

# Automatizacija modela industrijskog pogona izvedenog manipulatorima i pokretnim trakama

---

Nemčić, Marin

Undergraduate thesis / Završni rad

2016

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **Technical College in Bjelovar / Visoka tehnička škola u Bjelovaru**

Permanent link / Trajna poveznica: <https://um.nsk.hr/um:nbn:hr:144:477674>

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

Download date / Datum preuzimanja: **2025-04-02**



Repository / Repozitorij:

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



**VISOKA TEHNIČKA ŠKOLA U BJELOVARU**

ZAVRŠNI RAD br: 02/MEH/2016

**Automatizacija modela industrijskog  
pogona izvedenog manipulatorima i  
pokretnim trakama**

Marin Nemčić

Bjelovar, rujan 2016.

**VISOKA TEHNIČKA ŠKOLA U BJELOVARU**

ZAVRŠNI RAD br: 02/MEH/2016

**Automatizacija modela industrijskog  
pogona izvedenog manipulatorima i  
pokretnim trakama**

Marin Nemčić

Bjelovar, rujan 2016.



**Visoka tehnička škola u Bjelovaru**

**Trg E. Kvaternika 4, Bjelovar**

## 1. DEFINIRANJE TEME ZAVRŠNOG RADA I POVJERENSTVA

Kandidat: **Nemčić Marin**

Datum: 21.03.2016.

Matični broj: 000692

JMBAG: 0314006847

Kolegij: **AUTOMATIZACIJA STROJEVA I UREĐAJA 1**

Naslov rada (tema): **Automatizacija modela industrijskog pogona izvedenog manipulatorima i pokretnim trakama**

Mentor: **dr.sc. Igor Petrović**

zvanje: **viši predavač**

Članovi Povjerenstva za završni rad:

1. Zoran Vrhovski, mag.ing.el.techn.inf., predsjednik
2. dr.sc. Igor Petrović, mentor
3. mr.sc. Stjepan Golubić, član

## 2. ZADATAK ZAVRŠNOG RADA BROJ: 02/MEH/2016

U radu je potrebno:

- za postojeći sustav TTT manipulatora sa pokretnom trakom izraditi tehničku dokumentaciju izvedenog stanja sustava automatizacije
- izraditi programsku podršku za master PLC uređaj koji vodi model industrijskog pogona
- izraditi programsku podršku za slave PLC uređaj koji vodi TTT manipulator
- izraditi programsku podršku vizualizacije modela industrijskog pogona
- pustiti u rad model industrijskog pogona

Zadatak uručen: 21.03.2016.

Mentor: **dr.sc. Igor Petrović**



Zahvaljujem se svim profesorima Visoke tehničke škole u Bjelovaru na prenesenom znanju, uloženom trudu i strpljenu na predavanjima i konzultacijama. Posebno se zahvaljujem svom mentoru, dr. sc. Igoru Petroviću na savjetima i uputama koje su mi pomogle u izradi ovoga završnoga rada.

# Sadržaj

<b>Uvod.....</b>	<b>1</b>
<b>1. Programljivi logički kontroler .....</b>	<b>2</b>
1.1 Što je programljivi logički kontroler? .....	2
1.2 Povijest PLC-a .....	2
1.3 Arhitektura PLC-a.....	2
1.4 Princip rada PLC-a.....	3
1.5 Ulazni dio PLC-a .....	4
1.6 Izlazni dio PLC-a .....	5
<b>2. Pregled korištenih uređaja i komponenti .....</b>	<b>7</b>
2.1 Siemens S7-200 224XP .....	8
2.2 Napajanja.....	9
2.3 Motori .....	10
2.4 Driveri.....	10
2.5 Releji.....	11
2.6 Siemens S7-1200 1214C DC/DC/DC.....	11
2.7 Napajanje.....	12
2.8 Trakasti transporter .....	12
2.9 Releji za upravljanje trakastog transportera .....	13
2.10 Kompresor zraka.....	13
<b>3. Programski alat Step 7 Micro/WIN i uspostava komunikacije .....</b>	<b>14</b>
3.1 Općenito o programu Step 7 Micro/WIN .....	14
3.2 Pokretanje Step 7 Micro/WIN SP9.....	14
3.3 Uspostava komunikacije PLC-a i računala.....	14
3.4 Definiranje PLC-a i početak rada.....	15
3.4.1 Definiranje PLC-a.....	15
3.5 Osnovni simboli korišteni u programu Step 7 Micro/WIN.....	17
3.6 Početak pisanja programa .....	18

3.7	Tablica simbola.....	19
<b>4.</b>	<b>Programski alat TIA portal .....</b>	<b>20</b>
<b>5.</b>	<b>Step 7 Micro/WIN program upravljanja TTT manipulatora.....</b>	<b>24</b>
5.1	Prvi ciklus programa i definiranje korištenih memorijskih mjesta .....	24
5.2	Kretanje manipulatora prema krajnjim prekidačima.....	25
5.3	Generiranje impulsa pomoću timera.....	27
5.4	Generiranje signala za motore.....	28
5.5	Zbrajanje i oduzimanje impulsa pojedinih osi .....	31
5.6	Kretanje manipulatora prema poziciji predmeta .....	33
5.6.1	Kretanja po X osi.....	33
5.6.2	Kretanja po Y osi.....	34
5.6.3	Kretanja po Z osi .....	35
5.6.4	Reset korištenih bitova i priprema za prihvat predmeta .....	36
5.7	Prihvat predmeta i prijenos predmeta na pokretnu traku .....	36
5.8	Definiranje nove putanje manipulatora .....	41
<b>6.</b>	<b>TIA Portal program nadziranja cjelokupnog sustava.....</b>	<b>44</b>
<b>7.</b>	<b>WinCC Flexible Advanced.....</b>	<b>50</b>
7.1	Vizualni prikaz rada motora i pneumatske prihvatnice .....	50
7.2	Trend gibanja manipulatora po pojedinoj osi.....	52
<b>8.</b>	<b>TIA Portal WinCC RunTime Advanced.....</b>	<b>54</b>
<b>9.</b>	<b>Zaključak.....</b>	<b>60</b>
<b>10.</b>	<b>Literatura .....</b>	<b>61</b>
<b>11.</b>	<b>Sažetak.....</b>	<b>62</b>
<b>12.</b>	<b>Abstract .....</b>	<b>63</b>
<b>13.</b>	<b>Privitak .....</b>	<b>66</b>

## Uvod

Završni rad izabran je iz kolegija Automatizacija strojeva i uređaja 1. Automatizacija je prijenos rada čovjeka na stroj tehničkim putem. Omogućava tehnološki razvoj u proizvodnji, upravljanju i drugim tehnološkim procesima bez direktnog ljudskog djelovanja. Automatizacija smanjuje potreban rad čovjeka, ali i očuvava energiju i materijale radi kvalitete i preciznosti.

TTT manipulator jedna je od osnovnih robotskih struktura koja ima troosni koordinatni sustav. Takva struktura omogućava potpuno pozicioniranje hvataljke stroja u prostoru. Svaka os upravljana je pomoću koračnog motora, koje preko drivera upravlja PLC. Kao prihvatnica koristi se mehanička hvataljka s dva prsta. Upravlja se pomoću pneumatskih elektroventila koji su također spojeni na PLC. Glavna namjena ovakvog sustava je transport i pozicioniranje radnih komada prema potrebama i zahtjevima pojedinih procesa i zadataka, kao što je opisano u literaturi [1].

Cilj završnoga rada je automatizacija prijenosa predmeta, u ovome konkretnom slučaju drvene kocke, s pokretne trake na postolje korištenjem čeljusti pneumatske prihvatnice. Na kraju pokretne trake nalazi se senzor. Pokretna traka vozi predmet sve dok senzor ne detektira predmet. Detekcijom predmeta pali se vremenski brojač koji zaustavlja traku nekoliko stotina milisekundi nakon prolaska predmeta ispred senzora. Time se osigurava spuštanje predmeta u spremnik. Cijelu sustav zamišljen je u radu s robotskom rukom, koja se u ovom radu ne koristi. Dodatan timer predstavlja improvizirano vrijeme rada robotske ruke koja treba drvenu kocku postaviti na postolje. Nakon što timer izbroji zadano vrijeme, glavni PLC šalje signal podređenom PLC-u da krene s radom.



# 1. Programljivi logički kontroler

## 1.1 Što je programljivi logički kontroler?

Programljivi logički kontroler je digitalno računalo koje se tipično koristi u elektromehaničkim procesima. Princip rada temelji se na ulaznim signalima i programski opisanih naredbi kako bi se stvorili izlazni signali. Najvažniji dijelovi PLC-a (eng. *Programmable Logic Controller*) za rad su: procesor, memorija, ulazi i izlazi. Ulazi u PLC su razna tipkala i sklopke, senzori i pretvornici. Dizajniran je tako da zahtijeva minimalne uvjete održavanja i veoma je izdržljiv u industrijskim uvjetima rada gdje se pojavljuju razne elektromagnetne smetnje, nagle promjene u temperaturi, vlaga i slično. Prednosti PLC-a su jednostavno ožičenje i male dimenzije, kao što je opisano u literaturi [2].

## 1.2 Povijest PLC-a

Prvi programljivi logički kontroler proizvela je tvrtka „*General Motors*“. Cilj je bio zamijeniti relejnu tehniku koja je do kraja 60-ih godina bila dio svake industrijske tvrtke. Relejna tehnika bila je skup i dugotrajan proces pošto se za svaku promjenu u proizvodnom programu zahtijevala promjena relejnog rasporeda i ožičenja. Programiranje prvih PLC-a bilo je slično crtanju relejne sheme, dok se kasnijih godina nije počelo uvoditi dodatne komponente, kao što su vremenski sklopovi, aritmetičke operacije, brojila itd., kao što je opisano u literaturi [2].

## 1.3 Arhitektura PLC-a

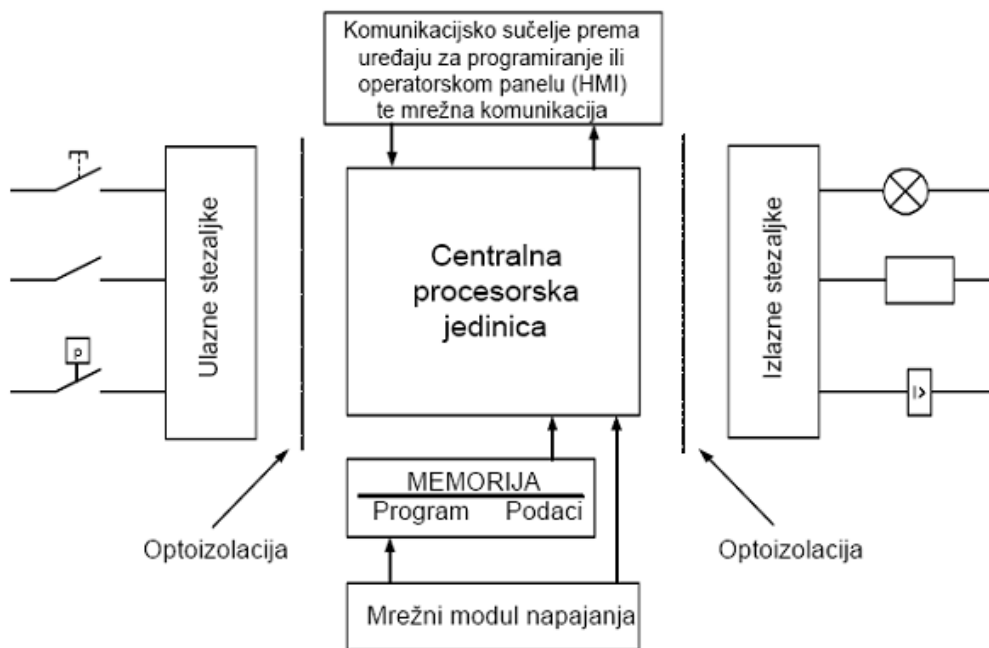
Glavni dio PLC-a je centralna procesorska jedinica (eng. *Central Processor Unit*). Ona čita podatke s digitalnih i analognih ulaza uređaja, zatim ih obrađuje logički i aritmetički te ih prosljeđuje na digitalne i analogne izlaze. PLC koristi i dvije vrste memorije, a to su RAM (eng. *Random Access Memory*) i EEPROM (eng. *Electrically Erasable Programmable Read-Only Memory*). RAM memorija je memorija koju procesor koristi za trenutno spremanje podataka u toku rada, pisanja ili izvođenja programa. Ova memorija gubi sav sadržaj prilikom gubitka napajanja, pa zbog toga postoji EEPROM memorija. EEPROM memorija čuva podatke trajno, makar i prilikom nestanka napajanja i namijenjena je za čitanje. Iz tog razloga, ova memorija se kod PLC-a koristi prilikom podizanja aplikacije.

Ovim putem se u slučaju gubitka programa iz RAM memorije koristi EEPROM memorija. Iz nje se učitava program u RAM memoriju prilikom svakog uključivanja PLC-a. Ulazi (eng. *Inputs*) odnose se na uređaje i pretvornike koji prikupljaju podatke iz fizičkog svijeta i šalju ih u PLC. PLC koristi te ulazne podatke kako bi izvršavao radnje upisane programom. Te odluke su pretežito paljenje ili gašenje određenog izlaza PLC-a. Potrebno je znati o različitim tipovima ulaza, a to su digitalni i analogni ulazi. Primjer za korištenje analognih ulaza je temperatura ili bilo koja mjerna veličina koja ima neki raspon vrijednosti. Digitalni ulazi su dva stanja kao što je stanje sklopke, 1 ili 0. Izlazi (eng. *Outputs*) također mogu biti analogni ili digitalni. Analogni izlazi koriste se za varijabilne raspone izlaza između potpuno ugašenog i potpuno upaljenog stanja. Digitalni izlazi su jednostavno upaljeno ili ugašeno stanje. Aktuatori su uređaji koji pretvaraju električni signal PLC-a u fizičku kretnju. Ovaj tekst opisan je u literaturi [2].

#### **1.4 Princip rada PLC-a**

Da bi se objasnio princip rada PLC-a, potrebno je dobro poznavati njegove osnovne cjeline. Osnovne cjeline PLC uređaja prikazane su na slici 1.1. Gotovo svi PLC uređaji imaju u principu istu sklopovsku strukturu. Tu sklopovsku strukturu čine:

- ulazni dio (digitalni i analogni ulazi),
- izlazni dio (digitalni i analogni izlazi),
- centralna procesorska jedinica,
- memorijski blok za program i podatke,
- mrežni dio za napajanje i komunikacijsko sučelje,
- moduli za proširenje.

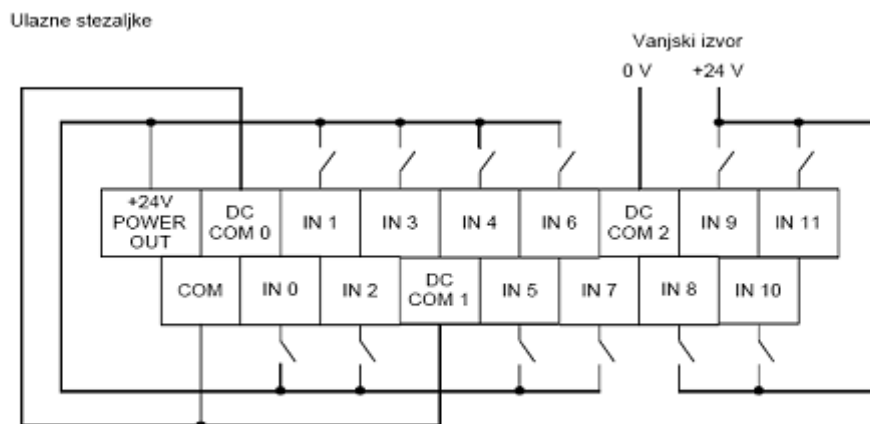


Slika 1.1: Osnovne cjeline PLC uređaja [2]

## 1.5 Ulazni dio PLC-a

Ulazni dio PLC-a čine stezaljke na koje se spajaju dolazni signali iz procesa. To je mjesto gdje se dovedeni vanjski signal prilagođava signalu kojeg razumije centralna procesorska jedinica PLC-a. Informacije koje PLC prima na svojim ulazima mogu biti digitalne ili analogne. Digitalna ulazna informacija može biti, na primjer, signal krajnjeg prekidača, tipkala i sl. Analogna ulazna informacija može biti naponski signal 0-10V s mjernog pretvornika temperature i sl. Za digitalnu informaciju visoko stanje iznosi 14-30 VDC, a nisko stanje 0-5 VDC. Prilagodba signala s uobičajenog ulaznog napona od 120-230 VAC ili 24 VDC na 5 VDC, tj. naponski nivo logike procesorske jedinice, uključuje optoizolaciju signala, što je vrlo važno kako bi se galvanski odvojili strujni krugovi, čime se sprječava protok struje uslijed potencijalnih razlika strujnih krugova unutarnje logike PLC-a i ulaznog kruga, te filtriranje signala kako bi se smanjile visokofrekventne smetnje. Ovaj tekst opisan je u literaturi [2].

Na slici 1.2 prikazan je primjer ožičenja ulaznih stezaljki PLC-a s 12 digitalnih ulaza podijeljenih u 3 grupe prema stezaljki mase. Stezaljke od IN 0 do IN 7 koriste izvor napajanja iz samog PLC-a dok stezaljke od IN 8 do IN 11 koriste vanjski izvor napajanja. Vanjski izvor napajanja se koristi ako izvor u PLC-u ne može zadovoljiti potrebe senzorske opreme za količinom električne energije.



Slika 1.2: Ožičenje ulaznih stezaljki PLC-a [2]

## 1.6 Izlazni dio PLC-a

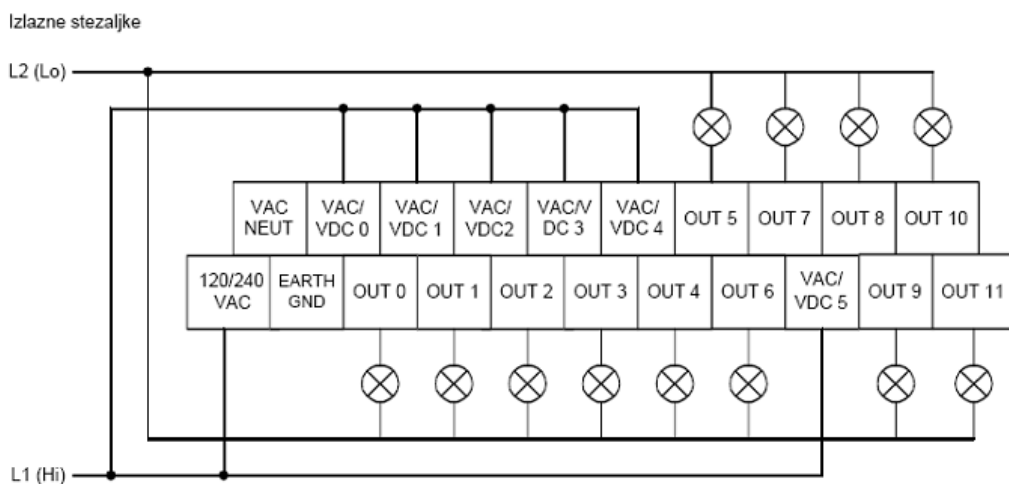
Izlazni dio PLC-a čine stezaljke na koje se spajaju izvršni uređaji iz procesa kojima PLC šalje digitalne i analogne signale te na taj način upravlja procesom. Na digitalne izlaze iz PLC-a najčešće su spojeni magnetni svici, releji, sklopnici, motorske sklopke, signalne lampe, pneumatski razvodnici i sl., dok na analogne izlaze mogu biti spojeni strujni signali za prikaz neke veličine na pokaznom instrumentu, referenca brzine za frekvencijski pretvarač, itd. Izlazne stezaljke također su optoizolirane od procesorske jedinice radi galvanske izolacije električnih krugova.

Digitalni izlazi najčešće su izvedeni kao relejni, tranzistorski ili pomoću trijaka, a svaki od njih ima svoje prednosti i mane:

- **relejni izlazi** mogu se koristiti za sklapanje istosmjernih i izmjeničnih tereta, za struje do nekoliko ampera. Releji dobro podnose naponske udare i obzirom na zračni razmak između njihovih kontakata ne postoji mogućnost pojave pulsirajućih struja. Releji su relativno spori prilikom sklapanja te imaju vijek trajanja (mjeren maksimalnim brojem sklapanja) manji od trijaka i tranzistora,

- **tranzistorski izlazi** služe za sklopanje istosmjernih tereta, nemaju pokretnih dijelova koji se troše i bešumni su. Vrijeme reakcije im je brzo, ali mogu sklopiti struje do 0.5 A,
  - **izlazi s trijacima** služe za sklopanje izmjeničnih tereta, a karakteristike su im slične kao kod izlaza s tranzistorima.
- Digitalni izlazi opisani su u literaturi [2].

Na slici 1.3 prikazan je primjer ožičenja izlaznih stezaljki PLC-a s 12 digitalnih izlaza koji su podijeljene u 6 grupa prema stezaljci napajanja. Releji koji se nalaze u PLC-u otvaraju, odnosno zatvaraju pojedine strujne krugove te na taj način uključuju, odnosno isključuju trošila.



Slika 1.3: Ožičenje izlaznih stezaljki PLC-a [2]

## 2. Pregled korištenih uređaja i komponenti

U ovom završnom radu korištena su dva PLC uređaja. Na slici 2.1 prikazan je u cijelosti stroj kojim se upravlja. U nastavku teksta opisan je dio po dio svake komponente koja se koristi.



Slika 2.1: TTT manipulator i pokretna traka

## 2.1 Siemens S7-200 224XP

Za upravljanje TTT manipulatora korišten je PLC *Siemens S7-200 224XP* prikazan na slici 2.2.



Slika 2.2: S7-200 224XP DC/DC/DC [3]

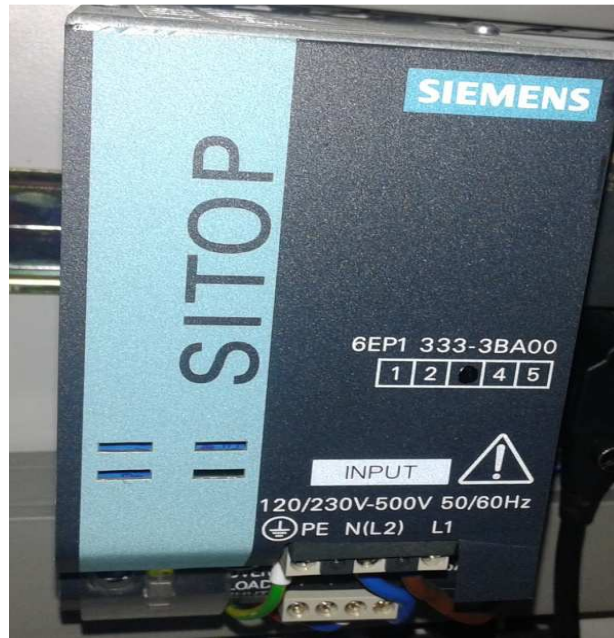
Karakteristike *Siemens S7-200 224XP* uređaja su:

- fizička dimenzija: 140 x 80 x 62 mm,
- digitalnih ulaza/izlaza: 14 ulaza i 10 izlaza,
- analognih ulaza/izlaza: 2 ulaza i 1 izlaz,
- modula za proširenje: 7,
- sat stvarnog vremena: ugrađen,
- komunikacijski portovi: 2xRS485.

PLC *Siemens S7-200 224XP* koristi se kao podređeni (eng. *Slave*) uređaj. Cijeli industrijski proces vođen je glavnim (eng. *Master*) uređajem, a to je PLC *S7-1214C*. Karakteristike PLC-a prikazane su u [3].

## 2.2 Napajanja

TTT manipulator koristi 2 napajanja. *Siemens 6EP1 333-3BA00* napajanje koristi se za PLC S7-200 i relej, a prikazano je na slici 2.3. Napajanje *Meanwell DRP-240-24* prikazano na slici 2.4 koristi se za drivere motora.



Slika 2.3: Napajanje *Siemens 6EP1 333-3BA00* [4]



Slika 2.4: Napajanje *Meanwell DRP\_240\_24* [5]



## 2.3 Motori

Koriste se tri unipolarna koračna motora *Nanotec-Munich ST6018* prikazana na slici 2.5. Karakteristike motora su sljedeće, a opisane su u [6].

- radni napon: 24V,
- nazivna struja: 2A,
- kutni korak: 1.8°,
- moment: 1.38Nm.



Slika 2.5: Koračni motor *Nanotec-Munich ST6018* [6]

## 2.4 Driveri

Tri koračna motora zahtijevaju tri drivera. Identični driveri koriste se za svaki motor posebno. Korišteni driveri za koračne motore *Nanotec SMC 133-2* prikazani su na slici 2.6.



Slika 2.6: Driveri za koračne motore *Nanotec SMC 133-2*

## 2.5 Releji

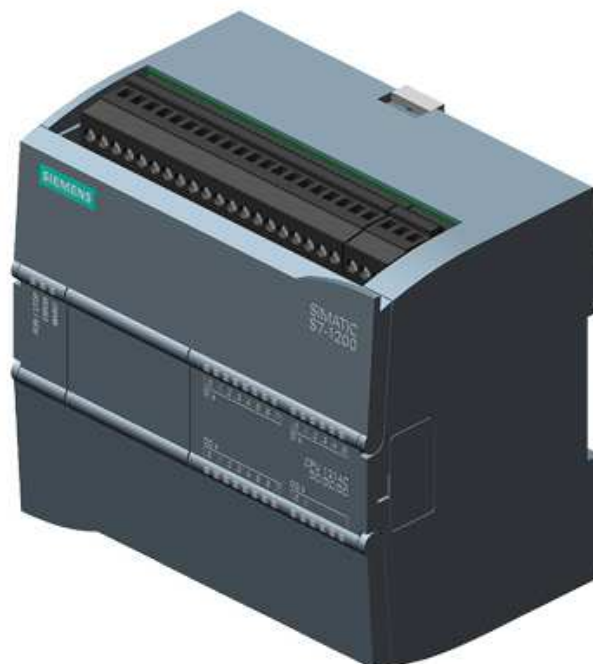
Zadatak pneumatske prihvatnice je da steže i otpušta čeljust kako bi prenesla predmet. Kako bi to bilo moguće, koristi se relej *Schrack PT570024* za uklapanje pneumatskog razvodnika, prikazan na slici 2.7.



Slika 2.7: Mini releji *Schrack PT570024*

## 2.6 Siemens S7-1200 1214C DC/DC/DC

Cijeli industrijski proces vodi *master* PLC *S7-1200 1214C* DC/DC/DC prikazan na slici 2.8. Glavni zadatak *master* PLC-a je nadziranje *slave* PLC-a i upravljanje malim automatiziranim procesom prijenosa kutije koji uključuje pokretnu traku.



Slika 2.8: PLC *S7-1200 1214C* DC/DC/DC [7]

## 2.7 Napajanje

Napajanje koje koristi PLC *S7-1200* i pokretna traka je *MSI KY-400ATX P4* prikazano na slici 2.9.



Slika 2.9: MSI KY-400ATX P4 [8]

## 2.8 Trakasti transporter

Trakasti transporter prikazan na slici 2.10 koristi se za prijenos materijala. Transporter se sastoji od dva bubnja i gumene trake. Jedan bubanj je pomičan, a drugi nepomičan.



Slika 2.10: Trakasti transporter

## 2.9 Releji za upravljanje trakastog transportera

Za odabir smjera vrtnje pokretne trake koriste se dva releja prikazana na slici 2.11.



Slika 2.11: Dva releja za upravljanje smjera pokretne trake

## 2.10 Kompresor zraka

Kako bi se upravljalo sa čeljusti pneumatske prihvatnice, potreban je komprimirani zrak. Kompresor AAC-WE2 spojen je na pneumatski dio stroja i prikazan na slici 2.12.



Slika 2.12: Kompresor zraka

### 3. Programski alat Step 7 Micro/WIN i uspostava komunikacije

#### 3.1 Općenito o programu Step 7 Micro/WIN

*Step 7 Micro/WIN SP9* programski je alat koji se koristi za programiranje PLC-a serije S7-200. Odlikuje se brzim i jednostavim programiranjem i nudi mogućnost programiranja u tri standardna jezika:

- STL (eng. *Statement List*)
- LAD (eng. *Ladder Logic*)
- FBD (eng. *Function Block Diagram*)

Program se instalira na računalo koristeći instalacijski CD kojeg tvrtka *Siemens* prilaže uz proizvod PLC S7-200.

#### 3.2 Pokretanje Step 7 Micro/WIN SP9

Pokretanje programa izvršava se dvoklikom na ikonu prikazanu na slici 3.1.



Slika 3.1: Ikona za pokretanje *Step 7 Micro/WIN*

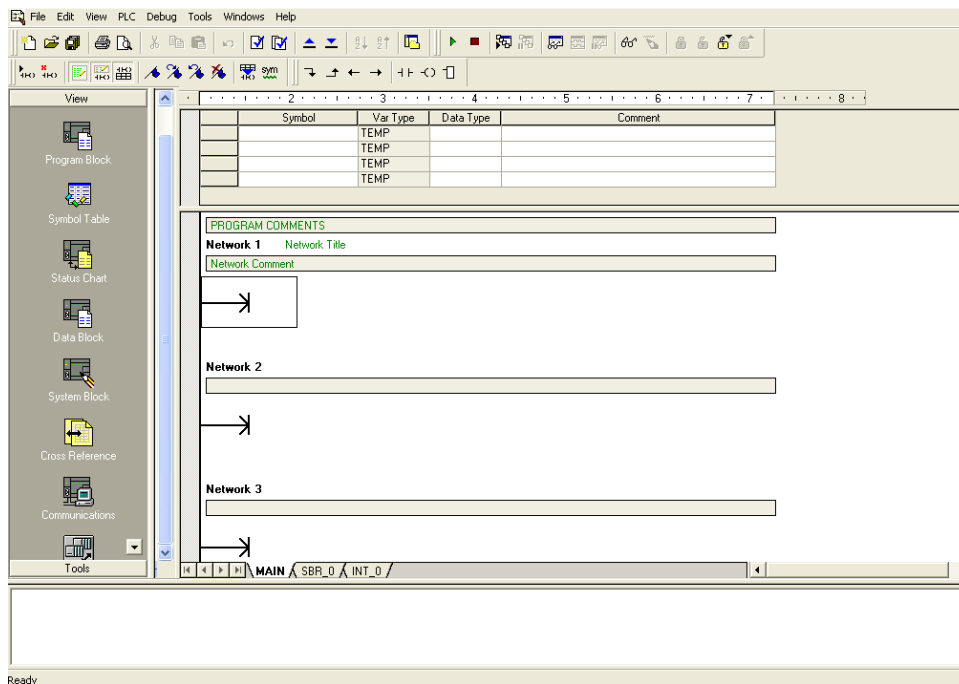
#### 3.3 Uspostava komunikacije PLC-a i računala

Za uspostavu komunikacije PLC-a i računala potreban je USB-PPI kabel prikazan na slici 3.2.



Slika 3.2: USB-PPI kabel [9]

Spajanjem PLC-a i PC-a kabelom i pokretanjem *Step 7 Micro/WIN* programskog alata, otvara se početni prozor za rad u programu prikazan na slici 3.3.

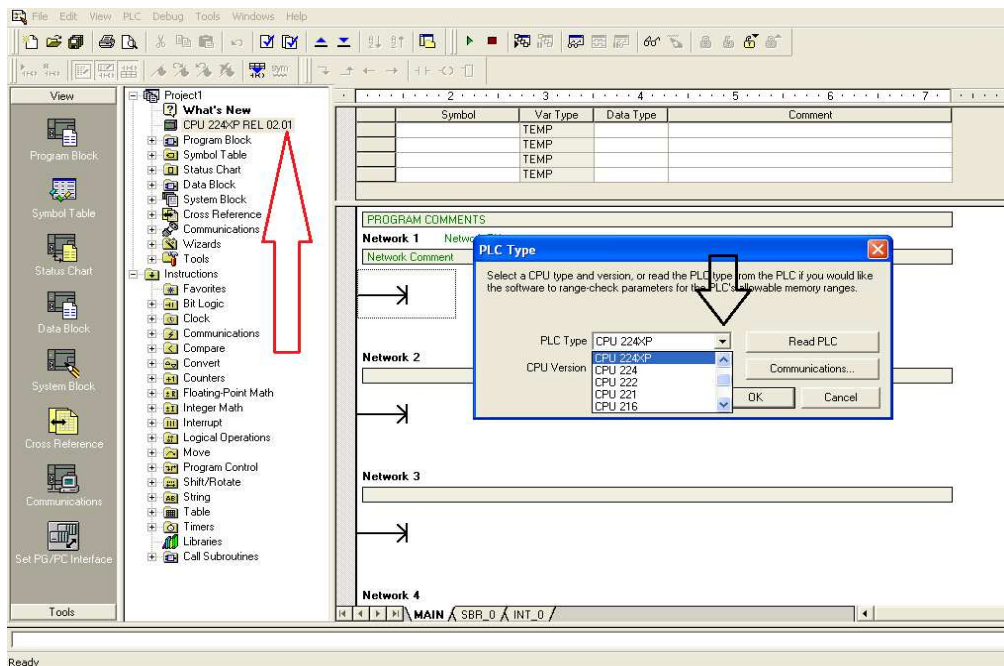


Slika 3.3: Početni zaslon programa *Step 7 Micro/WIN*

## 3.4 Definiranje PLC-a i početak rada

### 3.4.1 Definiranje PLC-a

Ulaskom u program *Step 7 Micro/WIN*, potrebno je definirati tip PLC-a koji se koristi za uspješno ostvarivanje komunikacije.



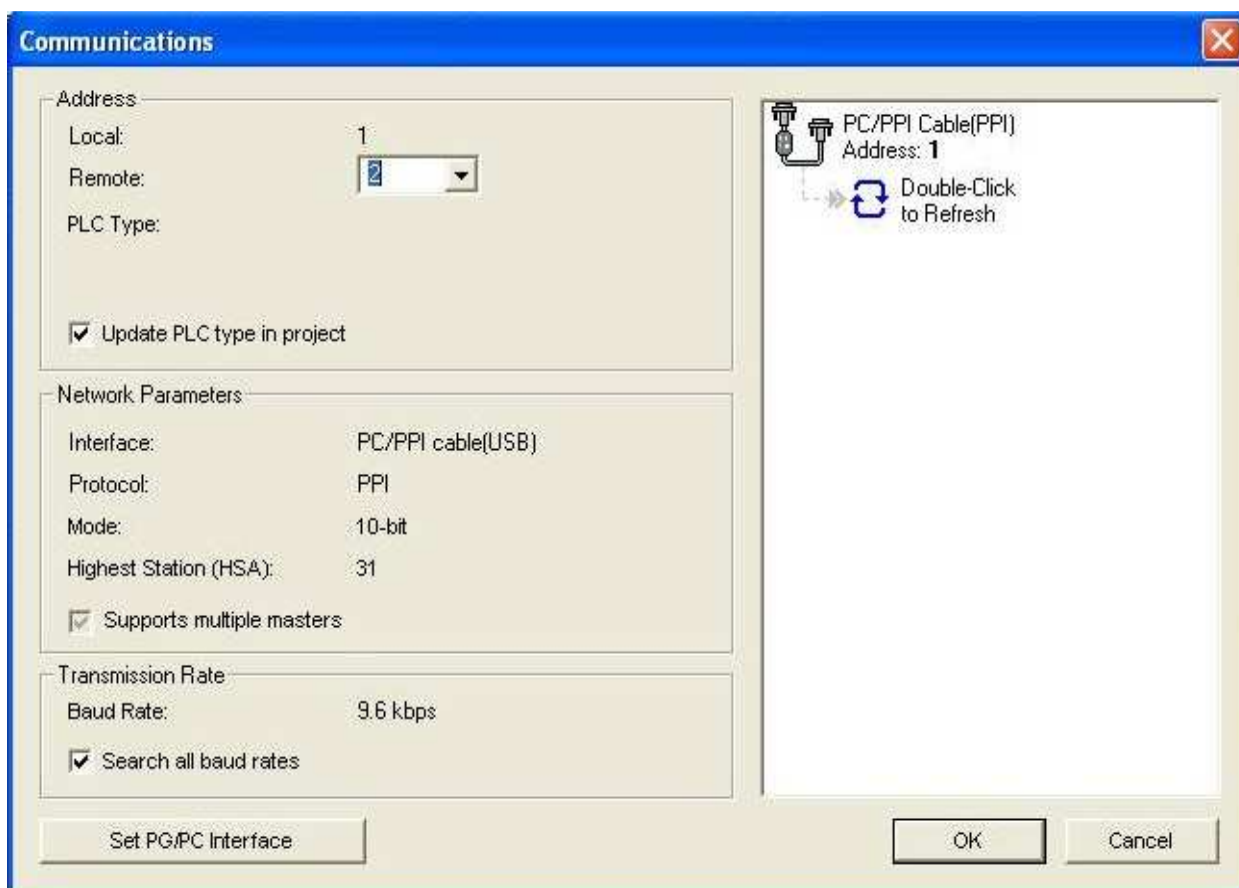
Slika 3.4: Definiranje PLC-a u programskom alatu *Step 7 Micro/WIN*

Dvoklikom na tab označen crvenom strelicom na slici 3.4 otvara se novi prozor. Opcija PLC Type nudi razne vrste PLC-a. Jedan klik otvara padajući izbornik s popisom različitih vrsta PLC-a. Za ovaj projekt odabran je CPU 224XP.

Sljedeći korak je podešavanje komunikacije. Klikom na „Communications“ otvara se novi prozor prikazan na slici 3.5. U ovom projektnom zadatku korištene su sljedeće postavke komunikacije:

- Brzina prijenosa podataka (eng. *Baud Rate*): 9.6 kbps
- Mrežno sučelje (eng. *Network Interface*): PC/PPI cable(USB)
- Mrežni protokol (eng. *Network Protocol*): PPI
- Lokalna adresa (eng. *Local Address*): 1
- Adresa za upravljanje na daljinu (eng. *Remote Address*): 2

Dvoklikom na tipku *Refresh* program automatski traži PLC spojen na računalo. Nakon što je pronađen željeni PLC, pristikom na tipku *OK* uspostavljena je komunikacija i moguće je prebacivati program sa računala na PLC i obratno.

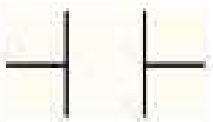
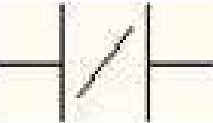

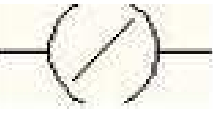

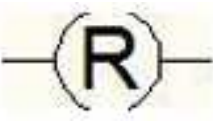




Slika 3.5: Postavke komunikacije u programskom alatu *Step 7 Micro/WIN*

### 3.5 Osnovni simboli korišteni u programu Step 7 Micro/WIN

Osnovni simboli korišteni u programskom okruženju *Step 7 Micro/WIN* prikazani su u tablici 3.1.

Tablica 3.1: Osnovni simboli u ljevičastom dijagramu i njihova namjena

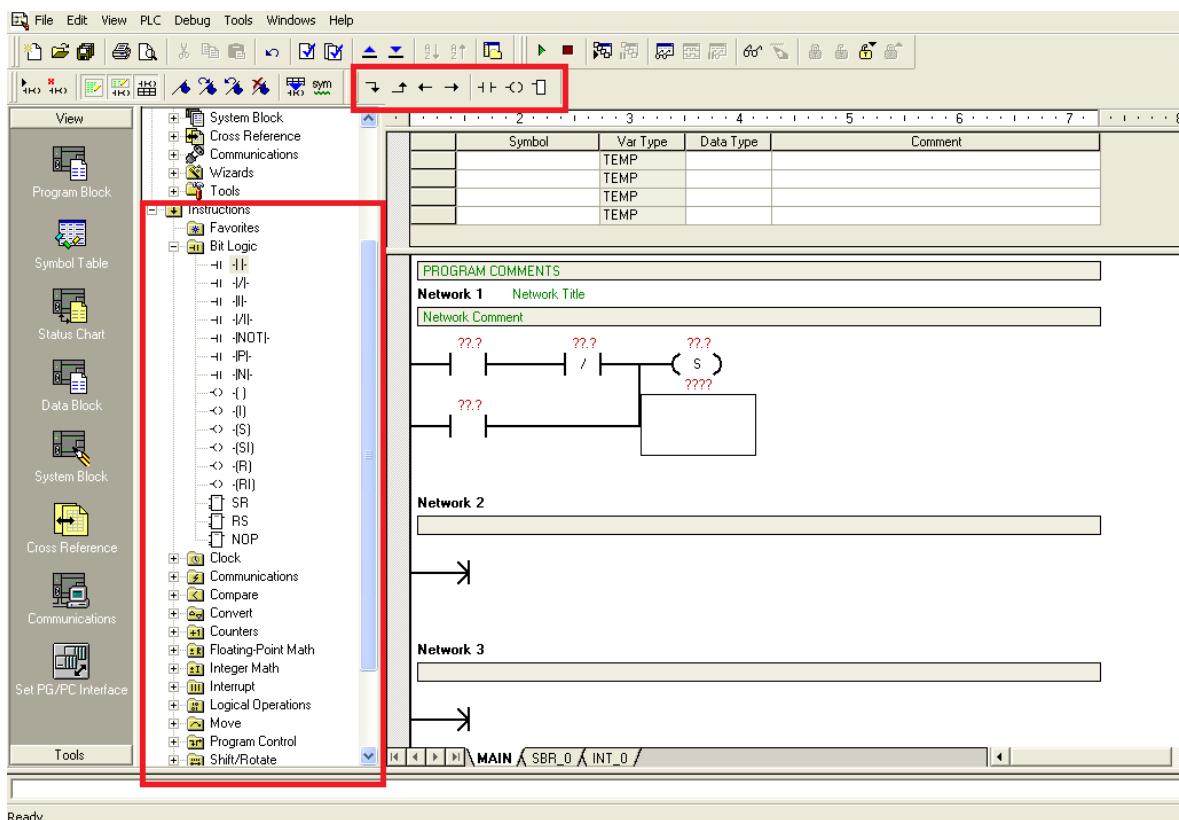
Simbol	Naziv simbola	Opis simbola
	Normalno otvoreni kontakt	Ako je kontakt aktivan, signal se propušta s lijeva na desno
	Normalno zatvoreni kontakt	Ako kontakt nije aktivan, signal se propušta s lijeva na desno
	Izlazni svitak	Ako se s lijeve strane pojavi signal, isti se prosljeđuje na svitak te je svitak uključen onoliko dugo koliko traje signal na lijevoj strani
	Invertirani izlazni svitak	Ako na lijevoj strani svitka nema signala, svitak ostaje uključen dok god se na lijevoj strani ne pojavi signal
	Set	Ako se s lijeve strane pojavi signal, uključuje se svitak i ostaje uključen dok god ga nešto ne isključi
	Reset	Ako se s lijeve strane pojavi signal, isključuje se svitak i ostaje isključen dok god ga nešto ne uključi
	Pozitivan brid	Pojavom pozitivnog brida na kontaktu, propušta se signal s lijeva na desno u trajanju duljine jednog ciklusa
	Negativan brid	Pojavom negativnog brida na kontaktu, propušta se signal s lijeva na desno u trajanju duljine jednog ciklusa



Simboli kao što su normalno otvoreni kontakt, normalno zatvoreni kontakt, pozitivni i negativni brid koriste se na ulazima PLC-a, dok se simboli kao izlazni svitak, invertirani izlazni svitak, *set* i *reset* izlaznog svitka koriste na izlazima i memorijskim riječima PLC-a. Normalno otvoreni kontakt prosljeđuje signal s lijeva na desno samo ako je pripadajuća adresa istom postavljena u visoko stanje, dok normalno zatvoreni kontakt prosljeđuje signal samo ako je na pripadajućoj adresi nisko stanje. Na primjer, ako je induktivni senzor povezan sa prvim ulazom PLC-a, normalno otvoreni kontakt sa dodijeljenom adresom %I0.1 propušta signal samo ako se predmet pojavi ispred senzora. Opis pojedinih simbola pisan je po uzoru na literaturu [10]

### 3.6 Početak pisanja programa

Nakon što je uspostavljena komunikacija, potrebno je napisati program. Na slici 3.6 označeni su izbornici za pisanje programa koji sadrže sve operacije potrebne za programiranje uređaja. U ovom projektu za programiranje PLC-a korišten je ljevičasti dijagram.



Slika 3.6: Izbornik operacija za programiranje PLC-a u programskom alatu *Step 7 Micro/Win*

Ovdje se nalaze operacije poput osnovnih logičkih funkcija kao što su normalno otvoreni kontakt, normalno zatvoreni kontakt, uključivanje ili isključivanje izlaza itd. Isto tako tu se nalaze i operacije konvertiranja, usporednih blokova, vremenskih brojača, blokova za manipuliranje memorijskim mjestima u PLC uređaju i sl.

### 3.7 Tablica simbola

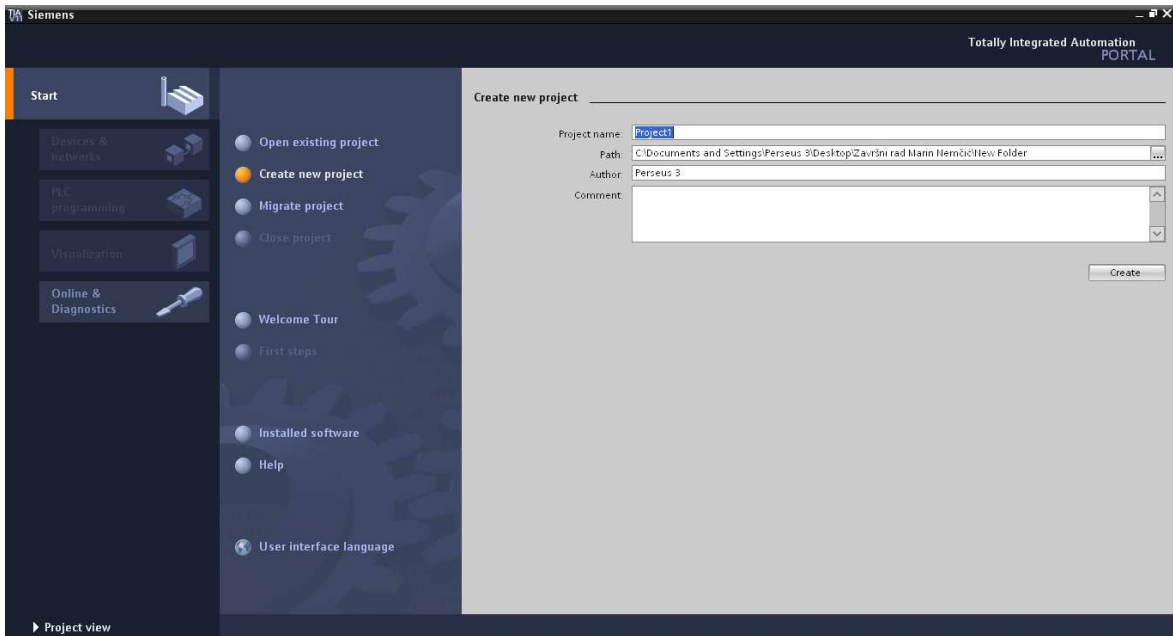
Tablica simbola (eng. *Symbol Table*) omogućava definiranje ulaza, izlaza, memorijskih bitova i sl. kako bi se osoba koja piše program lakše snalazila prilikom pisanja programa. Na slici 3.7 prikazana je tablica simbola korištena u ovom završnom radu. Jedan od primjera popunjavanja je da se u polje adrese upiše „I0.2“, što označava ulaz PLC-a na adresi dva prvog ulaznog bajta. U polje simbol (eng. *Symbol*) upisujemo naziv koji želimo koristiti za pripadajuću adresu. U ovom slučaju to je „X\_krajnji\_lijevo“ pošto se na toj adresi nalazi krajnji prekidač po X osi.

		Symbol	Address	Comment
1		Prvi_ciklus	SM0.1	Izvršava se samo prvi ciklus
2		Početni_bit	M0.0	Bit koji koristimo kod pokretanja programa
3		X_motor_gotov	M0.1	Aktiviran je krajnji prekidač X_lijevo
4		Y_motor_gotov	M0.2	Aktiviran je krajnji prekidač Y_iza
5		Z_motor_gotov	M0.3	Aktiviran je krajnji prekidač Z_gore
6			M0.4	
7		Start_T32	M1.0	Vremenski brojač T32
8		Start_T96	M1.1	Vremenski brojač T96
9		X_smjer_desno	M2.0	Motor X-osi vozi u desno
10		X_smjer_lijevo	M2.1	Motor X-osi vozi u lijevo
11		Y_smjer_naprijed	M2.2	Motor Y-osi vozi naprijed
12		Y_smjer_nazad	M2.3	Motor Y-osi vozi nazad
13		Z_smjer_dolje	M2.4	Motor Z-osi vozi dolje
14		Z_smjer_gore	M2.5	Motor Z-osi vozi gore
15		Kretnja	M10.0	Kreni prema predmetu
16		Kretnja_nazad	M11.0	Vrati predmet na pokretnu traku
17		Pneumatska_prihvatanica	M20.0	Aktiviraj čeljust pneumatske prihvatnice
18		Kretnja_nova	M22.0	Kreni ponovo po predmet
19		Kretnja_natrag_nova	M23.0	Vrati ponovo predmet nazad
20		Timer_T32	T32	Vremenski brojač T32
21		Timer_T96	T96	Vremenski brojač T96
22		X_brojanje_imp	VW0	Spremanje impulsa gibanja X-osi
23		Y_brojanje_imp	VW2	Spremanje impulsa gibanja Y-osi
24		Z_brojanje_imp	VW4	Spremanje impulsa gibanja Z-osi
25		Udaljenost_X	VW10	Udaljenost do predmeta po X-osi
26		Udaljenost_Y	VW12	Udaljenost do predmeta po Y-osi
27		Timer_signal	VW1000	Upravljanje brzine signala vremenskih brojača
28			MD0	
29			MD10	
30			MD20	
31		X_krajnji_lijevo	I0.2	Krajnji prekidač X os lijevi
32		Y_krajnji_iza	I0.5	Krajnji prekidač Y os iza

Slika 3.7: Tablica simbola u programu *Step 7 Micro/Win*

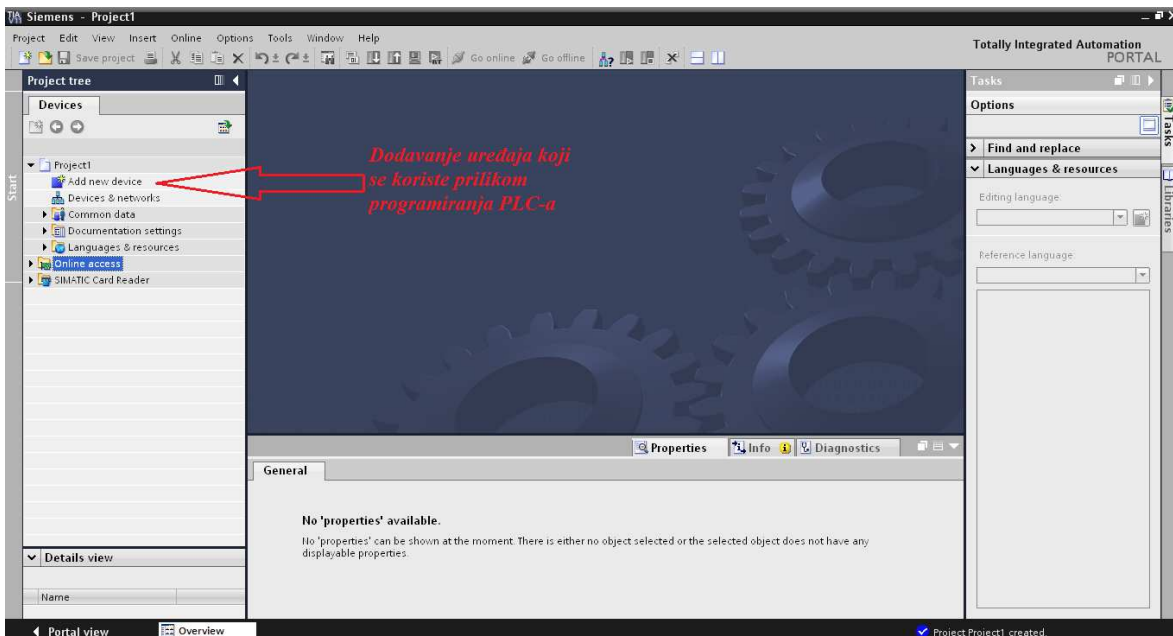
## 4. Programski alat TIA portal

TIA portal (eng. *Totally Integrated Automation Portal*) programski je alat koji se koristi za nadzor i upravljanje cijelim sustavom. Pokretanjem programa otvara se prozor za kreiranje novog projekta prikazan na slici 4.1.



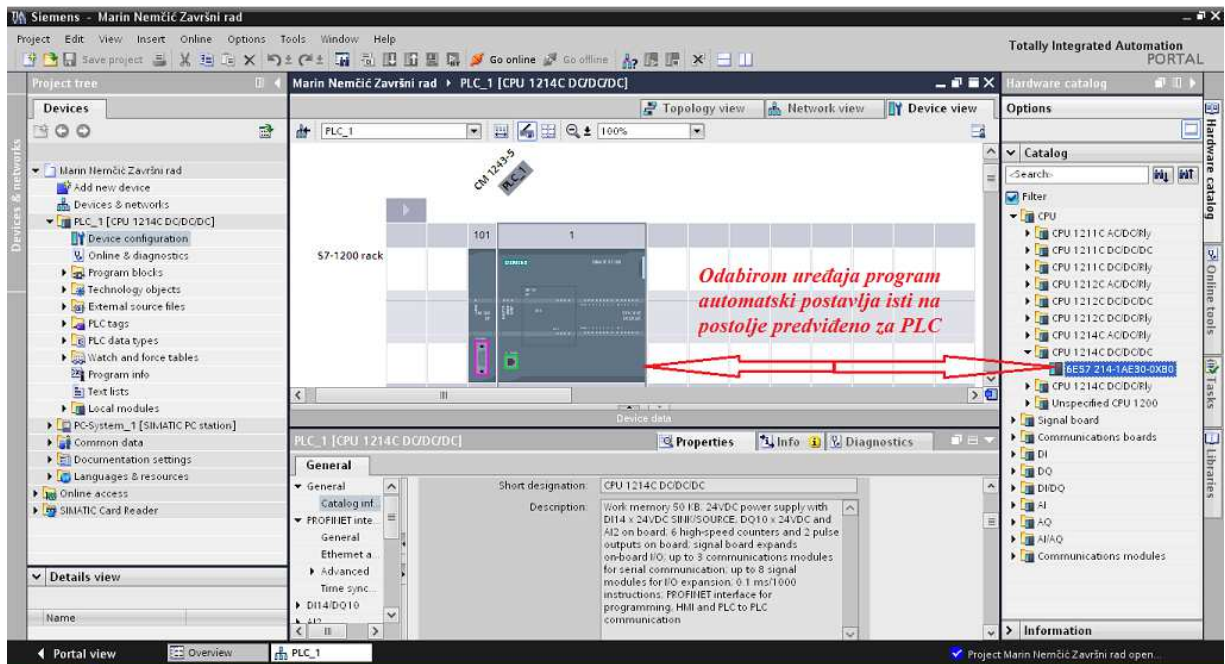
Slika 4.1: TIA portal kreiranje projekta

Kreiranjem novog projekta otvara se glavni prozor programa prikazan na slici 4.2. Potrebno je definirati korištene uređaje za programiranje PLC-a. Klikom na polje *Add new device* otvara se prozor dostupnih uređaja u programu. Konfiguracija uređaja opisana je u literaturi [11]



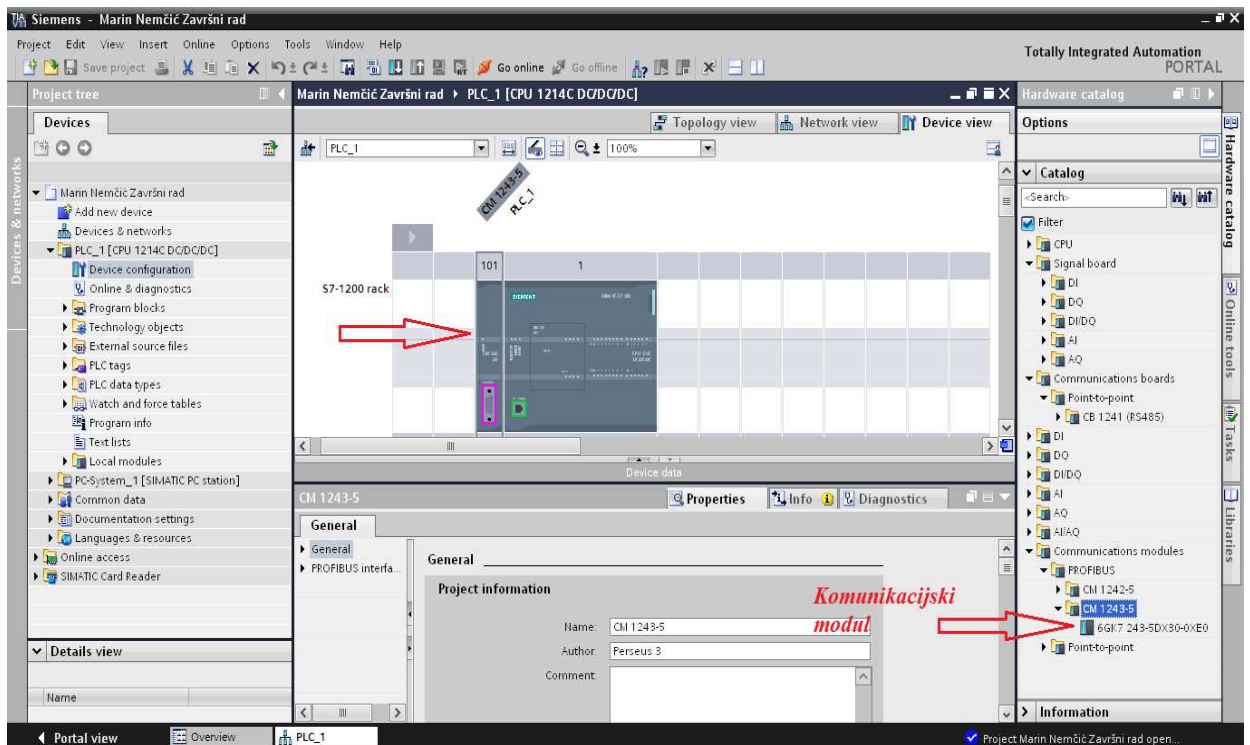
Slika 4.2 : Dodavanje novih uređaja

Prvi uređaj koji se definira je PLC. Na slici 4.3 prikazan je PLC uređaj korišten u ovom projektu, CPU 1214C DC/DC/DC.

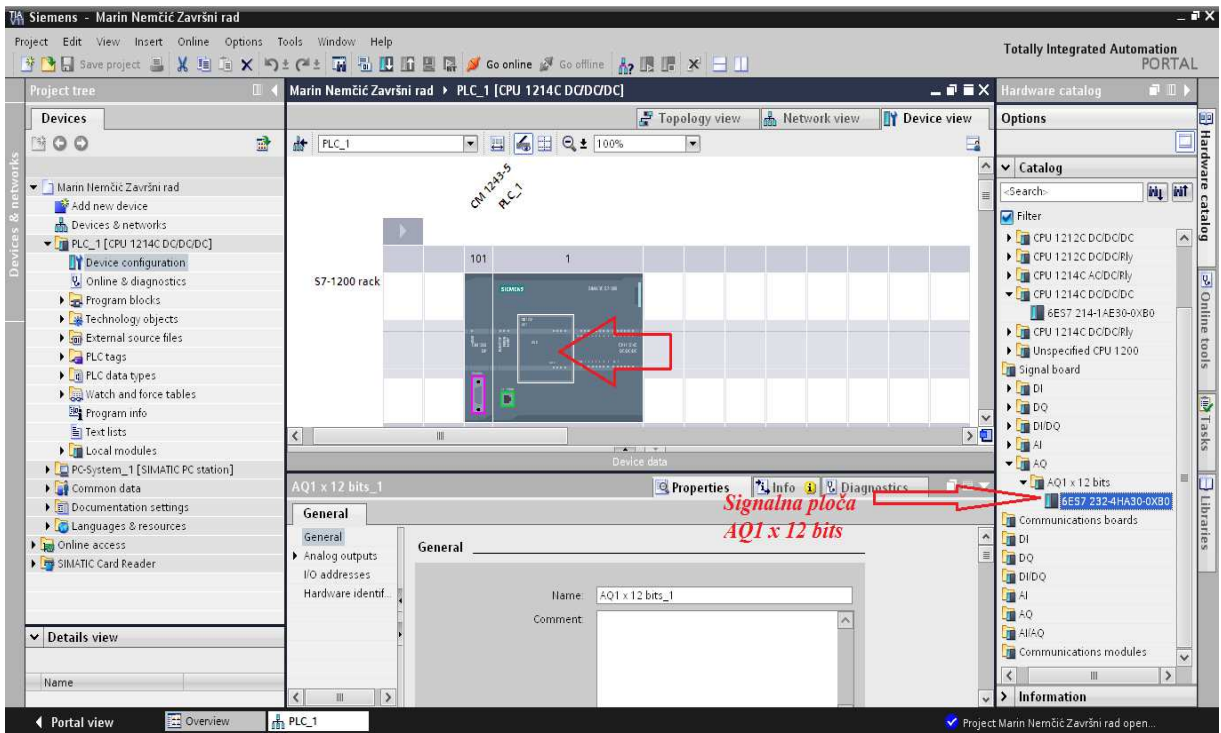


Slika 4.3: Odabir PLC uređaja

Isti princip koristi se za dodavanje svih ostalih komponenti koje se koriste. Na slikama 4.4, 4.5 i 4.6 prikazano je dodavanje svih komponenti kao što su komunikacijski moduli, signalne ploče i osobna računala korištena u radu.

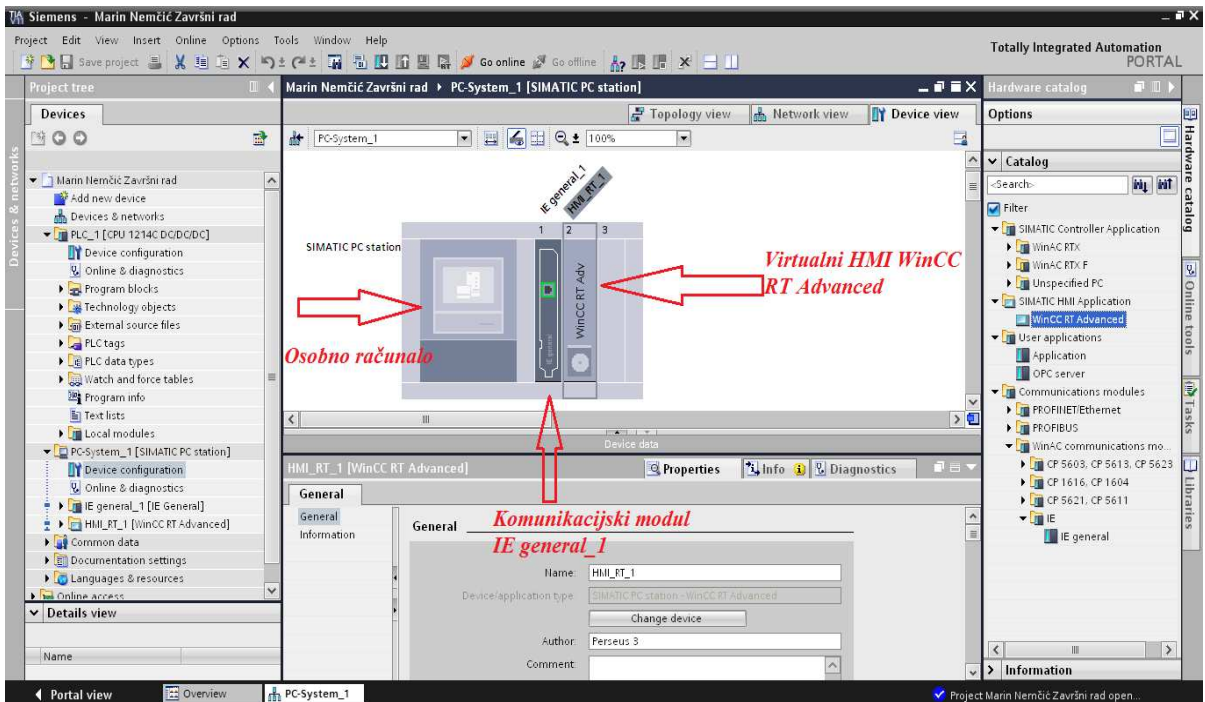


Slika 4.4: Komunikacijski modul CM 1243-5



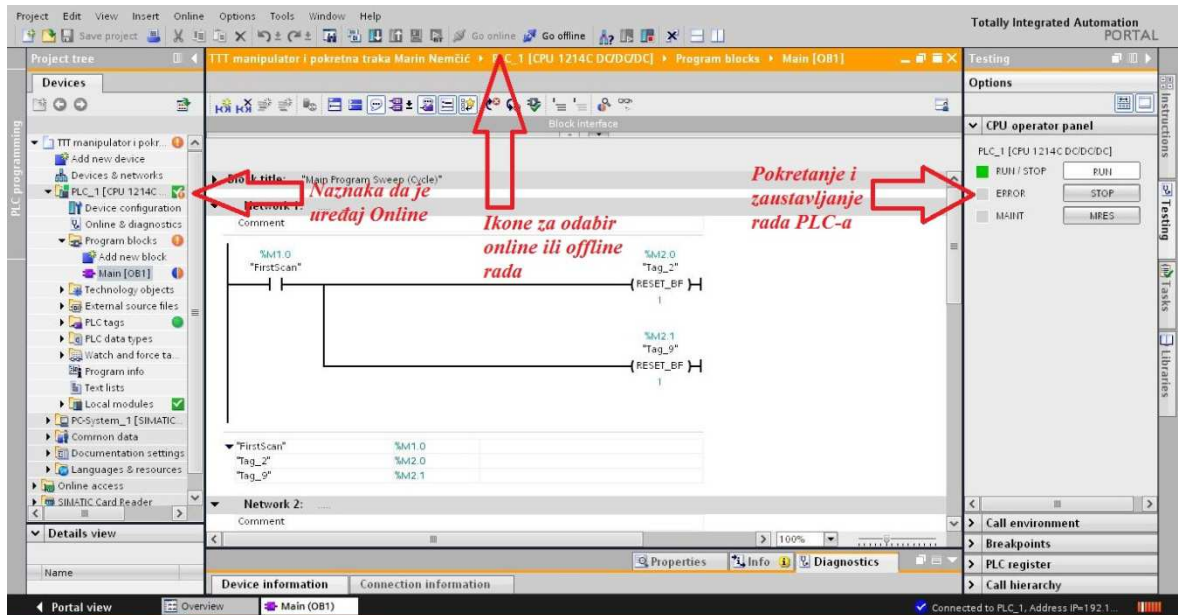
Slika 4.5: Signalna ploča *AO1 x 12 bits*

Također je potrebno i definirati osobno računalo korišteno za programiranje PLC uređaja. Uz osobno računalo, također se definira i komunikacija i program vizualizacije i simulacije.



Slika 4.6 Definiranje osobnog računala

Definiranjem svih potrebnih komponenti za rad potrebno je spojiti osobno računalo i PLC *Ethernet* kabelom. Nakon što je kabel spojen i IP adrese posložene, potrebno je pritisnuti tipku „Go online“ i uspostaviti komunikaciju. Slika 4.7 prikazuje *online* programiranje PLC uređaja i opisuje glavne dijelove programa koji se koriste za pokretanje i zaustavljanje rada PLC-a.

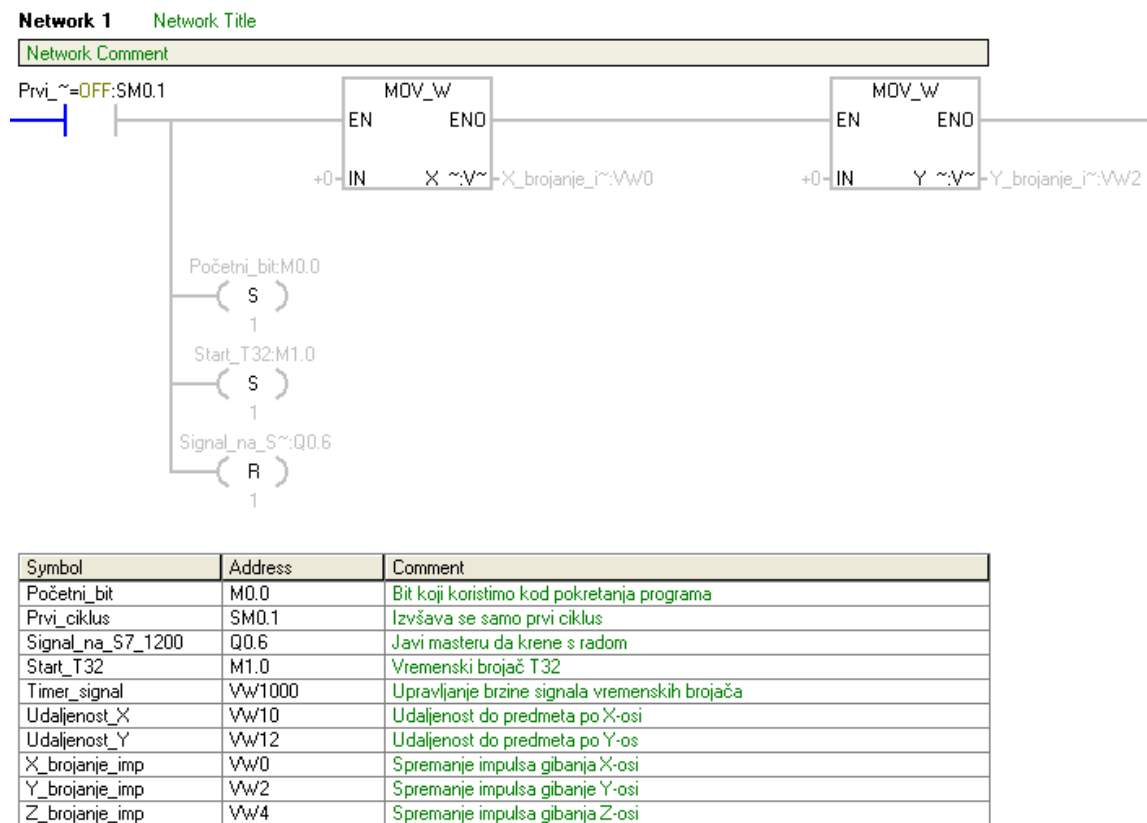


Slika 4.7: *Online* način rada TIA portal

## 5. Step 7 Micro/WIN program upravljanja TTT manipulatora

### 5.1 Prvi ciklus programa i definiranje korištenih memorijskih mjesta

U prvom ciklusu programa potrebno je sva korištena memorijska mjesta postaviti u nulu. Na slici 5.1 prikazan je način postavljanja svih korištenih bitova u logičku „0“. Ako je PLC već radio, u memorijska mjesta koja se koriste za poziciju i impulse gibanja već će biti upisana neka vrijednost. Kod ponovnog pokretanja PLC-a sva ta memorijska mjesta moraju biti postavljena u nulu. Sistemski bit SM0.1 izvršava se samo u prvom ciklusu programa i stoga se koristi u prvom *network*-u programa.

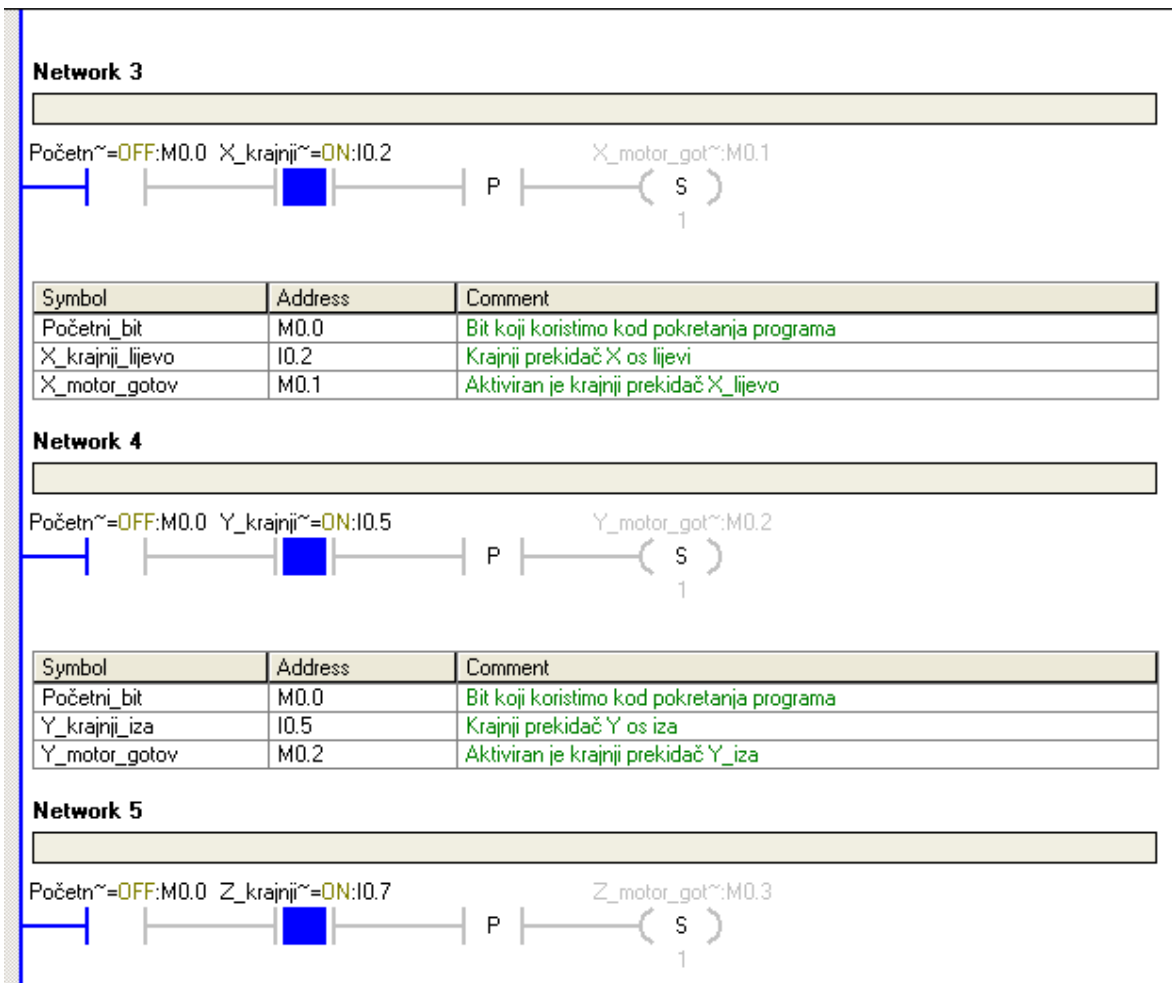


Slika 5.1: Definiranje korištenih memorijskih mjesta

Memorijsko mjesto VW0 koristi se za brojanje impulsa koje manipulator prođe po X osi. VW2 koristi se za brojanje impulsa koje manipulator prođe po Y osi, a VW4 se koristi za Z os. U programu je na početku isto tako postavljena varijabilna memorija VW1000 koja služi za upravljanje brzine signala vremenskih brojača. Ovo mjesto opisano je kasnije u programu. Također se u prvom ciklusu nalaze i memorijska mjesta VW10 i VW12 koja su postavljena u nulu i objašnjena kasnije u radu. Također, postavljen je memorijski bit M0.0 i memorijski bit M1.0. M0.0 koristi se kod početnog izvođenja programa, a M1.0 pokreće rad vremenskih brojača.

## 5.2 Kretanje manipulatora prema krajnjim prekidačima

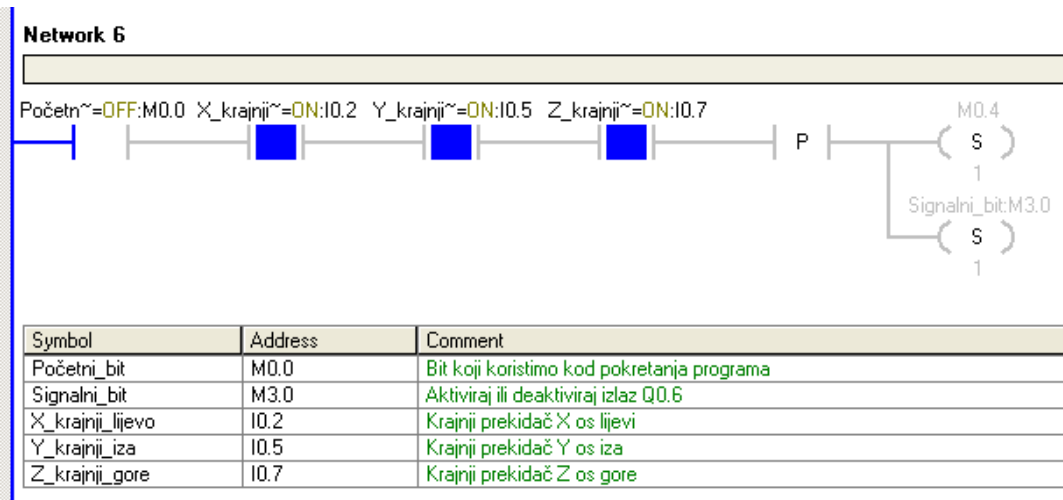
Kada se PLC postavi u „*RUN mode*“, manipulator se kreće prema krajnjim prekidačima. *Networks 2, 3 i 4* prikazani na slici 5.2 služe za to da kada se manipulator kreće po pojedinoj osi, zaustavi čim aktivira krajnji prekidač na pripadajućoj osi postavljanjem bitova u svakom networku. Na ulaze I0.2, I0.5 i I0.7 spojeni su krajnji prekidači korišteni kod pozicioniranja manipulatora.



Slika 5.2: Aktiviranje ulaza spojenih na krajnje prekidače

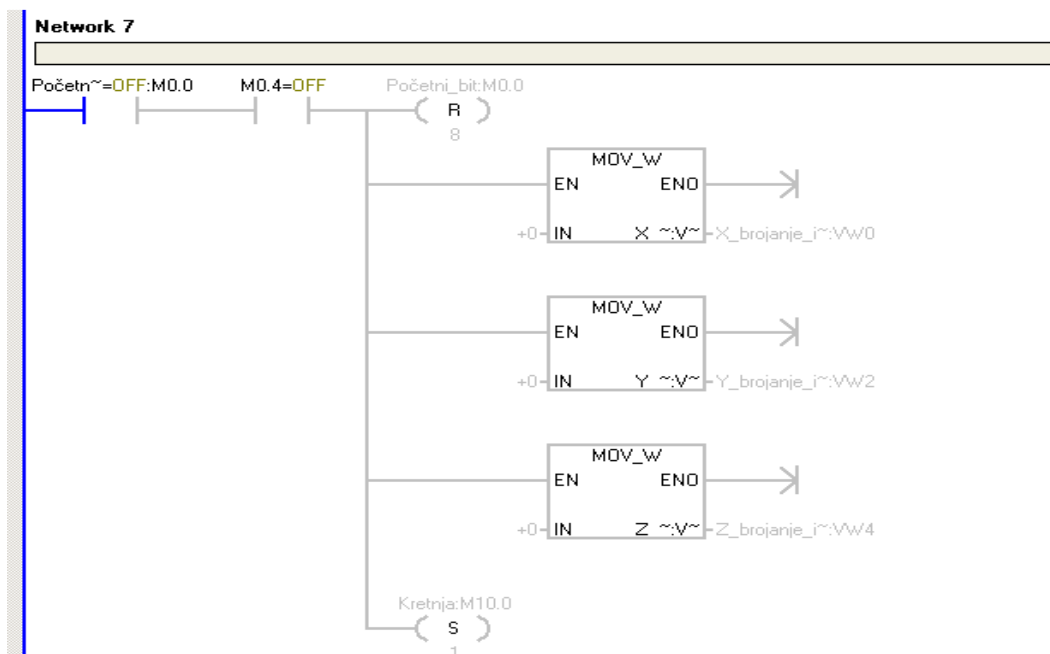


Postavljanjem bitova dobiva se informacija da je manipulator aktivirao sva tri krajnja prekidača. Kada su sva tri bita aktivirana, postavlja se memorijski bit M0.4 prikazan na slici 5.3 koji se koristi dalje u programu.



Slika 5.3: Aktiviranje memorijskog bita M0.4

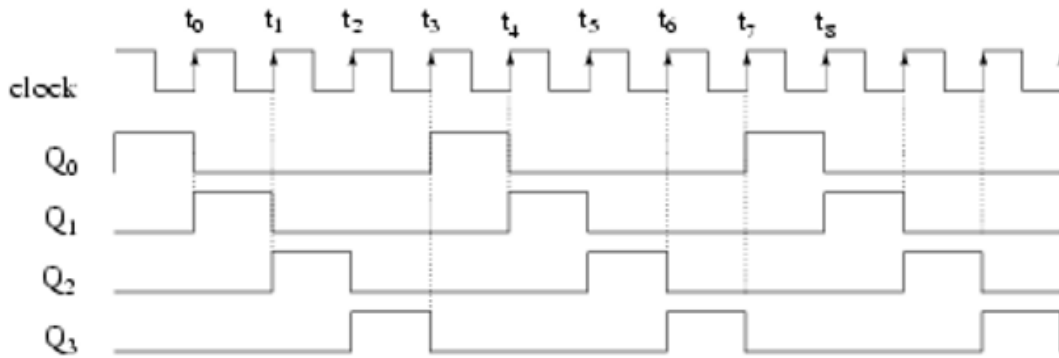
Network 7 prikazan na slici 5.4 služi za resetiranje memorijskog bita M0.0 koji se koristi za početno izvođenje programa. Memorijski bit M0.4 postavljen je u visoko stanje samo ako su sva tri krajnja prekidača u logičkoj jedinici. Memorijska mjesta VW0, VW2 i VW4 koriste se za brojanje impulsa koje manipulator prijeđe po pojedinoj osi, a pošto se već izvršavalo neko gibanje u njih je upisan neki broj, stoga ih je potrebno ponovno postaviti u nulu.



Slika 5.4: Kraj pozicioniranja manipulatora i postavljanje memorijskih varijabli

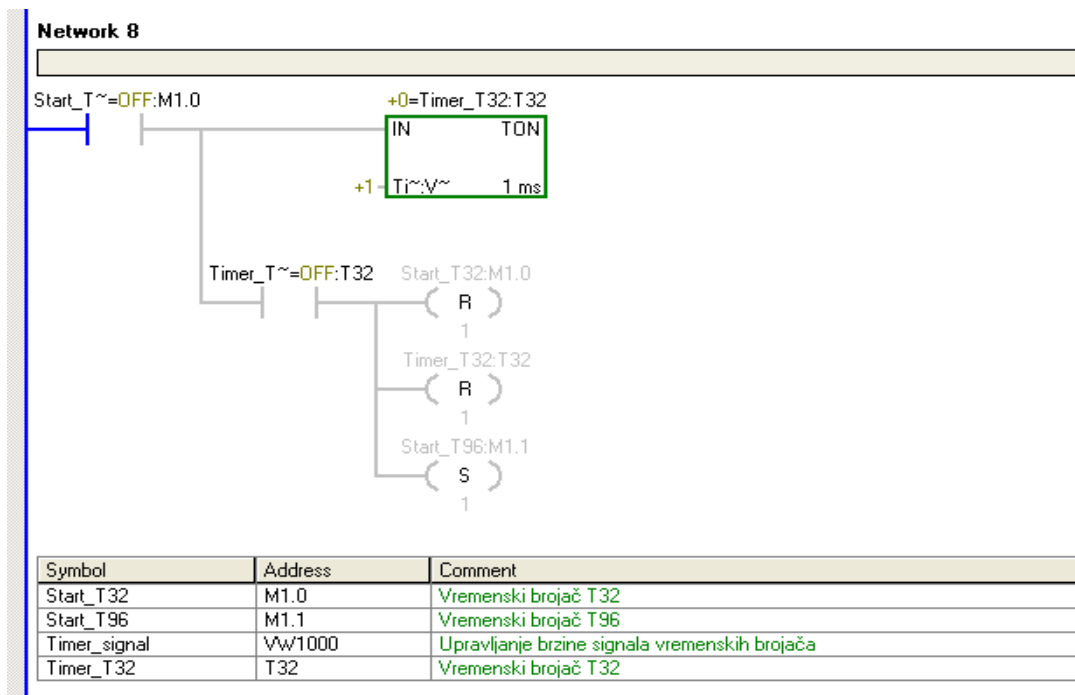
### 5.3 Generiranje impulsa pomoću timera

Na slici 5.5 prikazani su pravokutni impulsi potrebni za pokretanje koračnih motora.



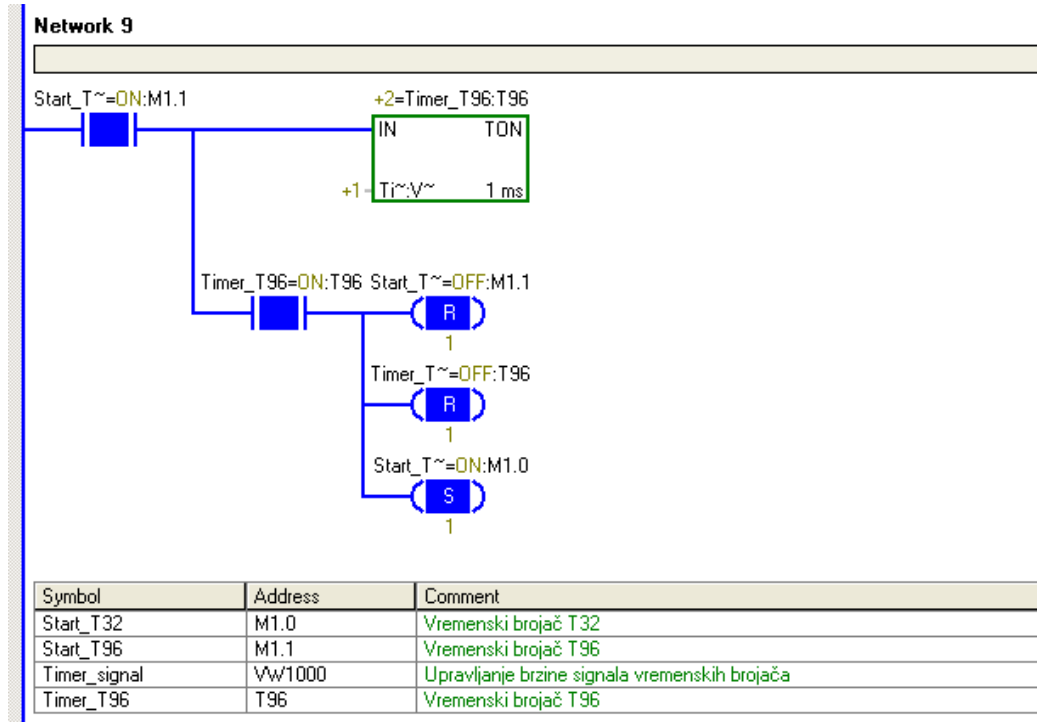
Slika 5.5: Pravokutni impulsi

Za ostvarivanje pravokutnih impulsa (eng. *Clock*) potrebna su dva vremenska brojača (eng. *Timer*). Vremenski brojači koji se koriste su TON tipa, T32 i T96, te imaju vremensku bazu od 1milisekunde. Ulazna varijabla korištena kod *timer*-a je VW1000. Na početku programa u tu varijablu postavljen je broj 1, taj broj množi se s vremenskom bazom *timer*-a i označava vrijeme nakon kojega će provesti signal. *Timer* T32 prikazan na slici 5.6 osigurava visoko stanje vođenja. Vremenski brojači i njihov način rada opisan je u [12].



Slika 5.6: Vremenski brojač T32

Memorijski bit M1.0 koji je postavljen na početku programa aktivira *timer* T32. Nakon što vremenski brojač izbroji zadane milisekunde, resetira bit M1.0 i T32 kako bi se zaustavilo brojanje. Memorijski bit M1.1 postavlja se za daljni nalog uključivanja *timer*-a T96 prikazanog na slici 5.7.

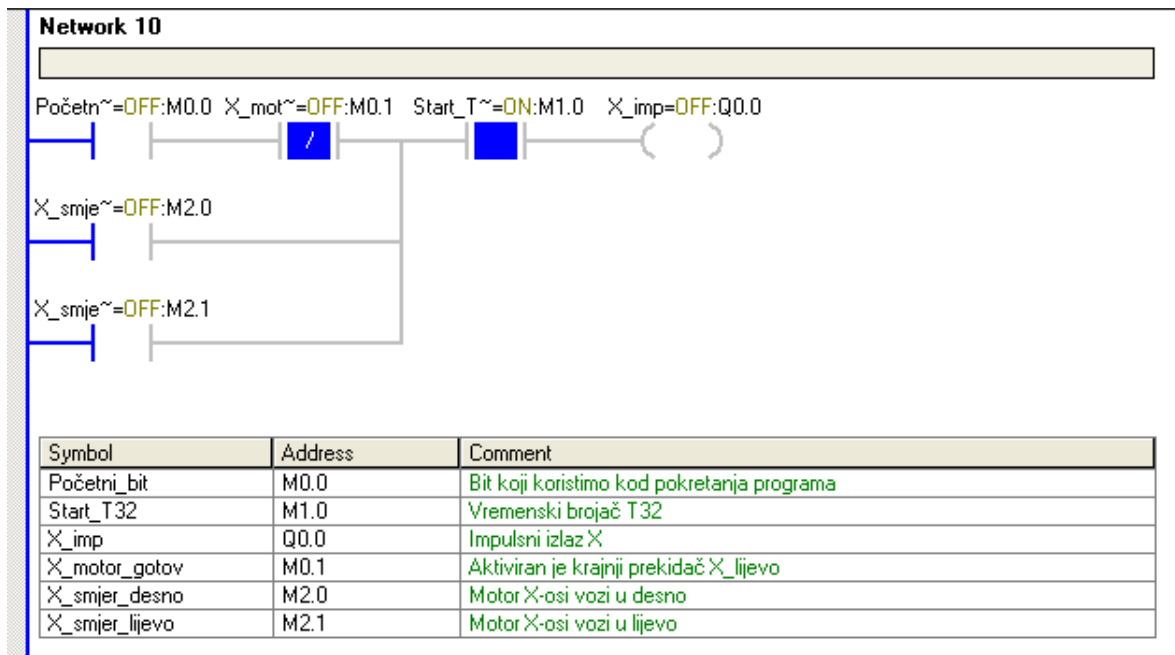


Slika 5.7: Vremenski brojač T96

Memorijski bit M1.1 uključuje vremenski brojač T96. Ulazna varijabla mora biti ista kao i kod vremenskog brojača T32 kako bi se dobio pravokutni signal. Nakon što vremenski brojač T96 izbroji zadane milisekudne, resetira korišteni memorijski bit M1.1, resetira vremenski brojač T96 kako ne bi i dalje nastavio brojati, a ponovno postavlja memorijski bit M1.0 kako bi *timer* T32 počeo brojati. *Timer* T96 osigurava nisko stanje vođenja. Ova dva networka rješavaju pravokutni signal potreban za koračne motore.

#### 5.4 Generiranje signala za motore

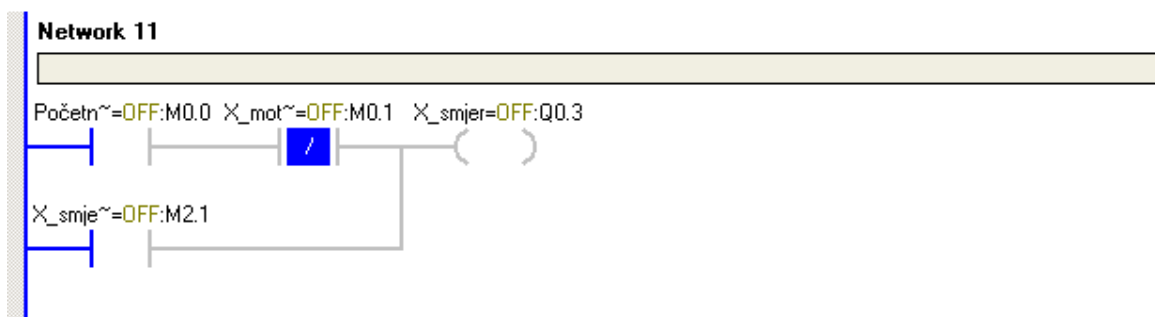
*Driver*-i za motore zahtjevaju dva signala, broj impulsa i smjer vrtnje motora. Na izlaze Q0.0, Q0.1 i Q0.2 dovode se impulsi za gibanje po pojedinoj osi. Izlaz Q0.0 povezan je s gibanjem po X osi, Q0.1 s gibanjem po Y osi i Q0.2 s gibanjem po Z osi. Nadalje, izlazi Q0.3, Q0.4 i Q0.5 određuju smjer vrtnje pojedinih motora, ovisno da li je izlaz u logičkoj „0“ ili „1“. *Network* 9 na slici 5.8 prikazuje dovođenje impulsa na izlaz Q0.0.



Slika 5.8: Izlaz Q0.0 povezan s impulsima motora po X osi

Memorijski bit M0.1 je negiran jer se postavlja u logičku jedinicu samo onda kada manipulator aktivira krajnji prekidač spojen na ulaz PLC-a I0.2. Ovaj dio povezan s *network*-om 2 isto tako osigurava da će se motori ugastiti kada se aktivira krajnji prekidač na ulazu I0.2 i time osigurava daljnju kretanju. U seriju je spojen i memorijski bit M1.0 koji je spojen na *timer* T32 jer osigurava visoko stanje vođenja. Bitovi M2.0 i M2.1 spojeni su paralelno i koriste se za pojedini smjer vrtnje motora. Koriste se kasnije u programu.

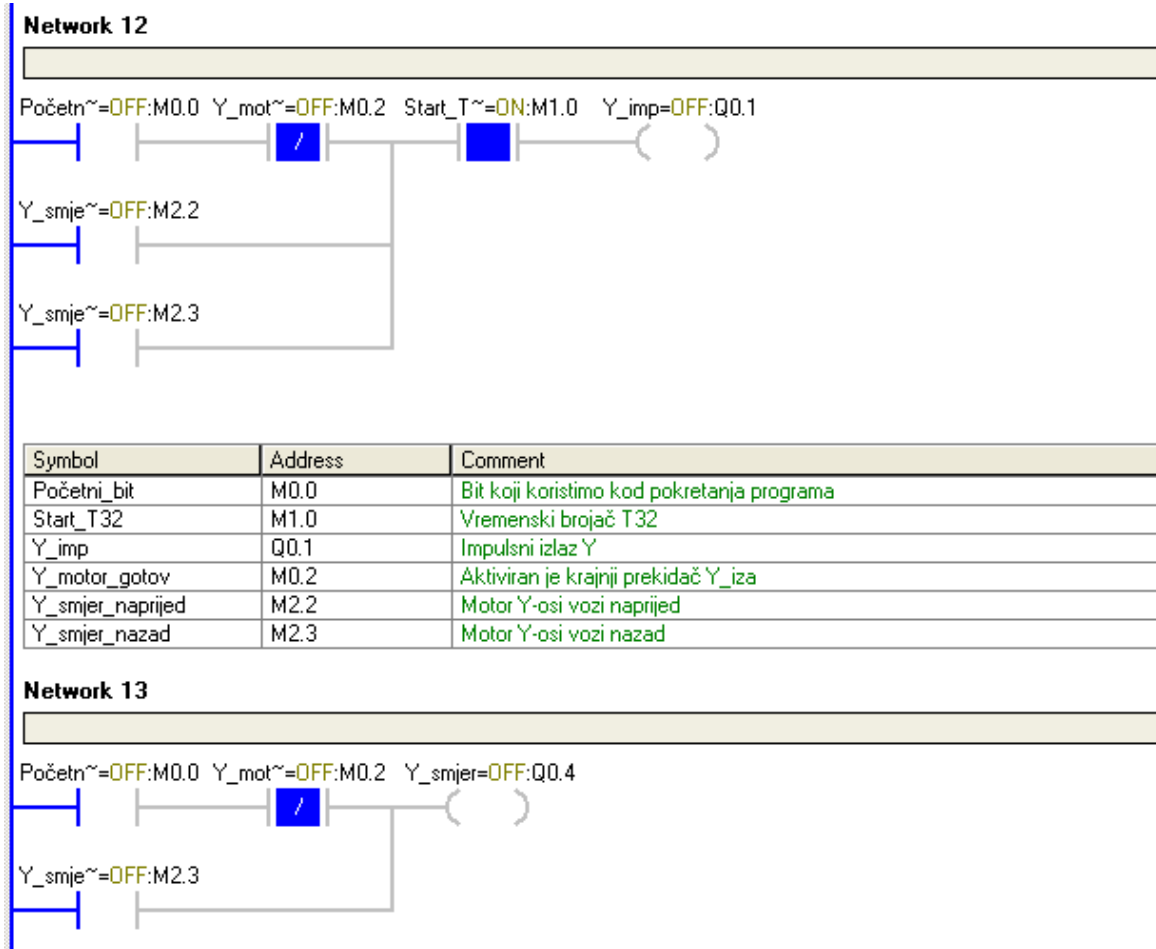
Nakon što je definiran impulsni izlaz Q0.0, potrebno je definirati smjer vrtnje motora na izlazu Q0.3. Isti je prikazan na slici 5.9.



Slika 5.9: Izlaz Q0.3 povezan s kretanjom motora X osi

Ako je izlaz Q0.3 u logičkoj jedinici motor se vrti u jednu stranu, a ako je u logičkoj nuli vrti se u drugu stranu. Bitovi M0.0 i M0.1 koriste se samo onda kada se manipulator pozicionira, a u drugom slučaju koristi se bit M2.1 kada se manipulator kreće prema zadanoj lokaciji koja je objašnjena kasnije u radu.

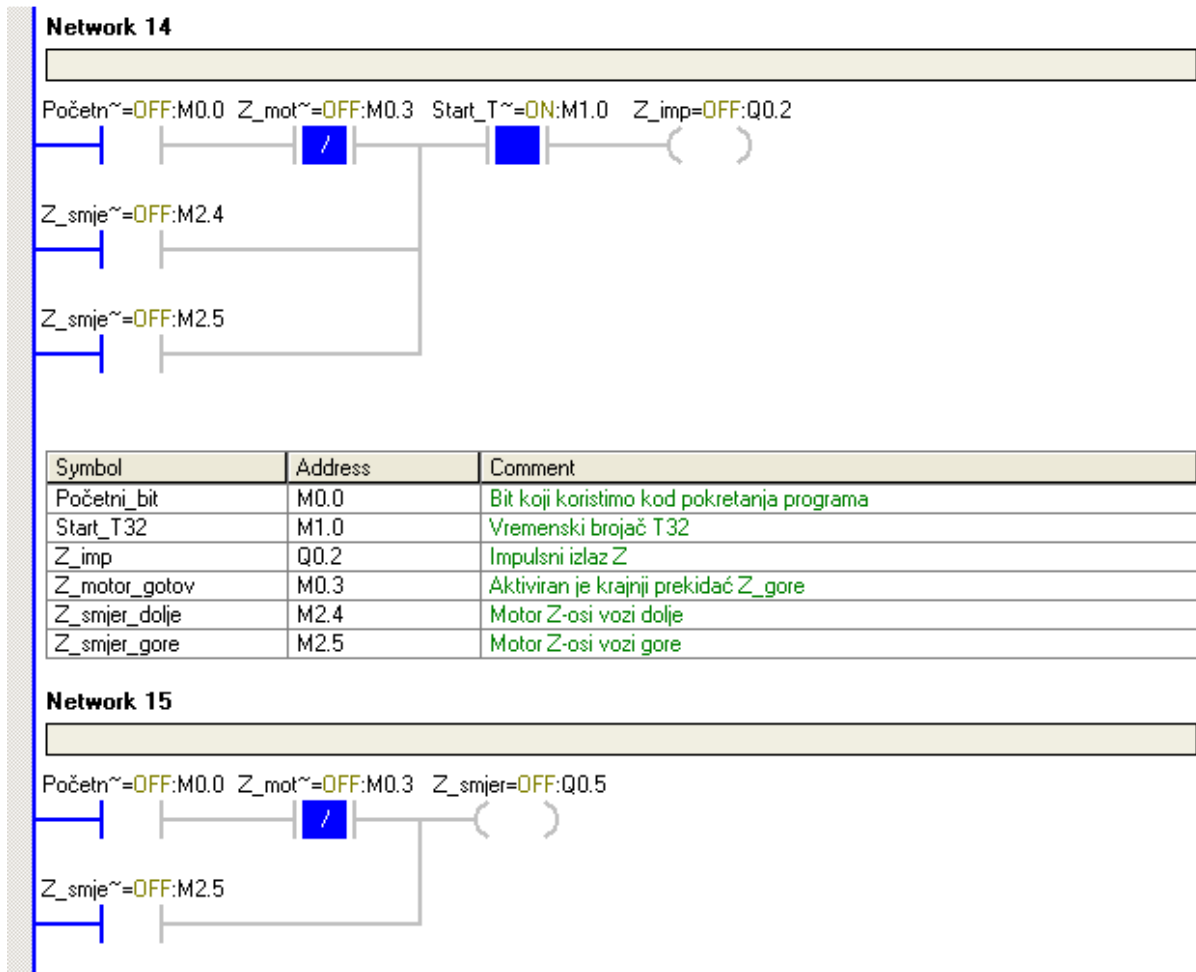
*Network 11* i *network 12* prikazani na slici 5.10 baziraju se na istom principu kao i prošla dva *network*-a, ali sada se upravlja impulsima i smjerom Y osi.



Slika 5.10: Izlazi Q0.1 i Q0.4 povezani s kretnjom i impulsima Y osi

Za kretnju po Y osi, na izlaz Q0.1 dovode se impulsi, a na izlaz Q0.4 smjer vrtnje motora.

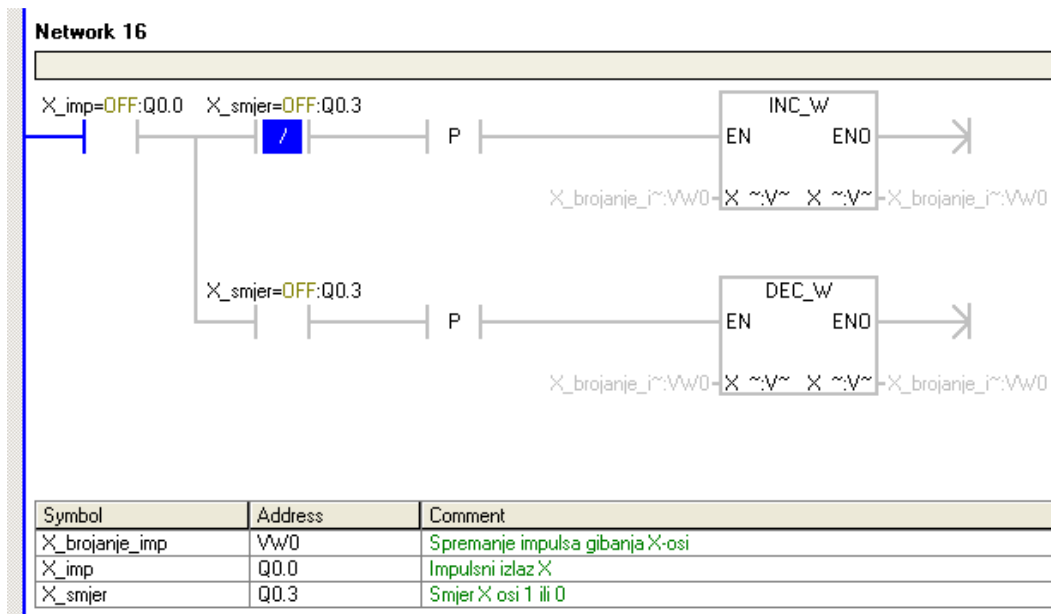
Network 13 i 14 prikazani na slici 5.11 imaju istu svrhu kao i prošla četiri *network*-a, a služe za impulse i smjer kretnje Z osi.



Slika 5.11: Izlazi Q0.2 i Q0.5 povezani s kretnjom i impulsima motora Z osi

## 5.5 Zbrajanje i oduzimanje impulsa pojedinih osi

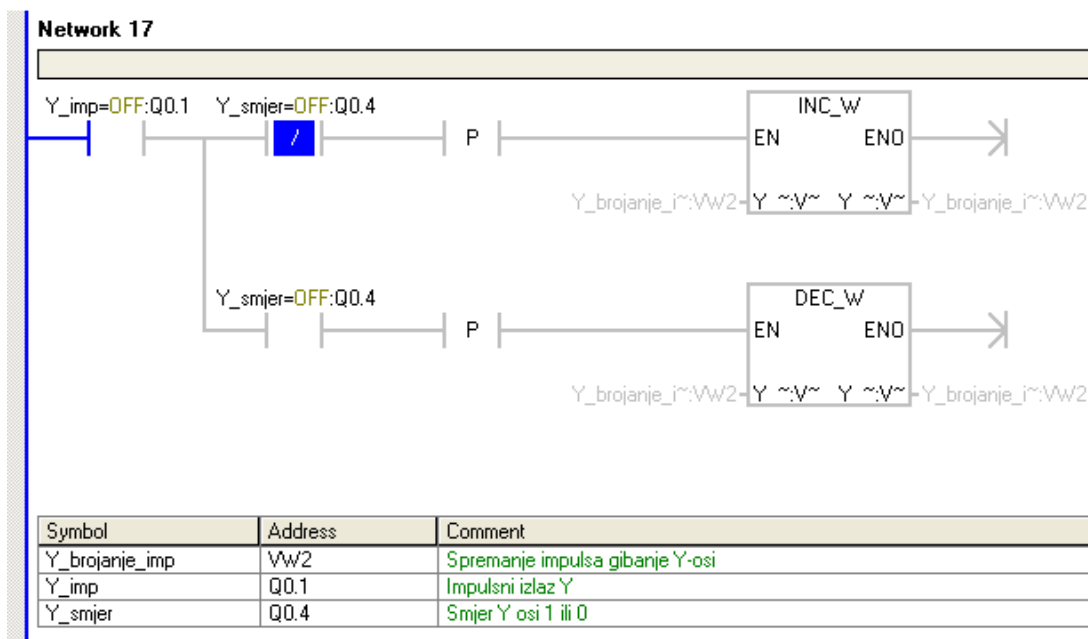
Kada manipulator vrši neko gibanje, impulsi se moraju zbrajati, odnosno oduzimati kako bi se upravljalo manipulatorom. Varijabilna memorija VW0 koja je korištena na početku programa služi za pohranjivanje impulsa gibanja X osi. Memorija VW2 koristi se za pohranjivanje impulsa gibanja Y osi, a memorija VW4 za pohranjivanje impulsa Z osi. Network 16 prikazan na slici 5.12 aktivan je kada se manipulator kreće po X osi.



Slika 5.12: Zbrajanje i oduzimanje impulsa X osi

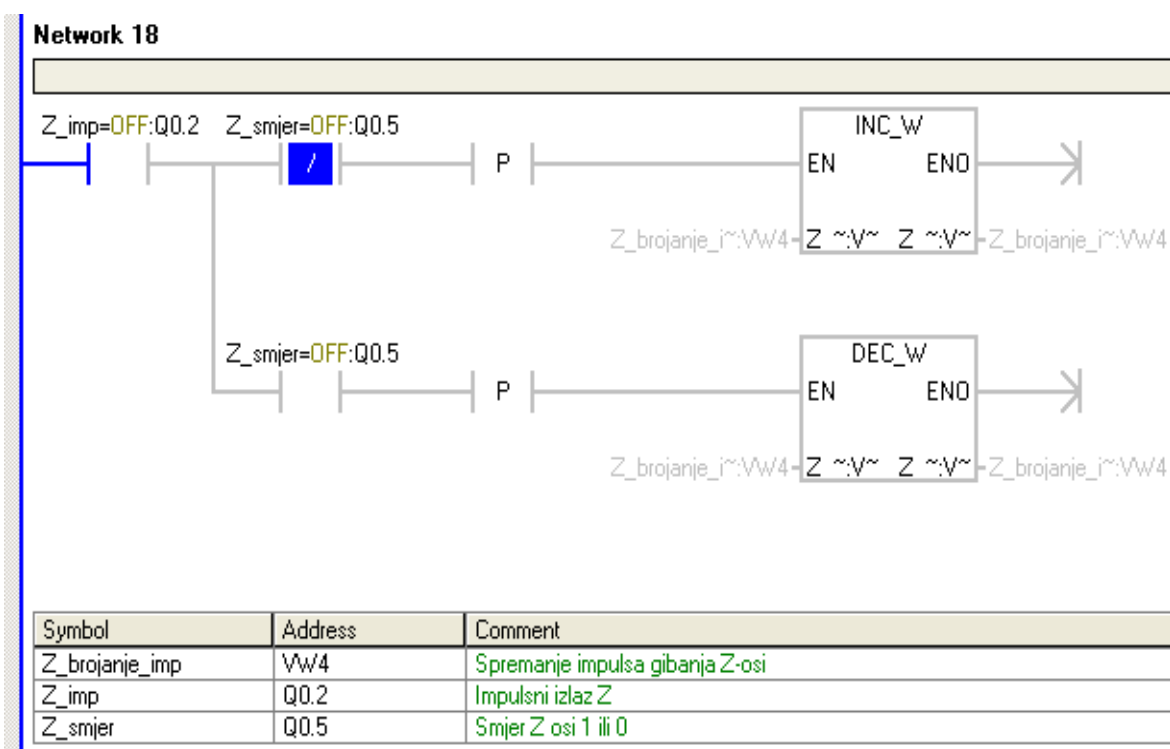
Izlaz Q0.0 generator je impulsa X osi, a izlaz Q0.3 određuje smjer vrtnje motora po X osi. Ovisno da li je izlaz Q0.3 u visokom ili niskom stanju definira se smjer vrtnje motora, stoga ako je izlaz Q0.3 negiran, u seriju je spojen *INC\_W* blok koji zbraja impulse, a u suprotnom kada je Q0.3 u visokom stanju, koristi se blok *DEC\_W* za oduzimanje impulsa X osi.

*Network 16* na slici 5.13 ima istu zadaću kao i *Network 15*. Sada se radi o zbrajanju i oduzimanja impulsa Y-osi.



Slika 5.13: Zbrajanje i oduzimanje impulsa Y osi

Network 18 na slici 5.14 odnosi se na zbrajanje i oduzimanje impulsa Z osi.



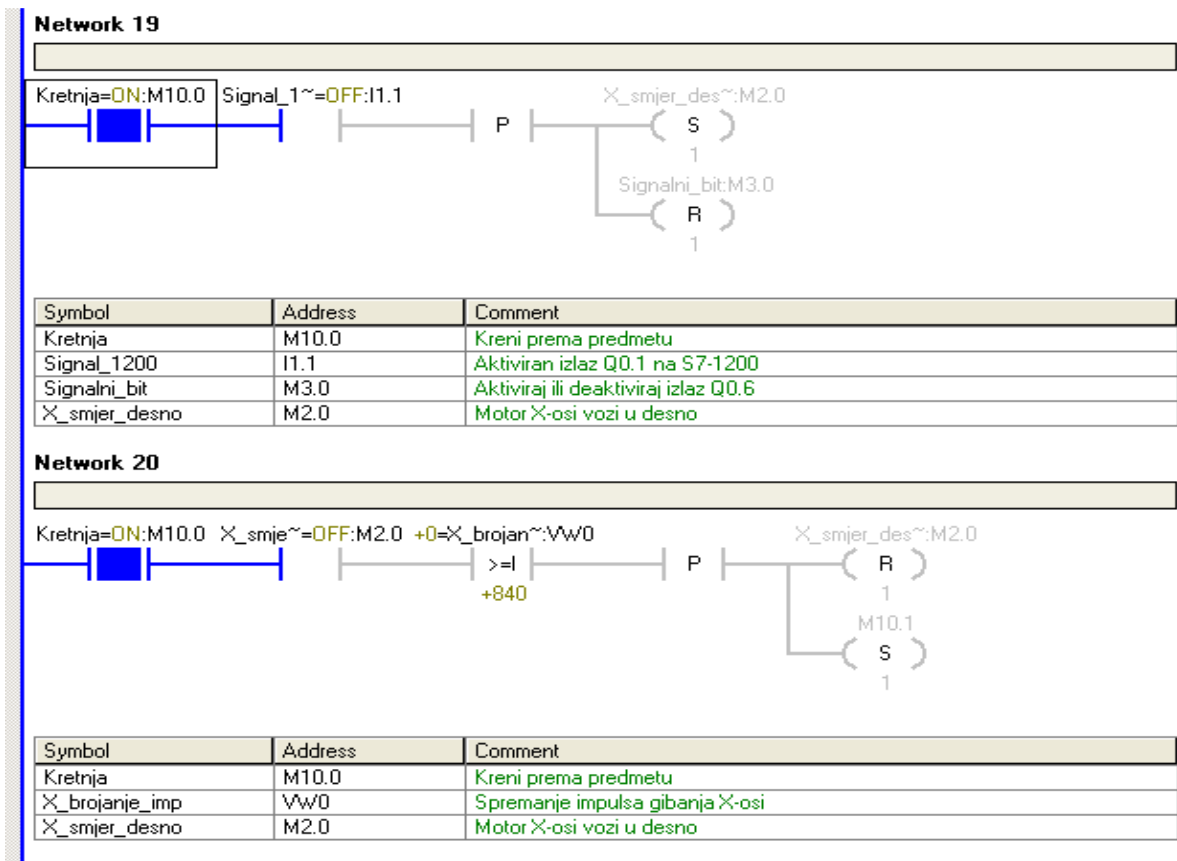
Slika 5.14: Zbrajanje i oduzimanje impulsa Z osi

## 5.6 Kretanje manipulatora prema poziciji predmeta

### 5.6.1 Kretanja po X osi

Glavni zadatak manipulatora je prihvatiti predmet sa postolja na kojem se isti nalazi. Da bi to bilo moguće, potrebno je znati udaljenost na kojoj se taj predmet nalazi od početne pozicije manipulatora. Ta udaljenost se zatim pretvara u impulse koje manipulator mora prijeći. Na slici 5.15 prikazana su dva *network*-a koja upravljaju kretanjem manipulatora po X osi. Memorijski bit M10.0 aktivira se nakon što se manipulator pozicionira u svoj početni položaj. Taj bit, uz uvjet da je stigao signal s *master PLC*-a, aktivira bit M2.0 koji određuje smjer u kojem se motor po kreće po X osi. Usporedni blok koji se koristi u *network* 19 pokazuje za koliko se impulsa manipulator po X osi mora pomaknuti. Kada se manipulator po X osi pozicionira za zadani broj impulsa, gasi se memorijski bit M2.0 koji ujedno i gasi motor.

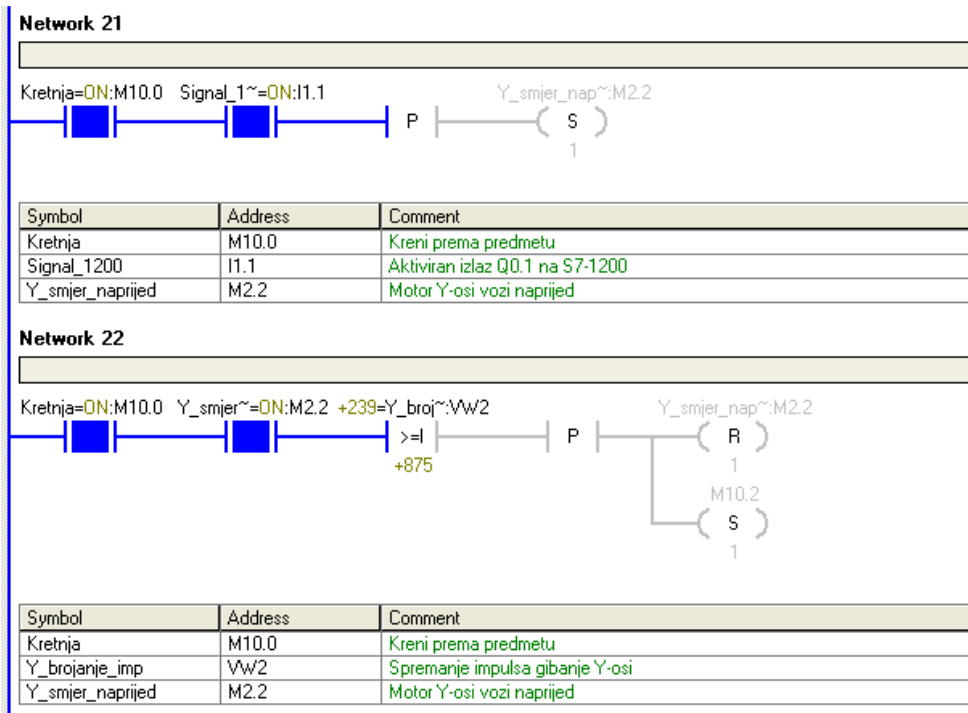




Slika 5.15: Kretanje manipulatora po X osi do pozicije predmeta

### 5.6.2 Kretanja po Y osi

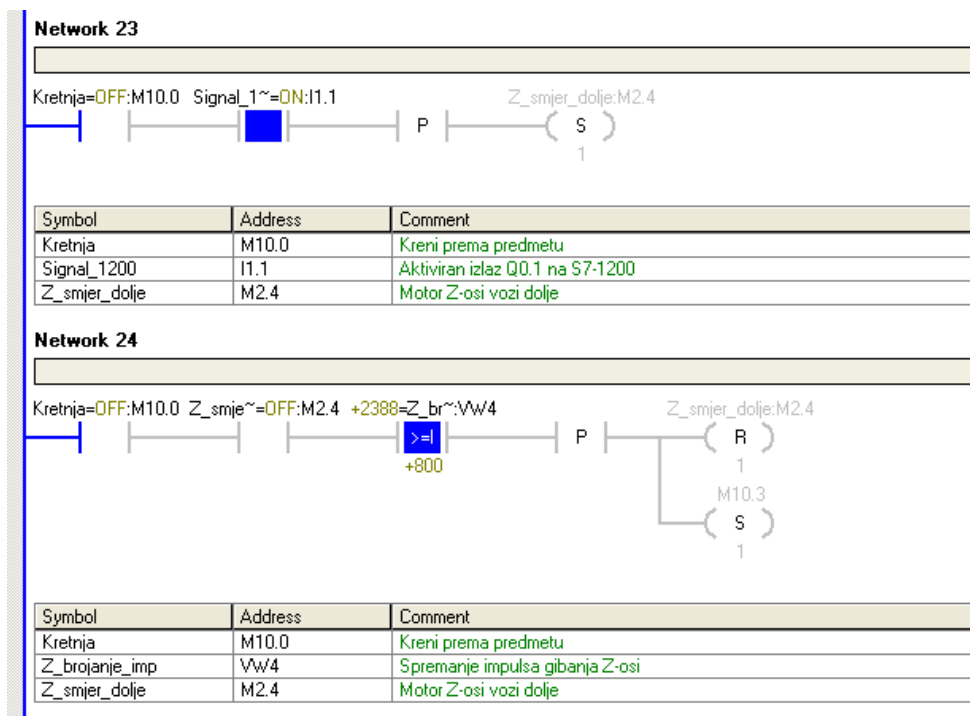
Uz kretanje manipulatora po X osi, istovremeno se izvodi i kretanja po Y i Z osi. Y os neće prelaziti iste impulse kao i X os zbog različite udaljenosti koju manipulator mora prijeći. Slika 5.16 prikazuje kretanju manipulatora po Y osi. Istovremena kretanja ostvarena je tako da se koristi isti memorijski bit M10.0 za pokretanje sve tri osi istovremeno. U ova dva *network*-a taj bit postavlja memorijski bit M2.0 koji služi za određivanje smjera kretnje motora Y osi. Isti usporedni blok se koristi kod sve 3 osi, ali za različiti broj impulsa.



Slika 5.16: Kretanje manipulatora po Y osi do pozicije predmeta

### 5.6.3 Kretanja po Z osi

Kretanja Z osi prikazana na slici 5.17 koristi se isključivo iz razloga što je Z os vrlo spora. Vrijeme koje manipulator prolazi kada se pozicionira po X i Y osi iskorišteno je da se Z os također spusti koliko je moguće trenutnom brzinom motora.

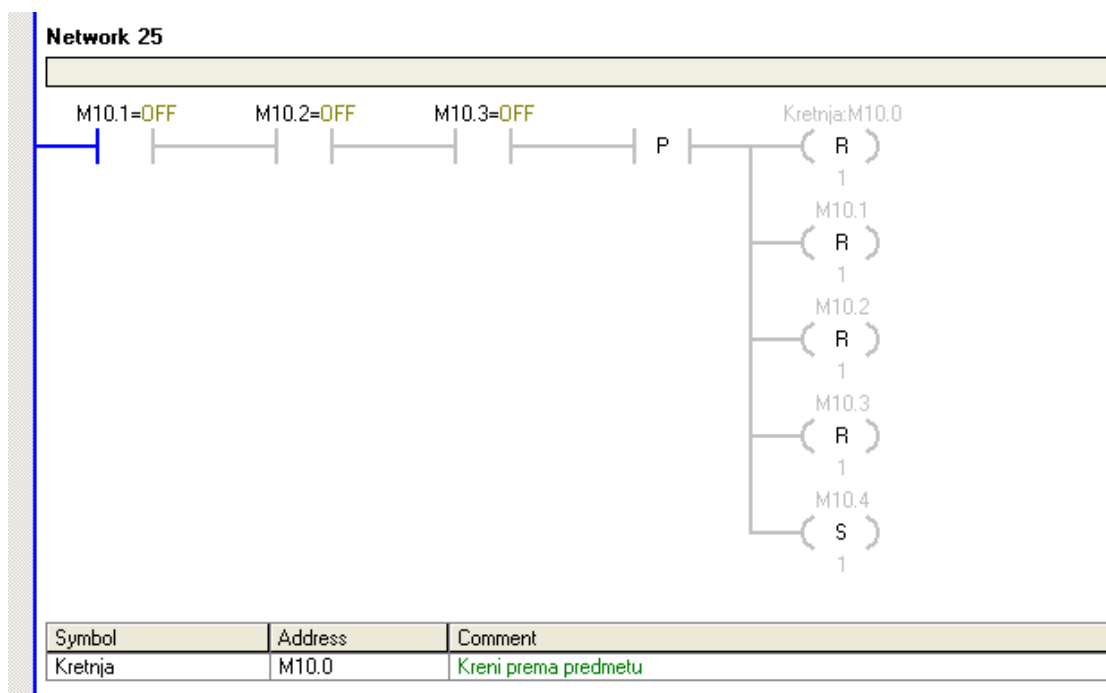


Slika 5.17 : Kratko spuštanje Z osi

Princip je isti kao i kod prethodne dvije osi. Sada je postavljen bit M2.4 koji postavlja smjer motora tako da se Z os spušta prema dolje.

#### 5.6.4 Reset korištenih bitova i priprema za prihvat predmeta

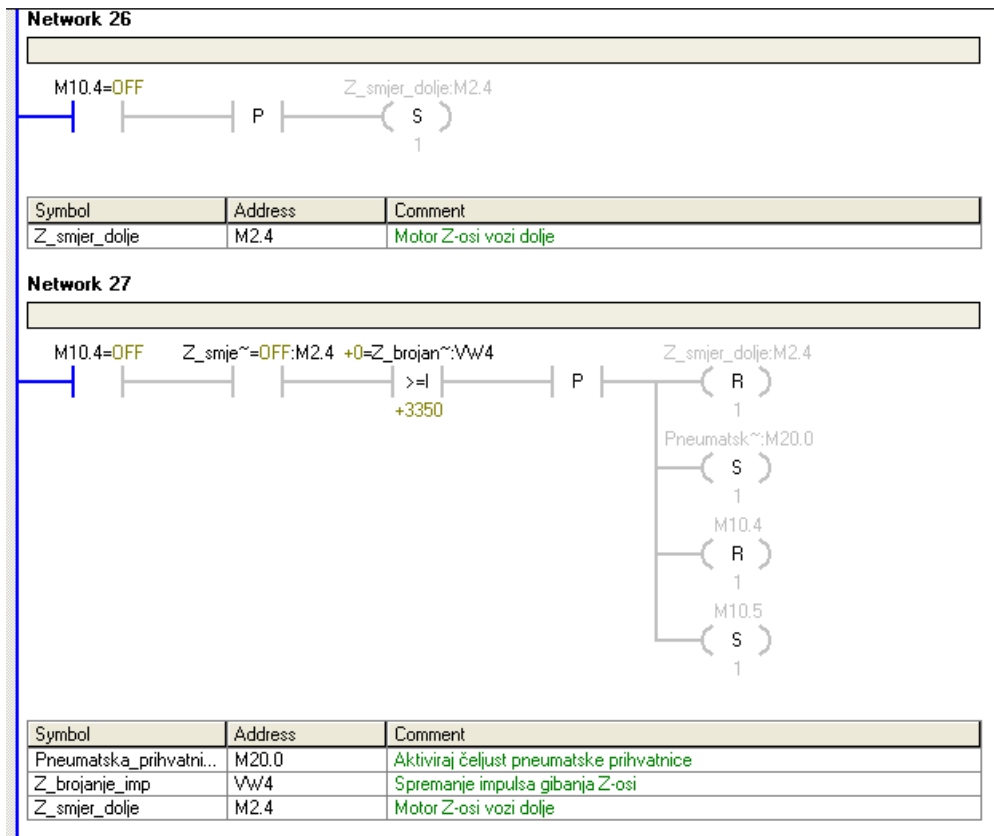
Nakon što je manipulator pozicioniran iznad predmeta koji se treba prihvatiti, resetiraju se svi korišteni bitovi koji su do sada bili postavljeni u svakom dijelu programa za pozicioniranje prema predmetu. To su bitovi M10.0, M10.1, M10.2 i M10.3, prikazani na slici 5.18.



Slika 5.18: Reset korištenih bitova i priprema za prihvat predmeta

### 5.7 Prihvat predmeta i prijenos predmeta na pokretnu traku

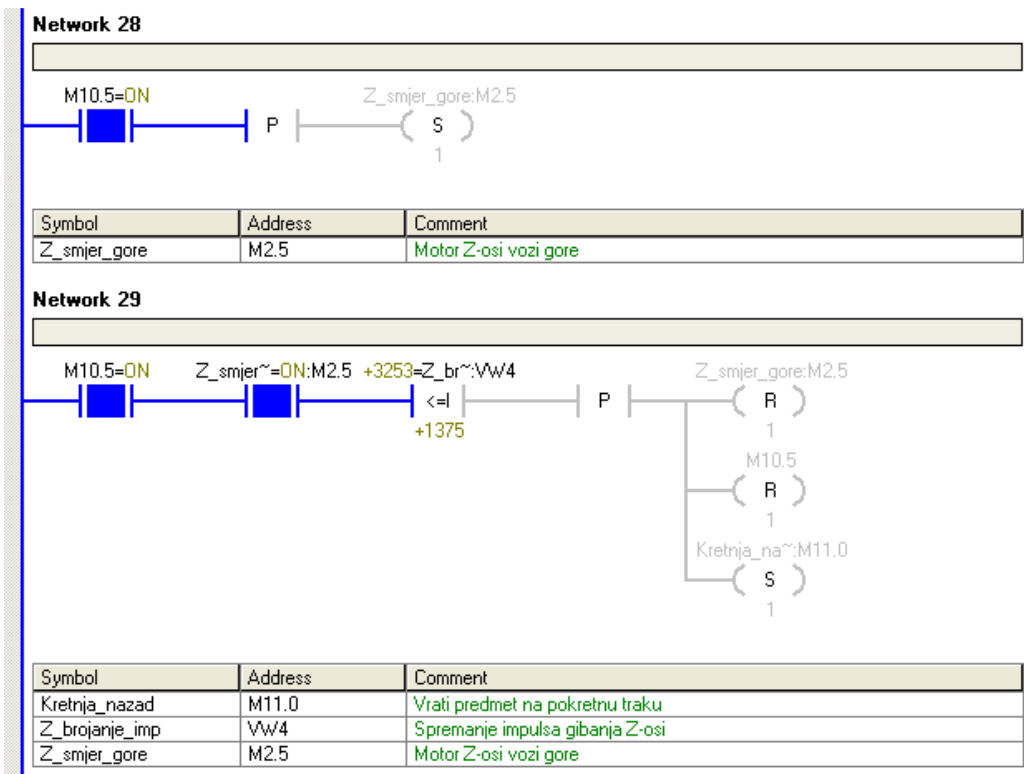
U nastavku programa opisan je prihvat predmeta i njegova putanja prema pokretnoj traci. Zadnji postavljeni memorijski bit u programu je M10.4. Sada taj bit aktivira bit M2.4 koji se koristi za ponovno spuštanje Z osi i prikazano je na slici 5.19. Z os se sada kreće puno brže zbog nove definirane brzine generiranja signala. Manipulator sada mora prijeći puno veću udaljenost pa je broj impulsa u usporednom bloku puno veći. Kada se Z-os spusti za zadani broj impulsa, resetiraju se korišteni bitovi jer sada samo jedna os vrši gibanje i aktiviramo memorijski bit M20.0. Memorijski bit M20.0 postavljen je u zadnjem *network*-u programa i on aktivira pneumatsku prihvatnicu na izlazu Q1.1.



Slika 5.19: Spuštanje Z-osi do predmeta i aktiviranje pneumatske prihvatnice

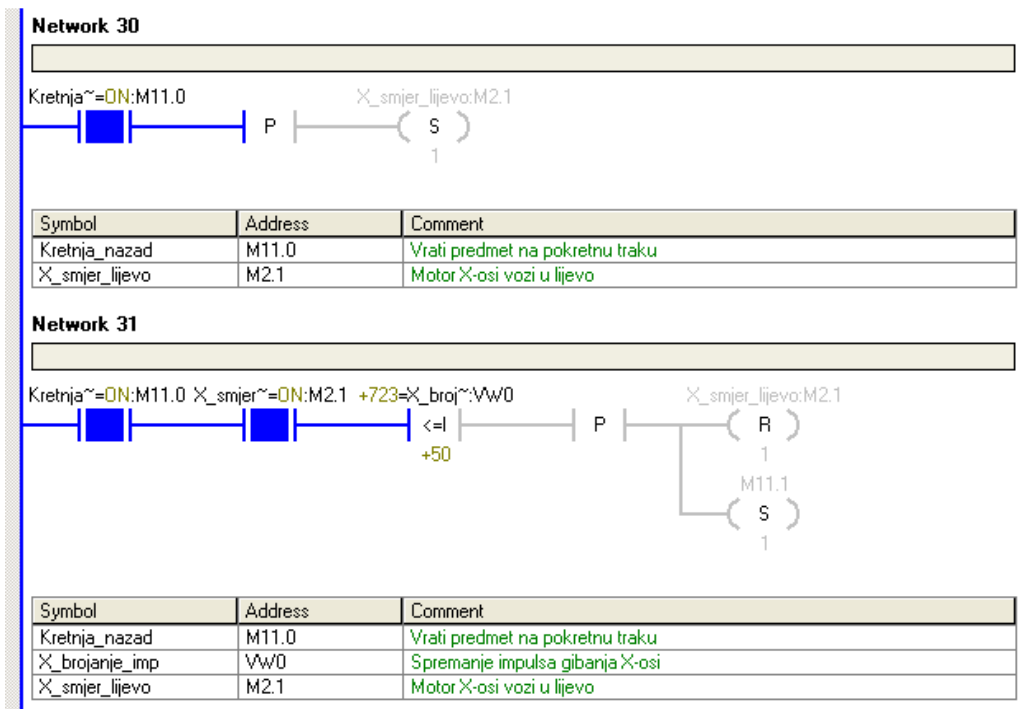
Nakon što je prihvatnica aktivirana i predmet zahvaćen, memorijski bit M10.5 aktivira memorijski bit M2.5 koji se koristi za kretanje Z osi prema gore koje je vidljivo na slici 5.20.

Princip ove kretnje je gotovo identičan kao i kod spuštanja. Usporedni blok koji se sada koristi drugačiji je nego do sada jer se Z os kreće u drugu stranu. Cilj ovog bloka je taj da kada se Z os vraća gore, vrati do zadanog broja impulsa. Bitno je znati razliku da se neće spuštati za broj impulsa koji je upisan, već trenutno upisana vrijednost impulsa će se smanjivati do upisane vrijednosti.



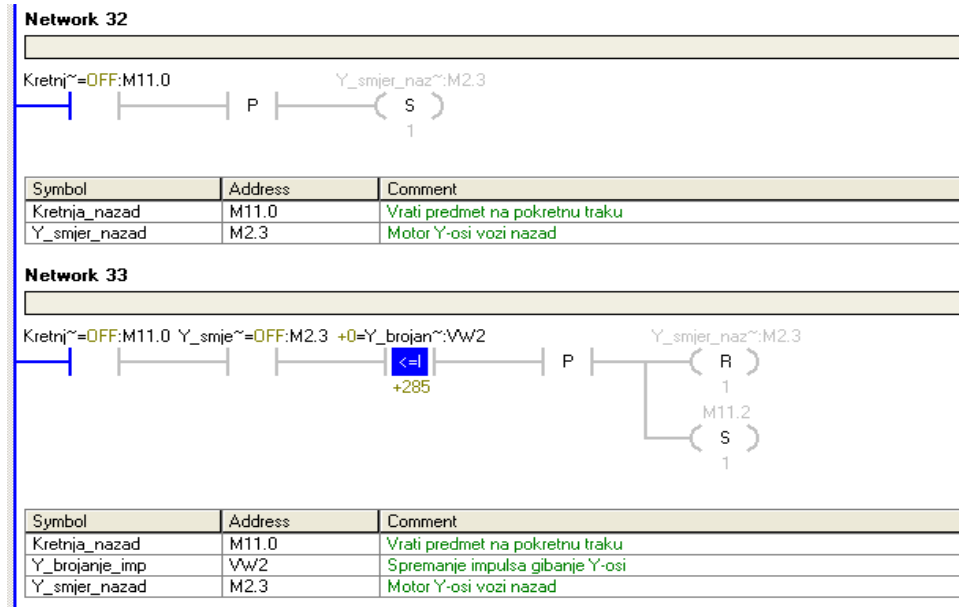
Slika 5.20: Gibanje Z-osi prema gore

Nakon što se manipulator vratio po Z osi do zadane vrijednosti, memorijski bit M11.0 aktivira bit M2.1, kao što je prikazano na slici 5.21. Memorijski bit M2.1 određuje smjer motora X-osi i manipulator se kreće u smjeru prema pokretnoj traci.



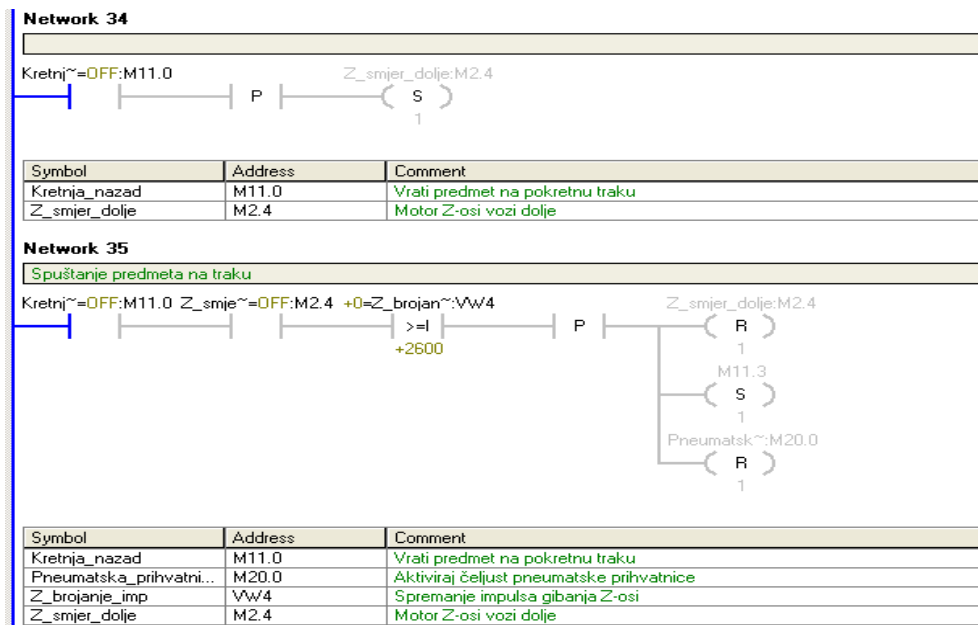
Slika 5.21: Kretnja prema pokretnoj traci po X-osi

Istovremeno se s X osi kreće i Y os. Manipulator po Y osi treba prijeći puno manju udaljenost, a to istovjetno predstavlja i manje impulsa. Memorijski bit M11.0 postavlja bit M2.3 prikazan na slici 5.22, koji je potreban da se manipulator kreće u smjeru prema pokretnoj traci.



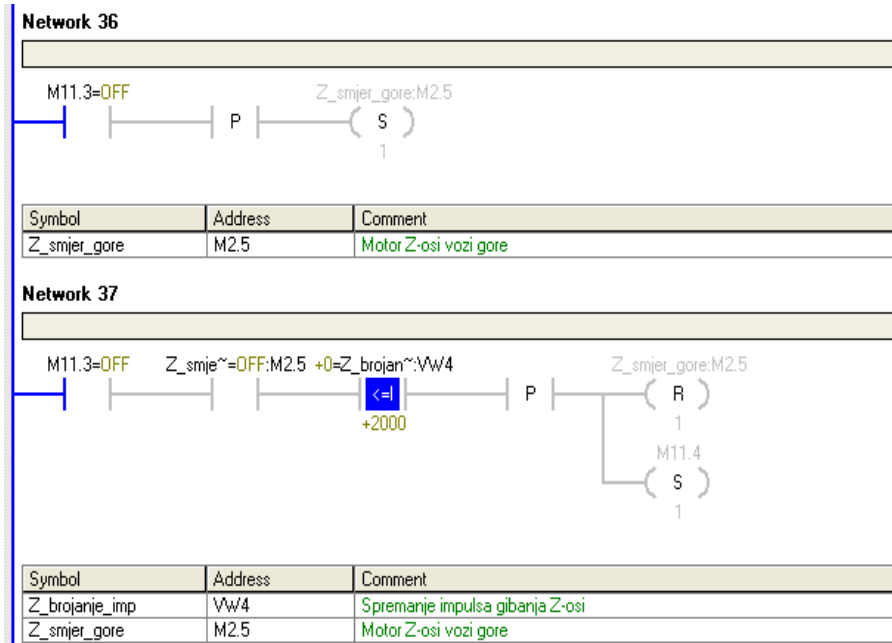
Slika 5.22: Kretanja prema pokretnoj traci po Y osi

Nakon što je pneumatska prihvatnica pozicionirana na željenom položaju iznad pokretne trake, Z os se spušta za mali broj impulsa kako bi se predmet ispustio iz hvataljke. Na slici 5.23 prikazan je dio programa koji se odnosi na spuštanje Z osi i ispuštanje predmeta na pokretnu traku.



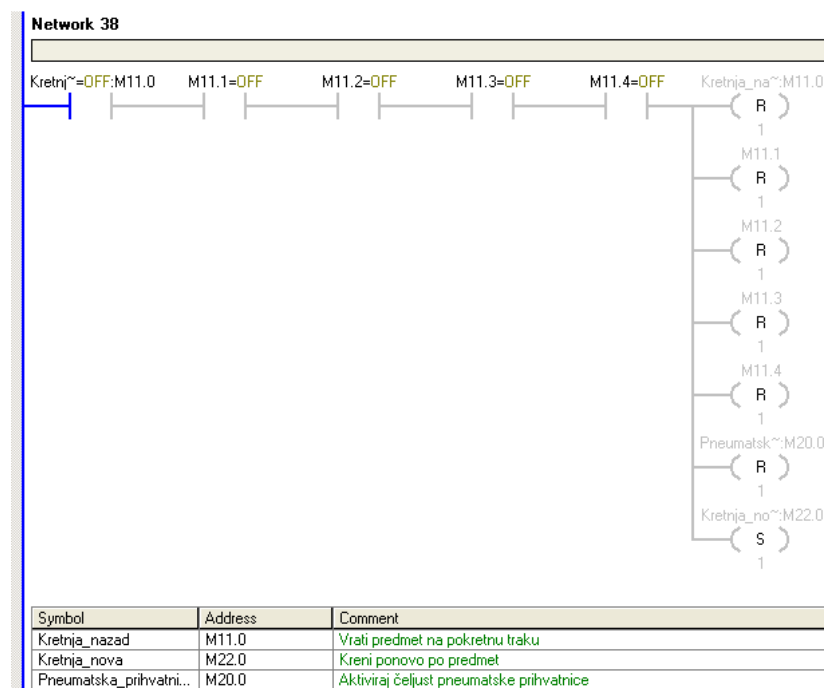
Slika 5.23: Spuštanje Z-osi i ispuštanje predmeta na pokretnu traku

Nakon što je predmet ispušten, pneumatska prihvatnica se još jednom podiže za određeni broj impulsa kako bi se osigurala sigurna kretanja u prostoru i to je prikazano na slici 5.24. Ovaj dio programa označava kraj trajektorije i u nastavku se definira nova putanja prihvatnice.



Slika 5.24: Podizanje po Z-osi nakon ispuštanja predmeta

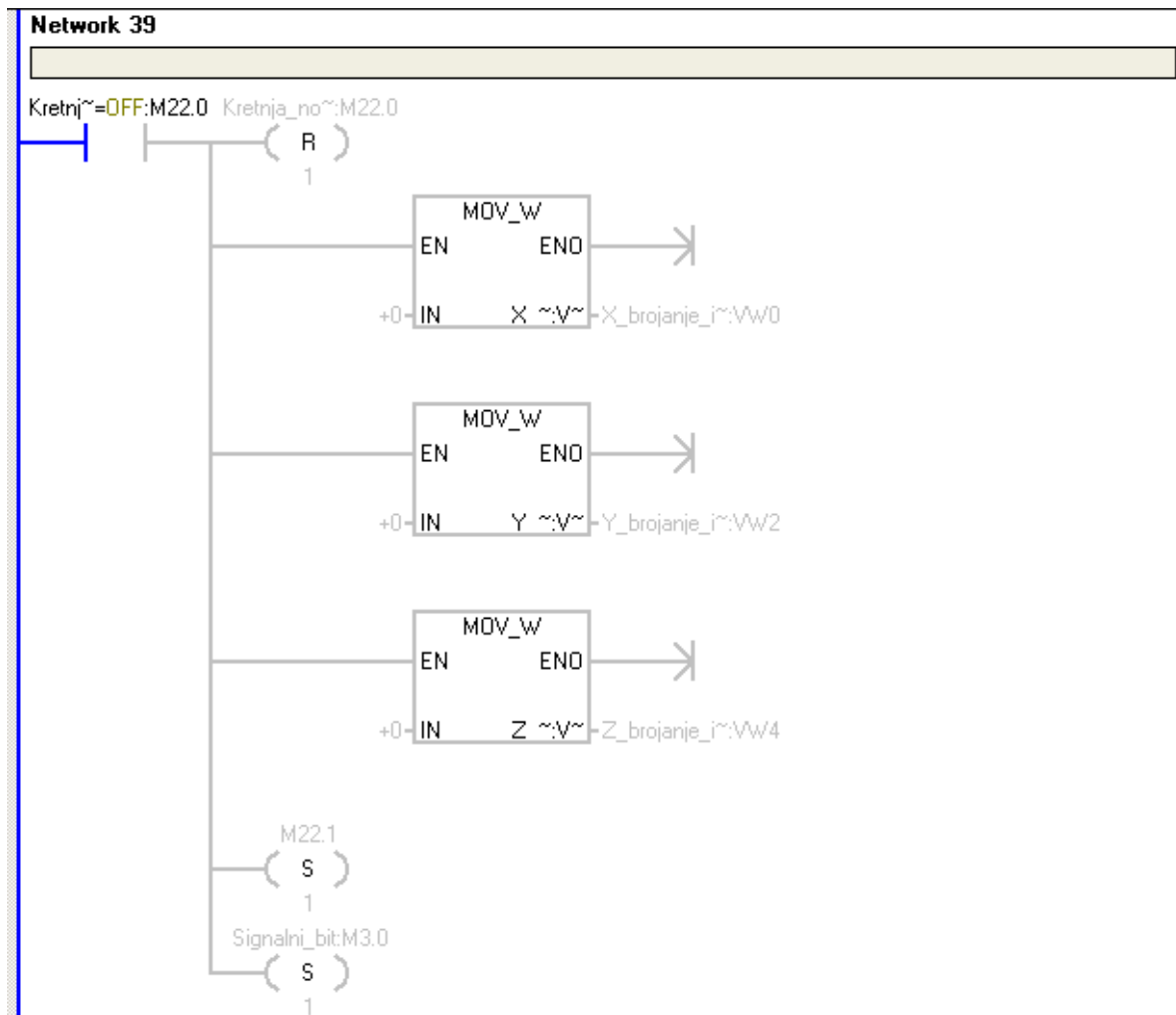
Po završetku ovog gibanja, sva korištena memorijska mjesta postavljaju se u logičku nulu i postavlja se novi memorijski bit koji označava početak nove trajektorije vidljiv na slici 5.25.



Slika 5.25: *Reset* memorijskih bitova i početak nove trajektorije

## 5.8 Definiranje nove putanje manipulatora

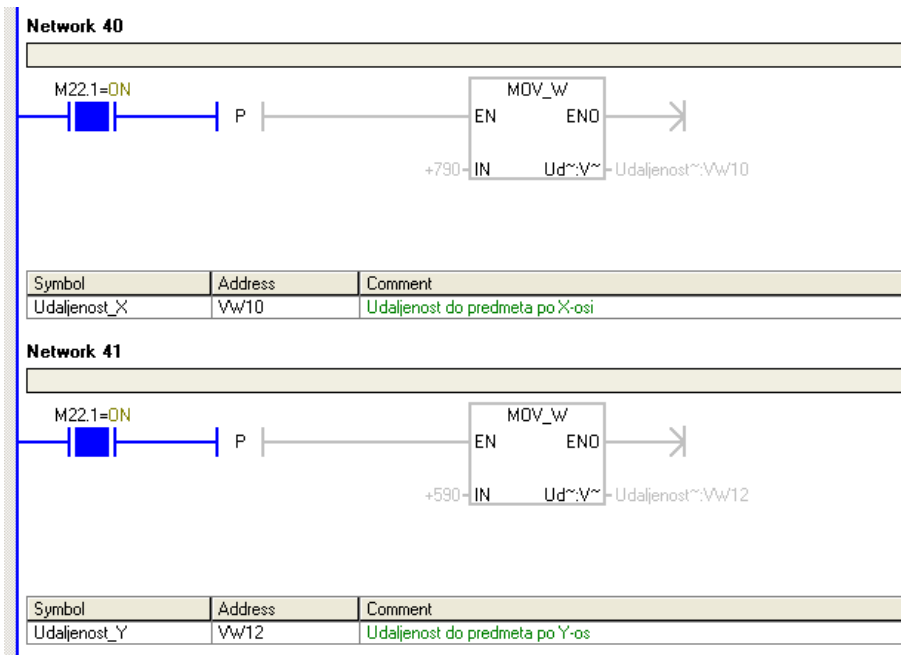
Nakon što manipulator prenese predmet na pokretnu traku, potrebno je resetirati svaku korištenu varijabilnu memoriju kako bi se mogla definirati nova putanja. Na slici 5.26 prikazan je memorijski bit M22.0 koji pokreće novu putanju. Istovremeno se pali i signalni bit M3.0 koji šalje signal *master* PLC-u da je pneumatska prihvatnica prazna te da može prihvatiti novi predmet.



Slika 5.26: Početak nove kretnje i *reset* korištene memorije

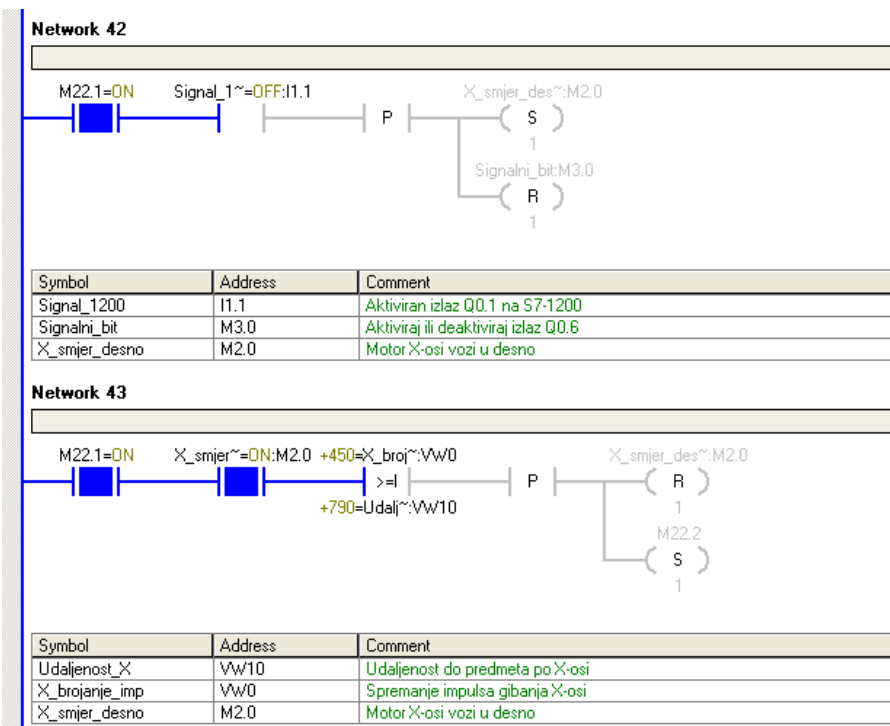


Pneumatska prihvatnica se u trenutnom položaju ne nalazi u *home* poziciji. Na slici 5.27 prikazan je unos nove udaljenosti do predmeta koji treba pokupiti. Kada je aktiviran memorijski bit M22.1, u memorije za udaljenost po X i Y osi upisuje se nova vrijednost.



Slika 5.27: Definiranje nove udaljenosti do predmeta

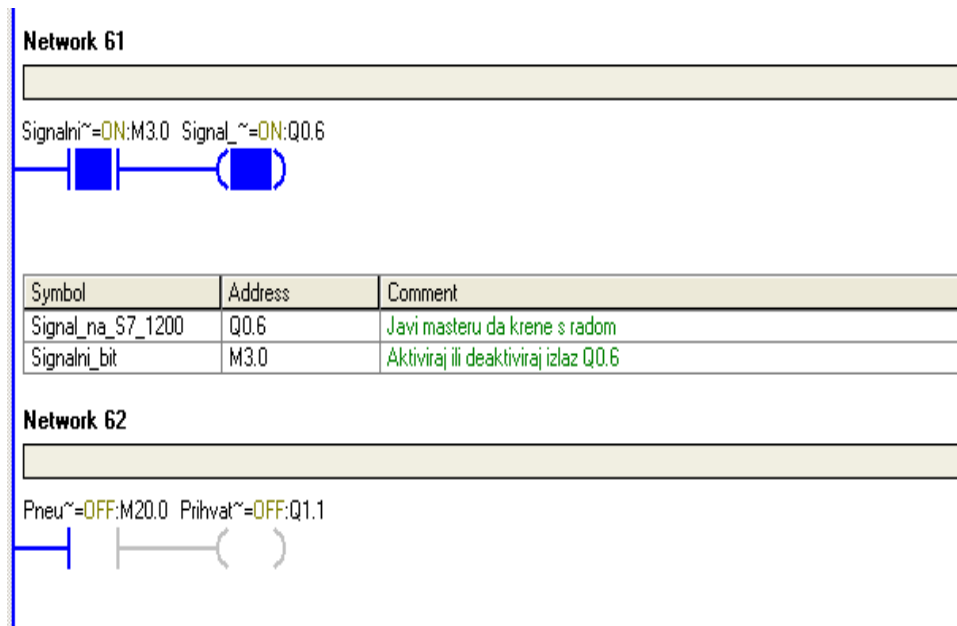
Daljni princip programiranja kretnje manipulatora je gotovo identičan kao i do sada. Na slici 5.28 prikazan je primjer kretnje manipulatora po X osi.



Slika 5.28: Kretnja prihvatnice po X-osi do predmeta

U usporedni blok prikazan na slici 5.28 upisana je udaljenost definirana u Network-u 40. Memorijska mjesta VW10 i VW12 definiraju udaljenost pneumatske prihvatnice do predmeta na postolju od onog trenutka kada je predmet prvi puta ispušten na traku.

Zadnja dva reda koda prikazana na slici 5.29 definiraju memorijske bitove korištene za slanje signala na *master* PLC i za aktiviranje i deaktiviranje čeljusti pneumatske prihvatnice.

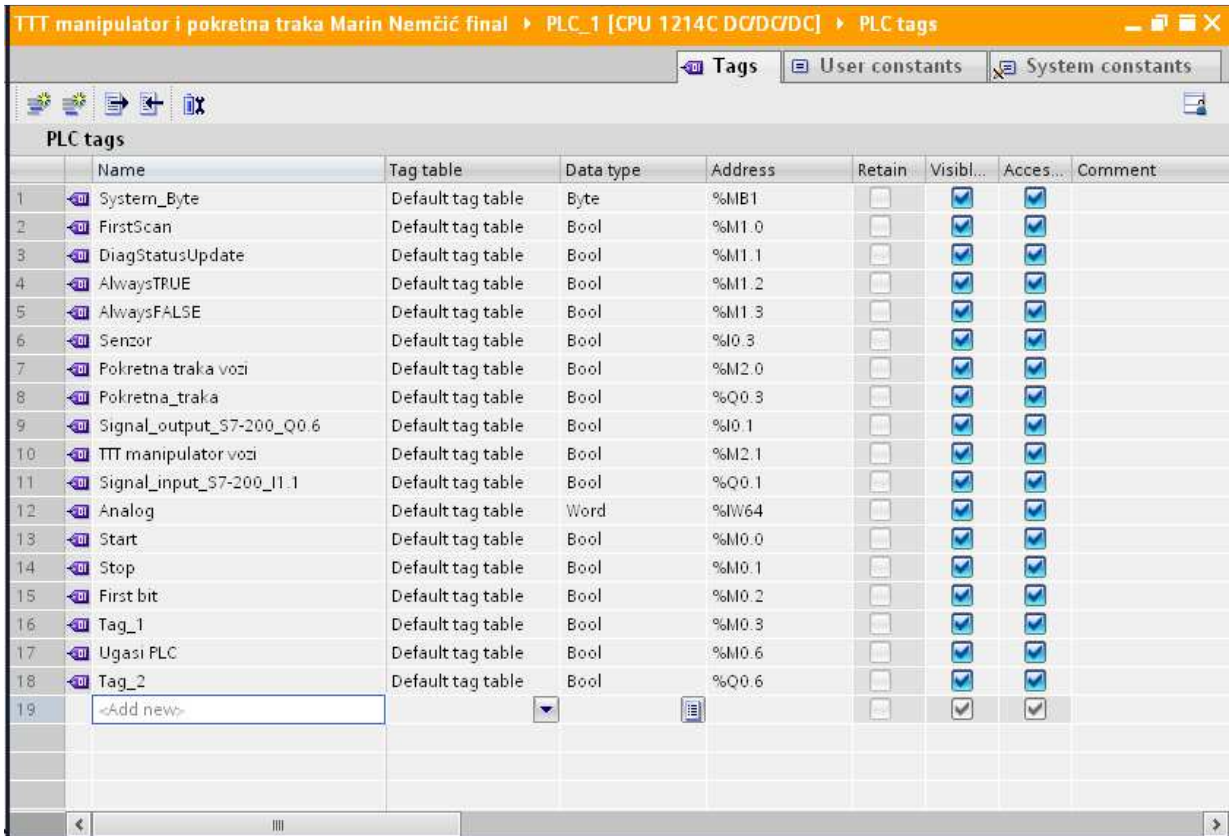


Slika 5.29: Memorijski bitovi M3.0 i M20.0

Signalni bit M3.0, kada je aktivan, pali izlaz Q0.6 slave PLC-a. Taj izlaz spojen je na ulaz *master* PLC-a i koristi se za komunikaciju istih.

## 6. TIA Portal program nadziranja cjelokupnog sustava

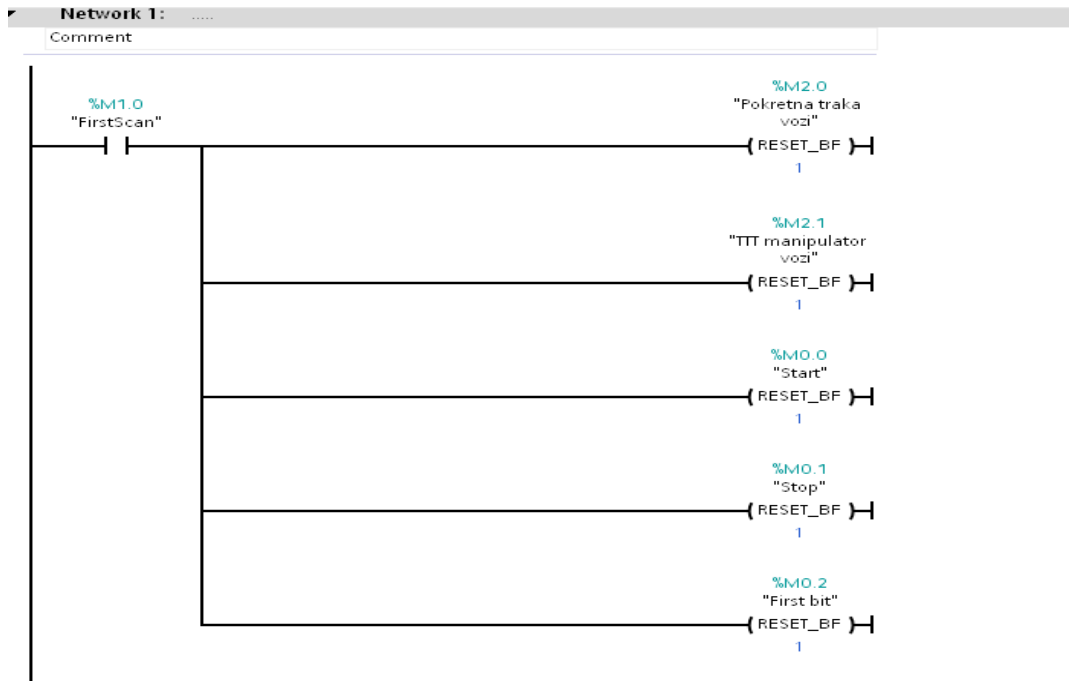
Cilj pisanja programa u *TIA Portal*-u je osigurati red izvođenja pojedinih cjelina. *Master* PLC upravlja pokretnom trakom i šalje signal na *slave* PLC kada predmet stigne do senzora. Na slici 6.1 prikazana je tablica simbola koja prikazuje ulaze, izlaze i memorijske riječi korištene za upravljanje PLC-a S7-1200.



	Name	Tag table	Data type	Address	Retain	Visibl...	Acces...	Comment
1	System_Byte	Default tag table	Byte	%MB1	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
2	FirstScan	Default tag table	Bool	%M1.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
3	DiagStatusUpdate	Default tag table	Bool	%M1.1	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
4	AlwaysTRUE	Default tag table	Bool	%M1.2	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
5	AlwaysFALSE	Default tag table	Bool	%M1.3	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
6	Senzor	Default tag table	Bool	%I0.3	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
7	Pokretna traka vozi	Default tag table	Bool	%M2.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
8	Pokretna traka	Default tag table	Bool	%Q0.3	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
9	Signal_output_S7-200_Q0.6	Default tag table	Bool	%I0.1	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
10	TTT manipulator vozi	Default tag table	Bool	%M2.1	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
11	Signal_input_S7-200_I1.1	Default tag table	Bool	%Q0.1	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
12	Analog	Default tag table	Word	%IW64	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
13	Start	Default tag table	Bool	%M0.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
14	Stop	Default tag table	Bool	%M0.1	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
15	First bit	Default tag table	Bool	%M0.2	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
16	Tag_1	Default tag table	Bool	%M0.3	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
17	Ugasi PLC	Default tag table	Bool	%M0.6	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
18	Tag_2	Default tag table	Bool	%Q0.6	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
19	<Add new>				<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	

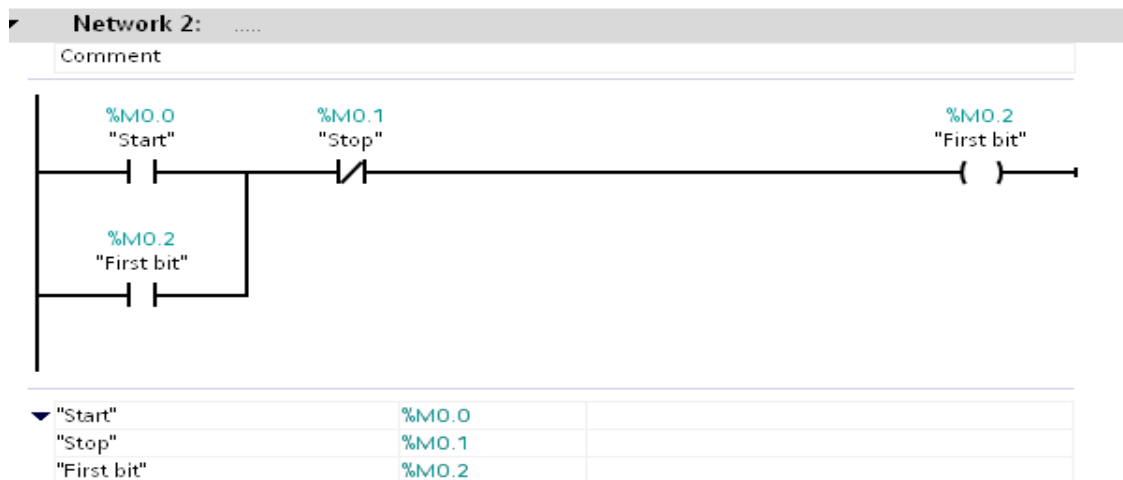
Slika 6.1: Tablica simbola u programskom okruženju TIA Portal

Prvi korak programiranja je u ovom slučaju sličan kao i u prethodnom programu. *Reset Bit Field* u prvom Network-u osigurava da se svi korišteni bitovi nalaze u logičkoj nuli kod pokretanja programa. Slika 6.2 prikazuje primjer korištenja ove naredbe.



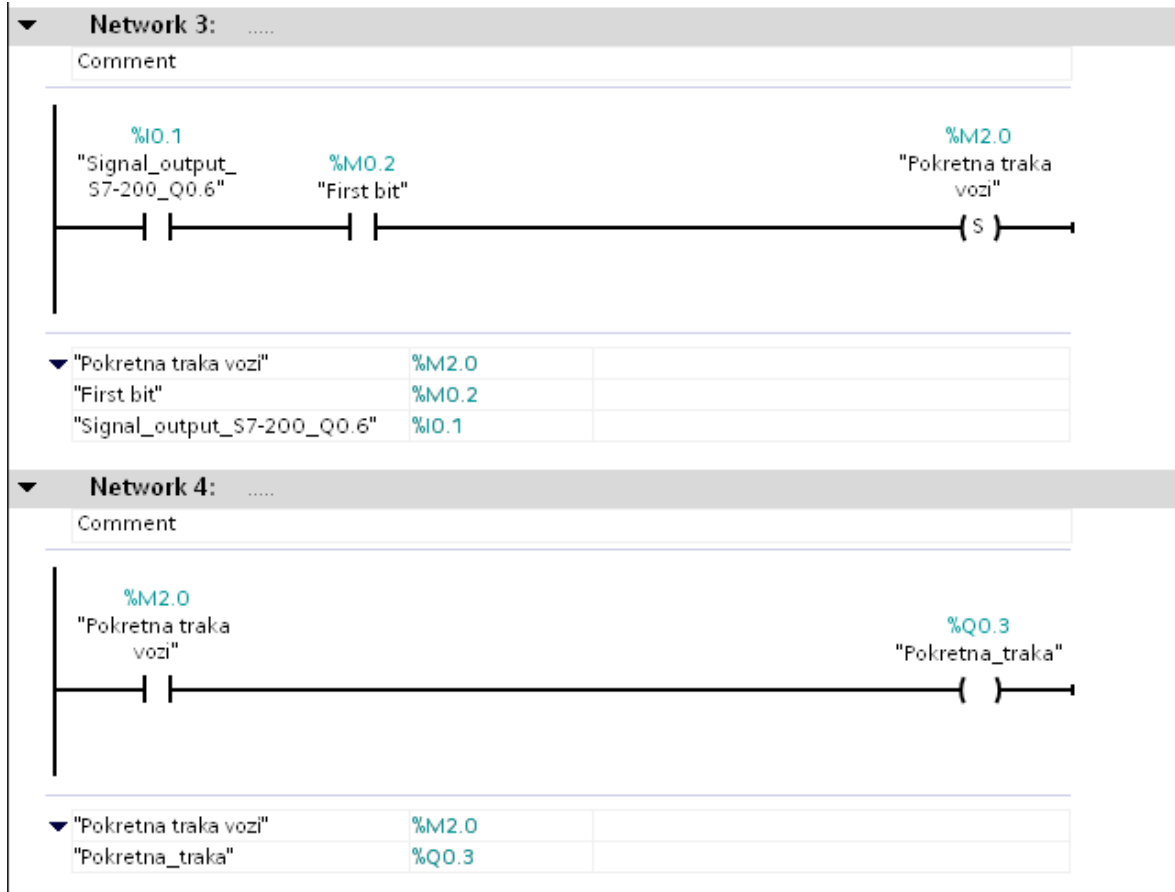
Slika 6.2: Reset svi korištenih bit-ova

U sljedećem koraku na slici 6.3 osigurava se rad tipke koja aktivira cijeli proces. Ova tipka nalazi se u *WinCC Runtime* aplikaciji *TIA Portal*-a koja je objašnjena dalje u tekstu. Memorijski bit M0.0, kada je u logičkoj jedinici, povezan uvjetom „I“ s memorijskim bitom M0.1, koji mora biti u logičkoj nuli, aktivira memorijski bit M0.2, koji označava da se program može početi izvršavati. U paralelu je na ulaz s M0.0 isto tako spojen taj isti memorijski bit M0.2 koji predstavlja ILI uvijet, što znači da se samo jednom mora pritisnuti tipka.



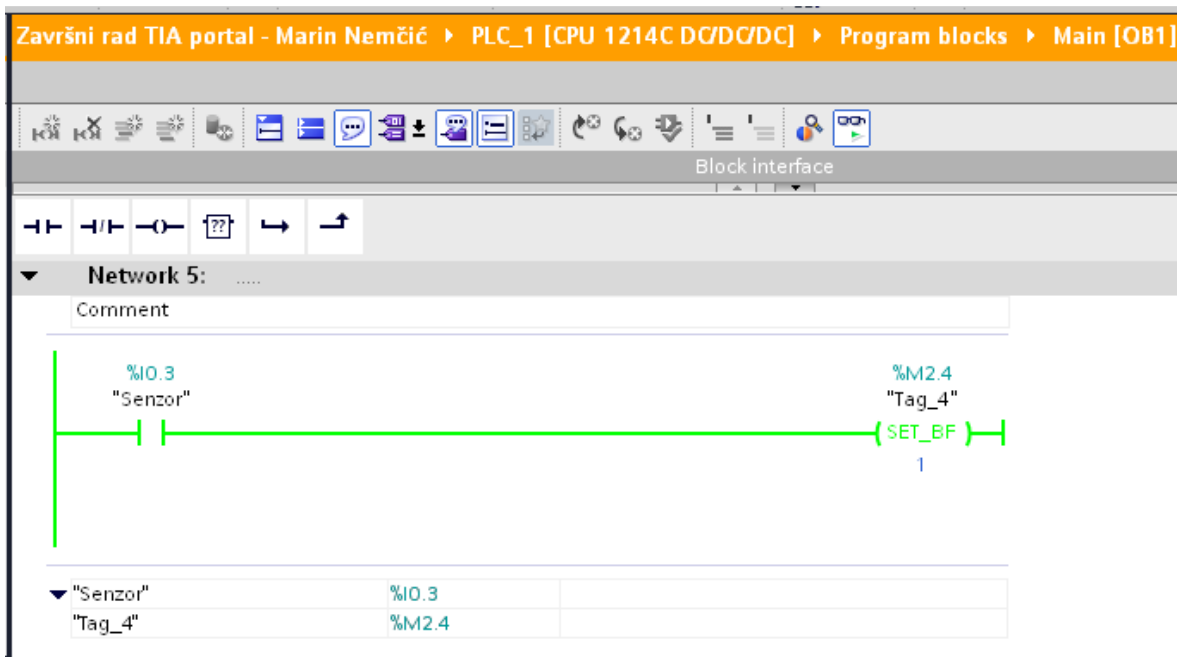
Slika 6.3: Osiguravanje početka programa

Sljedeća dva koraka programa odnose se na ulazni signal *master* PLC-a i pokretnu traku. Ulaz *master* PLC-a I0.1 spojen je na izlaz *slave* PLC-a na adresi Q0.6. U trenutku kada manipulator izvrši pozicioniranje šalje signal *master*-u da može krenuti s radom. Po primitku ovog signala, uključuje se pokretna traka kao što je vidljivo na slici 6.4.

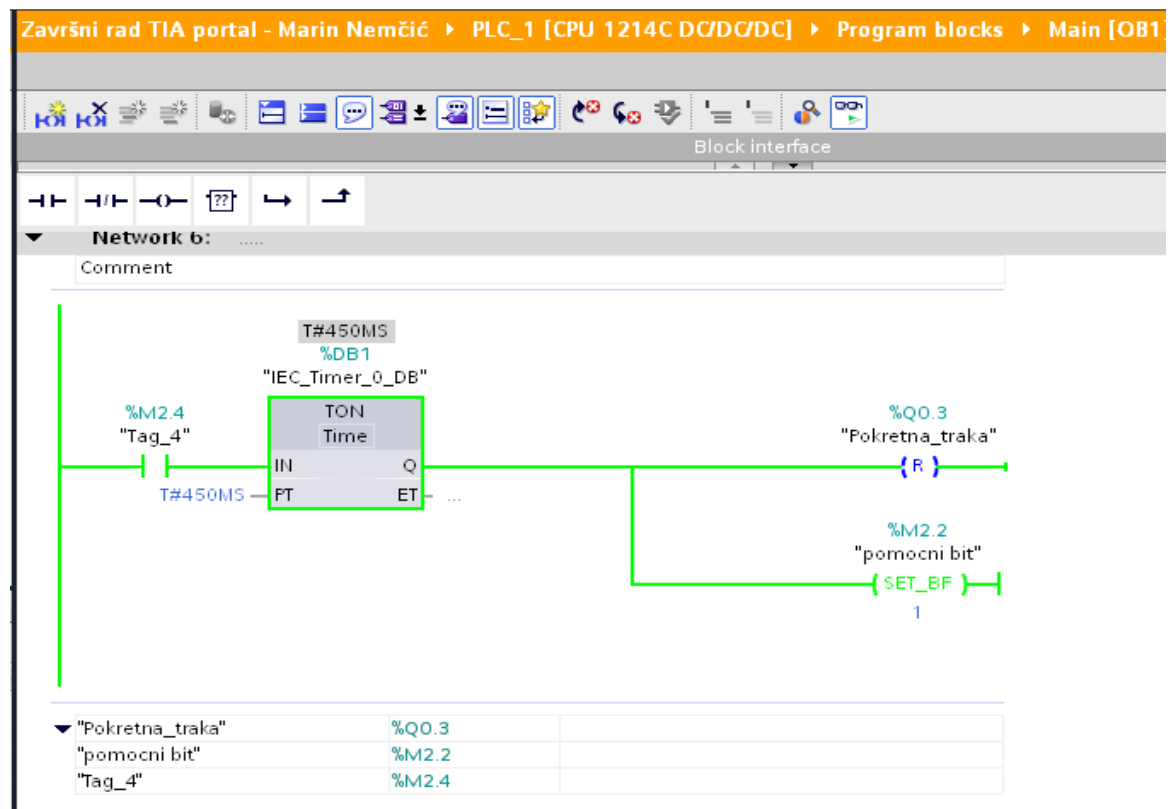


Slika 6.4: Komunikacija *Master-Slave* i pokretna traka

Na kraju pokretne trake nalazi se senzor. Kada isti detektira predmet, aktivira se memorijski bit M2.4 prikazan na slici 6.5. Taj memorijski bit služi za aktiviranje TON vremenskog brojača. Zbog nedostatka robotske ruke, dodan je vremenski brojač na slici 6.6 kako se predmet ne bi zaustavio u točnom trenutku detekcije senzora, već nešto kasnije kako bi se spustio u spremnik na kraju pokretne trake.

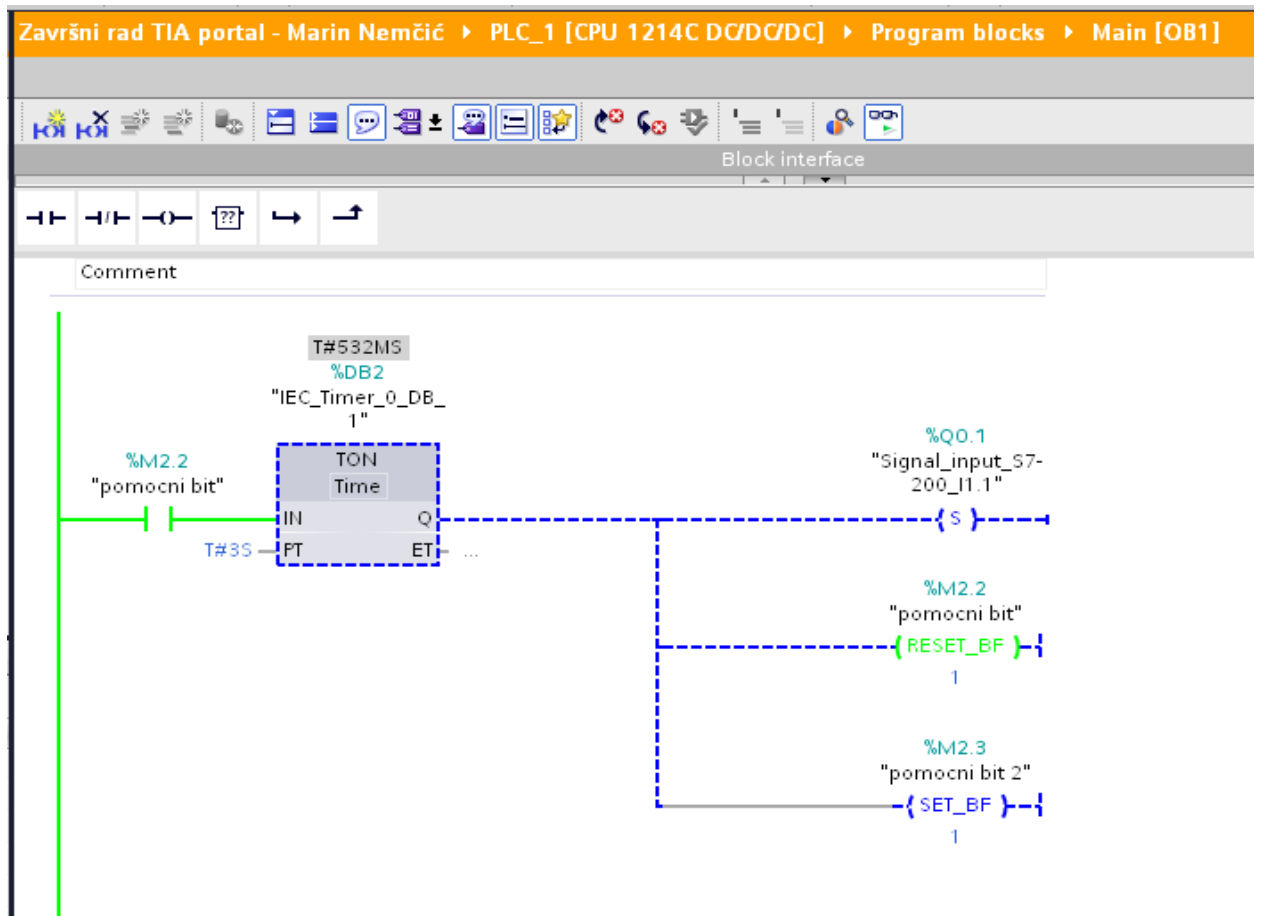


Slika 6.5: Ulazni signal senzora i memorijski bit M2.4



Slika 6.6: Vremenski brojač TON tipa

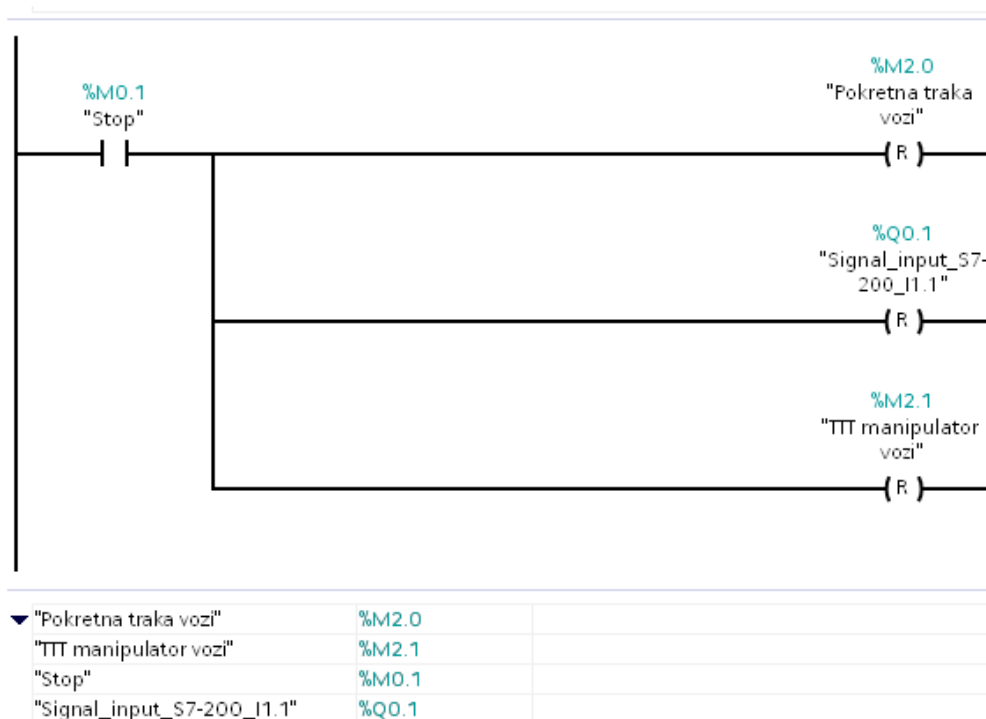
Dodatni vremenski brojač u sljedećem koraku na slici 6.7 koristi se zbog nedostatka robotske ruke. Cilj ovog brojača je imitiranje vremena potrebnog robotskoj ruci da obavi svoj dio posla.



Slika 6.7: Vremenski brojač kao zamjena robotskoj ruci

Nakon što je predmet ispušten u spremnik, aktivira se novi vremenski brojač koji broji tri sekunde. Nakon tri sekunde propušta signal na ulaz *slave* PLC-a da je predmet stigao na postolje.

Posljednji dio programa prikazan na slici 6.8 odnosi se na *Stop* tipku koja se nalazi u *WinCC RT Advanced*. Pritiskom na tipku *Stop* aktivira se memorijski bit M0.1, koji istovremeno gasi pokretnu traku, signalni izlaz na *slave* i memorijski bit povezan s kretanjem manipulatora.



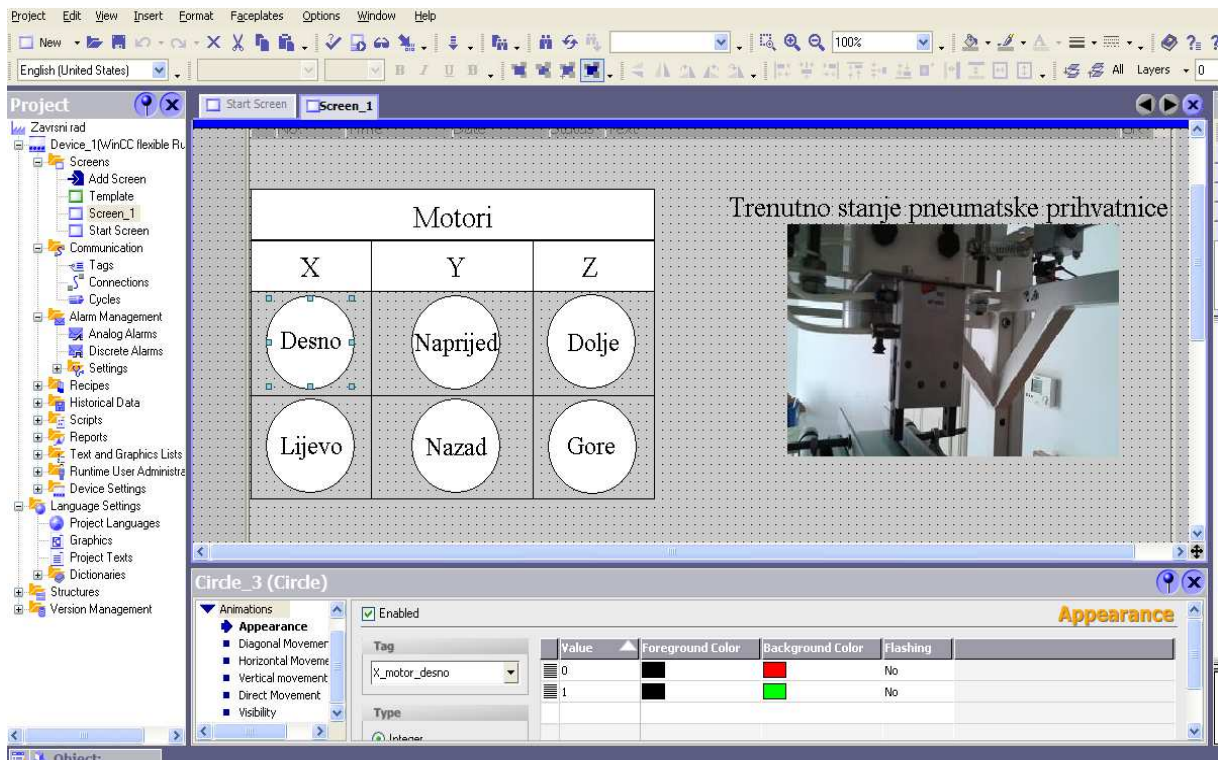
Slika 6.8: Tipka *Stop* i memorijski bit M0.1



## 7. WinCC Flexible Advanced

### 7.1 Vizualni prikaz rada motora i pneumatske prihvatnice

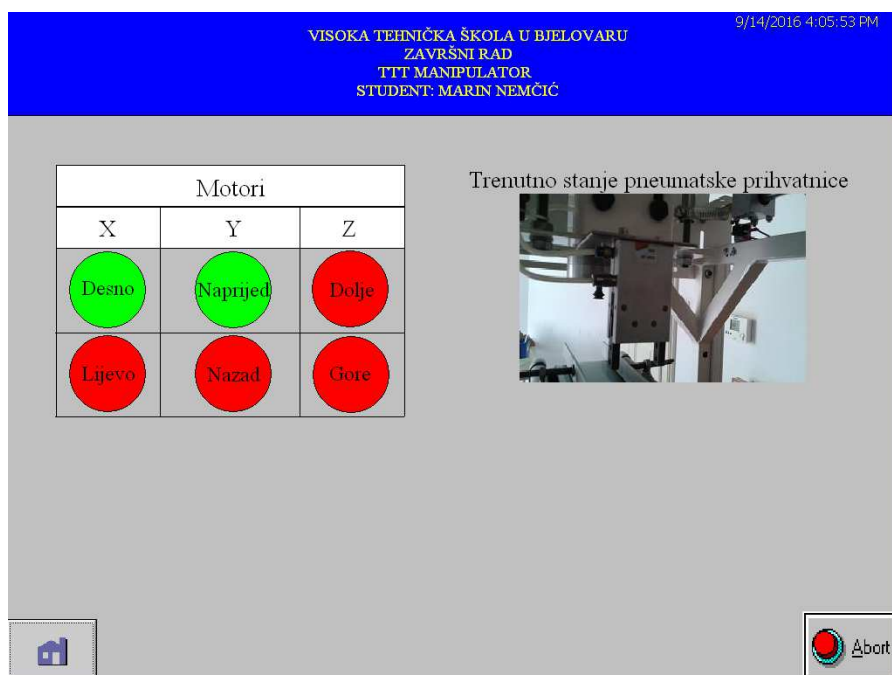
Program vizualizacije povezan s uređajem PLC *Siemens S7-200 224XP* je *WinCC Flexible*. Koraci izrade ove podrške gotovo su identični kao i u *TIA Portal SCADA* sustavu, koji je detaljnije objašnjen u sljedećem poglavlju ovoga rada. Na slici 8.1 prikazan je odabir i raspored simbola koji prikazuju kretnju pneumatske prihvatnice.



Slika 8.1: SCADA sustav kretnje pneumatske prihvatnice u *WinCC Flexible*

Cilj ovog sustava je nadzirati putanju kretnje pneumatske prihvatnice. Svaki motor može se kretati u dva smjera. Memorijski bitovi korišteni za kretnju u pojedinom smjeru prikazani su na slici 3.5. Primjer nadzora sa slike prikazuje motor koji pokreće stroj po X osi. Memorijski bit M2.0 označava da se motor vrti u stranu tako da se stroj kreće u desno, a memorijski bit M2.1 u lijevo po X osi. Ovisno koji je bit aktivan, pripadajući oblik u vizualizaciji svijetliti će ili zeleno ili crveno.

Na slikama 8.2, 8.3 i 8.4 prikazana je vizualizacija kretnje manipulatora u *online* načinu rada. Slika 8.2 prikazuje kretanje manipulatora prema predmetu. U ovom koraku manipulator se kreće po X i Y osi istovremeno.



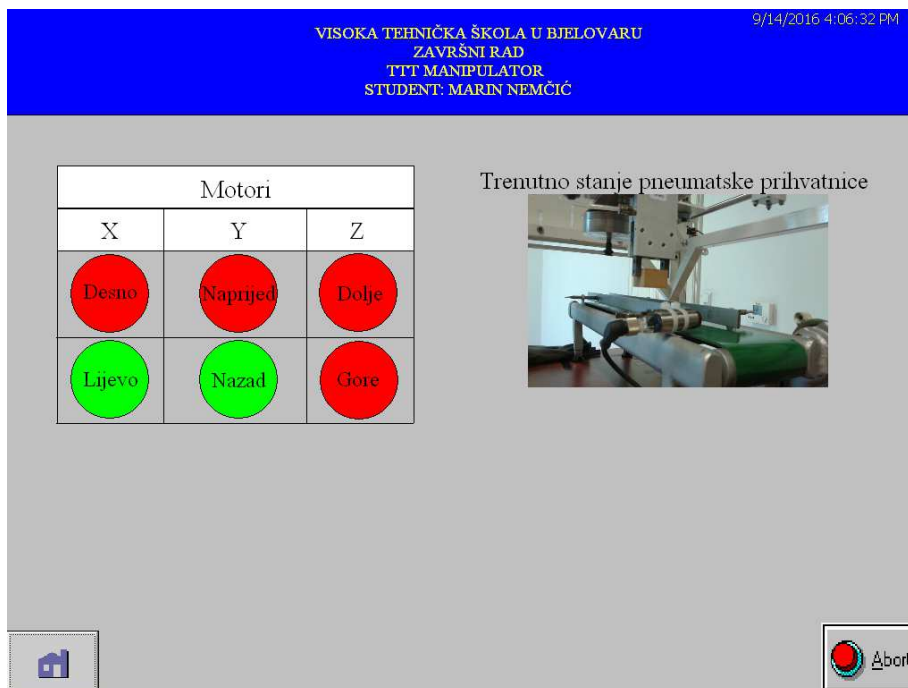
Slika 8.2: Kretnja manipulatora prema predmetu

Slika 8.3 prikazuje kretnju manipulatora nakon što je predmet zahvaćen. Manipulator se u ovom koraku kreće u suprotnom smjeru i spušta za određeni broj impulsa do pokretne trake



Slika 8.3: Pozicioniranje manipulatora iznad pokretne trake

Slika 8.4 prikazuje samo kretnju X i Y osi, bez spuštanja manipulatora po Z osi.



Slika 8.4: Kretnja manipulatora po X i Y osi nakon što je predmet zahvaćen

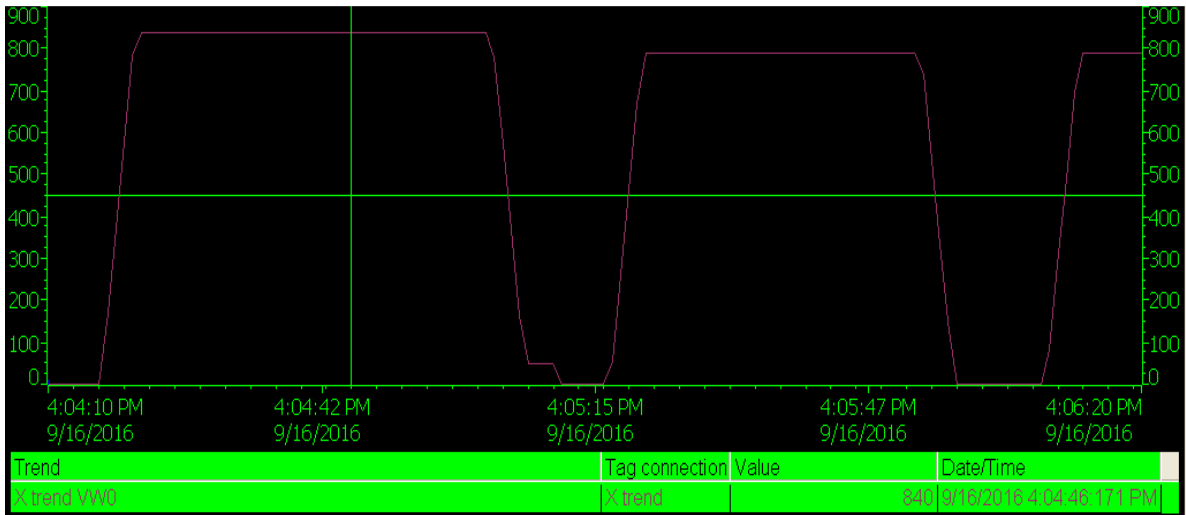
## 7.2 Trend gibanja manipulatora po pojedinoj osi

Na slikama 8.5, 8.6 i 8.7 prikazani su dijagrami gibanja TTT manipulatora u vremenu za X, Y i Z os. Slika 8.5 prikazuje dijagram gibanja manipulatora po X osi, slika 8.6 prikazuje dijagram gibanja manipulatora po Y osi, a slika 8.7 prikazuje dijagram gibanja manipulatora po Z osi.

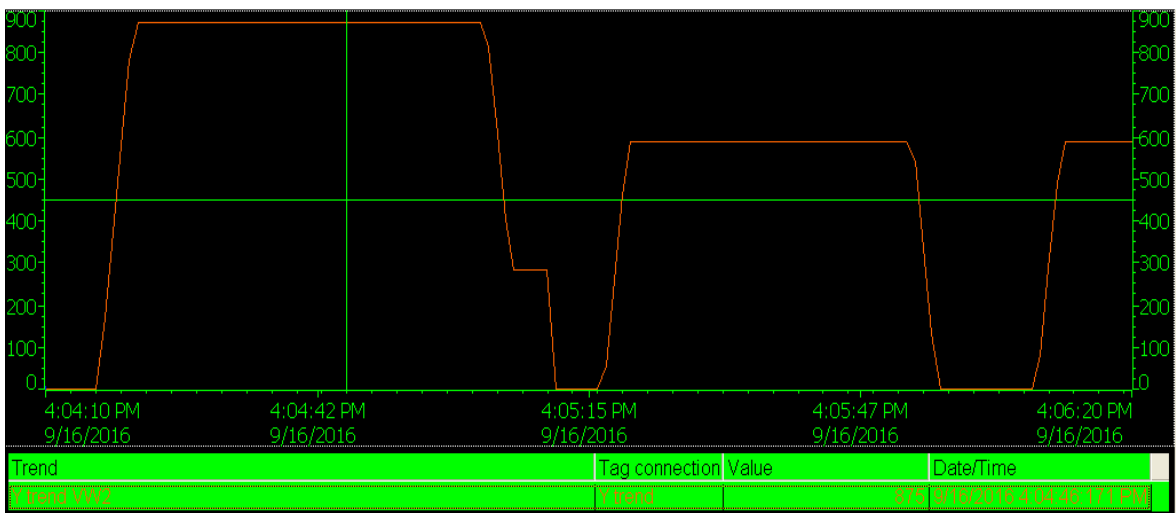
Nazivi pojedinih *trend*-ova koji se prate u ovim grafovima i korištene memorijske riječi su:

- *X trend*: prikazuje količinu impulsa koju motor prijeđe po X osi, a povezan je s varijabilnom memorijskom riječi VW0
- *Y trend*: prikazuje količinu impulsa koju motor prijeđe po Y osi, a povezan je s varijabilnom memorijskom riječi VW2
- *Z trend*: prikazuje količinu impulsa koju motor prijeđe po Z osi, a povezan je s varijabilnom memorijskom riječi VW4

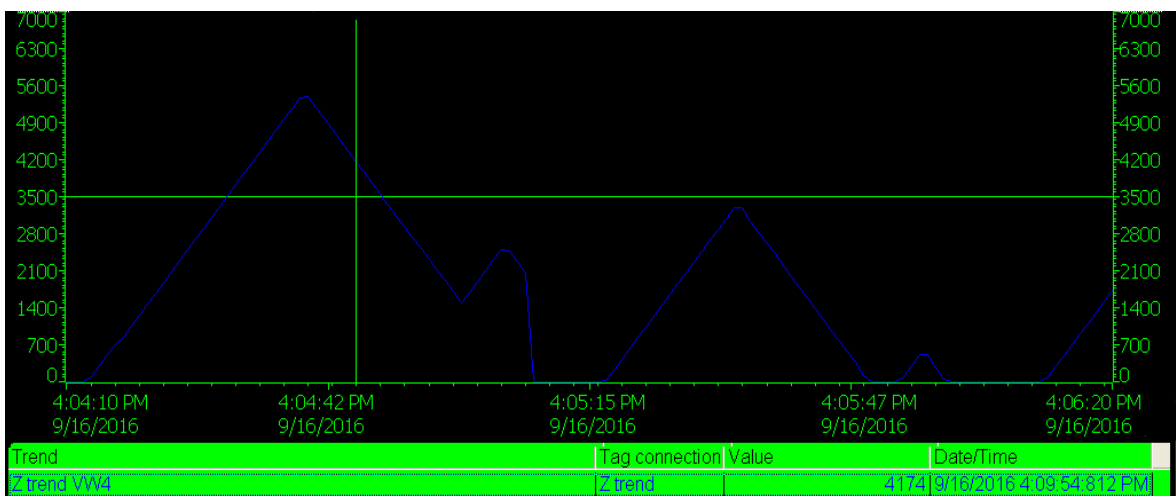
Pokazivač (eng. *Ruler*) okomita je crta zelene boje vidljiva na svakom grafu. Pokazivač je postavljen na svakom grafu na istom mjestu vremenske crte gibanja manipulatora kako bi razlika gibanja manipulatora po pojedinoj osi što vjernije bila prikazana.



Slika 8.5 Dijagram gibanja TTT manipulatora po X osi



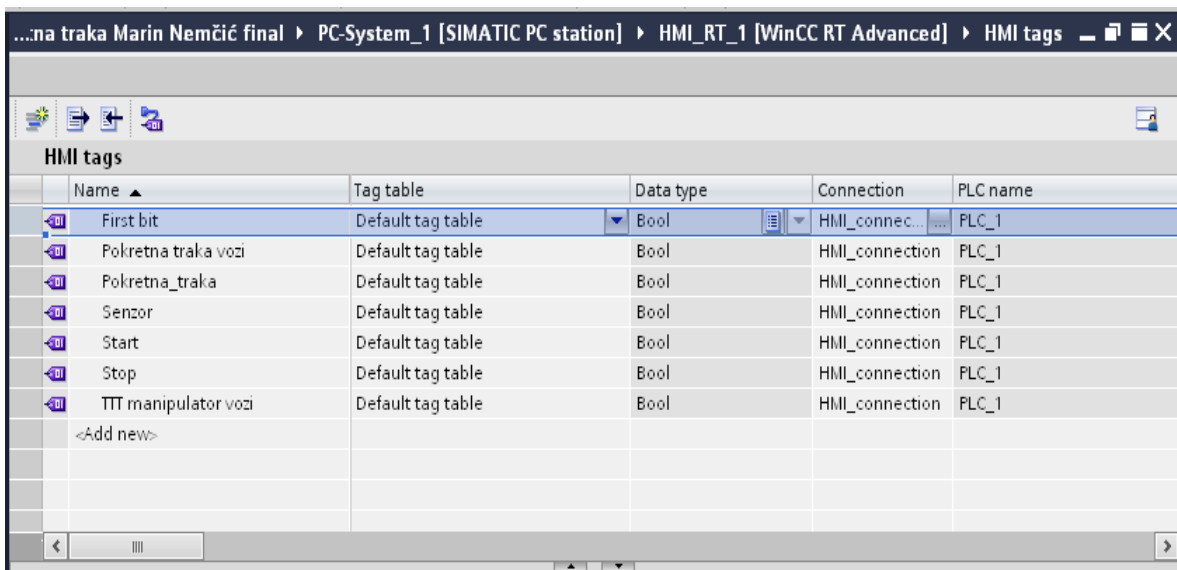
Slika 8.6: Dijagram gibanja TTT manipulatora po Y osi



Slika 8.7: Dijagram gibanja TTT manipulatora po Z osi

## 8. TIA Portal WinCC RunTime Advanced

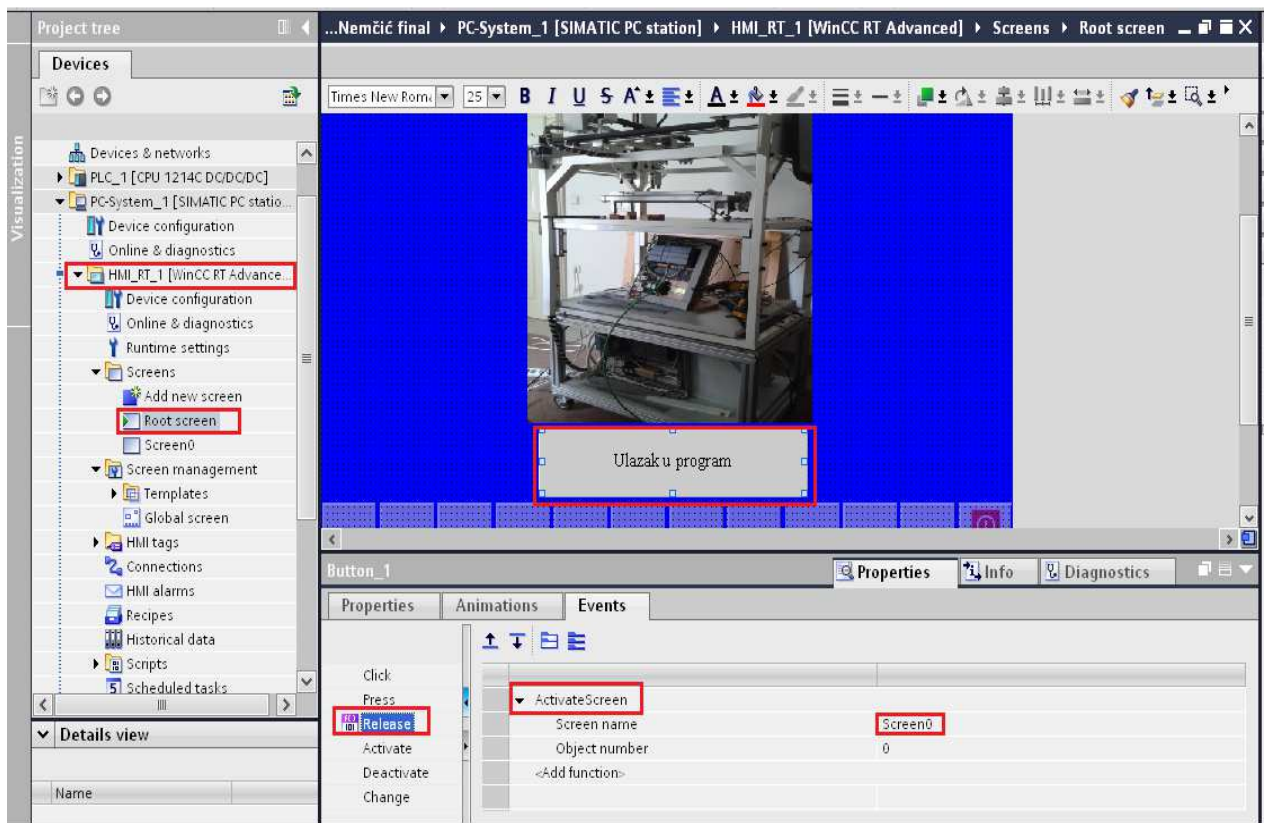
Program vizualizacije integriran je u program *TIA Portal*. Prvi korak izrade vizualizacije je podešavanje *tag*-ova, a to su slike memorijskih riječi PLC-a. Najjednostavniji način izrade slika memorijskih riječi jest putem otvaranja tablice simbola programa na PLC-u i otvaranjem „*HMI Tags*“. Opcijom *Float* na prozor tablice simbola odvađa se isti i moguće je izvršiti *Drag-and-Drop* s jednog mjesta na drugo. Na taj način osigurano je točno prenošenje memorijskih riječi. Slika 7.1 prikazuje tablicu slika memorijskih riječi korištenih u ovom SCADA sustavu.



Name	Tag table	Data type	Connection	PLC name
First bit	Default tag table	Bool	HMI_connection	PLC_1
Pokretna traka vozi	Default tag table	Bool	HMI_connection	PLC_1
Pokretna_traka	Default tag table	Bool	HMI_connection	PLC_1
Senzor	Default tag table	Bool	HMI_connection	PLC_1
Start	Default tag table	Bool	HMI_connection	PLC_1
Stop	Default tag table	Bool	HMI_connection	PLC_1
TTT manipulator vozi	Default tag table	Bool	HMI_connection	PLC_1
<Add new>				

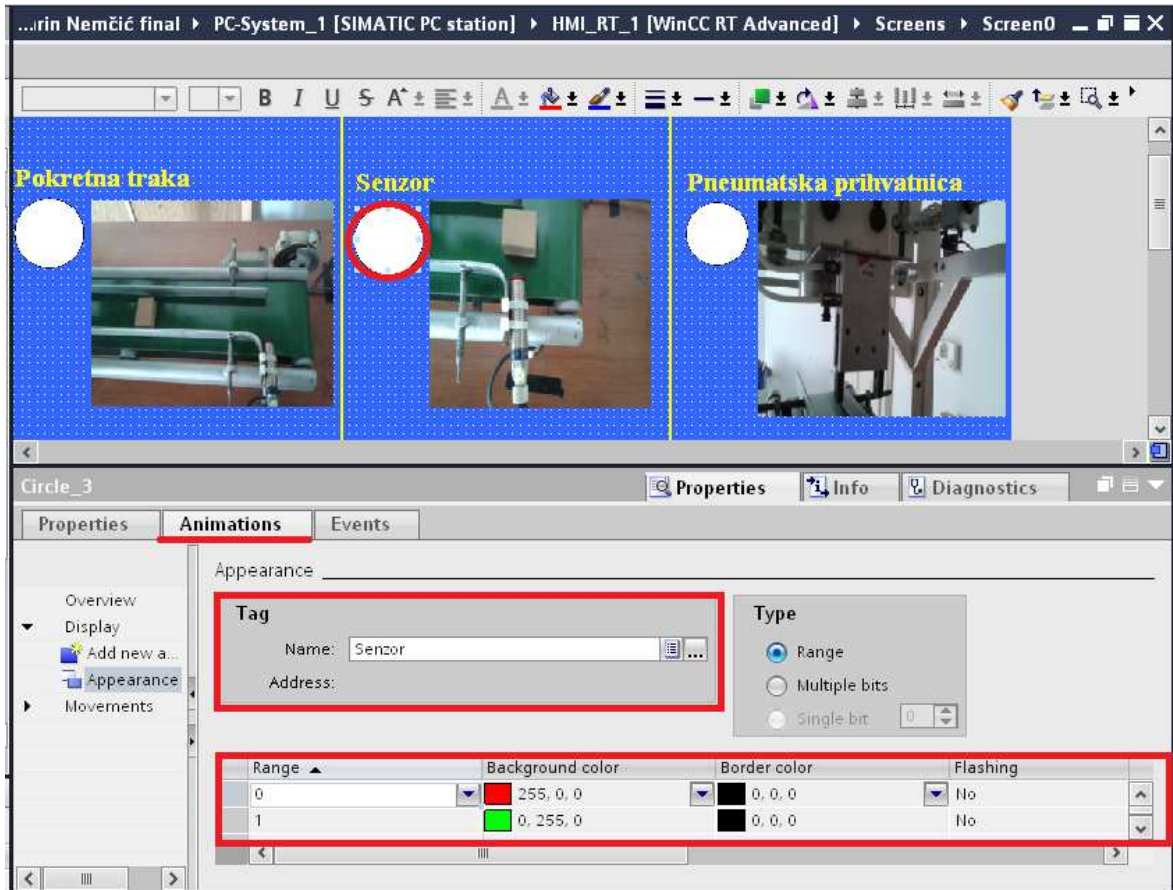
Slika 7.1: *HMI Tags* u programskom okruženju *TIA Portal*

Nakon definiranja *tag*-ova, realiziraju se prozori korišteni za prikaz sustava. U ovom konkretnom slučaju korištena su dva prozora, početni zaslon (eng. *Root screen*) i jedan dodatni zaslon na kojemu je prikazan cijeli proces. Kako bi se pristupilo pojedinom prozoru koristi se komanda „*ActivateScreen*“. Na slici 7.2 vidljiva je povezanost između tipke i događaja kojim se ista aktivira. Otpuštanjem (eng. *Release*) tipke miša aktivira se sljedeći zaslon.



Slika 7.2: Određivanje komande za prelazak na drugi zaslon

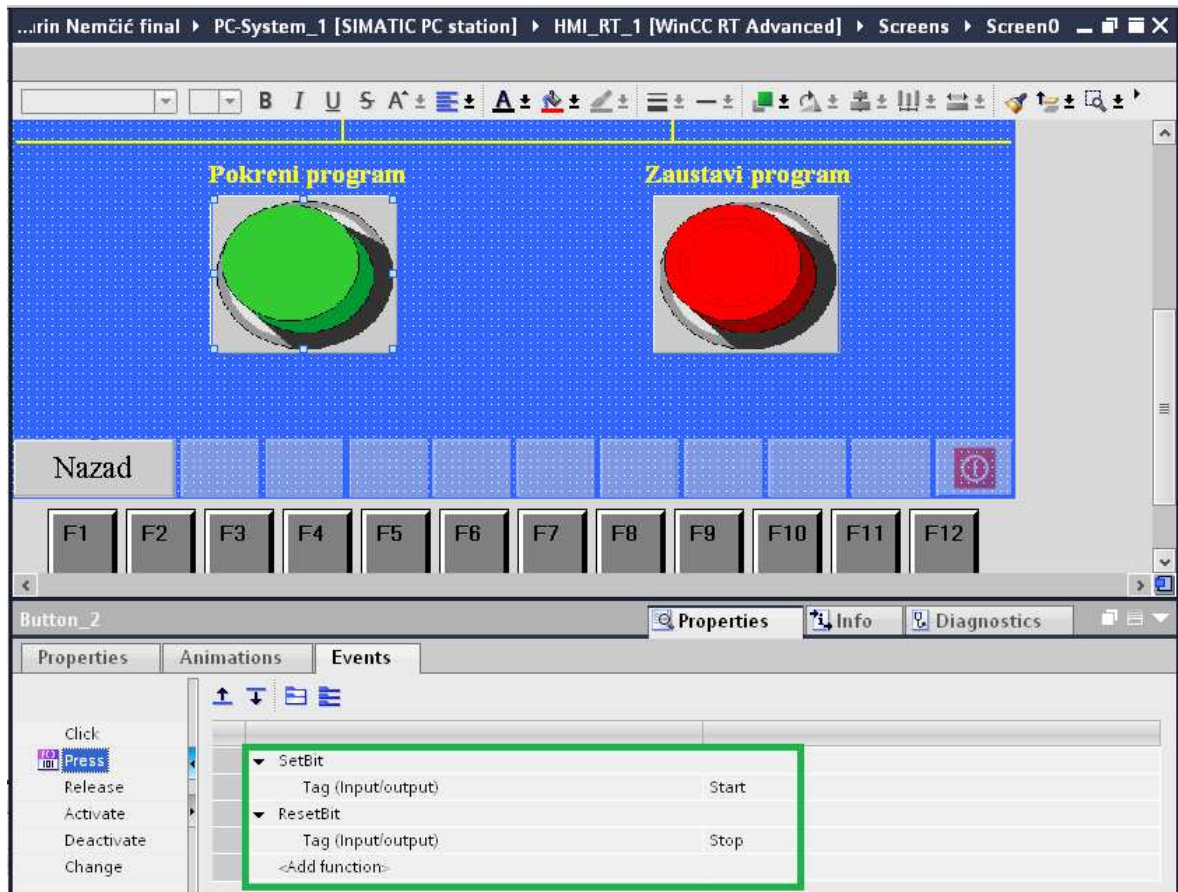
Zaslon „Screen0“ glavni je zaslon na kojemu se nadzire rad sustava. Na njemu se nalaze pojedini elementi koje je potrebno nadzirati. Elementi koji se prate su pokretna traka, senzor i kretnja pneumatske prihvatnice. Slika 7.3 prikazuje povezanost pojedinih stavki u prozoru sa slikama memorijskih riječi.



Slika 7.3: Povezivanje stavke sa slikom memorijske riječi

Povezivanje nekog predmeta kao što je elipsa, krug, kvadrat itd. sa slikom memorijske riječi omogućava upravljanje animacijom istog. Kao što je prikazano na slici 7.3, krug mijenja boje ovisno o stanju senzora. Ako se ispred senzora ne nalazi predmet, boja kruga je crvena, a ako senzor detektira predmet, boja postaje zelena. Isti princip je za rad pokretne trake i kretnju pneumatska prihvatnice.

Za pokretanje i zaustavljanje programa koriste se dva gumba prikazana na slici 7.4, pokreni program i zaustavi program. Pokretanje programa realizirano je tako da pritiskom na zeleni gumb memorijski bit M0.0 postavlja u visoko stanje, a memorijski bit M0.1 koji označava *Stop* naredbu, u nisko stanje. Ti bitovi, kao što je objašnjeno u samom programu pokreću, tj. zaustavljaju cijeli sustav.

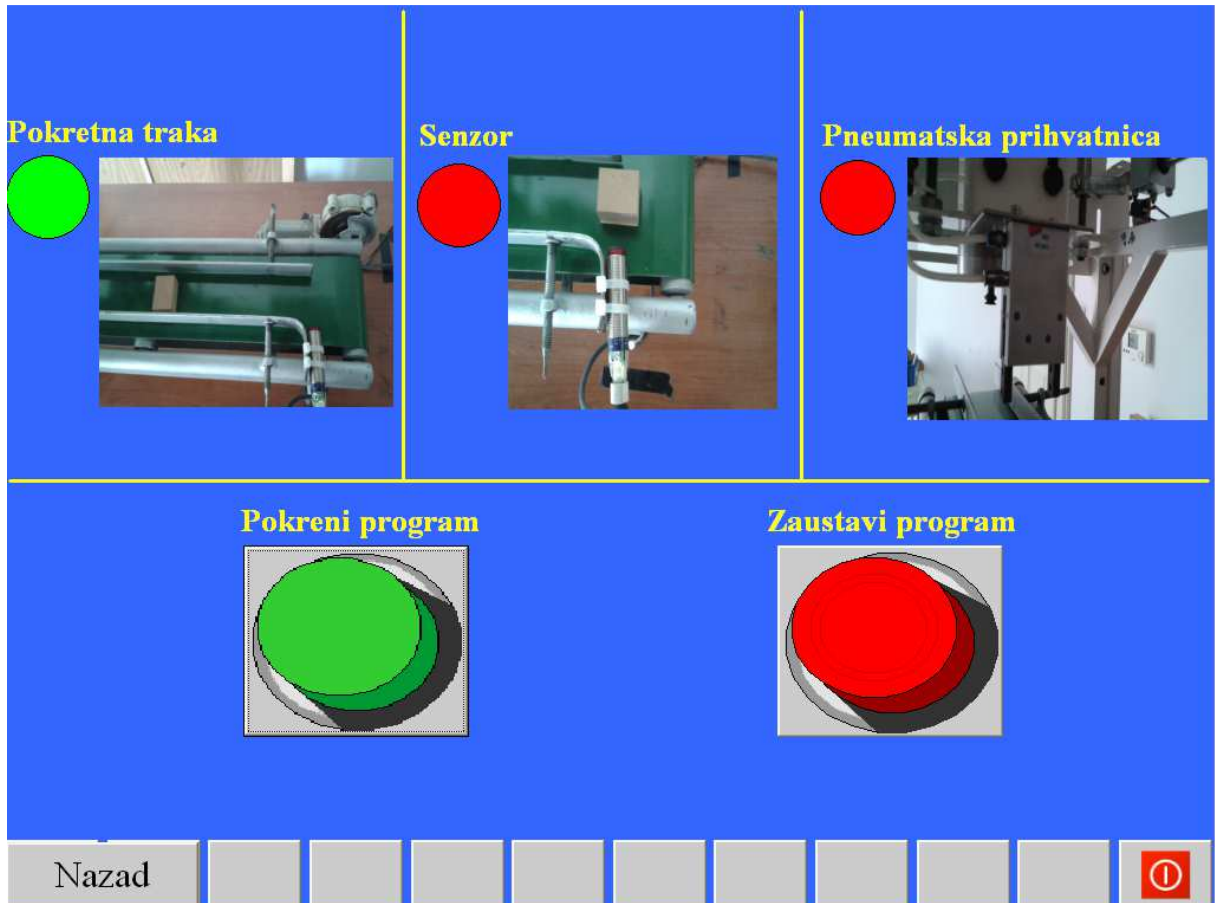


Slika 7.4: Start i stop programa

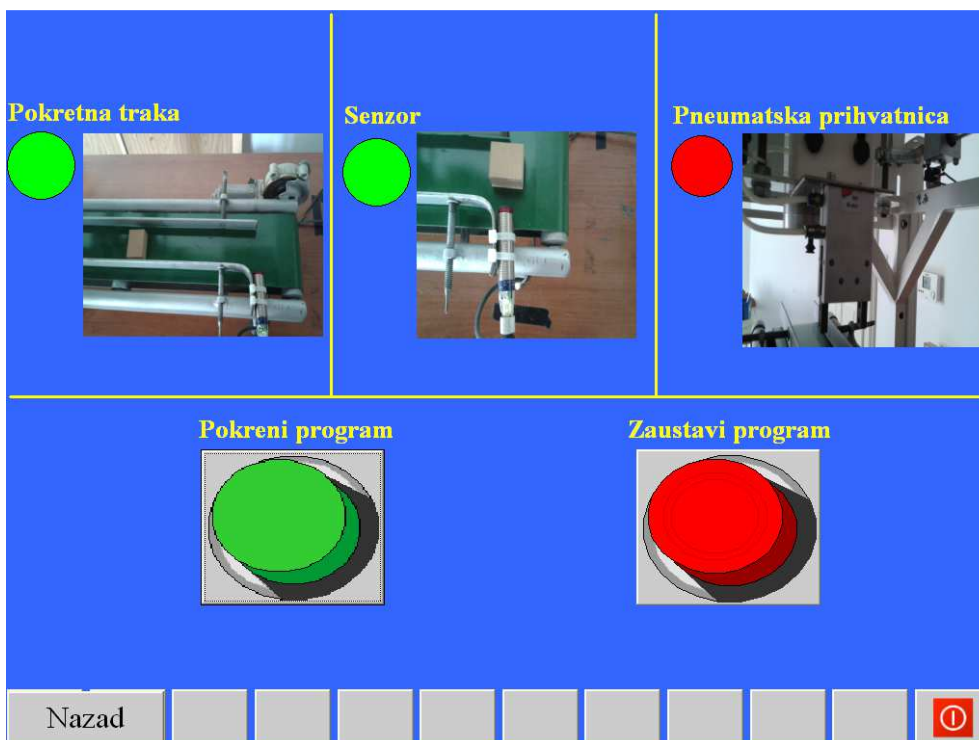
Identičan slučaj je kao i kod crvenog gumba, samo se za zaustavljanje programa koristi naredba *ResetBit*, povezana s memorijskom riječi M0.0 i *SetBit* povezana s memorijskom riječi M0.1.



Na slikama 7.5, 7.6 i 7.7 prikazan je sustav nadziranja rada pojedinih komponenti. Slika 7.5 prikazuje rad pokretne trake, dok su senzor i pneumatska prihvatnica i niskom stanju. Na slici 7.6 vidljiv je dolazak predmeta ispred senzora, ali zbog vremenskog brojača korištenog u programskom kodu pokretna traka nastavlja s radom još neko vrijeme.

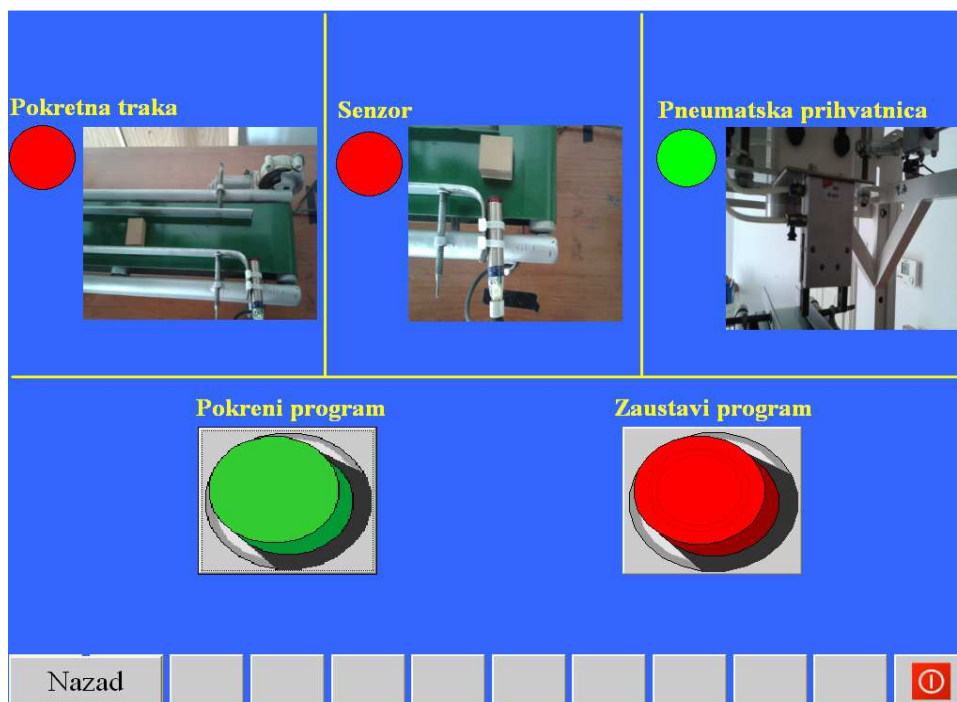


Slika 7.5: Rad pokretne trake u SCADA sustavu *TIA Portal*-a



Slika 7.6: Detekcija predmeta ispred senzora u SCADA sustavu *TIA Portal*-a

Slika 7.7 prikazuje kraj rada pokretne trake nakon određenog vremena detekcije predmeta senzorom. *Master* PLC šalje signal *slave* PLC-u i pneumatska prihvatnica počinje s radom.



Slika 7.7: Prihvatnica u radu prikazana SCADA sustavom u *TIA Portal*-u

## **9. Zaključak**

Automatizacija je vrlo složen postupak koji zahtijeva puno truda i rada. Poznavanje svakog dijela stroja koji se automatizira je neophodno. Ovaj konkretni zadatak automatizacije zahtijevao je puno učenja i prikupljanja materijala, široko poznavanje principa rada programljivih logičkih kontrolera i izrade SCADA sustava. Prilikom pisanja programa pojavilo se dosta grešaka u radu stroja koje je bilo potrebno otkloniti. Također, u nekoliko navrata improvizirani su dijelovi programa kako bi se zaobišle određene prepreke u radu. U cilju ovoga rada nije bitno kako je program napisan, već da odrađuje ono što je predviđeno, a to je i postignuto. Izrađeni SCADA sustav najjednostavnije predočuje stroj u radu.

## 10. Literatura

- [1] Šurina, Tugomir; Crneković, Mladen. *Industrijski roboti*, Školska knjiga, Zagreb, 1990.
- [2] Malčić, Goran. *Programljivi logički kontroleri*, Tehničko veleučilište u Zagrebu: Elektrotehnički odjel.
- [3] Siemens. *Simatic: S7-200 Programmable Controller System Manual*
- [4] <https://inverterdrive.com/file/Sitop-6EP1333-3BA00-datasheet-en> (dostupno 16. kolovoza 2016.)
- [5] <http://tinyurl.com/gmfsq7p> (dostupno 16. kolovoza 2016.)
- [6] <http://en.nanotec.com/products/541-st6018-stepper-motor-nema-24/> (dostupno 23. kolovoza 2016.)
- [7] <https://mall.industry.siemens.com/mall/en/WW/Catalog/Product/6ES7214-1AG40-0XB0> (dostupno 13. rujna 2016.)
- [8] <https://www.protis.hr/products/details/napajanje-400w-msi-ky400-atx-p4-hladenje-dva-ventilatora-od-8-cm/3794> (dostupno 14. rujna 2016.)
- [9] <http://www.sainsmart.com/plc-usb-ppi-programmer-communication-cable-for-siemens-s7-200-usb-isolated-ppi.html> (dostupno 03. rujna 2016.)
- [10] Vrhovski, Zoran. *Računalno upravljanje i vođenje procesima: Programiranje PLC-a*, Visoka tehnička škola u Bjelovaru
- [11] Petrović, Igor. *Automatizacija strojeva i uređaja 2: S7-1200 & TIA Portal*, Visoka tehnička škola u Bjelovaru, 2015.
- [12] <https://support.industry.siemens.com/cs/document/31044952/how-can-you-optimize-use-of-timers-in-step-7-micro-win-?dti=0&lc=en-WW> (dostupno 14. rujna 2016.)

## 11. Sažetak

Naslov: Automatizacija modela industrijskog pogona izvedenog manipulatorima i pokretnim trakama

Ovaj završni rad opisuje automatizaciju modela industrijskog pogona. Opisano je programiranje i upravljanje troosnim manipulatorom, te jednostavni SCADA sustav. Također, u uvodnom dijelu rada opisani su i vizualno prikazani elementi i dodatni uređaji korišteni u samom radu. Opisani su programski alati *Step 7 Micro/Win*, *TIA Portal* i *WinCC Flexible*.

Ključne riječi: automatizacija, manipulator, *Step 7 Micro/Win*, *TIA Portal*, *WinCC Flexible*, *S7-200*, *S7-1200*, *SCADA*

## 12. Abstract

Title: Automation of industrial facility model realized with manipulators and conveyor belts

This final thesis describes automation of industrial facility model. Programming and management of triaxial manipulator and a simple SCADA system is described. Moreover, in the introduction of thesis elements and additional devices used are described and visually represented. *Step 7 Micro/Win, TIA Portal* and *WinCC Flexible* description is included as well.

Keywords: automation, manipulator, *Step 7 Micro/Win, TIA Portal, WinCC Flexible, S7-200, S7-1200, SCADA*

Završni rad izrađen je u Bjelovaru 14.09.2016

---

(Potpis studenta)

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

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

Marin Nemčić

(Ime i prezime)

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

Svojim potpisom potvrđujem istovjetnost tiskane i elektroničke inačice završnog rada.

U Bjelovaru, 29.05.2016

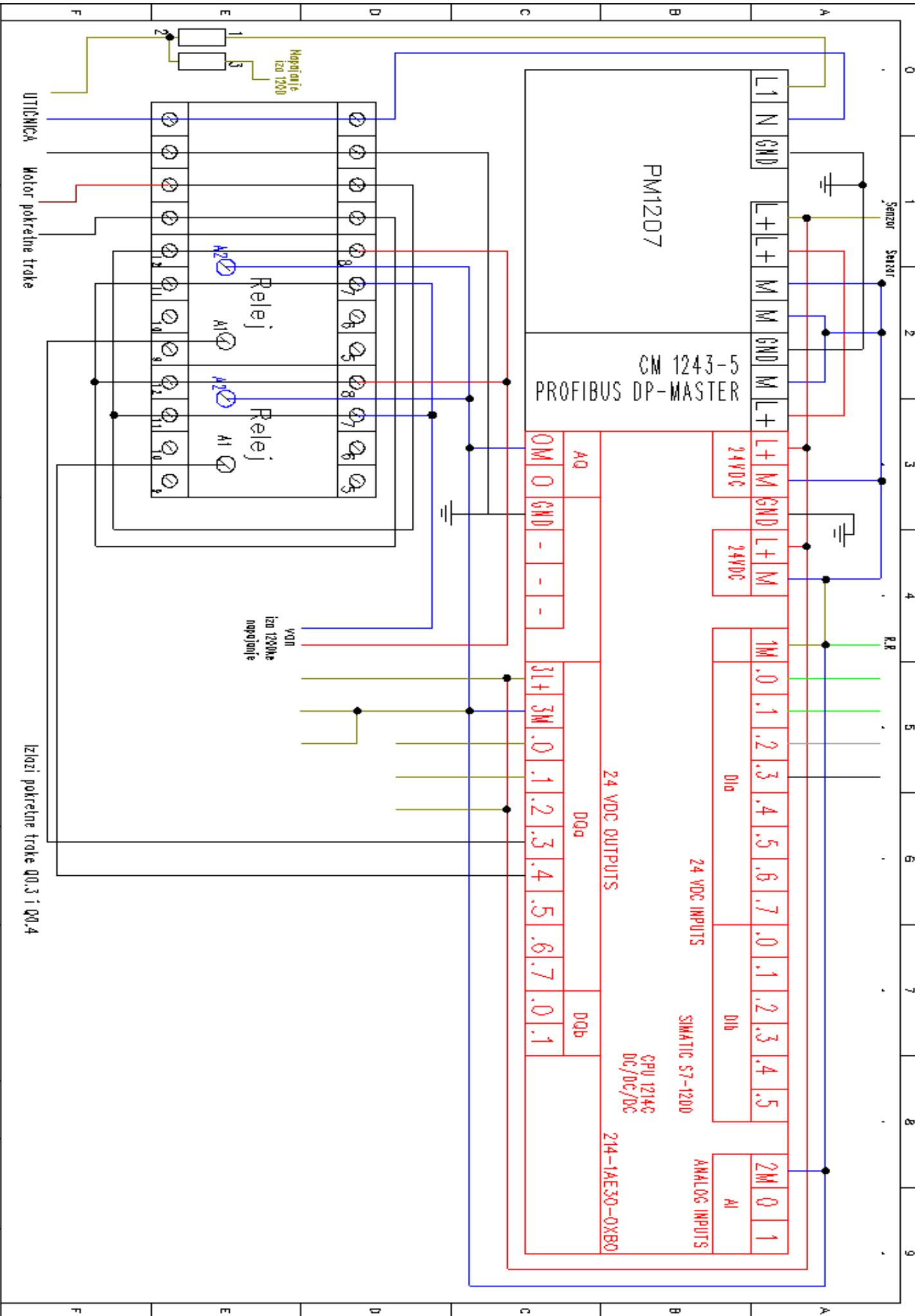
Nemčić

(potpis studenta/ice)

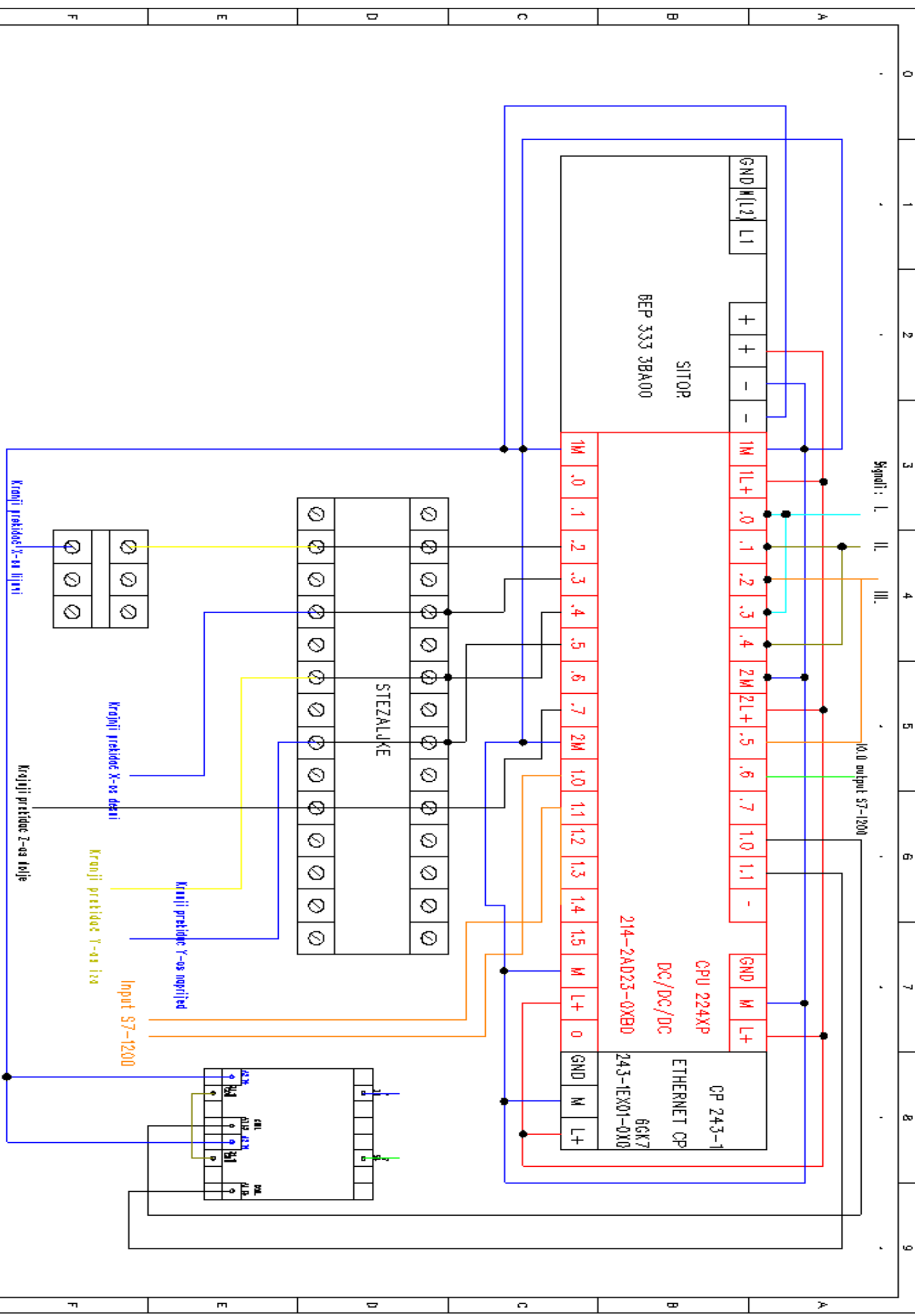


### **13. Prvitak**

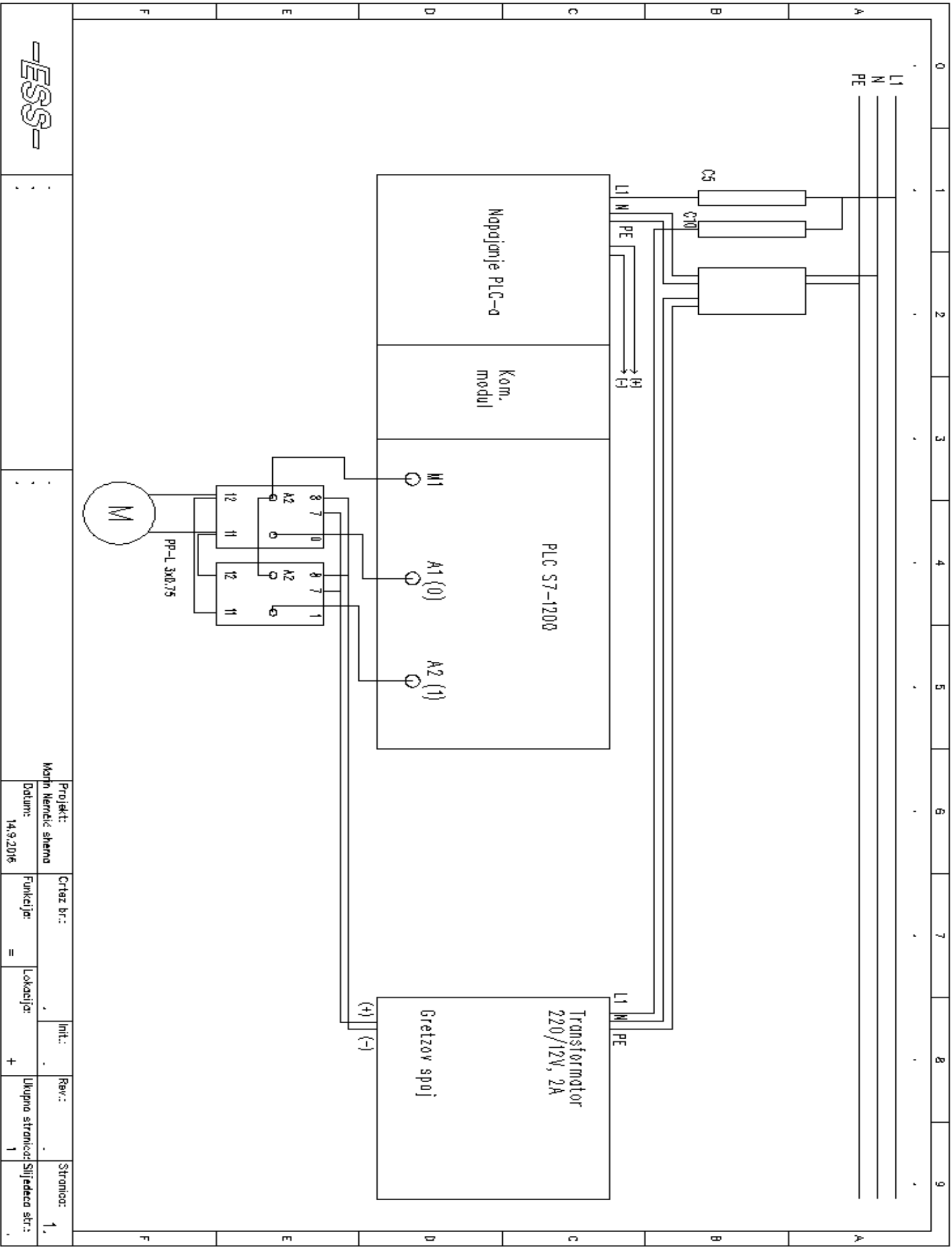
U prvitku se nalaze sheme ožičenja logičkih kontrolera izrađene u programskom jeziku *SEE Electrical*. Posljednja shema u prvitku izrađena je po uzoru na rad prošlih generacija Visoke tehničke škole u Bjelovaru



		ST-1200	Shema ST-1200 VTSBU 0314006847		
			Projekt: ST-1200 Marin Nemec	Crtač br.:	
		Datum: 27.10.2015	Funkcija:	Inti.:	Rev.:
			=	Lokacija:	Stranica:
			+	Ukupno stranica:	1.
				Sifredica str.:	



<b>-ESS-</b>		S7-200			
		Marin		Projekt: Memorij S7-200	
		Datum: 29.10.2015		Crtez br.:	
		Funkcija: =		Lokacija: +	
		Inti: .		Rev.: .	
		Ukupno stranica: 1		Stranica: 1.	
		Sifredica str.: .			



-ESS-

Projekt:	Mornj Neretiv skema		Inti:		Stranica:	1.	
Datum:	14.9.2016	Funkcija:	=	Lokacija:	+	Ukupno stranica: Sifredica str:	1