak može naglo porasti u slučaju da se materijal ne može isprešati, nakon čega slijedi usporavanje cilindra. No, ukoliko se tijekom tog usporavanja materijal ipak uspije isprešati, cilindar će početi ponovno ubrzavati. U drugom slučaju, ako se materijal nije uspio isprešati pritisak unutar sustava bi nastavio rasti i cilindar bi se u potpunosti usporio. Definira je logika koja simulira ovo djelovanje, koje osim vrijednosti pritiska ovisi i o vremenu koje je ono prisutno iznad određene granice.

Na slici 5.34 prikazano je izračunavanje argumenta kojim se otvorenost ventila mora smanjivati u slučaju visoke vrijednosti tlaka ulja unutar sustava. Vrijednost tlaka ulja je posebnom funkcijom skalirana na vrijednost koja je proporcionalna razini tlaka iznad granice, a u ovom slučaju ta vrijednost predstavlja vrijeme unutar kojeg je potrebno zaustaviti pokret cilindra. Što je vrijednost tlaka veća to je vrijeme (WARN\_TIME) manje jer prijeti veća opasnost od oštećenja sustava. Zatim se za to vrijeme određuje argument smanjenja otvorenosti ventila, opet što je pritisak veći to je i veće smanjenje otvorenosti ventila kako bi se cilindar zaustavio što prije.



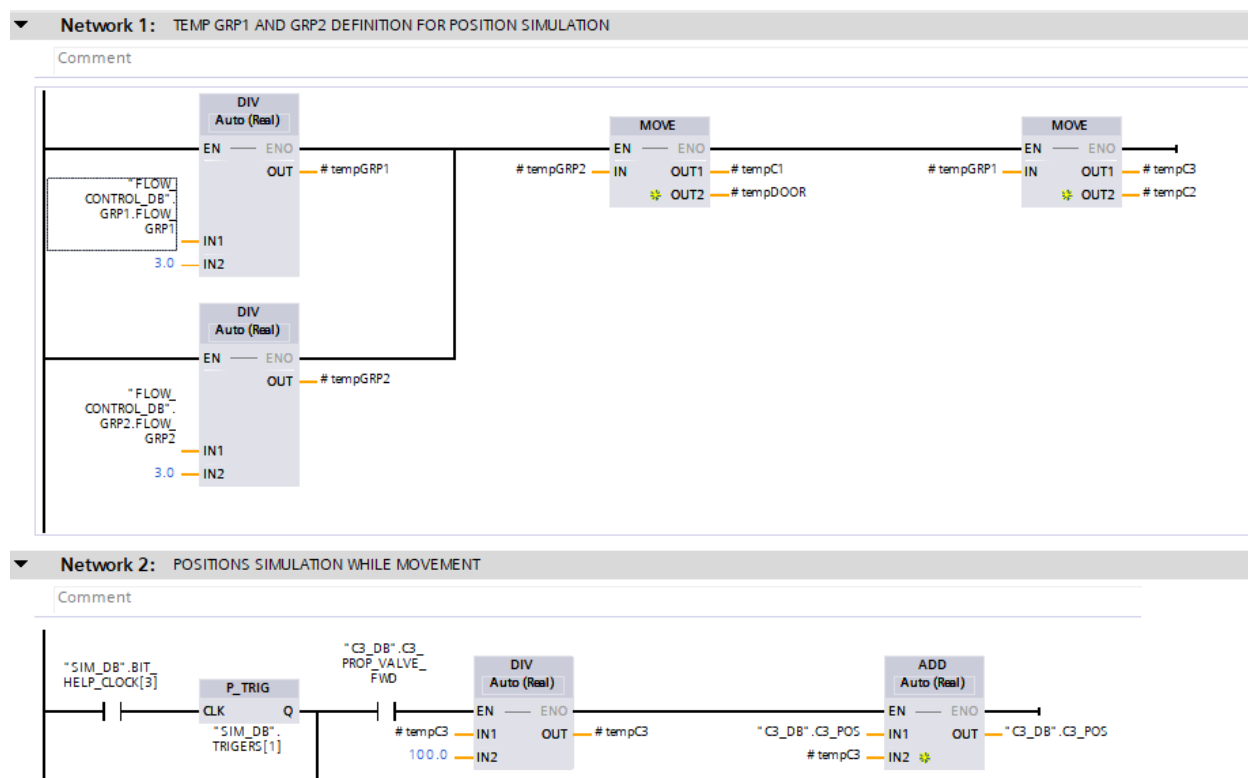
Slika 5.34. Određivanje vrijednosti vremena i argumenta u slučaju visoke vrijednosti pritiska ulja

Ukoliko je visoka vrijednost pritiska ulja prisutna dulje od vremena koje se dobije prema prethodnom opisu, cilindar se zaustavi te se aktivira alarm koji javlja neuspješni pokušaj prešanja. U automatskoj sekvenci nakon toga slijedi povratak cilindra, dok kod ručnim komandi se onemogućuje dodatno prešanje cilindrom, te je potrebno resetirati pogrešku. U slučaju da je vrijednost tlaka ulja smanjena ispod visoke vrijednost u vremenu kraćem od WARN\_TIME-a, cilindar se ne zaustavlja, nego počinje ubrzavati.

Visoka vrijednost tlaka ulja simulira se postavljanjem određene vrijednosti tlaka ulja tijekom rada preše, što će kasnije biti prikazano na primjeru prešanja cilindrom C2.

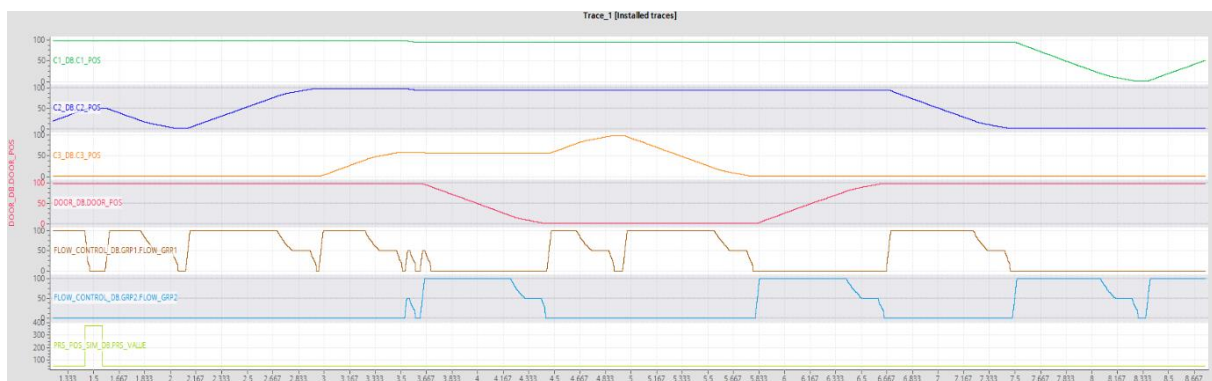
Osim visoke vrijednosti pritiska ulja, simulira se dekompresija koja je potrebna nakon svakog pokreta cilindra kako bi se hidraulični sustav rasteretio. Simulacija se ostvaruje postavljanjem vrijednosti od 50 bara nakon određenog vremena kada je završen pomak određenog cilindra. Vrijednost od 50 bara je odabrana jer je to vrijednost tlaka ulja koja bi bila pristuna u stvarnosti u slučaju da ni jedan od cilindar preše se ne pomiče. Obično je potrebno postići pritisak od 18-20 bara kako bi se pumpe mogle pokrenuti.

Pozicije cilindra, kako je već spomenuto simuliraju se ovisno o vrijednosti reference otvorenosti proporcionalnih ventila, jer što je ventil s veći postotkom otvoreniji to veći protok ulja putuje prema cilindru. Slika 5.35 prikazuje simulaciju pozicije cilindra C3 kada je aktiviran proporcionalni ventil za pomak cilindra unaprijed.



Slika 5.35. Simulacija pozicije

Na slici 5.36 dijagram prikazuje pozicije svih cilindara zajedno s protokom ulja (referencom otvorenosti ventila) i tlakom ulja unutar sustava tijekom rada preše u automatskoj sekvenci. U ovom slučaju povećanje tlaka tijekom prešanja nije simulirano, već je prikazan primjer visoke vrijednosti tlaka ulja tijekom prešanja cilindrom C2. Na početku dijagrama vidljivo je da je cilindar C1 u krajnoj poziciji, što znači da je izvršeno prešanje u toj osi. Nakon toga slijedi prešanje cilindrom C2, prilikom kojeg se javlja visoka vrijednost tlaka ulja što usporava i zaustavlja sam cilindar. Nakon povratka u nultu poziciju, cilindar ponovnim pokušajem stiže u svoju krajnju poziciju čime je odrađeno prešanje materijala i u drugoj osi. Zatim slijedi prešanje cilindrom C3 sve dok se ne pojavi visoki pritisak. Kod cilindra C3 to znači da se zaustavlja i ne pokušava ponovno prešanje jer je popunjena kompresijska komora preše, osim u slučaju da cilindar nije postignuo barem 50% svoje krajnje pozicije. Nakon toga je vidljiv istovremeni *depressure* cilindara C1 i C2 (mali povratak pozicije i impuls protoka ulja), zatim *depressure* C3 i podizanje cilindra vrata istovremeno. Dolazi se do izbacivanja isprešane bale metala pomakom cilindra C3 do krajnje pozicije, nakon čega slijedi repositioniranje svih cilindara te ponovni ciklus automatske sekvence koji započinje pomakom cilindra C1. Za potrebe ovog dijagrama pokrenuta je nova sekvenca odmah nakon završene prethodne, no inače je potrebno vrijeme za postavljanje novog metalnog otpada unutar preše te aktivacija početka sekvence nakon što je postavljanje izvršeno.

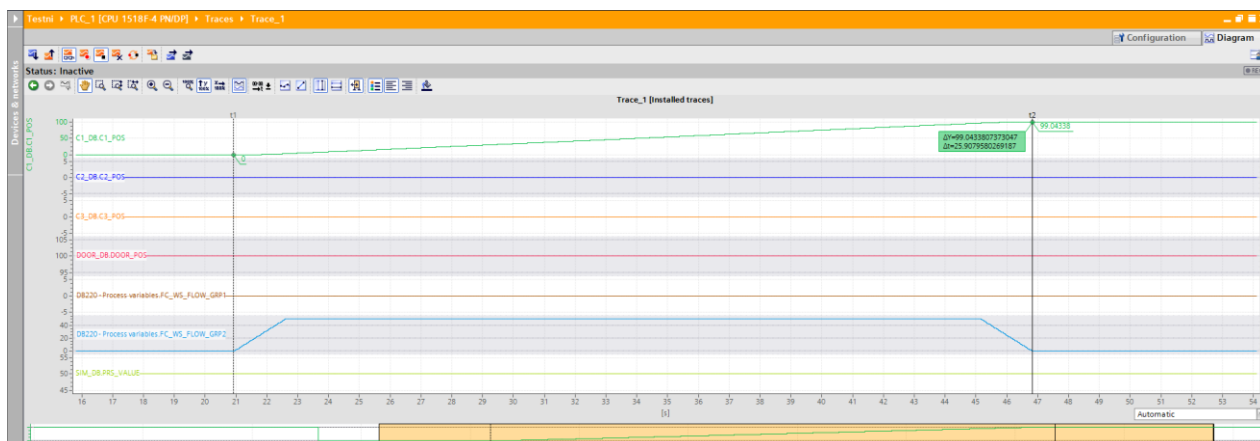


Slika 5.36. Simulacija automatske sekvence

## 5.8. Usporedba različitih modova kontrole automatskog pozicioniranja cilindra

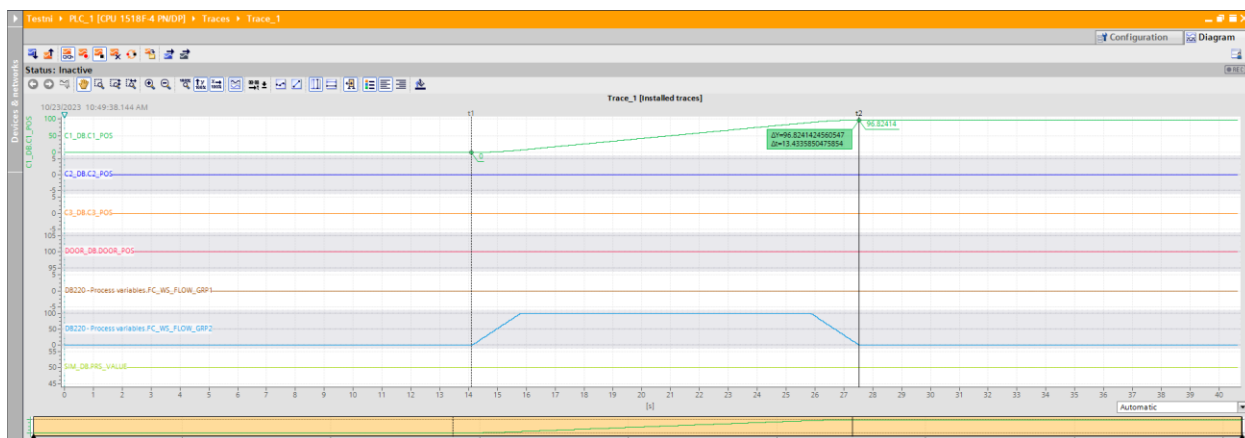
Usporedba različitih modova kontrole automatskog pozicioniranja hidrauličkih cilindra provodi se u odnosu na 2 načina kontrole koje se inače koriste u radu hidraulične preše. Ti načini su pozicioniranje tijekom kretanja cilindra s dvije različite brzine, a to su *low* i *high speed*, te treći način koji je za potrebe ovoga rada izrađen kombinacijom ta dva načina.

Na slici 5.37. prikazano je pozicioniranje cilindra tijekom kretanja sporijom brzinom odnosno *low speed*. Cilindar se pozicionira na 99.04% krajnje pozicije, što znači da je pogreška pozicioniranja manja od 1%. Vrijeme potrebno za taj pomak je 25.09 sekundi, što je očekivano zbog sporije brzine kretanja. Prosječna brzina tijekom tog moda iznosi 187.68 mm/s (maksimalna vrijednost), dobivena provedbom *machine selftest*-a za taj način automatskog pozicioniranja hidrauličkog cilindra C1.



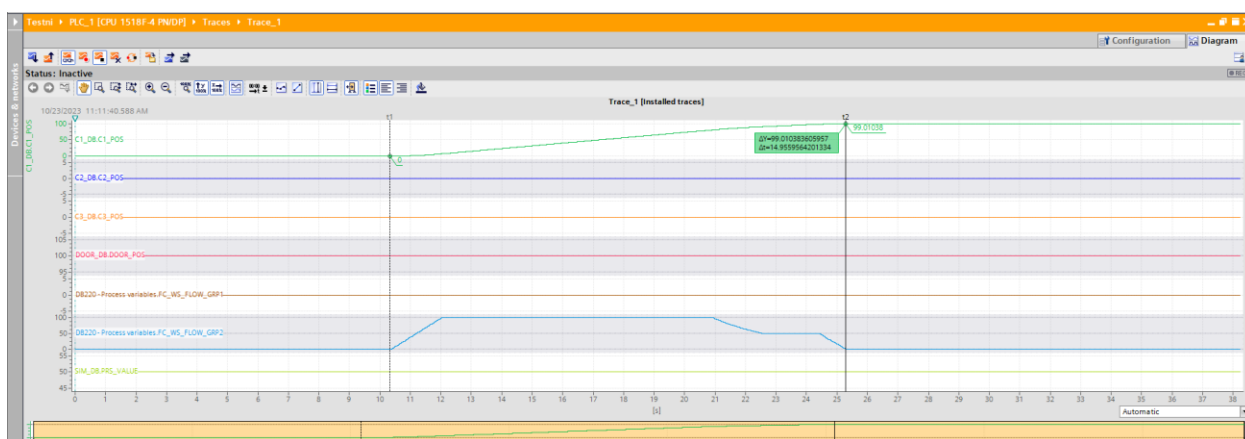
Slika 5.37. Automatsko pozicioniranje cilindra s kretanjem *low speed*

Na slici 5.38. provedeno je slično automatsko pozicioniranje cilindra, ali za slučaj kretanja s *high speed*. Cilindar se pozicionira na 96.82% krajnje pozicije, što rezultira većom pogreškom nego u prethodnom slučaju, što je očekivano zbog bržeg kretanja. Također primjetno je kraće vrijeme kretanja od 13.43 sekunde. Prosječna brzina u ovom slučaju iznosi 372.94 mm/s.



Slika 5.38. Automatsko pozicioniranje cilindra s kretanjem high speed

Te treći slučaj automatskog pozicioniranja dobiven kombinacijom dvaju prethodnih načina. Ovaj način automatskog pozicioniranja koristi za sve pokrete preše unutar ovoga rada i prikazan je na slici 5.39. Kako je prethodno opisano, protok se tijekom pokretanja cilindra rampira na maksimalnu vrijednost od 100%, što je ujedno i referenca otvorenosti proporcionalnog ventila, što je identično slučaju s *high speed*-om. Međutim, tijekom samog pozicioniranja u usporavanju prvo se sa *high speed* načina prelazi na *low speed*, nakon čega se cilindar u potpunosti zaustavlja na 99.01% krajnje pozicije, sa vremenom od 14.96 sekundi. To znači da ovaj način pozicioniranja pruža dobru preciznost pozicioniranja kao kod *low speed* načina, te kratko vrijeme trajanja pokreta koje je nešto duže nego kod *high speed* načina. Dakle ovaj način nudi prednosti oba načina automatskog pozicioniranja hidrauličkih cilindara u jednom.



Slika 5.39. Automatsko pozicioniranje cilindra s kombinacijom dvaju načina



## 6. Radio daljinska jedinica

Radio daljinska jedinica koja se u radu koristi je od proizvođača AUTEK, i to odašiljač TAJQ0343L0, a prijammnik DCMR24. Ova radio jedinica služi za daljinsku kontrolu i nadzor hidraulične preše, što omogućuje upravljanje prešom iz bilo kojeg položaja unutar radnog raspona prijammnika.

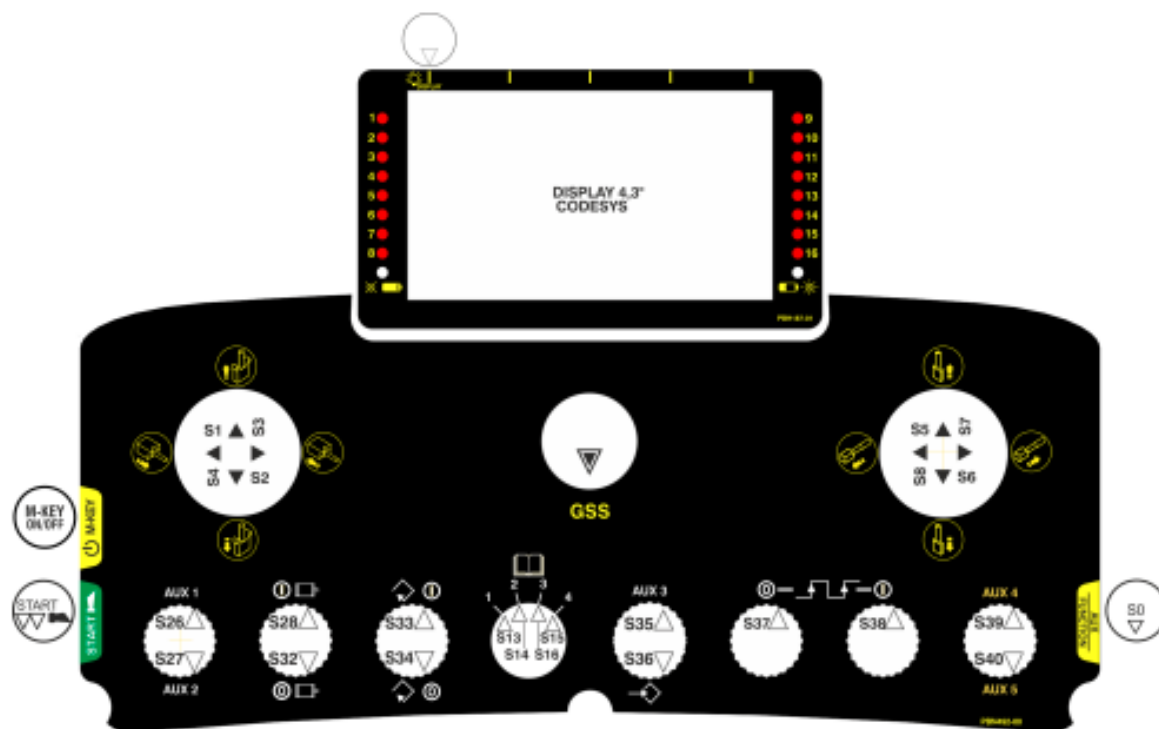
Radio odašiljač prikazan na slici 6.1. opremljen je zaslonom za prikaz statusa preše, kao što su tlak ulja unutar sustava, stanje preše, broj radnih sati, pozicije cilindara te aktiviranih alarma. Također, odašiljač posjeduje različite komande u obliku *joystick*-a, preklopnih prekidača te sigurnosne gljive pomoću kojih se postiže upravljanje pojedinim dijelom hidraulične preše. Maksimalni radni raspon može varirati ovisno o radnom okruženju.

Radio odašiljač komunicira s prijammnikom putem radio veze, dok prijammnika s PLC-om ostvaruje komunikaciju putem EtherNet-a. Svaka od komandi posjeduje odgovarajuću oznaku što je vidljivo na slici 6.2., po kojima se određuju adrese unutar PLC-a, po uputstvima proizvođača kako je prikazano na slici 6.3.



Slika 6.1. Radio odašiljač







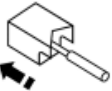
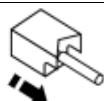
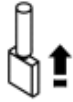
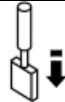
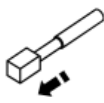
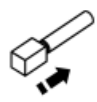


Slika 6.2. Oznake komandi radio odašiljača

Air - Ethernet Inputs									
Byte	Bit								Description
	7	6	5	4	3	2	1	0	
0					ENABLE				System info
1	S7	S6	S5	S4	S3	S2	S1	S31/START	Digital commands
2	S15	S14	S13	S12	S11	S10	S9	S8	
3	S23	S22	S21	S20	S19	S18	S17	S16	
4	S0	S30 / SAF	S29	S28	S27	S26	S25	S24	
5	S39	S38	S37	S36	S35	S34	S33	S32	
6			BAT	S44	S43	S42	S41	S40	
7									Reserved
8									
9									
10									
11									
12									
13									Reserved
14									
15									
16									
17									
18									
19									
20	Bytes	Message length							

Slika 6.3. Ulazne adrese po oznakama komandi

Svaka komanda posjeduje određenu oznaku, čije značenje se nalazi unutar tablice 6.1. Neke od komandi na radio odašiljaču se unutar rada ne koriste te nisu prikazane unutar tablice.

Tablica 6.1. Oznake komandi radio jedinice

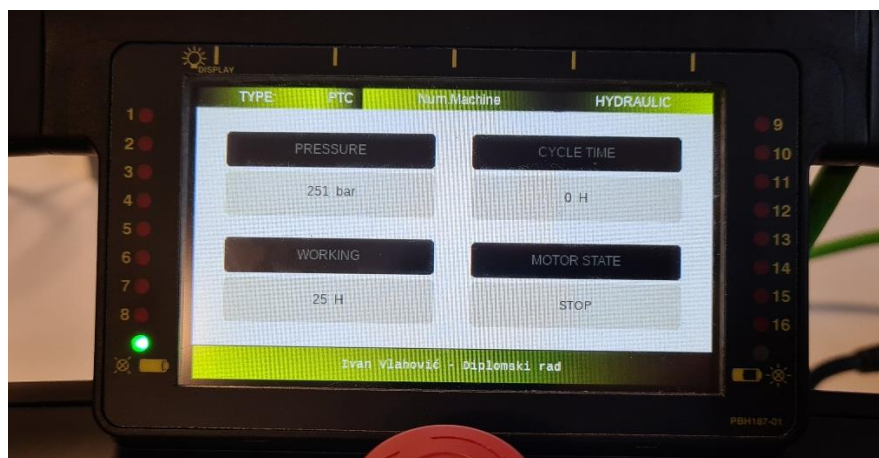
OZNAKA KOMANDE	OPIS	OZNAKA
S1	C2 GORE KOMANDA	
S2	C2 DOLJE KOMANDA	
S3	C1 NAPRIJED KOMANDA	
S4	C1 NATRAG KOMANDA	
S5	VRATA GORE KOMANDA	
S6	VRATA DOLJE KOMANDA	
S7	C3 NAPRIJED KOMANDA	
S8	C3 NATRAG KOMANDA	
S28, S32	HIDRAULIČNA JEDINICA START / STOP KOMANDA	
S37, S38	AUTOMATKSA SEKVENCE START / STOP KOMANDA	
GSS	SIGURNOSNA GLJIVA	

Unutar TIA Portala uvezana je GSD (*General Station Description*) datoteka koja odgovara korištenoj radio jedinici. Datoteka se postavlja unutar hardverske konfiguracije, gdje se postavlja odgovarajuća IP adresa, te ulazni/izlazni moduli koji odgovaraju onima na slici 6.3. (na slici prikazani samo ulazni). Za te module se definiraju željene adrese unutar softvera, te ulazne adrese radio jedinice unutar TIA Portala prikazane se na slici 6.4.

3	I_EMERGENCY_STOP_REMOTE	IO Map Table	Bool	%I200.2	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
4	I_REMOTE_ON	Default tag table	Bool	%I200.3	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
5	I_UNUSED_SW29	IO Map Table	Bool	%I200.4	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
6	I_START_REMOTE	IO Map Table	Bool	%I201.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
7	I_C2_UP_REMOTE	IO Map Table	Bool	%I201.1	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
8	I_C2_DOWN_REMOTE	IO Map Table	Bool	%I201.2	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
9	I_C1_FORWARD_REMOTE	IO Map Table	Bool	%I201.3	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
10	I_C1_BACKWARD_REMOTE	IO Map Table	Bool	%I201.4	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
11	I_DOOR_UPWARD_REMOTE	IO Map Table	Bool	%I201.5	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
12	I_DOOR_DOWNWARD_REMOTE	IO Map Table	Bool	%I201.6	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
13	I_C3_BACKWARD_REMOTE	IO Map Table	Bool	%I201.7	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
14	I_C3_FORWARD_REMOTE	IO Map Table	Bool	%I202.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
15	I_MOTOR_1_REMOTE	IO Map Table	Bool	%I204.4	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
16	I_MOTOR_0_REMOTE	IO Map Table	Bool	%I205.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
17	I_CONTINUOUS_CYCLE_REMOT	IO Map Table	Bool	%I205.1	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
18	I_CONTINUOUS_CYCLE_REMOT_OFF	IO Map Table	Bool	%I205.2	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
19	I_PHOCELL_START_REMOTE	IO Map Table	Bool	%I205.4	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
20	I_CYCLE_0_REMOTE	IO Map Table	Bool	%I205.5	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
21	I_CYCLE_1_REMOTE	IO Map Table	Bool	%I205.6	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

Slika 6.4. Ulazne adrese radio jedinice

Stranica na zaslonu radio odašiljača sa procesnim varijablama je prikazana na slici 6.5. Prikazuju se vrijednosti pritiska ulja unutar sustava, vrijeme trajanja zadnje izvršene automatske sekvence, ukupno radno vrijeme te status motora hidraulične jedinice. To su neke od glavnih vrijednosti koje treba nadzirati tijekom upravljanja s prešom, u slučaju da se javi potreba za detaljnim uvidom u stanje preše, pregled se mora obaviti putem HMI-a.

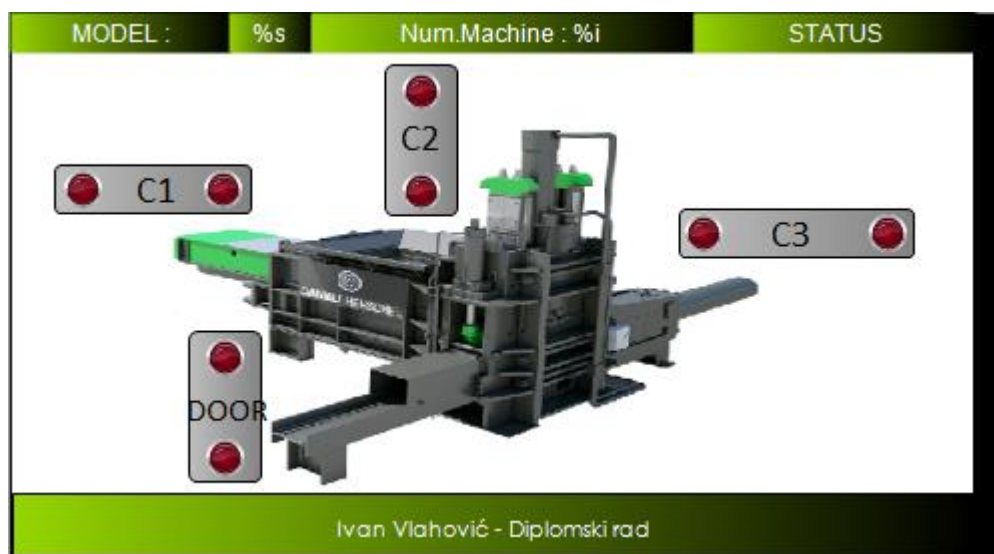


Slika 6.5. Stranica procesnih varijabli na radio odašiljaču

Ukoliko je neko od alarmnih stanja aktivno pripadajuća oznaka će biti crvene boje, upozoravajući na aktivno vanredno stanje kako je prikazano slikom 6.6.

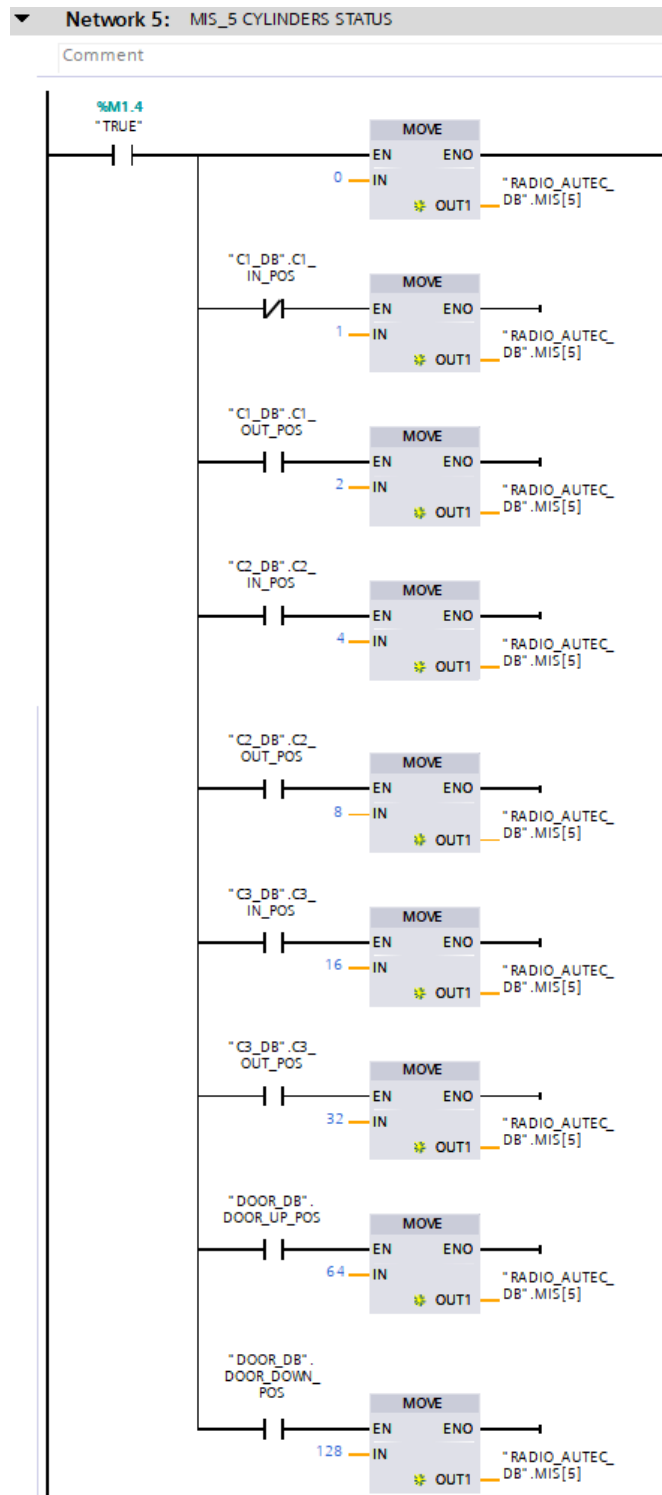


Slika 6.6. Stranica alarma na radio odašiljaču



Slika 6.7. Stranica pregleda preše unutar CODESYS-a

Program kojim se upravlja prikazom različitih statusa te same stranice zaslona radio odašiljača (kao što je stranica prikazana slikom 6.7.) je izrađen unutar CODESYS-a, a dok sam predložak aplikacije unutar CODESYS nudi proizvođač radio jedinice. Statusi preše se šalju preko MIS varijabli, te na slici 6.8. je prikazano slanje statusa pozicija cilindara.



Slika 6.8. Slanje statusa cilindara prema radio odašiljaču



Zatim se na strani odašiljača vrši provjera po bitovima MIS\_5, kako je prikazano slikom 6.9. Jer način definiranja statusa unutar TIA Portal-a upravo odgovara postavljanju određenog bit-a unutar varijable MIS\_5 (*byte*) u vrijednost 1.

```
MIS[2,3]:=MIS_5 ;  
LED_CI_INFO_REMOTE_1:=MIS[2,3].0 ;  
LED_CI_INFO_REMOTE_2:= MIS[2,3].1;  
LED_CII_INFO_REMOTE_1:=MIS[2,3].2;  
LED_CII_INFO_REMOTE_2:=MIS[2,3].3;  
LED_CIII_INFO_REMOTE_1:=MIS[2,3].4;  
LED_CIII_INFO_REMOTE_2:=MIS[2,3].5;  
LED_DOOR_INFO_REMOTE_1:=MIS[2,3].6;  
LED_DOOR_INFO_REMOTE_2:=MIS[2,3].7;
```

*Slika 6.9. Statusi cilindara unutar CODESYS-a*

Radio jedinica koja se sastoji od radio odašiljača i radio prijmnika te je oboje prikazano na slici 6.10. Prijamnik se obično postavlja u blizini ili unutar samog električnog ormara unutar kojeg se nalazi PLC-a s kojim komunicira. Na slici s lijeve strane prijmnika se vidi antena, dok pored nje priključeni EntherNet kabel s M8 4 pinovnim konektorom, koji je na PLC postavljen na port X3.



*Slika 6.10. Radio jedinica*

## 7. Vizualizacija

Vizualizacija predstavlja grafički prikaz stanja sustava hidraulične preše, koji je vidljiv pomoću HMI-a ili SCADA sustava. HMI panel se obično postavlja na električni ormar u radnom okruženju hidraulične preše te omogućuje nadzor i upravljanje prešom. Dok kod SCADA sustava, obično se prikazuje cijelo postrojenje sa svim podacima pojedinačnih sustava. Pa primjerice ukoliko se nakon hidraulične preše nalazi *reshredder* i/ili *shredder*, svi podaci svih strojeva bi bili vidljivi na jednom mjestu, obično unutar centralnih prostorija postrojenja.

Vizualizacija za potrebe ovoga rada izrađena je unutar programskog paketa AVEVA InTouch, koji je također poznat i po svom prethodnom imenu Wonderware. Aveva InTouch paket softverskih programa za HMI vizualizaciju se sastoji od nekoliko programa. U okviru ovog rada korišteni su InTouch HMI Application Manager, WindowMaker, WindowViewer te System Platform Management Console. HMI Application Manager koristi se za stvaranje novih HMI aplikacija te omogućuje rad na više aplikacija, WindowMaker služi za izradu stranica vizualizacije i skripti te definiranje radnji koje se obavljaju na tim stranicama. WindowViewer omogućuje simulaciju vizualizacije, dok System Platform Management Console omogućuje uspostavljanje komunikacije s PLC-om.

## 7.1. LANDEF

LANDEF (*LAN DEFinition*) je Microsoft Excel datoteka koja se sastoji od niza *sheets*-a (tablica), unutar kojih se definiraju podaci koji se razmjenjuju između PLC-a i sustava vizualizacije putem komunikacijskog protokola.

Za potrebe ovog rada, podaci unutar LANDEF-a su organizirani u nekoliko grupa:

- Komande
- Alarmi
- Statusi
- Dopuštenja za rad (*permissives*)
- Procesne varijable

Komande se šalju s HMI-a prema PLC-u, dok ostali podaci putuju sa PLC-a prema HMI-u. Na taj način je ostvarena komunikacija između sustava nadzora i sustava upravljanja hidraulične preše. U Prilogu 3 priložen je jedan *sheet* iz LANDEF-a s definicijom statusnih varijabli.

Prilikom početne definicije varijabli unutar LANDEF-a potrebno je ostaviti dovoljno rezervnih mjesta unutar svake podgrupe, kako bi se tijekom rada moglo dodati novi podaci ukoliko bude bilo potrebno. Iako je to prvotno učinjeno, s razvojem projekta i dodavanjem značajnog broja novih varijabli popunila su se rezervna mjesta unutar nekoliko podgrupa, što je rezultiralo dodavanjem nove AUX podgrupe.

Nakon izrade pojedinih *sheet*-ova unutar LANDEF tablice, podaci su eksportirani pomoću posebnih *export* alata koji su na korištenje dobiveni od strane tvrtke Danieli Systec. Ovi *export*-irani podaci su prilagođeni za uvoz unutar TIA Portal-a i InTouch-a. Za potrebe TIA Portala iz LANDEF-a generirane su .AWL datoteke koje se uvoze unutar TIA Portal-a pod *External source files* nakon čega se generiraju odgovarajući blokovi podataka. Generirani blokovi podataka su opisane u poglavlju 5.3. Uvoz tag-ova unutar InTouch-a provodi se putem programa Application Manager korištenjem DB Load. funkcije.



## 7.2. Izrada stranica vizualizacije

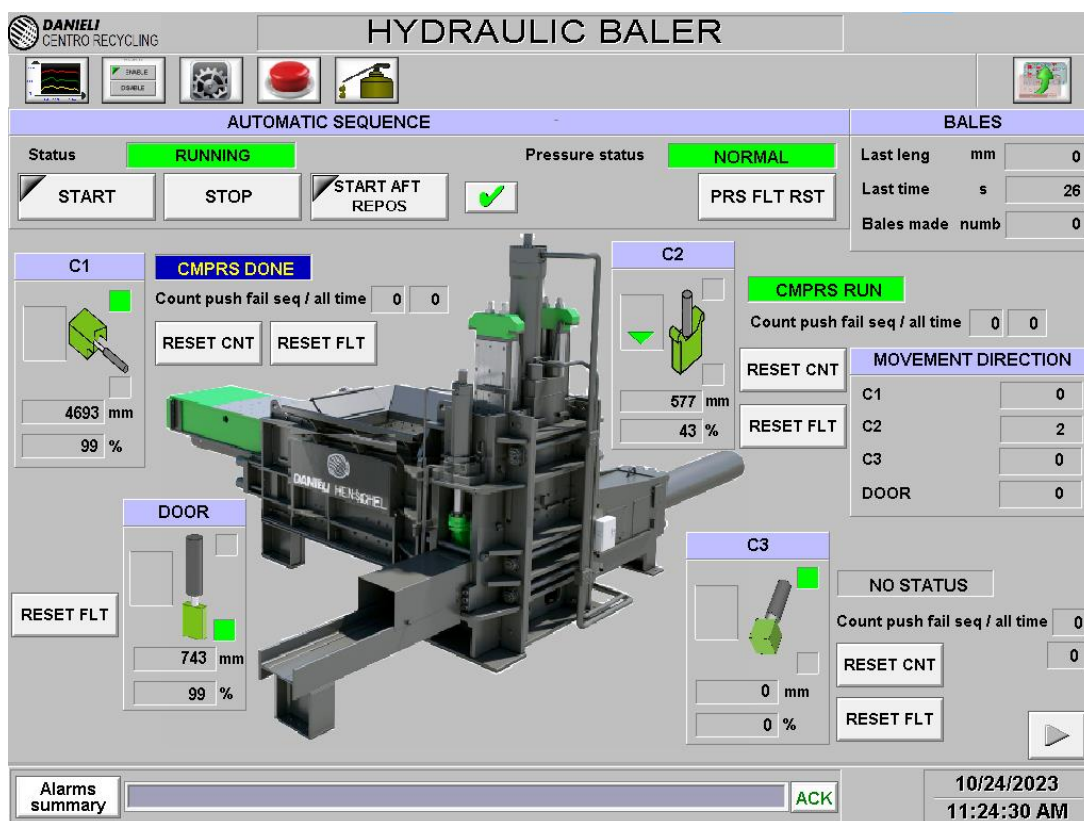
Pripadajuća vizualizacija hidraulične preše podijeljena je na nekoliko stranica koje prikazuju funkcionalnosti cjeline iste, pa tako se razlikuju stranice općenitog pregleda stanja preše, stranica ručnih komandi, stranica hidraulične jedinice, stranica *machine selftesta*, stranica s grafovima, *Emergency* stranica te stranica za pregled povijesti alarma.

Na svakoj od stranica se prikazuje stanje sustava pomoću različitih statusnih i numeričkih vrijednosti. Primjerice pozicije cilindara, status automatske sekvence, duljina zadnje izrađene bale, vrijeme trajanja sekvence i dopuštenja za rad. Povrh toga, na stranicama se nalaze komande za ručno upravljanje prešom, pokretanje automatske i *home* sekvence te selekciju pumpi. Stranice s grafovima prikazuju brzine pojedinog cilindra tijekom *machine selftest-a*, pozicije hidrauličkih cilindara, tlak ulja unutar sustava te protok za pojedinu hidrauličnu grupu.

Na svakoj od stranica gornja traka služi za prelazak između stranica, dok donja traka daje na uvid zadnje aktivni alarm uz mogućnost potvrde istog, te otvaranje stranice povijesnog pregleda svih alarma s navedenim vremenom pojavljivanja te opisom pojedinog alarma.

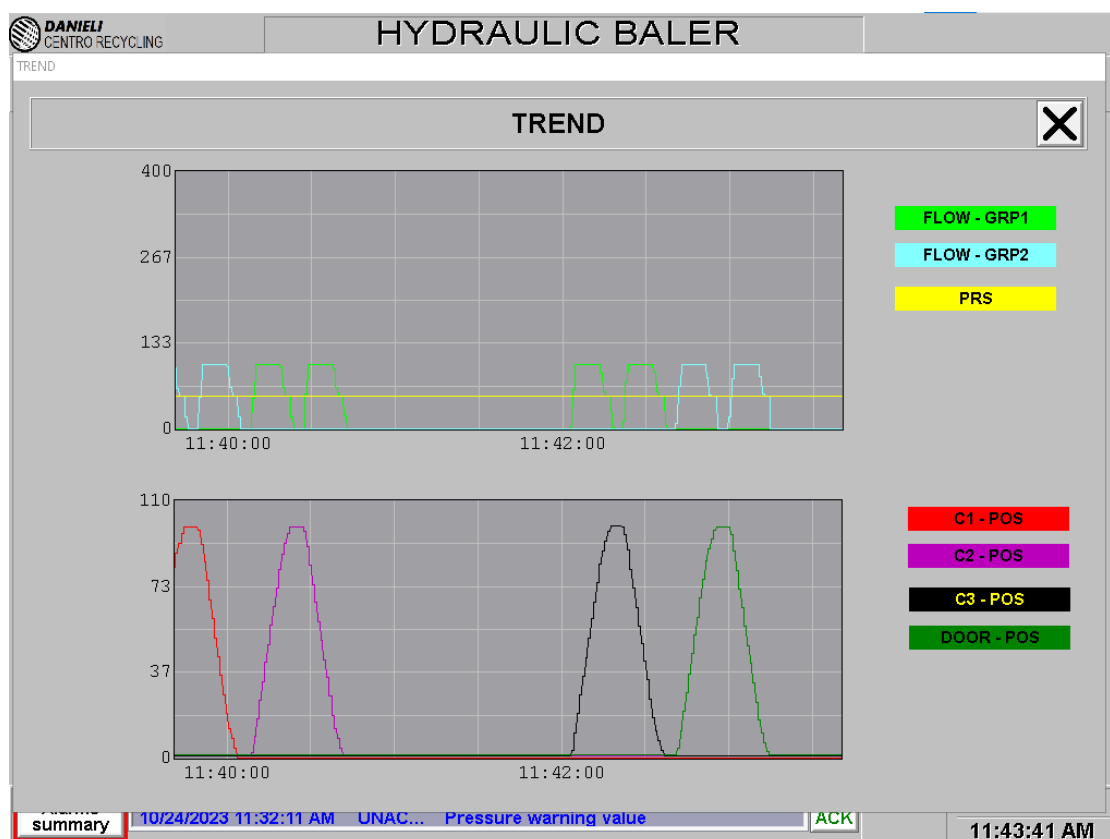
### 7.3. Izrađene stranice vizualizacije

Na slici 7.1. prikazana je početna stranica vizualizacija, koja se prikazuje nakon uspješne prijave. Inače su prisutne različite mogućnosti prijave, odnosno različite razine pristupa (razina operatera, razina konfiguracije, razina sistemskog inženjera) koje nude različite mogućnosti. Viša razina omogućuje pristup nekim funkcionalnostima koje nisu dostupne na nižim razinama prijave. Time se ostvaruje zaštita opreme od neovlaštene upotrebe. Međutim, u svrhu ovoga rada koristi se samo jedna razina prijave koja omogućuje pristup svim funkcionalnostima. Prikazano je stanje automatske sekvence s komandama za pokretanje i zaustavljanje iste. Automatska sekvenca je aktivna što se osim status „RUNNING“ može prepoznati po odrađenoj kompresiji cilindra C1 te aktivnoj kompresiji C2. Također kod statusa cilindra se nalaze kućice koje postaju aktivne (zeleno) kada je cilindar u početnom ili krajnjem položaju. Konkretno, C1 je u krajnjem, C3 u početnom položaju dok cilindar C2 izvodi kompresiju što se može primijetiti po smjeru njegovog pomaka prema dolje. Dopuštenja za rad automatske sekvence su sva zadovoljavajuća, očituje se putem zelene kvačice unutar kućice. Pritiskom na kućicu *permissives* otvara se prozor (*pop-up*) na kojem se mogu vidjeti statusi svih dopuštenja za rad.



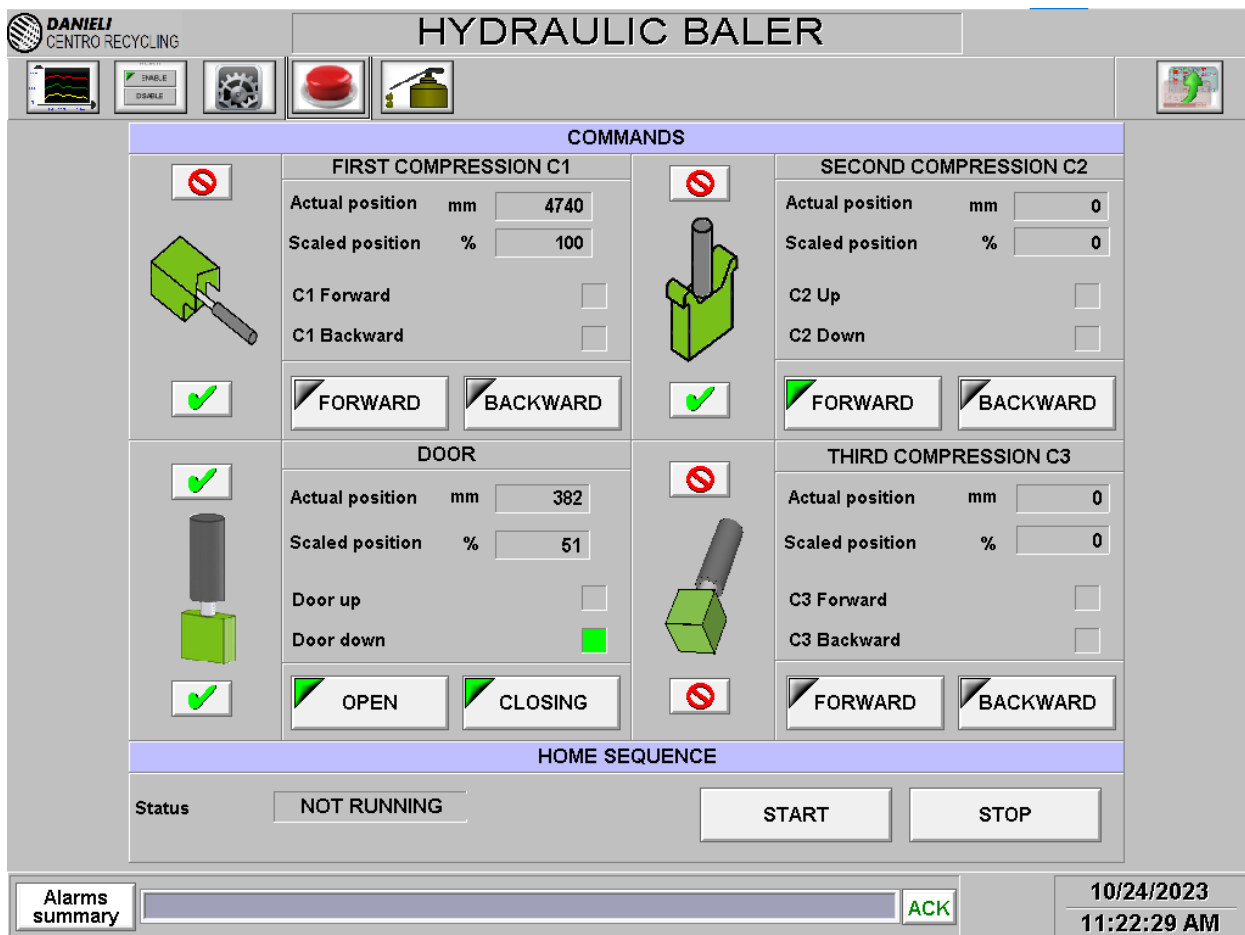
Slika 7.1. Stranica općeg pregleda hidraulične preše

Na ovoj stanici također je moguće pregledati status tlaka ulja unutar sustava, vrijednosti vezane uz posljednu izprešanu balu, broj neuspješnih pokušaja prešanja pojedinog cilindra(u sekvenci i ukupno), te smjer aktivnog pokreta. Definirano je da broj 1 označava repositioniranje, broj 2 kompresiju ,broj 3 označava povratak nakon neuspješne kompresije, te samo kod cilindra C3 broj 4 označava *ejecting*. Za prelazak na druge stranice potrebno je pritisnuti ikonu na gornjoj traci. Pa tako pritiskom na ikonu grafa otvara se trend stranica prikazana slikom 7.2. Trendovi prikazuju vrijednosti protoka, tlaka ulja te pozicije svih cilindara unutar dva grafa.



Slika 7.2. Stranica trenda

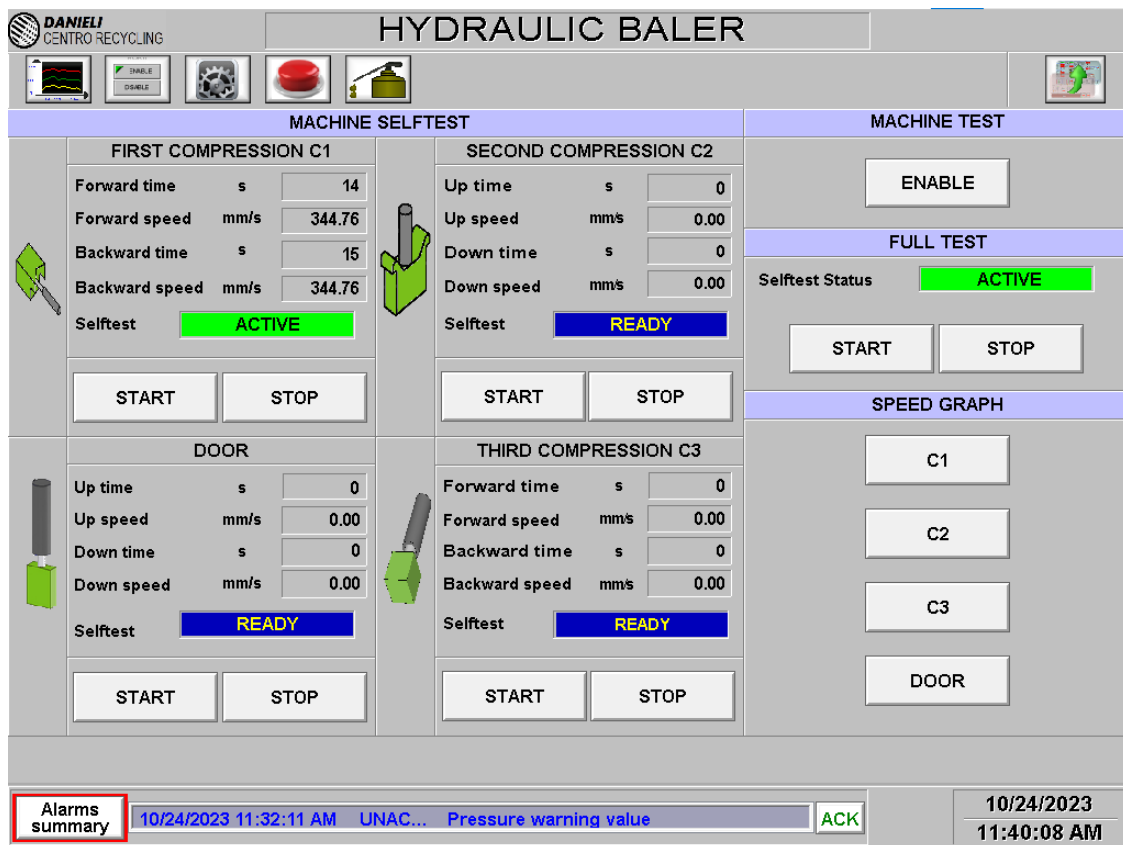
Pritiskom na ikonu „Enable/Disable“ otvara se stranica ručnih komandi (slika 7.3.),pomoću kojih je omogućeno ručno upravljanje hidrauličnim cilindrima preše. Osim samih komandi, uz njih nalaze se polja koja prikazuju je li neki od pokreta moguće inicirati. Ako je, polje unutar trokuta označeno je zelenom bojom. U ovom slučaju, vidi se ručno pomicanje vrata prema dolje, što onemogućuje upravljanje s cilindrom C1, budući da pripadaju istoj hidrauličnoj grupi. Dok C3 nije moguće pomicati jer mu dopuštenja za rad nisu zadovoljena, što je posljedica pomicanja vrata pa postoji opasnost od sudara ta dva cilindra. Osim komandi za ručno upravljanje, dostupne su i komande za start/stop „home“ sekvence koja dovodi prešu u potpuno „otvoreno“ stanje.



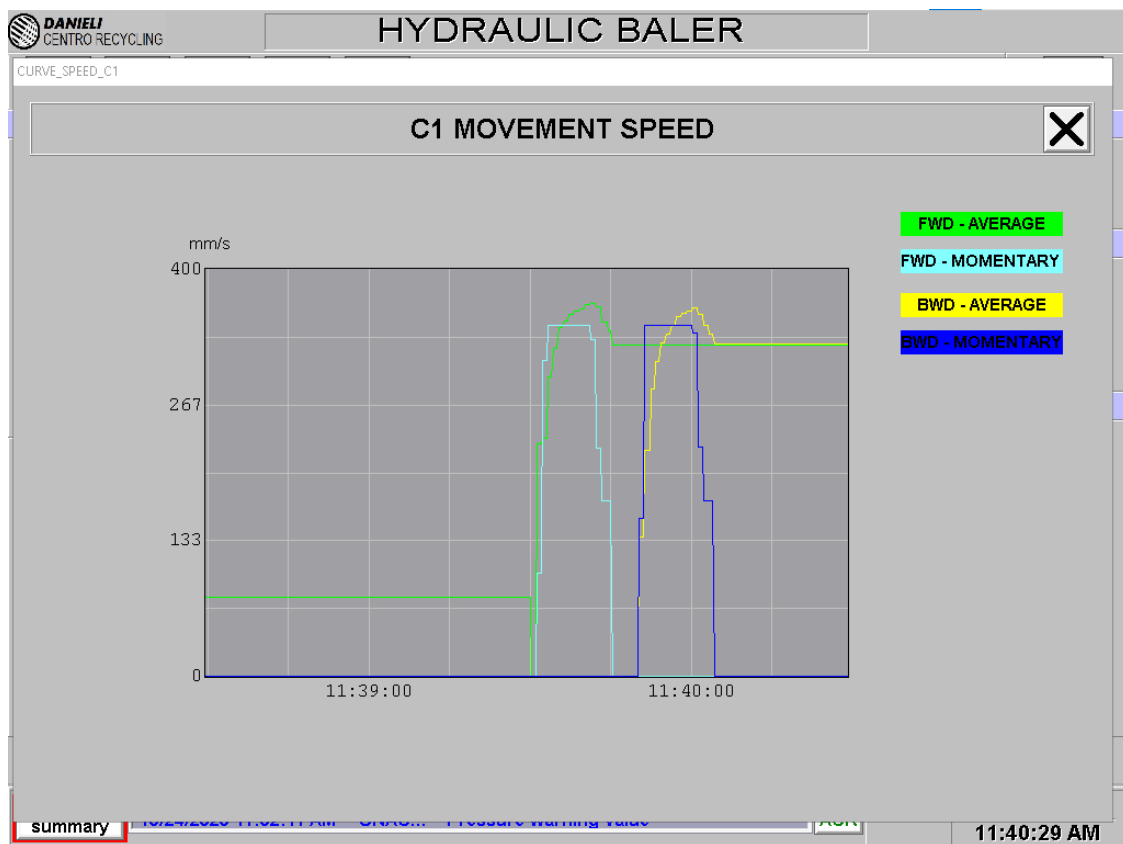
Slika 7.3. Stranica ručnih komandi

Stranica *machine selftest*-a prikazana slikom 7.4. se na HMI-u otvara pomoću ikone zupčanika. Za pokretanje samog testa potrebno ga je omogućiti (tipkalom „Enable“) nakon čega slijedi odabir ili kompletnog testa ili testa pojedinog cilindra. Na slici se vidi odabrani *full test*, te da se trenutno izvodi za cilindar C1, dok su ostali spremni za izvođenje testa. Prikazuju se vremena za oba moguća poketa svih cilindara uz brzinu koja je ostvarena u testu. Tijekom testa vrijednost brzine se mijenja prema trenutnoj vrijednosti, a kada se test za cilindar završi prikazuje se maksimalna ostvarena brzina. Uz to moguće je i uvid u graf brzine za svaki cilindar pomoću odgovarajuće tipke.

Na slici 7.4. prikazan je graf brzina (prosječne i trenutne) tijekom pokreta cilindra C1 prema naprijed te pokreta unatrag.

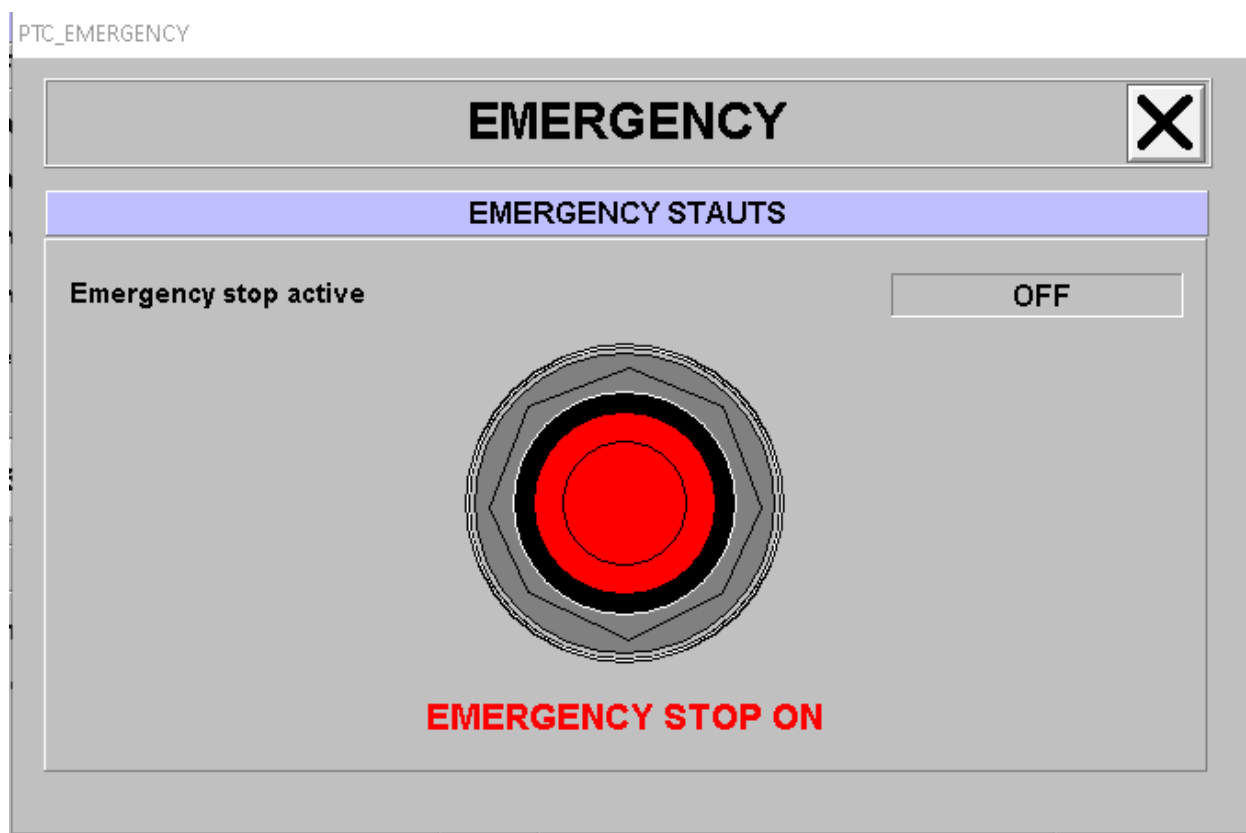


Slika 7.4. Stranica machine selftesta



Slika 7.5. Graf brzina pokreta cilindra C1 tijekom machine selftest-a

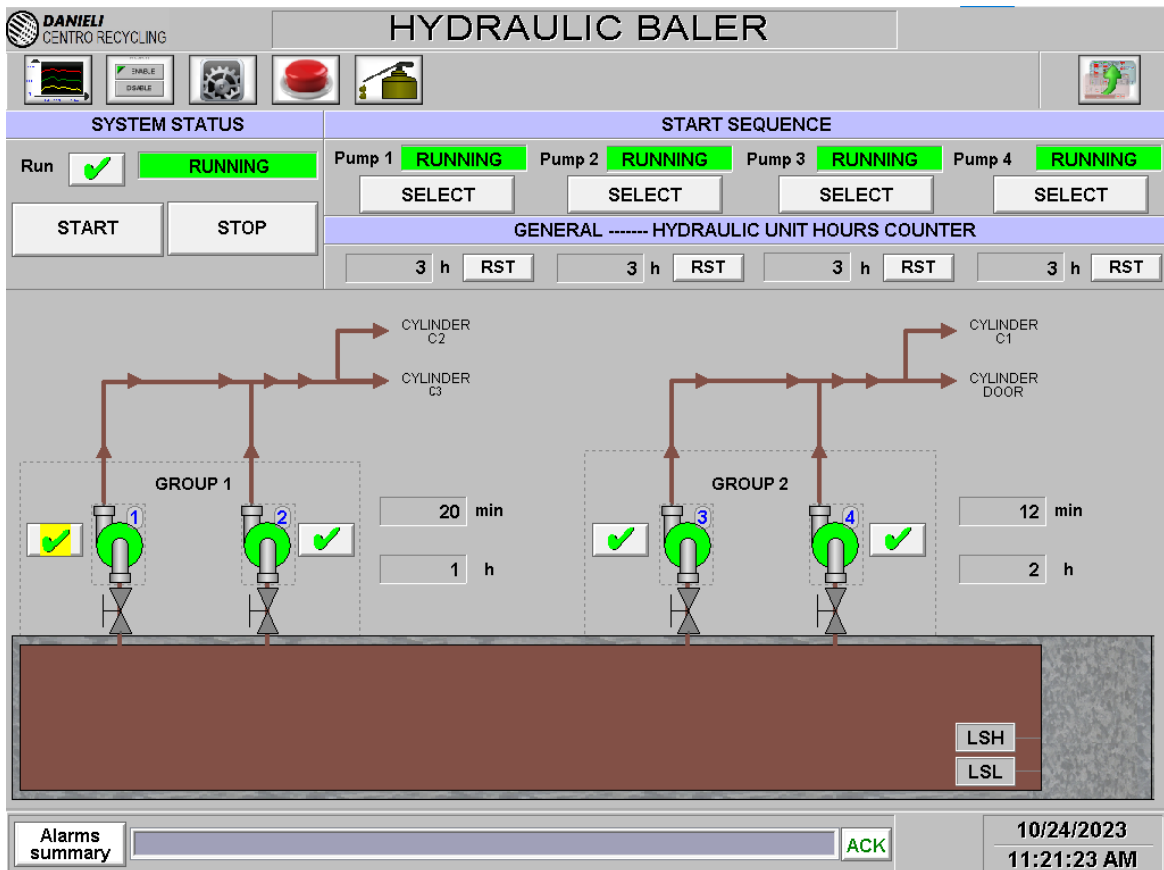
Emergency stranica prikazana na slici 7.6. omogućuje preko HMI-a aktivaciju alarmnog stanja koje zaustavlja sve aktivne procese preše.



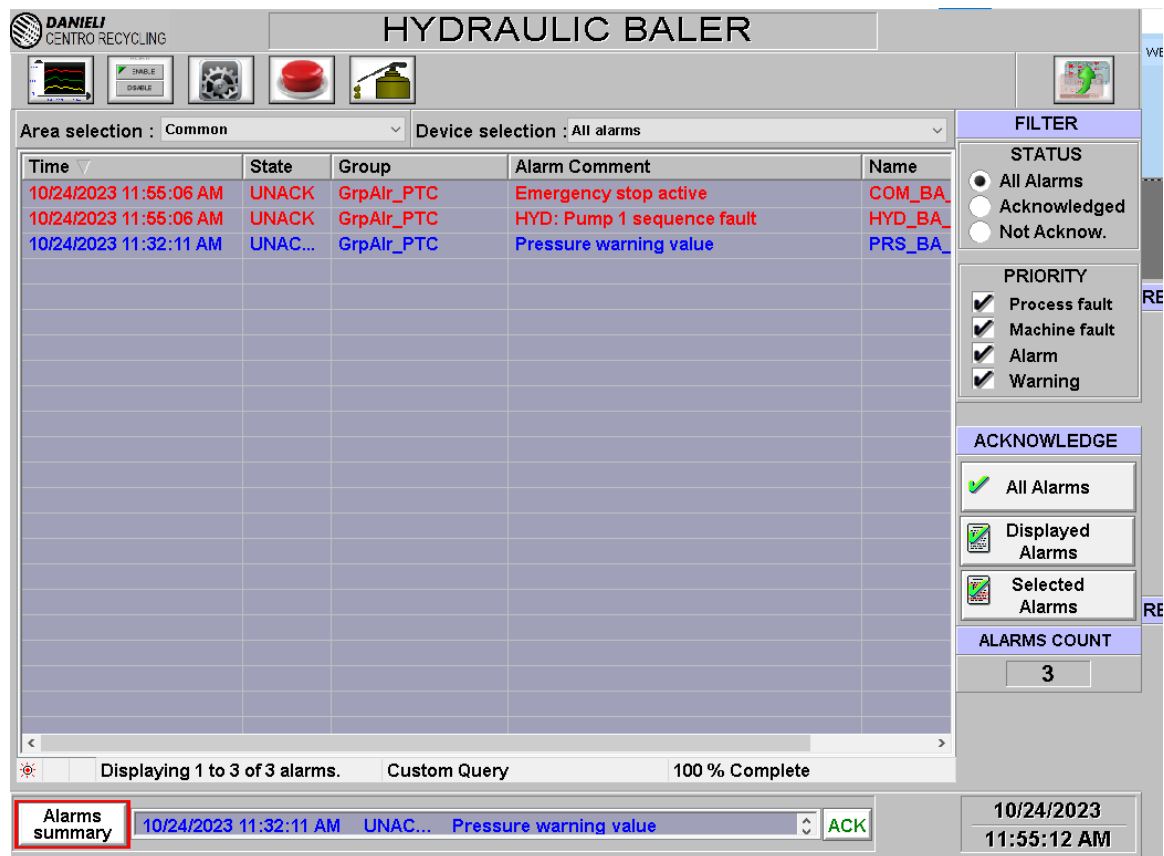
Slika 7.6. Emergency stranica

Slika 7.7. prikazuje stranicu hidraulične jedinice s svim potrebnim informacijama o stanju hidraulične jedinice te pojedinačnih pumpi. Omogućena je selekcija pumpi prije pokretanje rada hidraulične jedinice. Prikazani su statusi te vrijeme rada svake pumpe, kao i grupe koja se sastoji od 2 pumpe.

Na slici 7.8. prikazana je alarmna stranica koja prikazuje 3 alarma. Od toga su dva (crvena) aktivna te jedan (plavi) neaktivan. Neaktivan alarm znači da je alarmno stanje završeno. Moguća je potvrda pojedinačnog ili skupine alarma, nakon čega potvrđeni alarmi postaje sive boje. Alarm se prikazuje s vremenom njegovog pojavljivanja, stanjem (potvrđeno/nepotvrđeno), pripadnosti grupi alarma , opisom istog te oznakom *tag*-a.



Slika 7.7. Stranica hidraulične jedinice



Slika 7.8. Stranica alarma

## 8. Zaključak

Ovaj rad opisuje rad hidraulične preše kontrolirane putem PLC-a. Prvi dio rada upravo detaljnije opisuje sam PLC, uključujući njegovu strukturu te princip rada, s posebnim osvrtom na tip PLC-a korištenog za potrebe rada.

Hidraulična preša sa hidrauličnom jedinicom opisana u radu je model PTC (*engl. Press triple compression*) te je opisana u posebnom poglavlju. Unutar poglavlja je opisana primjena preše te način rada, prikazan je izgled same preše te izlazne bale nakon prešanja metalnog otpada. Također, prikazana je tehnička specifikacija PTC hidraulične preše.

Upravljački algoritam za upravljanje rada preše je razvijen u okruženje TIA Portal-a, a neki od najznačajnijih dijelova istog prikazani i opisani su unutar posebnom poglavlju. Posebna pozornost posvećena je automatskoj sekvenci rada preše, koja je izrađena u vidu *Graph*-a. Automatska sekvenca slijedi niz koraka ovisno o tijeku procesa prešanja metalnog otpada. Osim automatskog načina rada omogućen je ručni način, bilo putem HMI-a ili putem radio upravljačke jedinice. Oba načina ručnog upravljanja su opisana u posebnim poglavljima.

Sustav vizualizacije koristi se za nadzor stanja hidraulične preše i omogućuje operateru različite mogućnosti upravljanja iste. Također putem HMI-a, operatera se obavještava u slučaju alarmnih stanja za koja neka od njih postoji potreba za njegovom intervencijom. Sam sustav vizualizacije je izrađen unutar InTouch-a, unutar kojega je izrađeno nekoliko stranica koje prikazuju sve potrebne informacije o stanju preše te nude komande za upravljanje istom.



## 9. Literatura

- [1] Bolton W.: “Programmable Logic Controllers”, Elsevier Newnes , 2006.
- [2] ADJ , “S7-1500 Central processing units CPU 1518-4PN/DP CPU 1515-2 PN of SIEMENS”, s Interneta, <https://adjditec.com/es/noticias/s7-1500-central-processing-units-cpu-1518-4pn-dp-cpu-1515-2-pn-of-siemens>, 09.listopada 2023.
- [3] Kolegij : “Elementi automatizacije postrojenja – Uvod u PLC” , Tehnički fakultet Sveučilišta u Rijeci, 2020
- [4] “Uvod u programibilne logičke kontrolere”, s Interneta, [http://www.mikroknjiga.rs/Knjige/PLC/01\\_PLC.pdf](http://www.mikroknjiga.rs/Knjige/PLC/01_PLC.pdf) , 05.listopada .2023.
- [5] Automation Direct, “PLC Handbook”
- [6] FESTO, “ Programmable logic controllers”
- [7] Automation Direct, “Sinking and Sourcing Concepts”, s Interneta, [Sinking and Sourcing for the PLC Explained \(automationdirect.com\)](https://www.automationdirect.com/usa/learning-center/sinking-and-sourcing-for-the-plc-explained), 06.listopada 2023
- [8] Siemens, “Basics of PLCs“
- [9] SIMATIC, “ SM 1238 Energy Meter 480VAC”
- [10] REALPARS, “How a PLC power supply works”, s Interneta, [https://realpars.com/plc\\_power\\_supply/](https://realpars.com/plc_power_supply/) , 07.listopada 2023
- [11] Temposonics, “Temposonics Techonology”, s Interneta, <https://www.temposonics.com/Products/Technology/TemposonicsTechnology>, 15. listopada 2023

## 10. Sažetak i ključne riječi

Hidraulična preša se sastoji od 4 cilindra pomoću kojih se obavlja prešanje metalnog otpada u tri osi, čime se metalni otpad kompresira u obliku bale. Upravljačka logika rada hidraulične preše izrađena je pomoću SIEMENS-ovog PLC-a programiranog unutar TIA Portala. Za pomicanje cilindra simulirani su proporcionalni ventili, pomoću kojih se znatno smanjuju mehanički trzaji i udarci tijekom pokretanja i zaustavljanja cilindra, u usporedbi sa ON/OFF ventilima. Uspoređena su tri načina automatskog pozicioniranja cilindra u željenu poziciju. I to gibanje cilindra sporom brzinom (50% maksimalne brzine) čime se ostvaruje precizno pozicioniranje no potreban je dugi vremenski period za isto, gibanje maksimalnom brzinom čime se cilindar uz kratko vrijeme pozicionira sa većom pogreškom nego u prethodnom slučaju. Te kombinacija dva moda čime se ostvaruju prednosti oba, odnosno dobra preciznost u kratkom vremenskom periodu. Hidraulična preša razlikuje automatski i ručni način rada. Za nadzor i ručno upravljanje rada preše koristi se sustav vizualizacije (HMI) izrađen u AVEVA InTouch-u. Što omogućuje nadziranje i upravljanje različitim dijelova same preše poput pojedinih cilindra, *machine selftest*-a i alarmnih stanja. Osim toga, ručno upravljanje prešom moguće je i putem radio upravljačke jedinice.

**Ključne riječi:** PLC, HMI, automatizacija, hidraulična preša, metalni otpad

## 11. Summary and key words

The hydraulic baler consists of four cylinders used for compacting metal waste in three axes, compressing it into bales. The control logic of the hydraulic baler is implemented using SIEMENS PLC programmed within the TIA Portal. Proportional valves are simulated in order to move the cylinders, significantly reducing mechanical jolts and impacts during starts and stops of cylinders movements compared to ON/OFF valves. Three automatic cylinder positioning modes were compared, slow speed movement (with 50% maximum speed) for precise positioning but requiring a longer time, maximum speed movement for quicker cylinder positioning with a higher error in position compared to the previous mode. And a combination of both modes to leverage the advantages of each, achieving good precision in a shorter time frame. The hydraulic baler distinguishes between automatic and manual operation modes. A visualisation system (HMI) created in AVEVA InTouch is used for monitoring and it enables manual control of various components of the baler, including individual cylinders, machine selftest and alarms statues. Additonaly, manual control of the baler is possible through a radio control unit.

**Key words:** PLC, HMI, automation, hydraulic baler, metal waste

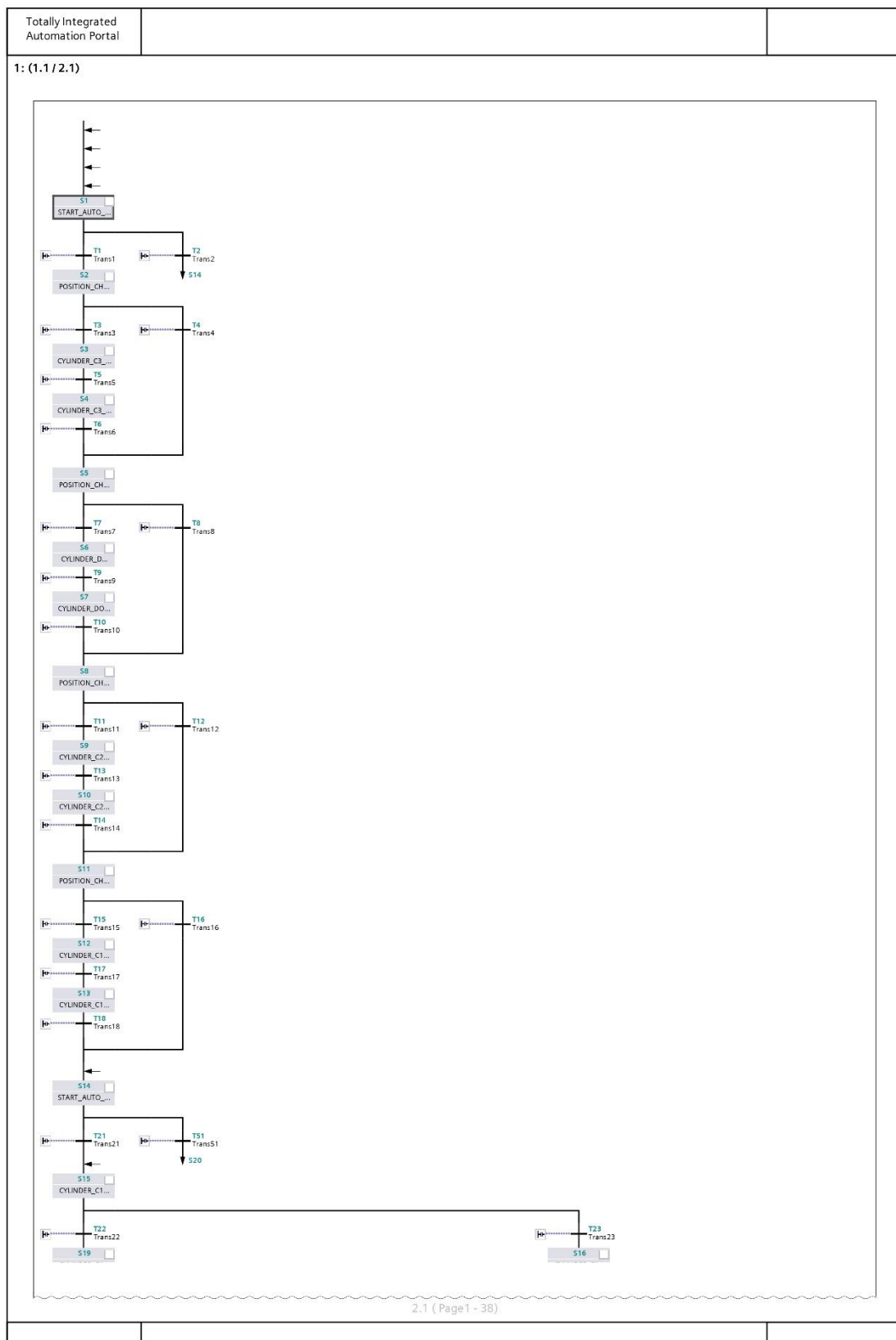
## 12. Popis slika i tablica

Slika 3.1. PLC [2].....	4
Slika 3.2. Relejni ormar [3].....	5
Slika 3.3. Arhitektura CPU-a [6].....	7
Slika 3.4. Pregled segmenata snage .....	10
Slika 3.5. Prikaz PLC-a CPU 1518F -4 PN/DP .....	15
Slika 4.1. PTC hidraulična preša s hidrauličnom jedinicom .....	16
Slika 4.2. PTC hidraulična preša .....	17
Slika 4.3. Bala sprešanog metalnog otpada.....	17
Slika 4.4. Hidraulična jedinica .....	19
Slika 4.5. Temposonic senzor unutar cilindra C3 .....	21
Slika 5.1. Project tree unutar TIA Portal-a.....	22
Slika 5.2. Automatska sekvenca prvi dio .....	23
Slika 5.3. Prvi korak automatske sekvence .....	24
Slika 5.4. Drugi korak automatske sekvence .....	25
Slika 5.5. Treći korak automatske sekvence – repozicioniranje cilindra C3 .....	26
Slika 5.6. Četvrti korak automatske sekvence – završeno repozicioniranje cilindra C3 .....	26
Slika 5.7. Automatska sekvenca drugi dio .....	27
Slika 5.8. Četrnaesti korak automatske sekvence – start ostatka sekvence nakon repozicioniranja ili ručnog rada.....	28
Slika 5.9. Automatska sekvenca treći dio .....	29
Slika 5.10. Petnaesti korak automatske sekvence – prešanje cilindrom C1.....	30
Slika 5.11. Šesnaesti korak automatske sekvence – neuspješno prešanje cilindra C1.....	31
Slika 5.12. Sedamnaesti korak automatske sekvence – povratak cilindra C1 nakon neuspješnog prešanja.....	31
Slika 5.13. Automatska sekvenca četvrti dio .....	32
Slika 5.14. Trideseti korak automatske sekvence – depressure cilindra C1 .....	33
Slika 5.15. Tridesetprvi korak automatske sekvence – izvršen depressure C1 .....	34
Slika 5.16. Tridesetčetvrti korak automatske sekvence – pomak vrata gore .....	34
Slika 5.17. Tridesetosmi korak automatske sekvence – pomak C3 to eject pozicije.....	35
Slika 5.18. „Home“ sekvenca.....	36
Slika 5.19. Ulazno/izlazne varijable DB-a automatske sekvence .....	37
Slika 5.20. Mogućnosti pokretanje automatske sekvence.....	38

Slika 5.21. Izračun duljine bale .....	39
Slika 5.22. Mapiranje varijabli automatske sekvence unutar blokova za komunikaciju sa HMI-om .....	40
Slika 5.23. Zahtjev za pomak cilindra C1 unaprijed .....	41
Slika 5.24. Ručna komanda za pomak cilindra .....	42
Slika 5.25. Uvjeti za pomak cilindra 1 naprijed .....	43
Slika 5.26. Referenca pozicije i otvorenosti ventila za pomak C1 unaprijed.....	44
Slika 5.27. Referenca pozicije i otvorenosti ventila za pomak C1 unatrag.....	44
Slika 5.28. Referenciranje i aktiviranje hidraulične grupe dva za pomak cilindra C1 .....	45
Slika 5.29. Greške visoke razine pritiska .....	46
Slika 5.30. Blok podataka cilindra C1.....	47
Slika 5.31. Sekvenca machine selftest-a .....	49
Slika 5.32. Logika pokretanje pumpi 1 dio .....	50
Slika 5.33. Logika pokretanje pumpi 2 dio .....	51
Slika 5.34. Određivanje vrijednosti vremena i argumenta u slučaju visoke vrijednosti pritiska ulja .....	53
Slika 5.35. Simulacija pozicije .....	54
Slika 5.36. Simulacija automatske sekvence.....	55
Slika 5.37. Automatsko pozicioniranje cilindra s kretanjem low speed .....	56
Slika 5.38. Automatsko pozicioniranje cilindra s kretanjem high speed .....	57
Slika 5.39. Automatsko pozicioniranje cilindra s kombinacijom dvaju načina .....	57
Slika 6.1. Radio odašiljač .....	58
Slika 6.2. Oznake komandi radio odašiljača .....	59
Slika 6.3. Ulazne adrese po oznakama komandi .....	59
Slika 6.4. Ulazne adrese radio jedinice .....	61
Slika 6.5. Stranica procesnih varijabli na radio odašiljaču .....	61
Slika 6.6. Stranica alarma na radio odašiljaču .....	62
Slika 6.7. Stranica pregleda preše unutar CODESYS-a.....	62
Slika 6.8. Slanje statusa cilindara prema radio odašiljaču .....	63
Slika 6.9. Statusi cilindara unutar CODESYS-a .....	64
Slika 6.10. Radio jedinica .....	64
Slika 7.1. Stranica općeg pregleda hidraulične preše.....	68
Slika 7.2. Stranica trenda.....	69
Slika 7.3. Stranica ručnih komandi .....	70
Slika 7.4. Stranica machine selftesta.....	71

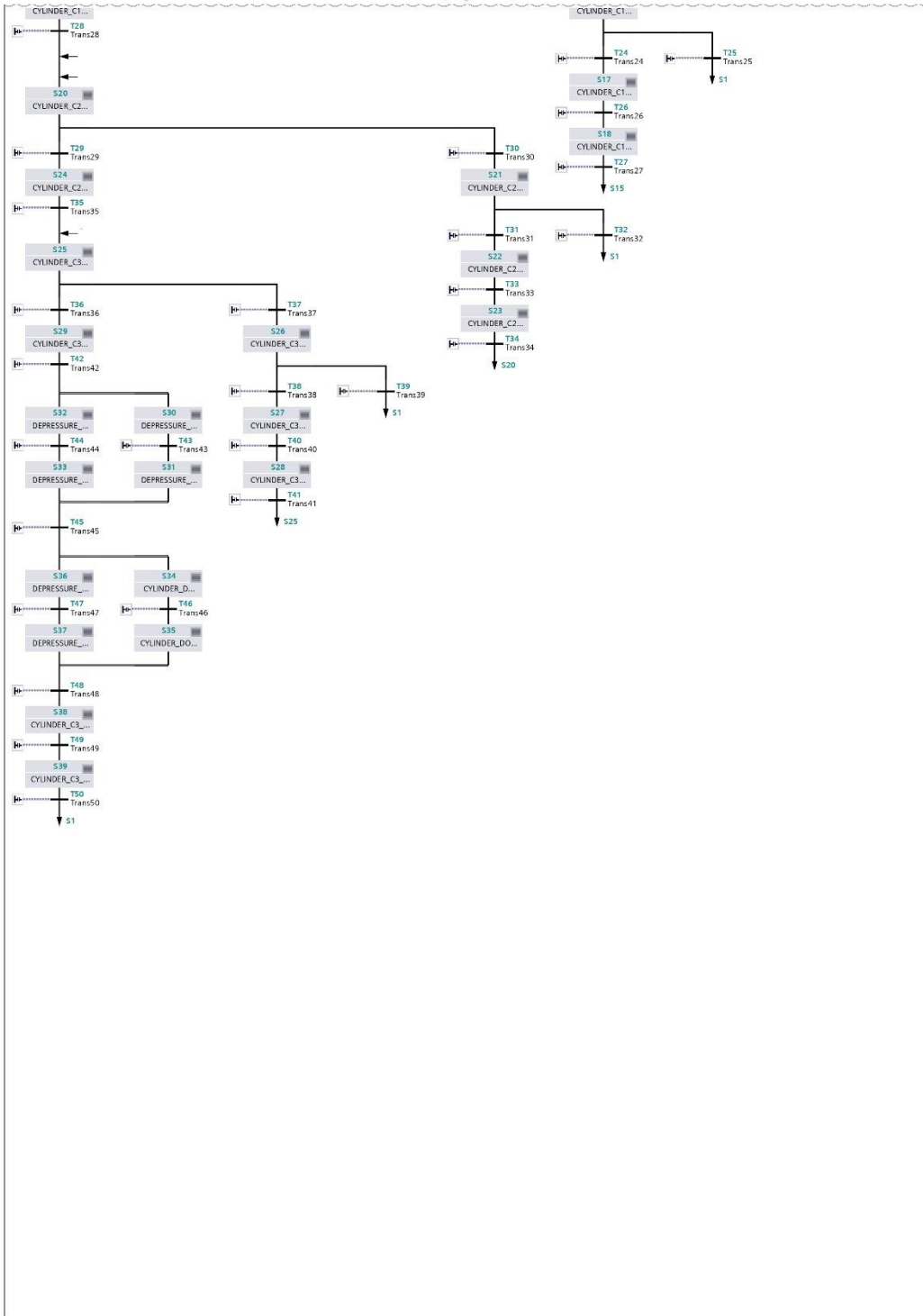
Slika 7.5. Graf brzina pokreta cilindra C1 tijekom machine selftest-a .....	71
Slika 7.6. Emergency stranica .....	72
Slika 7.7. Stranica hidraulične jedinice .....	73
Slika 7.8. Stranica alarma.....	73
Tablica 4.1. Slijed automatske sekvence.....	18
Tablica 4.2. Tehnička specifikacija hidraulične preše .....	20
Tablica 6.1. Oznake komandi radio jedinice.....	60

# Dodatak A. Automatska sekvenca



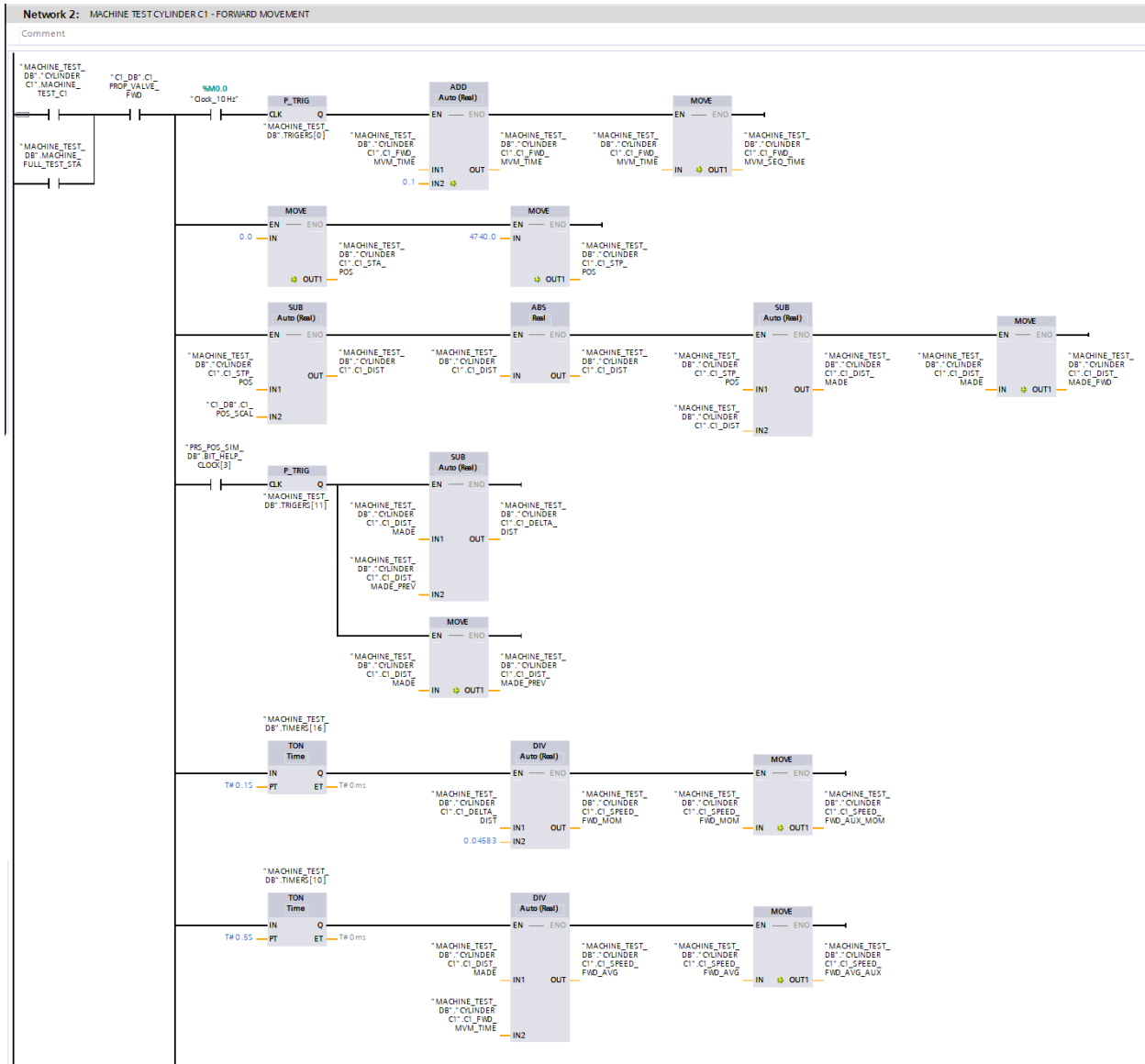
1: (2.1/2.1)

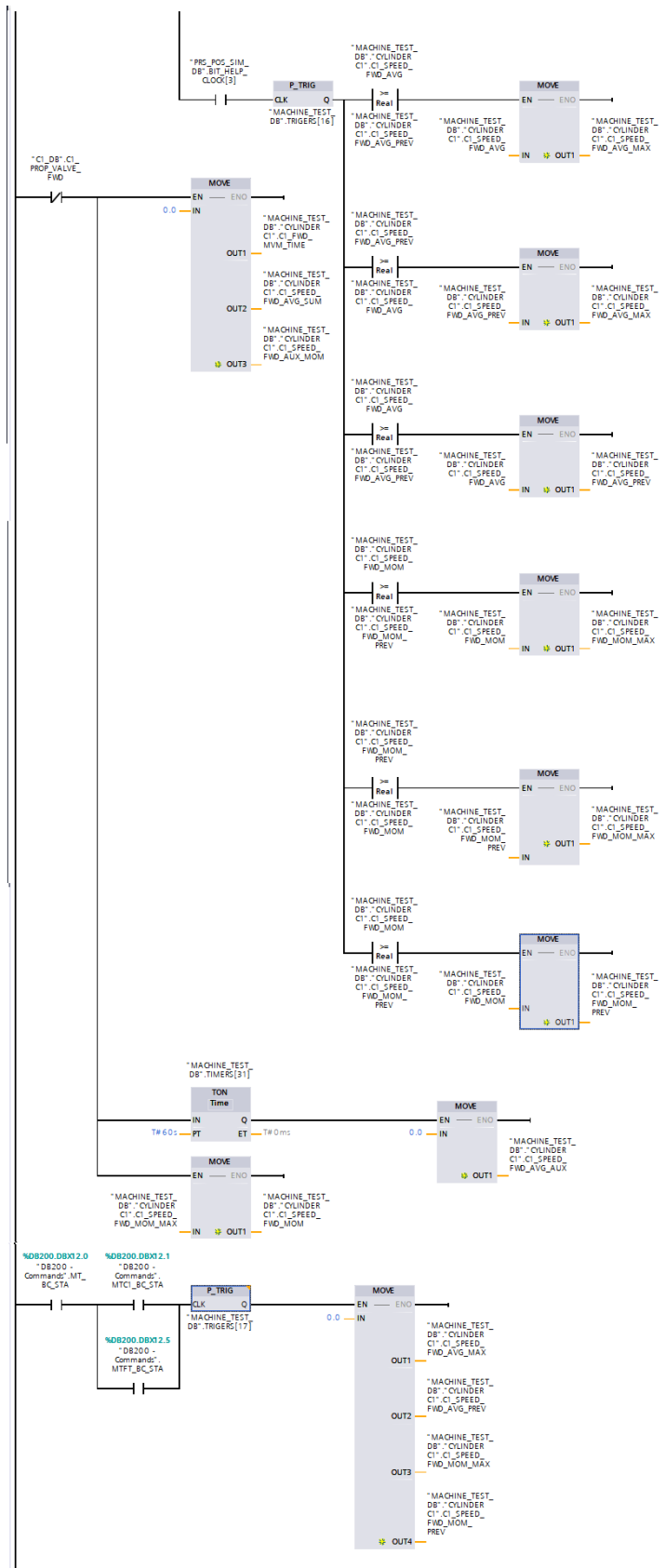
1.1 (Page1 - 37)





# Dodatak B. Funkcija machine selftest za pomak cilindra C1 unaprijed





## Dodatak C. LANDEF sheet statusnih varijabli

Message Source:	PTC_PLC	DB215
Message Destination:	PTC_HMI	HMI Database

Address	Tag	Description	Fmt	Range	Remarks	Rev	Ver	Properties
	<b>1</b>	<b>COMMON</b>						
DBW000	COM_WS_PLC_WDOG	PLC watch-dog	IN	- 32768..32767		00	A	
DBX002.0	COM_BS_HMI_WDOG_TOUT	Com.parts: HMI watchdog timeout	BB	0..1	1=timeout	00	A	
DBX002.1	COM_BS_GEN_ALM	Generic alarm	BB	0..1		00	A	
DBX002.2	COM_BS_TRP_ALR	Trip alarm	BB	0..1		00	A	
DBX002.3	COM_BS_ALR_ACK	Alarms acknowledge	BB	0..1	1=true	00	A	
DBX002.4	COM_BS_HN_ENAB	Horn enabled	BB	0..1	1=true	00	A	
DBX002.5	COM_BS_PASS_REQ	Password required	BB	0..1	1=true	00	A	
DBX002.6	COM_BS_PASS_OK	Password ok	BB	0..1		00	A	
DBX002.7	COM_BS_SAFE_PERM_OK	Safety permissive ok	BB	0..1		00	A	
DBX003.0	COM_BS_SET_MODE	Setting mode	BB	0..1		00	A	
DBX003.1	COM_BS_WORK_MODE	Working mode	BB	0..1		00	A	
DBX003.2	COM_BS_AUTO_SEQ_ON	Auto seq on	BB	0..1		00	A	
DBX003.3	COM_BS_AUTO_PERM_OK	Auto seq perm ok	BB	0..1		00	A	
DBX003.4	COM_BS_DP1_FLT1	Not used	BB	0..1		00	A	
DBX003.5	SPARE	Spare	BB	0..1		00	A	
DBX003.6	SPARE	Spare	BB	0..1		00	A	
DBX003.7	SPARE	Spare	BB	0..1		00	A	
DBX004.0	COM_BS_EMG_STP	Emergency stop	BB	0..1		00	A	
DBX004.1	COM_BS_CP1_EMG	Not used	BB	0..1		00	A	
DBX004.2	COM_BS_REM_EMG	Not used	BB	0..1		00	A	
DBX004.3	COM_BS_TC1_EMG	Not used	BB	0..1		00	A	
DBX004.4	COM_BS_CP2_EMG	Not used	BB	0..1		00	A	
DBX004.5	COM_BS_CP3_EMG	Not used	BB	0..1		00	A	
DBX004.6	SPARE	Spare	BB	0..1		00	A	
DBX004.7	SPARE	Spare	BB	0..1		00	A	
		<b>HYDRAULIC</b>						
DBX005.0	HYD_BS_P1_FLT	HYD: Pump 1 fault	BB	0..1		00	A	
DBX005.1	HYD_BS_P1_RUN	HYD: Pump 1 running	BB	0..1		00	A	
DBX005.2	HYD_BS_P1_SS_FLT	HYD: Pump 1 soft starter fault	BB	0..1		00	A	
DBX005.3	HYD_BS_SQ1_OPN	HYD: Pump 1 suction valve opened	BB	0..1		00	A	
DBX005.4	HYD_BS_P1_SEL	HYD: Pump 1 selected	BB	0..1		00	A	
DBX005.5	HYD_BS_P1_SEQ_RDY	HYD: Pump 1 sequence ready	BB	0..1		00	A	
DBX005.6	HYD_BS_P1_SEQ_RUN	HYD: Pump 1 sequence run	BB	0..1		00	A	
DBX005.7	HYD_BS_P1_SEQ_FLT	HYD: Pump 1 sequence fault	BB	0..1		00	A	
DBX006.0	HYD_BS_P2_FLT	HYD: Pump 2 fault	BB	0..1		00	A	
DBX006.1	HYD_BS_P2_RUN	HYD: Pump 2 running	BB	0..1		00	A	
DBX006.2	HYD_BS_P2_SS_FLT	HYD: Pump 2 soft starter fault	BB	0..1		00	A	
DBX006.3	HYD_BS_SQ2_OPN	HYD: Pump 2 suction valve opened	BB	0..1		00	A	
DBX006.4	HYD_BS_P2_SEL	HYD: Pump 2 selected	BB	0..1		00	A	
DBX006.5	HYD_BS_P2_SEQ_RDY	HYD: Pump 2 sequence ready	BB	0..1		00	A	
DBX006.6	HYD_BS_P2_SEQ_RUN	HYD: Pump 2 sequence run	BB	0..1		00	A	
DBX006.7	HYD_BS_P2_SEQ_FLT	HYD: Pump 2 sequence fault	BB	0..1		00	A	
DBX007.0	HYD_BS_P3_FLT	HYD: Pump 3 fault	BB	0..1		00	A	
DBX007.1	HYD_BS_P3_RUN	HYD: Pump 3 running	BB	0..1		00	A	
DBX007.2	HYD_BS_P3_SS_FLT	HYD: Pump 3 soft starter fault	BB	0..1		00	A	
DBX007.3	HYD_BS_SQ3_OPN	HYD: Pump 3 suction valve opened	BB	0..1		00	A	
DBX007.4	HYD_BS_P3_SEL	HYD: Pump 3 selected	BB	0..1		00	A	
DBX007.5	HYD_BS_P3_SEQ_RDY	HYD: Pump 3 sequence ready	BB	0..1		00	A	
DBX007.6	HYD_BS_P3_SEQ_RUN	HYD: Pump 3 sequence run	BB	0..1		00	A	
DBX007.7	HYD_BS_P3_SEQ_FLT	HYD: Pump 3 sequence fault	BB	0..1		00	A	
DBX008.0	HYD_BS_P4_FLT	HYD: Pump 4 fault	BB	0..1		00	A	
DBX008.1	HYD_BS_P4_RUN	HYD: Pump 4 running	BB	0..1		00	A	
DBX008.2	HYD_BS_P4_SS_FLT	HYD: Pump 4 soft starter fault	BB	0..1		00	A	
DBX008.3	HYD_BS_SQ4_OPN	HYD: Pump 4 suction valve opened	BB	0..1		00	A	
DBX008.4	HYD_BS_P4_SEL	HYD: Pump 4 selected	BB	0..1		00	A	
DBX008.5	HYD_BS_P4_SEQ_RDY	HYD: Pump 4 sequence ready	BB	0..1		00	A	
DBX008.6	HYD_BS_P4_SEQ_RUN	HYD: Pump 4 sequence run	BB	0..1		00	A	
DBX008.7	HYD_BS_P4_SEQ_FLT	HYD: Pump 4 sequence fault	BB	0..1		00	A	
DBX009.0	HYD_BS_RUN_SEQ_RDY	HYD : Run sequence ready	BB	0..1		00	A	
DBX009.1	HYD_BS_RUN_SEQ_RUN	HYD : Run sequence run	BB	0..1		00	A	
DBX009.2	HYD_BS_RUN_SEQ_FLT	HYD : Run sequence fault	BB	0..1		00	A	
DBX009.3	SPARE	Spare	BB	0..1		00	A	
DBX009.4	SPARE	Spare	BB	0..1		00	A	
DBX009.5	SPARE	Spare	BB	0..1		00	A	

DBX009.6	SPARE	Spare	BB	0.1		00	A
DBX009.7	SPARE	Spare	BB	0.1		00	A
		CYLINDER C1					
DBX010.0	C1_BS_POS_IN	Cylinder C1 position in (zero position)	BB	0.1		00	A
DBX010.1	C1_BS_POS_OUT	Cylinder C1 position out (end position)	BB	0.1		00	A
DBX010.2	C1_BS_DEPRESS_POS	Cylinder C1 depressure position	BB	0.1		00	A
DBX010.3	C1_BS_AUTO_SEQ_REPOS_IN	Cylinder C1 return request to in pos from auto seq reposition	BB	0.1		00	A
DBX010.4	C1_BS_HOME_SEQ_REPOS_IN	Cylinder C1 return request to in pos from home seq reposition	BB	0.1		00	A
DBX010.5	C1_BS_AUTO_SEQ_PUSH	Cylinder C1 push request from auto seq	BB	0.1		00	A
DBX010.6	C1_BS_AUTO_SEQ_RETURN_PUSH_FAIL	Cylinder C1 push failed from auto seq, return request	BB	0.1		00	A
DBX010.7	C1_BS_AUO_SEQ_DEPRESS	Cylinder C1 depressure request from auto seq	BB	0.1		00	A
DBX011.0	C1_BS_PUSH_FWD_MVM	Cylinder C1 push forward movement	BB	0.1		00	A
DBX011.1	C1_BS_RETURN_BWD_MVM	Cylinder C1 return backward movement	BB	0.1		00	A
DBX011.2	C1_BS_PUSH_FWD_MAN	Cylinder C1 manual request to push forward movement	BB	0.1		00	A
DBX011.3	C1_BS_RET_BWD_MAN	Cylinder C1 manual request to return backward movement	BB	0.1		00	A
DBX011.4	C1_BS_POS_ADJ	Cylinder C1 position adjusted	BB	0.1		00	A
DBX011.5	C1_BS_PRS_FLT_MAN	Cylinder C1 pressure fault in manual	BB	0.1		00	A
DBX011.6	C1_BS_AUTO_SEQ_PRS_FLT	Cylinder C1 pressure fault in auto sequence	BB	0.1		00	A
DBX011.7	C1_BS_AUTO_SEQ_PUSH_FAIL	Cylinder C1 push failed in auto sequence	BB	0.1		00	A
DBX012.0	C1_BS_AUTO_SEQ_3_FAIL_PUSH_FLT	Cylinder C1 fault after 3 failed pushes in auto sequence	BB	0.1		00	A
DBX012.1	C1_BS_MVM_REQ	Cylinder C1 movement request	BB	0.1		00	A
DBX012.2	C1_BS_FWD_MVM_FLT	Cylinder C1 forward movement fault	BB	0.1		00	A
DBX012.3	C1_BS_BWD_MVM_FLT	Cylinder C1 backward movement fault	BB	0.1		00	A
DBX012.4	C1_BS_AUTO_SEQ_PUSH_DONE	Cylinder C1 push successfully done in auto sequence	BB	0.1		00	A
DBX012.5	C1_BS_FWD_MVM_ENAB	Cylinder C1 forward movement enable	BB	0.1		00	A
DBX012.6	C1_BS_BWD_MVM_ENAB	Cylinder C1 backward movement enable	BB	0.1		00	A
DBX012.7	SPARE	Spare	BB	0.1		00	A
		CYLINDER C2					
DBX013.0	C2_BS_POS_IN	Cylinder C2 position in (zero position)	BB	0.1		00	A
DBX013.1	C2_BS_POS_OUT	Cylinder C2 position out (end position)	BB	0.1		00	A
DBX013.2	C2_BS_DEPRESS_POS	Cylinder C2 depressure position	BB	0.1		00	A
DBX013.3	C2_BS_AUTO_SEQ_REPOS_IN	Cylinder C2 return request to in pos from auto seq reposition	BB	0.1		00	A
DBX013.4	C2_BS_HOME_SEQ_REPOS_IN	Cylinder C2 return request to in pos from home seq reposition	BB	0.1		00	A
DBX013.5	C2_BS_AUTO_SEQ_PUSH	Cylinder C2 push request from auto seq	BB	0.1		00	A
DBX013.6	C2_BS_AUTO_SEQ_RETURN_PUSH_FAIL	Cylinder C2 push failed from auto seq, return request	BB	0.1		00	A
DBX013.7	C2_BS_AUO_SEQ_DEPRESS	Cylinder C2 depressure request from auto seq	BB	0.1		00	A
DBX014.0	C2_BS_PUSH_FWD_MVM	Cylinder C2 push forward movement	BB	0.1		00	A
DBX014.1	C2_BS_RETURN_BWD_MVM	Cylinder C2 return backward movement	BB	0.1		00	A
DBX014.2	C2_BS_PUSH_FWD_MAN	Cylinder C2 manual request to push forward movement	BB	0.1		00	A
DBX014.3	C2_BS_RET_BWD_MAN	Cylinder C2 manual request to return backward movement	BB	0.1		00	A
DBX014.4	C2_BS_POS_ADJ	Cylinder C2 position adjusted	BB	0.1		00	A
DBX014.5	C2_BS_PRS_FLT_MAN	Cylinder C2 pressure fault in manual	BB	0.1		00	A
DBX014.6	C2_BS_AUTO_SEQ_PRS_FLT	Cylinder C2 pressure fault in auto sequence	BB	0.1		00	A
DBX014.7	C2_BS_AUTO_SEQ_PUSH_FAIL	Cylinder C2 push failed in auto sequence	BB	0.1		00	A
DBX015.0	C2_BS_AUTO_SEQ_3_FAIL_PUSH_FLT	Cylinder C2 fault after 3 failed pushes in auto sequence	BB	0.1		00	A
DBX015.1	C2_BS_MVM_REQ	Cylinder C2 movement request	BB	0.1		00	A
DBX015.2	C2_BS_FWD_MVM_FLT	Cylinder C2 forward movement fault	BB	0.1		00	A
DBX015.3	C2_BS_BWD_MVM_FLT	Cylinder C2 backward movement fault	BB	0.1		00	A
DBX015.4	C2_BS_AUTO_SEQ_PUSH_DONE	Cylinder C2 push successfully done in auto sequence	BB	0.1		00	A
DBX015.5	C2_BS_FWD_MVM_ENAB	Cylinder C2 forward movement enable	BB	0.1		00	A
DBX015.6	C2_BS_BWD_MVM_ENAB	Cylinder C2 backward movement enable	BB	0.1		00	A
DBX015.7	SPARE	Spare	BB	0.1		00	A
		CYLINDER C3					
DBX016.0	C3_BS_POS_IN	Cylinder C3 position in (zero position)	BB	0.1		00	A
DBX016.1	C3_BS_POS_OUT	Cylinder C3 position out (position 60%)	BB	0.1		00	A
DBX016.2	C3_BS_POS_EJECT	Cylinder C3 position eject (end position)	BB	0.1		00	A
DBX016.3	C3_BS_DEPRESS_POS	Cylinder C3 depressure position	BB	0.1		00	A
DBX016.4	C3_BS_AUTO_SEQ_REPOS_IN	Cylinder C3 return request to in pos from auto seq reposition	BB	0.1		00	A
DBX016.5	C3_BS_HOME_SEQ_REPOS_IN	Cylinder C3 return request to in pos from home seq reposition	BB	0.1		00	A
DBX016.6	C3_BS_AUTO_SEQ_PUSH	Cylinder C3 push request from auto seq	BB	0.1		00	A
DBX016.7	C3_BS_AUTO_SEQ_RETURN_PUSH_FAIL	Cylinder C3 push failed from auto seq, return request	BB	0.1		00	A
DBX017.0	C3_BS_AUO_SEQ_DEPRESS	Cylinder C3 depressure request from auto seq	BB	0.1		00	A
DBX017.1	C3_BS_AUO_SEQ_PUSH_EJC	Cylinder C3 push to eject position from auto seq	BB	0.1		00	A
DBX017.2	C3_BS_PUSH_FWD_MVM	Cylinder C3 push forward movement	BB	0.1		00	A
DBX017.3	C3_BS_RETURN_BWD_MVM	Cylinder C3 return backward movement	BB	0.1		00	A
DBX017.4	C3_BS_PUSH_FWD_MAN	Cylinder C3 manual request to push forward movement	BB	0.1		00	A
DBX017.5	C3_BS_RET_BWD_MAN	Cylinder C3 manual request to return backward movement	BB	0.1		00	A
DBX017.6	C3_BS_BWD_MVM_ENAB	Cylinder C3 backward movement enable	BB	0.1		00	A
DBX017.7	C3_BS_PRS_FLT_MAN	Cylinder C3 pressure fault in manual	BB	0.1		00	A
DBX018.0	C3_BS_AUTO_SEQ_PRS_FLT	Cylinder C3 pressure fault in auto sequence	BB	0.1		00	A
DBX018.1	C3_BS_AUTO_SEQ_PUSH_FAIL	Cylinder C3 push failed in auto sequence	BB	0.1		00	A
DBX018.2	C3_BS_AUTO_SEQ_3_FAIL_PUSH_FLT	Cylinder C3 fault after 3 failed pushes in auto sequence	BB	0.1		00	A
DBX018.3	C3_BS_MVM_REQ	Cylinder C3 movement request	BB	0.1		00	A
DBX018.4	C3_BS_FWD_MVM_FLT	Cylinder C3 forward movement fault	BB	0.1		00	A

DBX018.5	C3_BS_BWD_MVM_FLT	Cylinder C3 backward movement fault	BB	0.1		00	A
DBX018.6	C3_BS_AUTO_SEQ_PUSH_DONE	Cylinder C3 push successfully done in auto sequence	BB	0.1		00	A
DBX018.7	C3_BS_FWD_MVM_ENAB	Cylinder C3 forward movement enable	BB	0.1		00	A
CYLINDER DOOR							
DBX019.0	DOOR_BS_POS_UP	Cylinder DOOR position in (zero position)	BB	0.1		00	A
DBX019.1	DOOR_BS_POS_DOWN	Cylinder DOOR position out (end position)	BB	0.1		00	A
DBX019.2	DOOR_BS_AUTO_SEQ_REPOS_DOWN	Cylinder DOOR return request to down pos from auto seq reposition	BB	0.1		00	A
DBX019.3	DOOR_BS_HOME_SEQ_REPOS_UP	Cylinder DOOR return request to up pos from home seq reposition	BB	0.1		00	A
DBX019.4	DOOR_BS_AUTO_SEQ_UP	Cylinder DOOR bwd to up request from auto seq	BB	0.1		00	A
DBX019.5	DOOR_BS_DOWN_FWD_MVM	Cylinder DOOR forward movement to down	BB	0.1		00	A
DBX019.6	DOOR_BS_UP_BWD_MVM	Cylinder DOOR backward movement to up	BB	0.1		00	A
DBX019.7	DOOR_BS_DOWN_FWD_MAN	Cylinder DOOR manual request forward movement to down	BB	0.1		00	A
DBX020.0	DOOR_BS_UP_BWD_MAN	Cylinder DOOR manual request backward movement to up	BB	0.1		00	A
DBX020.1	DOOR_BS_POS_ADJ	Cylinder DOOR position adjusted	BB	0.1		00	A
DBX020.2	DOOR_BS_PRS_FLT_MAN	Cylinder DOOR pressure fault in manual	BB	0.1		00	A
DBX020.3	DOOR_BS_MVM_REQ	Cylinder DOOR movement request	BB	0.1		00	A
DBX020.4	DOOR_BS_FWD_MVM_FLT	Cylinder DOOR forward movement fault	BB	0.1		00	A
DBX020.5	DOOR_BS_BWD_MVM_FLT	Cylinder DOOR backward movement fault	BB	0.1		00	A
DBX020.6	DOOR_BS_FWD_MVM_ENAB	Cylinder DOOR forward movement enable	BB	0.1		00	A
DBX020.7	DOOR_BS_BWD_MVM_ENAB	Cylinder DOOR backward movement enable	BB	0.1		00	A
MACHINE TEST							
DBX021.0	MT_BS_RUN	Machine test running	BB	0.1		00	A
DBX021.1	MT_BS_ENAB	Machine_test_enable	BB	0.1		00	A
DBX021.2	MTC1_BS_FWD_FLT	Machine test for C1 fault in forward movement	BB	0.1		00	A
DBX021.3	MTC1_BS_BWD_FLT	Machine test for C1 fault in backward movement	BB	0.1		00	A
DBX021.4	MTC2_BS_FWD_FLT	Machine test for C2 fault in forward movement	BB	0.1		00	A
DBX021.5	MTC2_BS_BWD_FLT	Machine test for C2 fault in backward movement	BB	0.1		00	A
DBX021.6	MTC3_BS_FWD_FLT	Machine test for C3 fault in forward movement	BB	0.1		00	A
DBX021.7	MTC3_BS_BWD_FLT	Machine test for C3 fault in backward movement	BB	0.1		00	A
DBX022.0	MTDOOR_BS_FWD_FLT	Machine test for DOOR fault in forward movement	BB	0.1		00	A
DBX022.1	MTDOOR_BS_BWD_FLT	Machine test for DOOR fault in backward movement	BB	0.1		00	A
DBX022.2	MTC1_BS_FLT	Machine test for C1 fault	BB	0.1		00	A
DBX022.3	MTC2_BS_FLT	Machine test for C2 fault	BB	0.1		00	A
DBX022.4	MTC3_BS_FLT	Machine test for C3 fault	BB	0.1		00	A
DBX022.5	MTDOOR_BS_FLT	Machine test for DOOR fault	BB	0.1		00	A
DBX022.6	MTFT_BS_FLT	Machine full test fault	BB	0.1		00	A
DBX022.7	MTC1_BS_ACT	Machine test for C1 active	BB	0.1		00	A
DBX023.0	MTC2_BS_ACT	Machine test for C2 active	BB	0.1		00	A
DBX023.1	MTC3_BS_ACT	Machine test for C3 active	BB	0.1		00	A
DBX023.2	MTDOOR_BS_ACT	Machine test for DOOR active	BB	0.1		00	A
DBX023.3	MTFT_BS_ACT	Machine full test active	BB	0.1		00	A
DBX023.4	MT_BS_RDY	Machine test ready	BB	0.1		00	A
DBX023.5	MTC1_BS_DONE	Machine test for C1 successfully done	BB	0.1		00	A
DBX023.6	MTC2_BS_DONE	Machine test for C2 successfully done	BB	0.1		00	A
DBX023.7	MTC3_BS_DONE	Machine test for C3 successfully done	BB	0.1		00	A
PRESSURE SIMULATION							
DBX024.0	PRS_BS_DECOMP	Pressure after decompress (less than TBD bars)	BB	0.1		00	A
DBX024.1	PRS_BS_WARN_STP_MVM	Pressure high value to stop movement	BB	0.1		00	A
DBX024.2	PRS_BS_DECR_FLOW_TO_HALF	Pressure high value need to decrease flow to half it's value	BB	0.1		00	A
DBX024.3	PRS_BS_DECR_FLOW	Pressure high value need to decrease flow	BB	0.1		00	A
DBX024.4	PRS_BS_HHIGH	Pressure high high value	BB	0.1		00	A
DBX024.5	PRS_BS_WARN	Pressure warning high value	BB	0.1		00	A
DBX024.6	PRS_BS_THRS_DET	Pressure threshold detected	BB	0.1		00	A
DBX024.7	SPARE	Spare	BB	0.1		00	A
AUTO SEQUENCE							
DBX025.0	AS_BS_REPOS_RUN	Auto sequence reposition running	BB	0.1		00	A
DBX025.1	AS_BS_RUN	Auto sequence running	BB	0.1		00	A
DBX025.2	AS_BS_REPOS_C1	Auto sequence reposition cylinder C1	BB	0.1		00	A
DBX025.3	AS_BS_REPOS_C2	Auto sequence reposition cylinder C2	BB	0.1		00	A
DBX025.4	AS_BS_REPOS_C3	Auto sequence reposition cylinder C3	BB	0.1		00	A
DBX025.5	AS_BS_REPOS_DOOR	Auto sequence reposition cylinder DOOR	BB	0.1		00	A
DBX025.6	AS_BS_END	Auto sequence successfully done	BB	0.1		00	A
DBX025.7	AS_BS_STA_WAIT	Auto sequence waiting to start after repos/man	BB	0.1		00	A
DBX026.0	AS_BS_FLT	Auto sequence fault	BB	0.1		00	A
DBX026.1	AS_BS_PRS_FLT	Auto sequence pressure fault	BB	0.1		00	A
DBX026.2	AS_BS_ENAB	Auto sequence enabled	BB	0.1		00	A
DBX026.3	SPARE	Spare	BB	0.1		00	A
DBX026.4	SPARE	Spare	BB	0.1		00	A
DBX026.5	SPARE	Spare	BB	0.1		00	A
DBX026.6	SPARE	Spare	BB	0.1		00	A
DBX026.7	SPARE	Spare	BB	0.1		00	A
HOME SEQUENCE							

DBX027.0	HC_BS_RUN	Home cycle running	BB	0..1		00	A
DBX027.1	HC_BS_END	Home cycle end	BB	0..1		00	A
DBX027.2	HC_BS_FLT	Home cycle fault	BB	0..1		00	A
DBX027.3	HC_BS_STPD	Home cycle stopped	BB	0..1		00	A
DBX027.4	SPARE	Spare	BB	0..1		00	A
DBX027.5	SPARE	Spare	BB	0..1		00	A
DBX027.6	SPARE	Spare	BB	0..1		00	A
DBX027.7	SPARE	Spare	BB	0..1		00	A
<b>FLOW CONTROL</b>							
DBX028.0	FC_BS_GRP1_ENAB	Enable flow control group 1	BB	0..1		00	A
DBX028.1	FC_BS_GRP1_MVM_DONE	Movement done group 1	BB	0..1		00	A
DBX028.2	FC_BS_GRP2_ENAB	Enable flow control group 2	BB	0..1		00	A
DBX028.3	FC_BS_GRP2_MVM_DONE	Movement done group 2	BB	0..1		00	A
DBX028.4	SPARE	Spare	BB	0..1		00	A
DBX028.5	SPARE	Spare	BB	0..1		00	A
DBX028.6	SPARE	Spare	BB	0..1		00	A
DBX028.7	SPARE	Spare	BB	0..1		00	A
<b>STATUS - AUX</b>							
DBX029.0	MTDOOR_BS_DONE	Machine test for DOOR successfully done	BB	0..1		00	A
DBX029.1	MTFT_BS_DONE	Machine full test successfully done	BB	0..1		00	A
DBX029.2	C1_BS_OVRSZ_BALE	Cylinder C1 oversize bale detected	BB	0..1		00	A
DBX029.3	C2_BS_OVRSZ_BALE	Cylinder C2 oversize bale detected	BB	0..1		00	A
DBX029.4	C3_BS_OVRSZ_BALE	Cylinder C3 oversize bale detected	BB	0..1		00	A
DBX029.5	SPARE	Spare	BB	0..1		00	A
DBX029.6	SPARE	Spare	BB	0..1		00	A
DBX029.7	SPARE	Spare	BB	0..1		00	A