

# VODIČ KROZ SCADA PROGRAMIRANJE S PRIMJEROM TIA PORTALA

---

**Dobrenić, Ivo**

**Undergraduate thesis / Završni rad**

**2020**

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

*Permanent link / Trajna poveznica:* <https://um.nsk.hr/um:nbn:hr:128:083706>

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

*Download date / Datum preuzimanja:* **2025-03-11**



**VELEUČILIŠTE U KARLOVCU**  
Karlovac University of Applied Sciences

*Repository / Repozitorij:*

[Repository of Karlovac University of Applied  
Sciences - Institutional Repository](#)



zir.nsk.hr



DIGITALNI AKADEMSKI ARHIVI I REPOZITORIJ

Veleučilište u Karlovcu

Strojarski odjel

Stručni studij Mehatronike

Ivo Dobrenić

# **VODIČ KROZ SCADA PROGRAMIRANJE S PRIMJENOM TIA PORTALA**

ZAVRŠNI RAD

Karlovac, 2020.

Karlovac University of Applied Sciences  
Mechanic Department

Professional undergraduate study of Mechatronics

Ivo Dobrenić

# **TIA PORTAL APPLICATION IN SCADA PROGRAMMING**

FINAL PAPER

Karlovac, 2020.

Velečilište u Karlovcu  
Strojarski odjel

Stručni studij Mehatronike

Ivo Dobrenić

# **VODIČ KROZ SCADA PROGRAMIRANJE S PRIMJENOM TIA PORTALA**

ZAVRŠNI RAD

Mentor:

dr. sc. Vladimir Tudić, prof. v.š.

Karlovac, 2020.



**VELEUČILIŠTE U KARLOVCU**  
KARLOVAC UNIVERSITY OF APPLIED SCIENCES  
Trg J.J. Strossmayera 9  
HR-47000, Karlovac, Croatia  
Tel. +385 - (0)47 - 843 – 510  
Fax. +385 - (0)47 - 843 – 579



## **VELEUČILIŠTE U KARLOVCU**

Stručni studij: STROJARSTVO

Karlovac, 28.02.2020.

Usmjerenje: MEHATRONIKA

### **ZADATAK ZAVRŠNOG RADA**

Student: IVO DOBRENIĆ

Matični broj: 0112616079

Naslov: VODIČ KROZ SCADA PROGRAMIRANJE S PRIMJENOM TIA PORTALA

Opis zadatka:

U teorijskom dijelu zadatka opisati koncept i funkcioniranje SCADA sustava. Napraviti vodič kroz programiranje SCADA sustava u TIA Portal programskom paketu. U eksperimentalnom dijelu rada potrebno je razviti programsku logiku i simulirati SCADA sustav koji se sastoji od njegovih ključnih dijelova: upravljačke logike, korisničkog sučelja te mogućnosti nadzora i upravljanja procesa.

Zadatak zadan:  
28.02.2020.

Rok predaje rada:  
10.04.2020.

Predviđeni datum obrane:  
29.04.2020.

Mentor:

dr.sc. Vladimir Tudić, prof.v.š.

Predsjednik Ispitnog povjerenstva:

dr.sc. Damir Kralj, prof.v.š.

## **PREDGOVOR**

Izjavljujem da sam završni rad na temu „Vodič kroz SCADA programiranje s primjenom TIA Portala“ izradio samostalno, koristeći navedenu stručnu literaturu i znanje stečeno tijekom studija.

Zahvaljujem se svojoj obitelji na strpljenju i povjerenju koje su mi ukazali tokom studijskog obrazovanja.

Posebno se zahvaljujem profesoru i mentoru dr. sc. Vladimiru Tudiću na korisnim savjetima, prijedlozima i uputama koje su mi pomogle pri izradi završnog rada.

Ivo Dobrenić

## **SAŽETAK:**

Sadržaj rada obrađuje temu nadzora, kontrole i automatskog upravljanja industrijskih sustava. Konkretno jedan od danas najzastupljenijih sustava automatizacije poznatog kao SCADA (Supervisory Control And Data Acquisition). U radu je detaljno obrazložena povijest razvoja, funkcija i implementacija sustava kao i mane istog. Drugi dio rada odnosi se na vodič s generalnim uputama za programiranje u jednom od najzastupljenijih programskih sučelja za ove sustave (TIA Portal). Sučelje je namijenjeno za konfiguraciju, logičko programiranje, testiranje i dijagnozu svih SIMATIC kontrolera, odnosno kontrolera proizvedenih od njemačke kompanije Siemens. Za potrebe ovog rada koristit će se simulacija SIMATIC S7-1500 PLC kontrolera s posebnim modulima za digitalni i analogni input/output, a za HMI (Human-Machine Interface) odnosno grafičko sučelje za nadzor i upravljanje korišten je program WinCC. Kao primjer programiran je i simuliran sustav upravljanja autopraonice. Rad je također nadopunjen teorijskim cjelinama koje objašnjavaju komponente SCADA sustava kao što su PLC, RTU, komunikacijska infrastruktura i HMI.

Ključne riječi: SCADA, TIA Portal, vodič, HMI, PLC

## **SUMMARY:**

Content of the paper deals with the topic of supervision, commission and automation control of industrial systems. In particular, one of the most common automation systems today known as SCADA (Supervisory Control And Data Acquisition). The history of system development, functions and implementation as well as its flaws is explained in detail in the paper. The second part of the paper deals with a guide with general programming instructions in one of the most common programming interfaces for these systems (TIA Portal). The interface is intended for the configuration, logical programming, testing and diagnosis of all SIMATIC controllers, which are manufactured by German company Siemens. For the purposes of this work, it will be used simulation of the SIMATIC S7-1500 PLC controller with special modules for digital and analog input / output, and for HMI (Human-Machine Interface), graphical interface for monitoring and control the WinCC program was used. As an example, the car wash management system was programmed and simulated. The paper is also complemented by theoretical units that explain the components of SCADA systems such as PLC, RTU, communication infrastructure and HMI.

Key words: SCADA, TIA Portal, guide, HMI, PLC



## **SADRŽAJ:**

<b>1. UVOD .....</b>	<b>1</b>
<b>2. OPĆENITO O SCADA SUČELJU .....</b>	<b>2</b>
Sl. 2.1. Predodžba suvremenog SCADA sustava. Izvor [7]. .....	2
2.1. Povijest razvoja .....	2
2.2. Koncept SCADA sustava .....	4
2.2.1. Centralna stanica .....	5
2.2.2. Komunikacijska infrastruktura .....	5
2.2.3. PLC / RTU .....	6
2.3. Alarmni sustav i zaštita.....	9
2.4. Sigurnosni problemi .....	11
<b>3. VODIČ KROZ TIA PORTAL.....</b>	<b>12</b>
3.1. Uspostava hardware-a .....	12
3.1. Razvoj upravljačke logike .....	15
3.2. Tags and data types .....	18
3.3. HMI .....	20
<b>4. IZRADA SUSTAVA UPRAVLJANJA I NADZORA AUTOMATSKE AUTOPRANICE.....</b>	<b>24</b>
<b>5. ZAKLJUČAK .....</b>	<b>32</b>
<b>LITERATURA.....</b>	<b>33</b>
<b>POPIS SLIKA .....</b>	<b>35</b>

## 1. UVOD

Završni rad obrađuje temu sustava za nadzor, kontrolu i prikupljanje informacija (SCADA) koji su široko primijenjeni u današnjoj automatizaciji. Zbog konstantnog istraživanja i razvoja u području robotike, računarstva, pneumatike i elektronike dostupnost i efikasnost automatiziranih mehanizama postaje sve veća. Upravo zbog prethodno spomenutog konstantnog razvoja potražnja postaje sve veća i zahtjevnija. Konkurentnost na tržištu dostiže se ravnotežom između produktivnosti, kvalitete, cijene i fleksibilnosti, a što se stabilnija ravnoteža postigne ostvaruje se veća kompetitivna prednost na globalnom tržištu.<sup>1</sup> Kako bi ponuditelji ostali u koraku s konkurencijom obvezni su unaprjeđivati svoje sustave sa suvremenom tehnologijom. Jedna od tih tehnologija je SCADA sustav upravljanja i nadzora koji je detaljno obrazložen u nastavku ovog rada.

S razvojem automatizacije struka ovog područja prisiljena je pratiti trendove programiranja sustava. Proporcionalno s povećanjem potražnje proizvoda i usluga dolazi i do usavršavanja već postojećih načina programiranja, grafičkog prikaza i upravljanja složenih procesa, stoga se ovim radom čitatelju s osnovnim znanjem logičkog upravljanja nastoje približiti današnji trend projektiranja SCADA sustava. Programski zadatak izrađen je u sučelju TIA portal (Totally Integrated Automation Portal), namijenjenom za konfiguraciju, logičko programiranje, testiranje i dijagnozu svih SIMATIC kontrolera, odnosno kontrolera proizvedenih od njemačke kompanije Siemens. Rad PLC uređaja simuliran je programom S7-PLCSIM, a grafičko sučelje za upravljanje i nadzor WinCC Runtime aplikacijom.

## 2. OPĆENITO O SCADA SUČELJU

Akronim SCADA dolazi od početnih slova riječi koje opisuju sustav - Supervisory Control and Data Acquisition. „SCADA je sustav koji omogućava operatoru upravljanje i nadgledanje jedne ili više povezanih industrijskih jedinica, kao što su naftna ili plinska polja, cijevovodi, elektrane i slično, slanjem i primanjem signala u ta postrojenja, gdje god se oni nalazili na terenu (npr. otvaranje i zatvaranje ventila, sklopki, praćenje alarma, prikupljanje mjerenih informacija itd.).“ [2]. (Boyer, 2004.)



Sl. 2.1. Predodžba suvremenog SCADA sustava. Izvor [7].

### 2.1. Povijest razvoja

Razvoj telemetrijske znanosti usko je vezan uz razvoj SCADA-e. Sredinom 20. stoljeća dolazi do prvih razvoja zrakoplova i raketa, istraživanja vremenskih pojava. To je zahtijevalo sakupljanje informacija s opreme do koje je promatračima prostorno bilo jako

teško ili nemoguće doći. Primjerice, u prvim letjelicama bilo je mjesta za pilota, ali ne i za konstrukcijske i testne inženjere kako bi pratili informacije sa stotine senzora postavljenih oko motora i konstrukcije letjelice prilikom leta. Također, prve rakete nisu imale mjesta za pilota, ali i da su imale, zbog manjka sigurnosti, teško bi bilo naći osobu za mjerenje.

Rješenje za neke od ovih problema je došlo iz područja koje se već koristilo u to vrijeme. Primjerice, na slici 2.2. prikazano je slanje signala žicom u željeznici kako bi centralna stanica pratila poziciju vlakova, zatim telegramima slala upute mjesnim kontrolorima. Ipak, prethodno navedeno rješenje je bilo korisno samo za dvije dimenzije odnosno kopno, a kako bi se proširili i na letjelice, rakete i druga nedostupna mjesta bio je potreban potpuno drugi pristup. Rješenje je pronađeno u radiju koji se u to vrijeme paralelno razvijao. Prvotno radio je mogao samo primati informacije, no ne i slati ih. Prvo primanje i slanje signala omogućeno je žicom. To dostignuće uvelike je pomoglo naftnim i vodoopskrbnim kompanijama čiji su se senzori i aktuatori na mreži nalazili pod zemljom ili na teško dostupnim mjestima [2]. Tako je kombinacija žice i radija postala najefektivnijom metodom sakupljanja i slanja signala. Novi veliki iskorak u razvoju SCADA-e dogodio se izumom računala. Već početkom 1960. godina računala su imala veliku ulogu u nadzoru i upravljanju. Mrežom žica i radija omogućeno je primanje svih informacija u jedno računalo daleko udaljeno od mjesta procesa. Nadalje, razvojem radija zamjenjuju se žice osjetljive na hrđu, glodavce i vanjske radove. Također, razvojem radija sve je veća zasićenost frekvencija te postaje sve jasnije kako je nova metoda komuniciranja obavezna. Integrirana je satelitska komunikacija, kao i komunikacija mikrovalovima. Daljnjim razvojem računala, mikroprocesora, senzora, mrežnog povezivanja omogućena je decentralizacija i veća kvaliteta sustava [2].

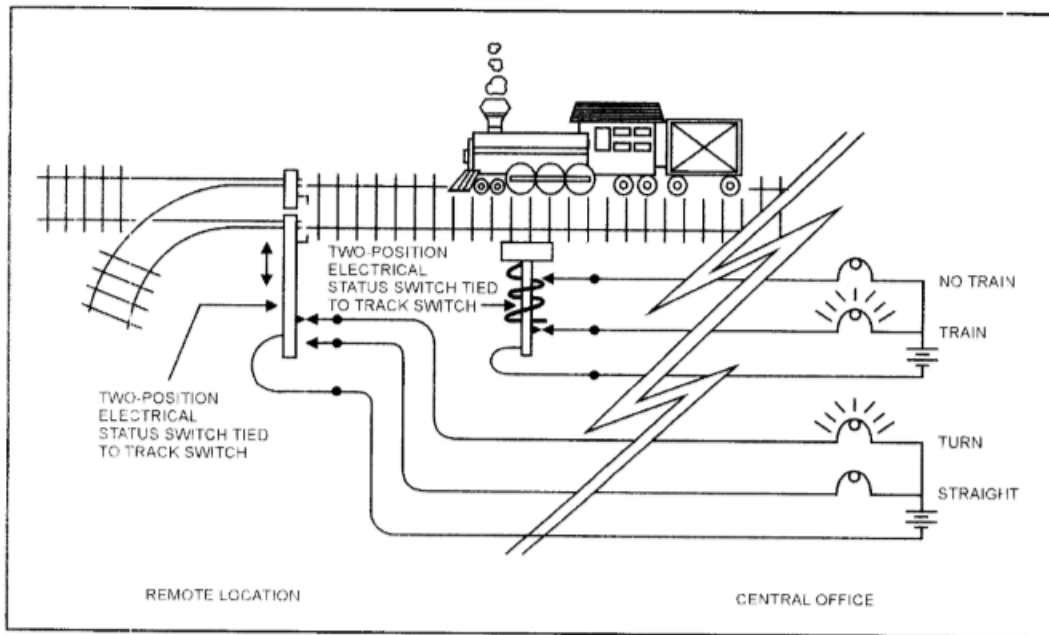
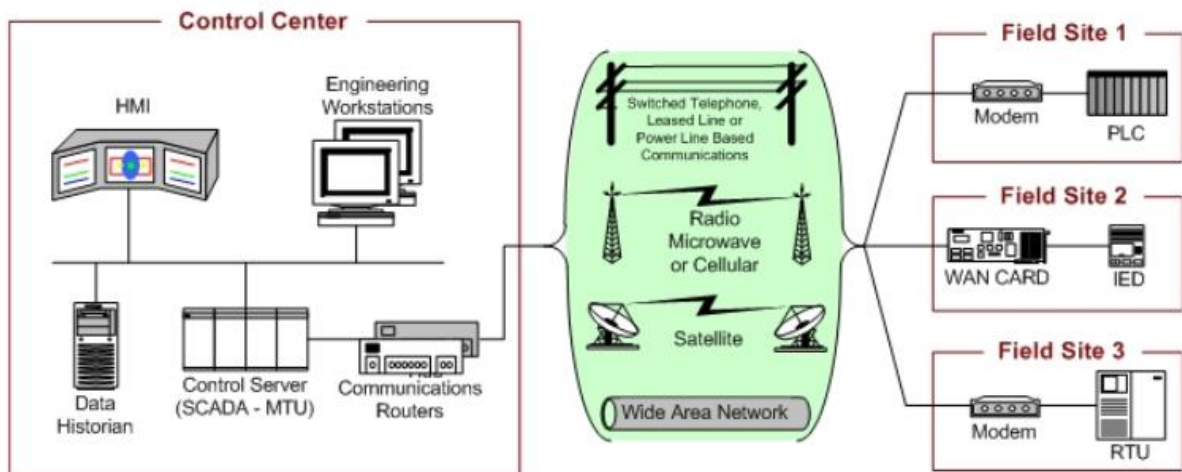


Figure 3-1. Early Telemetry

Sl. 2.2. Predodžba prvotnih slanja signala u željeznici. Izvor [2].

## 2.2. Koncept SCADA sustava

SCADA sustav se generalno sastoji od centralne stanice i podstanica koje prikupljaju i šalju sve informacije i naredbe, komunikacijske infrastrukture i lokalnih kontrolera PLC, RTU ili IED koji primaju signale od senzora ili upravljaju aktuatorima kao što je predočeno na slici 2.3.



Sl. 2.3. Predodžba: Struktura SCADA sustava. Izvor [2].

### 2.2.1. Centralna stanica

Centralna stanica prati i kontrolira procese koji se dešavaju na terenu. Prikazuje ih na korisničkom sučelju i može automatizirano ili naredbama operatera odgovoriti na određene događaje. Sastoji se od SCADA servera (MTU), komunikacijskih rutera, korisničkog sučelja (HMI), a po potrebi stanice posjeduju i dodatne radne stanice, spremište informacija ili alarmni sustav [3]. Centralna stanica je također zadužena za dijagnostiku.

### 2.2.2. Komunikacijska infrastruktura

Cijeli SCADA sustav oslanja se u potpunosti na komunikaciju između svih njegovih komponenti. Danas komunikacija može biti ostvarena raznim načinima kao što su bakrene

žice, optička vlakna, radio signali, sateliti, Wi-Fi i dr. Značajan broj uređaja korištenih u automatizaciji izumljen je prije standardizacije u komunikaciji, a rezultat toga je bilo mnoštvo protokola u istoj industriji koji nisu mogli međusobno komunicirati. Današnji standardizirani protokoli za SCADA-u su Modbus, DNP3 (Distributed Network Protocol V3) te protokoli iz standarda IEC (The International Electrotechnical Commission), odnosno IEC 60870-5 i IEC 61850 [4].

### 2.2.3. PLC / RTU

RTU (Remote Terminal Unit) – je samostalni elektronički uređaj koji skuplja podatke od okolnih senzora i upravlja aktuatorima (analogni ili digitalni inputi/outputi). Podatke uglavnom prenosi dalje centralnoj stanici. RTU se danas obično koriste pri težim vremenskim uvjetima i udaljenim nedostupnim mjestima (moguće solarno napajanje kao na slici 2.4. Za razliku od prvotnih RTU-a oni moderni imaju ugrađene dodatne komunikacijske uređaje i mikrokontrolere kako bi se omogućila dvosmjerna komunikacija [5].



Sl. 2.4. Predodžba: RTU uređaj napajan solarnom energijom. Izvor [8].

PLC (Programmable Logic Controller) je sličan RTU-u, ali je mogućnost funkcija i kontrola mnogo sofisticiranija. Specijalni je oblik mikroprocesorskog kontrolera koji koristi memoriju za spremanje instrukcija i izvršavanje funkcija kao što su logika, mjerenja, računske operacije s ciljem upravljanja procesa. Može izvršavati jednostavne i vrlo složene operacije. Obično nemaju ekran tj. *display*, tipkovnicu jer je namijenjen upravljanju sustava



koristeći jedan program. Za razvijanje programa potrebno je programsko sučelje koje zavisi od proizvođača PLC-a. Određenim podacima s PLC-a se može pristupiti pomoću korisničkog sučelja HMI koje je također dizajnirano u posebnom programskom sučelju. U eksperimentalnom dijelu rada prikazati će se primjer programiranja Siemensovog PLC-a S7-1500 prikazanog na slici 2.5., u programskom sučelju namijenjenom ovom tipu PLC-a, TIA Portalu.



2.5. Predodžba SIEMENS PLC S7-1500 s modulima. Izvor [9].

HMI je sučelje koje prikazuje informacije prikupljene od SCADA sustava operateru kako bi mogao upravljati procesima. Također alarmira korisnika te omogućuje hijerarhiju ovlasti raznih operatera. Slika 2.6. prikazuje primjer HMI uređaja. Primjerice, simbol pumpe pokazuje kada je pumpa u radu, a zapisi pokraj spremnika pokazuju mjerene veličine sustava (temperaturu, protok, popunjenost spremnika itd.). Operator može uključiti ili isključiti pumpu, zatvoriti ili otvoriti elektro-ventile i slično.

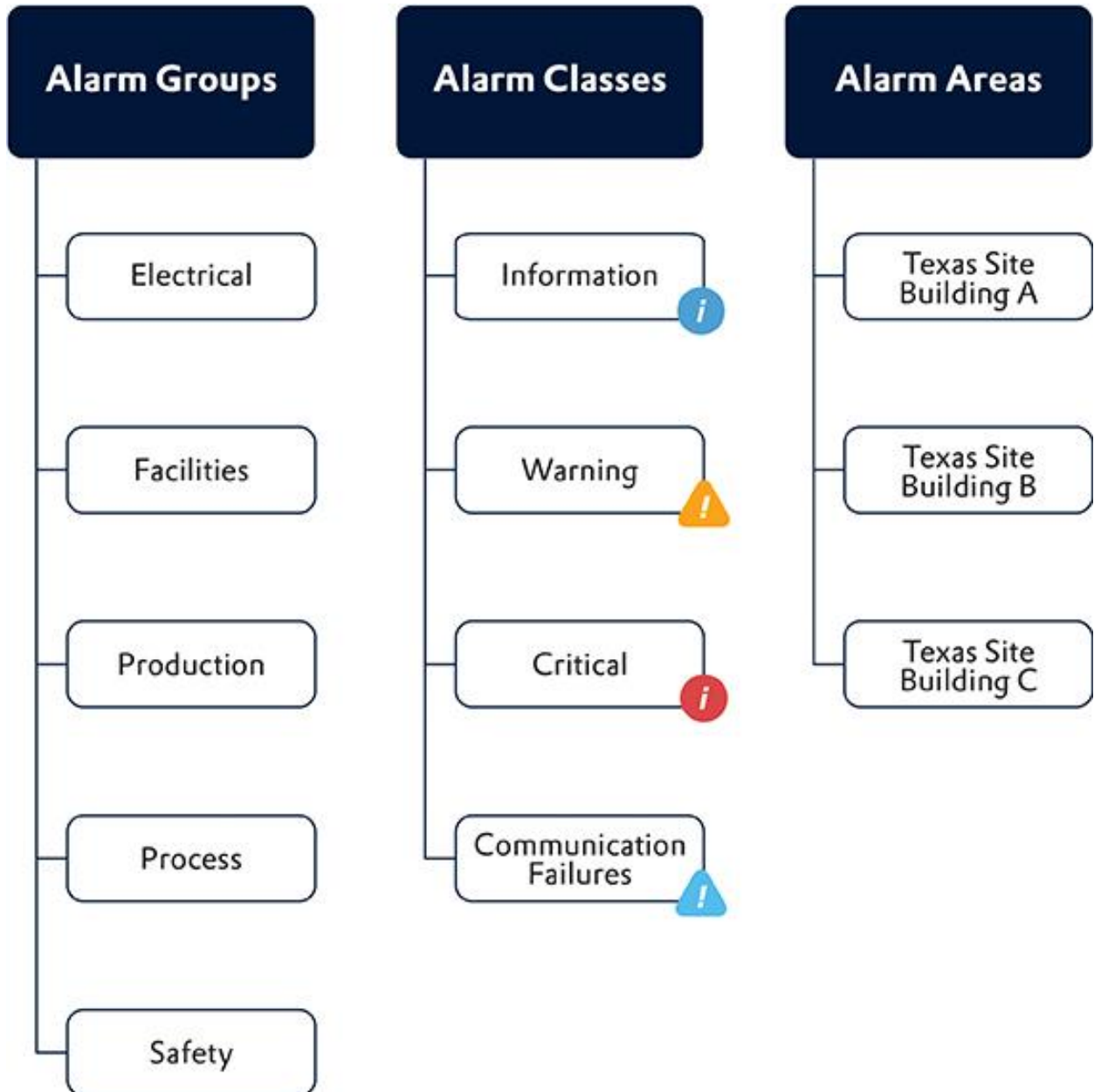


Sl. 2.6. Predodžba HMI uređaja. Izvor [10].

### 2.3. Alarmni sustav i zaštita

Svi procesi sustava trebaju biti opremljeni sustavom zaštite jer prilikom kvara može doći do ozbiljnih povreda zaposlenika, drugih osoba ili može oštetiti opremu. Zaštitni program treba biti napravljen tako da ima prioritet u izvršavanju svih naredbi. Funkcionira tako da prati zadane parametre i izvršava potrebne radnje kako bi sustav ostao u sigurnim granicama tih parametara. Kada zaštitni sustav reagira i zaustavi dio procesa operator bi trebao što prije biti obaviješten. Alarmni sustav upozorava operatora na sve nepravilnosti

nadziranih procesa. Ovisno o hijerarhiji alarma prikazanih na slici 2.7. izvršavaju se odgovarajuće akcije. Za niže razine dovoljan je svjetlosni signal u dijelu ekrana, za nešto ozbiljnije vidljiva poruka na ekranu, a za najozbiljnije čak i automatsko slanje SMS poruke, poziva i e-maila operateru [2].



Sl. 2.7. Hijerarhija alarma. Izvor [11].

## 2.4. Sigurnosni problemi

Kada uzmemo u obzir da s korisničkog sučelja možemo u trenu ugasiti ili preusmjeriti ključne radnje, recimo naftne platforme ili električne distribucije, dolazimo do zaključka kako je sigurnost takvih sustava izuzetno bitna. Zbog sve većih proširenja, ozbiljnosti procesa i udaljenosti pojedinih komponenti sustava, potpuna izolacija ovih upravljačkih veza više nije moguća. Zbog sve veće izloženosti i probijanja zaštita SCADA sustava ISA (Međunarodno društvo za automatizaciju) razvilo je nekoliko standarda i smjernica kojeg se kompanije trebaju pridržavati [5].

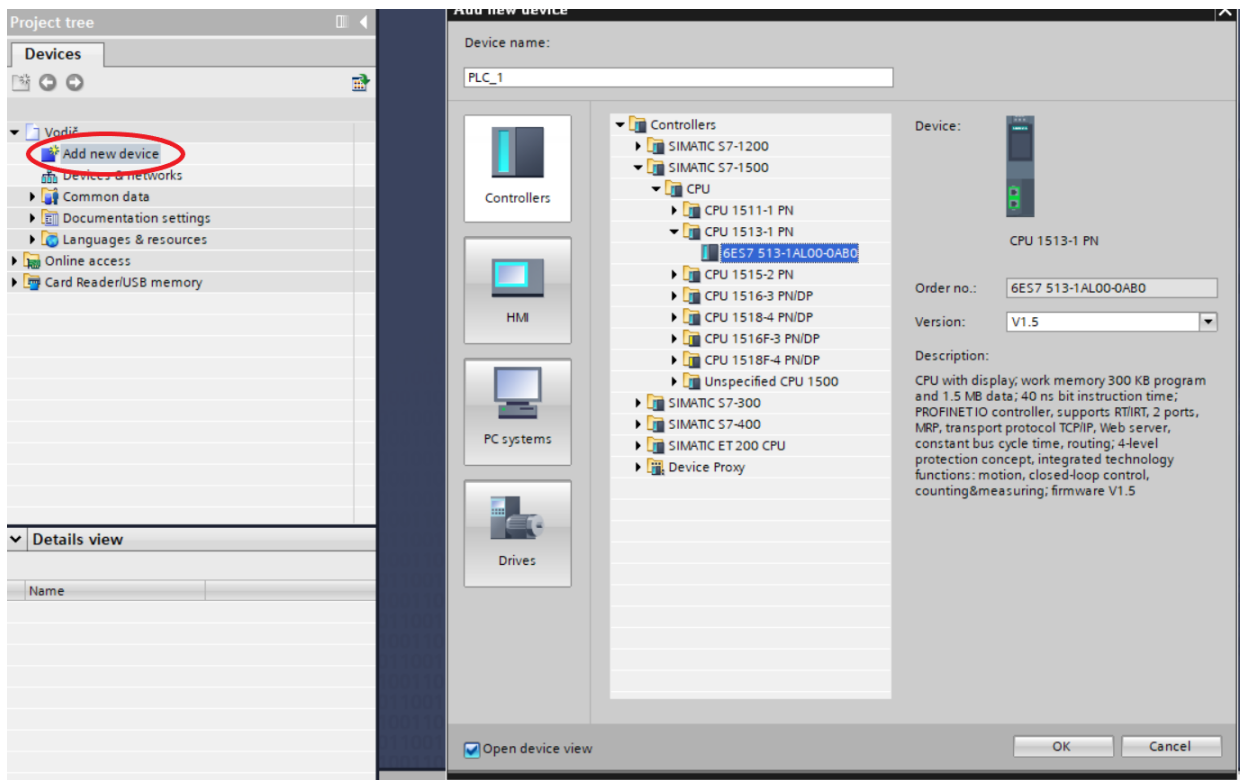
### 3. VODIČ KROZ TIA PORTAL

*„Totally Integrated Automation Portal (TIA Portal) objedinjuje SIMATIC Totally Integrated Automation (TIA) proizvode u jedinstvenu programsku aplikaciju. Svi TIA proizvodi rade zajedno unutar istog programskog okruženja i pružaju podršku korisniku u svim segmentima potrebnim za stvaranje rješenja automatizacije. TIA Portal se koristi za konfiguraciju i programiranje PLC uređaja, ali i za vizualizaciju procesa u jedinstvenoj razvojnoj okolini. Svi podaci se spremaju u zajedničku projektnu datoteku, koristi se zajedničko korisničko sučelje za sve zadaće preko kojeg je moguće pristupiti svim programskim i vizualizacijskim funkcijama u svakom trenutku.“ [6]. (Maršić, Malčić, Vlašić, 2014.)*

#### 3.1. Uspostava hardware-a

Na početnom zaslonu TIA Portala prikazan je izbor odabira otvaranja izrađenog projekta, razvoj potpuno novog programa te Online & Diagnostic koji se koristi za konfiguraciju sustava u radu (ograničene mogućnosti konfiguriranja).

Odabere se novi projekt te se klikom na „Project view“ u lijevom donjem kutu ulazi u prozor za uređivanje projekta. Prvo se dodaje novi uređaj klikom na „Add new device“.



Slika 3.1. Odabir odgovarajućeg hardwarea. Izvorno autor.

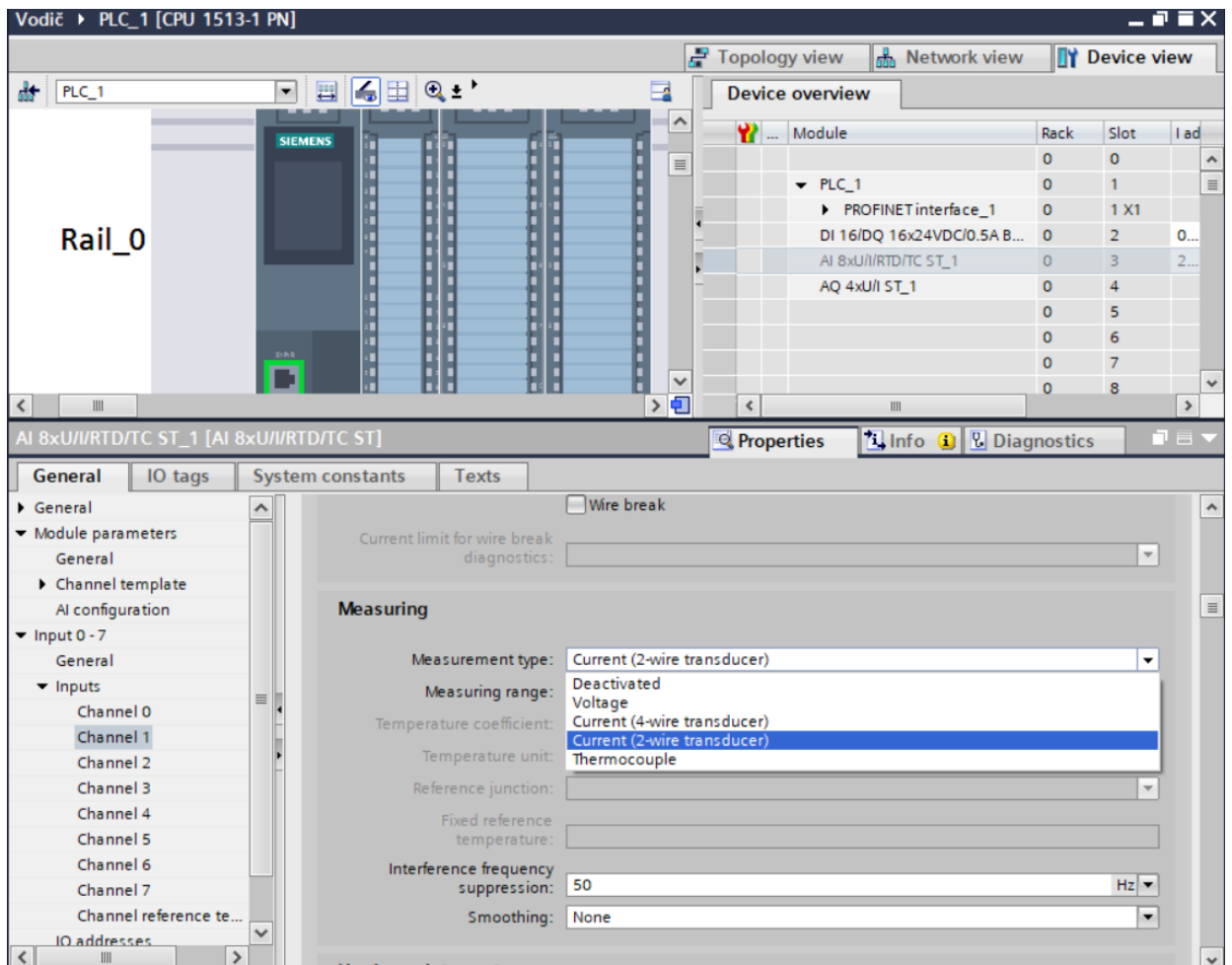
Potom se otvara prozor s izborom kontrolera, korisničkog sučelja, računalnog sklopovlja i „Drivera“ (služe za upravljanje izmjeničnim motorima). Ikona s nazivom „Controllers“ predstavlja grupu SIMATIC kontrolera. Odabire se odgovarajući kontroler ovisno o potrebama ili uređaju kojeg imamo. Bitno je odabrati isti CPU uređaja kojeg posjedujemo inače komunikacija između (PORTALA) TIA-e i fizičkog uređaja neće biti moguća. U slučaju kada se ne zna koji se tip CPU-a posjeduje, odabire se opcija „Unspecified CPU“, gdje će Portal kasnije sam prepoznati povezani uređaj. Za potrebe ovog vodiča odabran je „CPU 1513-1 PN“ koji će u eksperimentalnom dijelu rada biti simuliran programom S7-PLC SIM.

Akronim CPU stoji za Central Processing Unit, broj 15 označava S7-1500, 13 predstavlja tip ovog PLC-a (generalno veći broj na ovom mjestu označava veću funkcionalnost) i PN označava tip komunikacije – ProfiNet.

Konkretno ovaj tip PLC-a ne posjeduje digitalne i analogne ulaze ili izlaze, stoga se moraju dodati odgovarajuće kompatibilni moduli. Na desnoj strani prozora nalazi se mapa „Catalog“ u kojoj se mogu odabrati željeni moduli kao što su: izvori izmjeničnog ili istosmjernog napajanja, digitalni/analogni ulazi i izlazi, komunikacijski moduli, tehnološki moduli (npr. služe za brzo brojanje impulsa inkrementalnih enkodera).

Klikom na PLC, zatim Properties dolazi se do raznih postavki uređaja od kojih bi se u ovom radu izdvojio „System and clock memory“. Klikom na „Enable the use of clock memory byte“ omogućuje se korištenje stvarnog sata PLC-a (brzina od 10 do 0.5 HZ je spremljena u prvi byte bit memorije) kojeg se kasnije može implementirati u program (recimo za brojanje radnih sati procesa). Neke od ostalih mogućnosti: upravljenje komunikacijom, pristup web serveru, načini pokretanja nakon prisilnog gašenja PLC-a, konfiguracija ovlasti korisnika, napajanje sustava itd.

Klikom na „Properties“ modula s analognim inputima zatim „Inputs“ dolazi se do mogućnosti konfiguriranja svakog ulaza modula. Ovdje je bitno uskladiti signale senzora koji su dovedeni na ulaze. S obzirom na vrstu senzora kojeg očitavamo uspostavljamo željeno mjerenje (napon, struja, termopar ili bez mjerenja) na tom ulazu kako je prikazano na slici 3.2. Danas je standardizirana upotreba 4-20 mA strujne petlje.

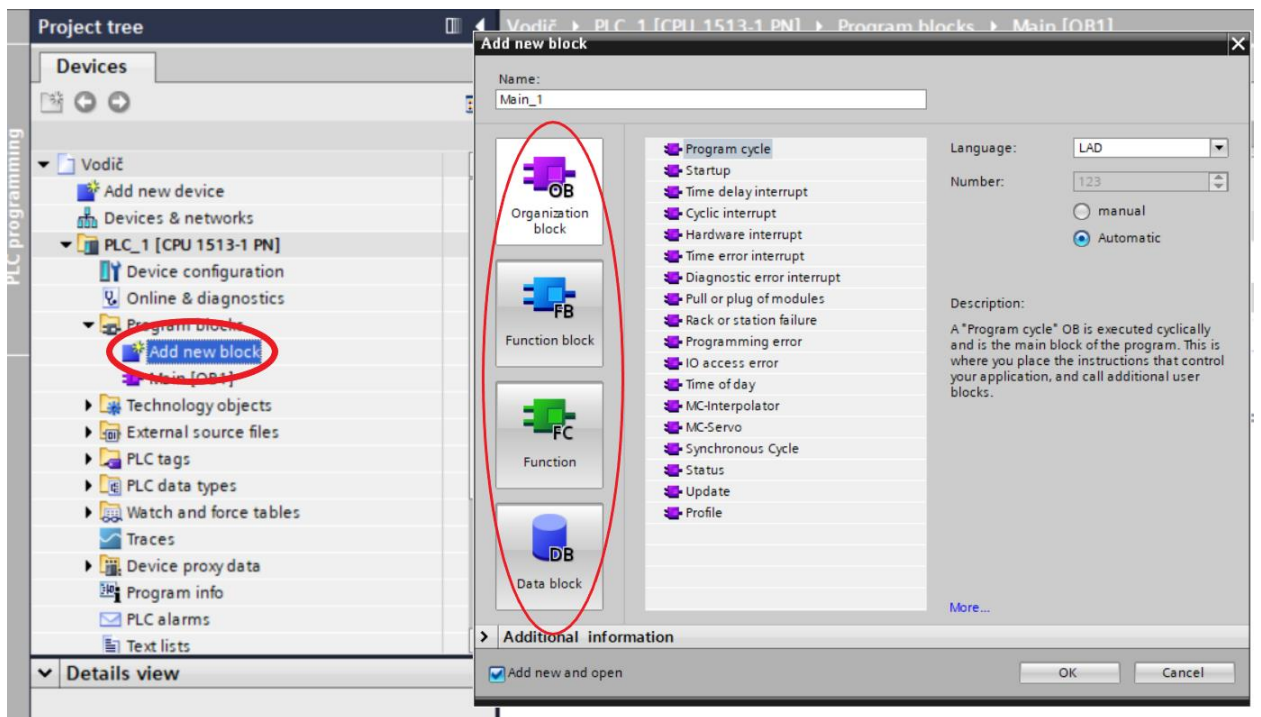


Slika 3.2. Prikaz konfiguracije analognih ulaza. Izvorno autor.

### 3.1. Razvoj upravljačke logike

Nakon podešavanja temeljnih komponenti PLC-a slijedi razvoj upravljačke logike. Klikom na „Program blocks, Add new block“ otvara se prozor s opcijama prikazanim na slici 3.3.





Slika 3.3. Prikaz programskih blokova. Izvorno autor.

Neki od organizacijskih blokova:

- Main[OB1] je glavni organizacijski blok, sve što je u njemu izrađeno će se izvršiti.
- Startup[OB100] je blok koji se se izvršava samo jednom i to prije početka našeg programa.
- Time delay interrupt[OB20] blok kojeg kada se pozove prekida glavni program na željeno vrijeme i izvršava razvijenu logiku.
- Cyclic interrupt[OB30] se koristi kod procesa za koji nam je bitno vrijeme izvršavanja. Main[OB1] obično ima vrijeme izvršavanja 1-150 ms, ovisno o kompleksnosti programa. No za neke procese nam je izuzetno bitno vrijeme izvršavanja (npr. striktno svakih 5ms), u tu svrhu koristimo Cyclic interrupt koji izvršava svoj program konstantno prema željenim parametrima. Redovito se koristi za PID kontrolere.
- Time error interupt[OB80] blok koji se implementira u svrhu nastavka programa. Izvršava se kad je program dužeg vremena izvšenja od zadanih

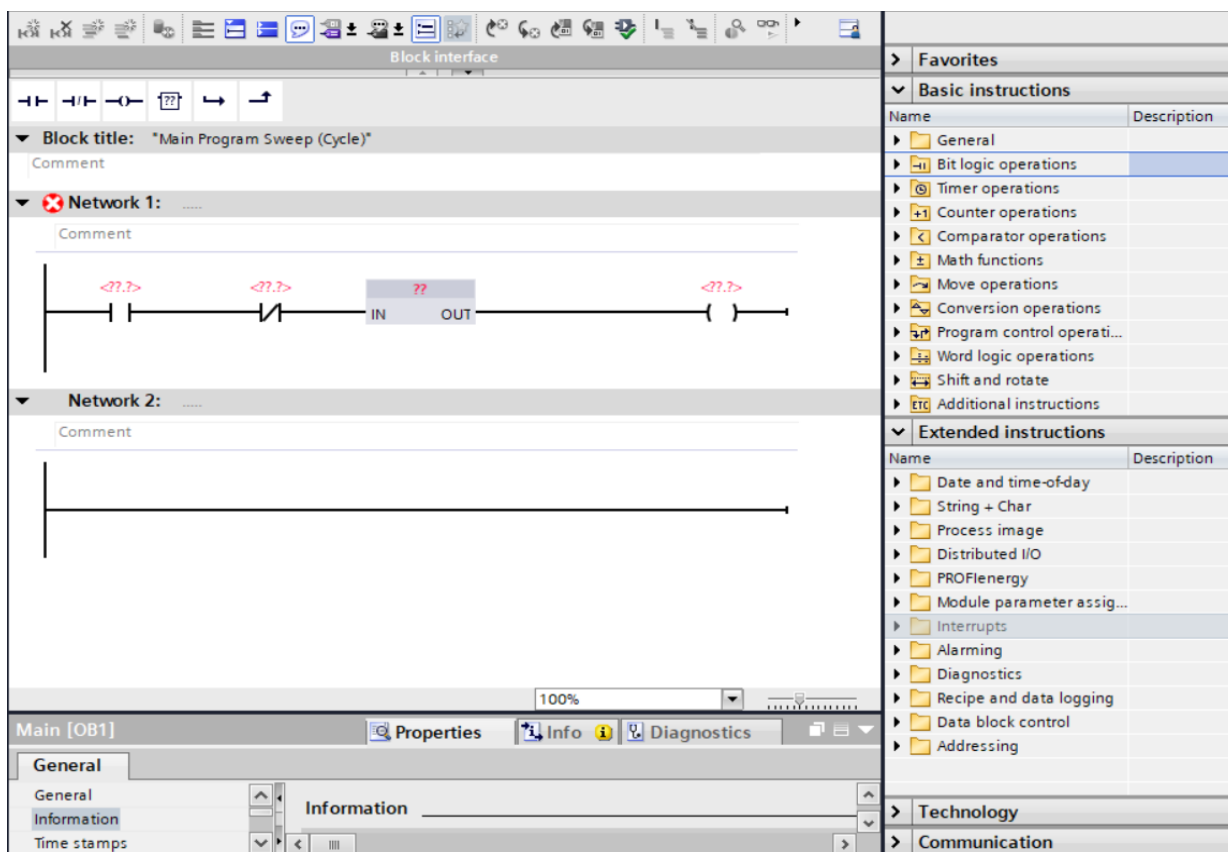
granica jednog kruga (obično 1-150ms) ili kada postoji beskonačna petlja. Bez ovog bloka PLC prilikom spomenutih radnji staje s programom.

Razlika između Function i Function Block je tu tome što Function Block ima spremljenu memoriju za svoje varijable. Nakon svakog ciklusa naredba Function gubi svoje vrijednosti pa je tako često korištena za računске operacije.

Data block su blokovi za spremanje željenih informacija.

TIA portal omogućuje razvijanje logike u četiri programska jezika. Ladder logic, Function block diagram, Statement list i Structured control. Ovaj završni rad bazira se na Ladder logic programskom jeziku.

Upravljačka logika razvija se pomoću mnogobrojnih instrukcija koje su u TIA-i podjeljene na osnovne i proširene instrukcije, zatim tehnološke objekte te komunikacijske naredbe.



Slika 3.4. Prikaz instrukcija. Izvorno autor.

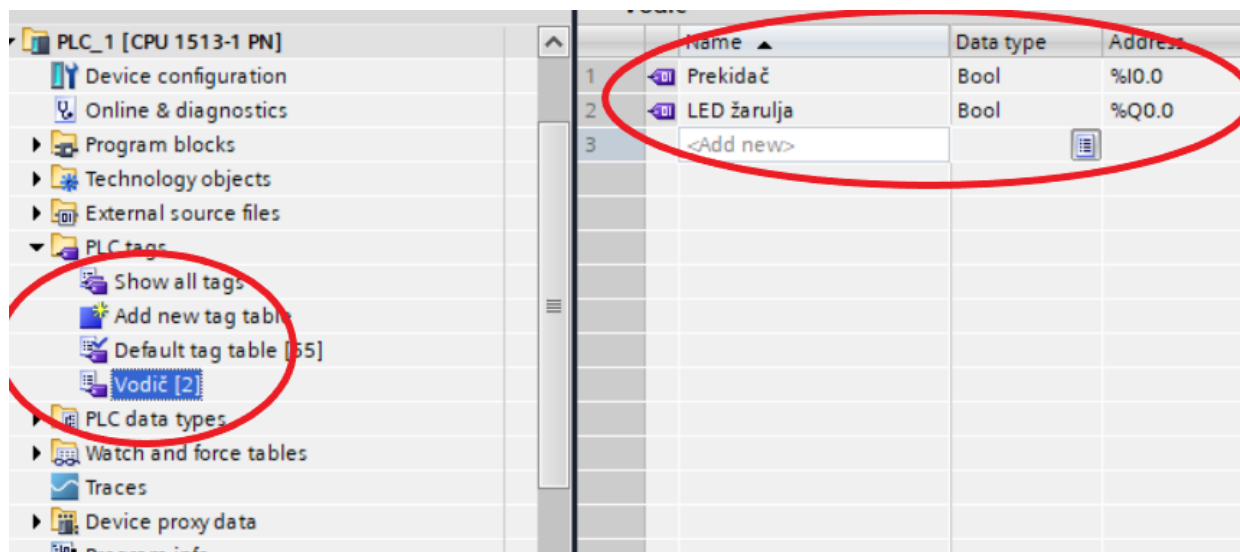
Slika 3.4. nam prikazuje lijevo prostor u kojem razvijamo ljestvičasti dijagram, a desno podjelu instrukcija.

Kategorija tehnološki objekti nudi kompleksna rješenja kao što su kontrola pokreta (koristi se za CNC strojeve, robotske ruke), funkciju PID regulatora (kontroler s povratnom vezom koji konstantno mjeri stanje) i brze brojače (brojanje impulsa).

### 3.2. Tags and data types

Svakoj varijabli moraju se odrediti odgovarajuća adresa i oblik informacije koju nose. Tako se razlikuju adrese inputa, memorije i outputa.

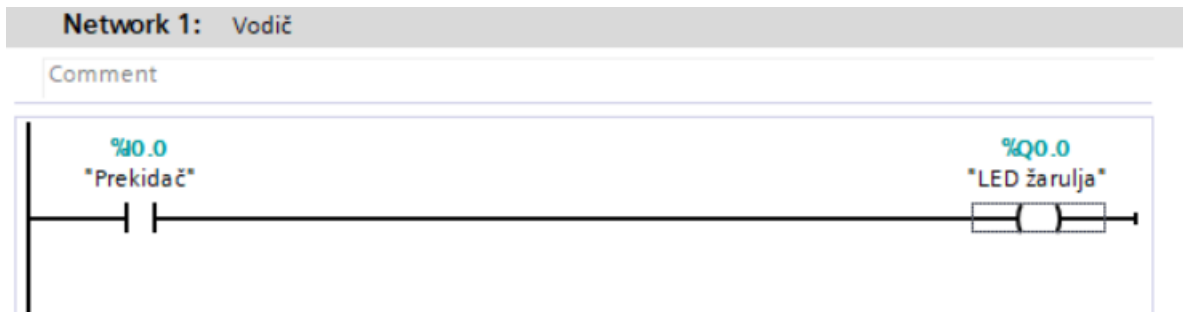
Kao što je rečeno varijablu opisuju adresa i oblik informacije. Primjerice fizički je spojen prekidač na mjesto 0.0 modula digitalnih inputa i LED žarulja na mjesto 0.0 digitalnog outputa. Da bi ih se opisalo i moglo precizno razvijati logiku navodi im se ime, adresa i tip informacije kao što je prikazano na slici 3.5.



Slika 3.5. Prikaz označavanja varijabli. Izvorno autor.

Adresa prekidača je I0.0 (Input na mjestu 0.0), a tip informacije Bool. LED žarulja ima adresu Q0.0 (Output na mjestu 0.0) i također tip Bool.

Slika prikazuje jednostavan primjer upravljačke logike spomenutih varijabli.



Slika 3.6. Prikaz upravljačke logike. Izvorno autor.

Dovođenjem logičke jedinice na I0.0 dobiva se logička jedinica na Q0.0. Oznaka adrese varijable govori na kojem bitu/byteu ta varijabla počinje.

Informacije mogu zauzimati:

- Bit (Boolean) – primjer adrese M0.0
- 8 bitova (Byte) – MB0
- 16 bitova (Word) – MW0
- 32 bita (Double Word) – MD0
- 64 bita (Long Word) – M0.0 – za 64 bitnu riječ Portal nema posebno adresiranu oznaku za memoriju već označavamo na kojem byte-u ta riječ počinje

Neke od često korištenih vrsta informacija u Portalu opisane su na slici 3.7.

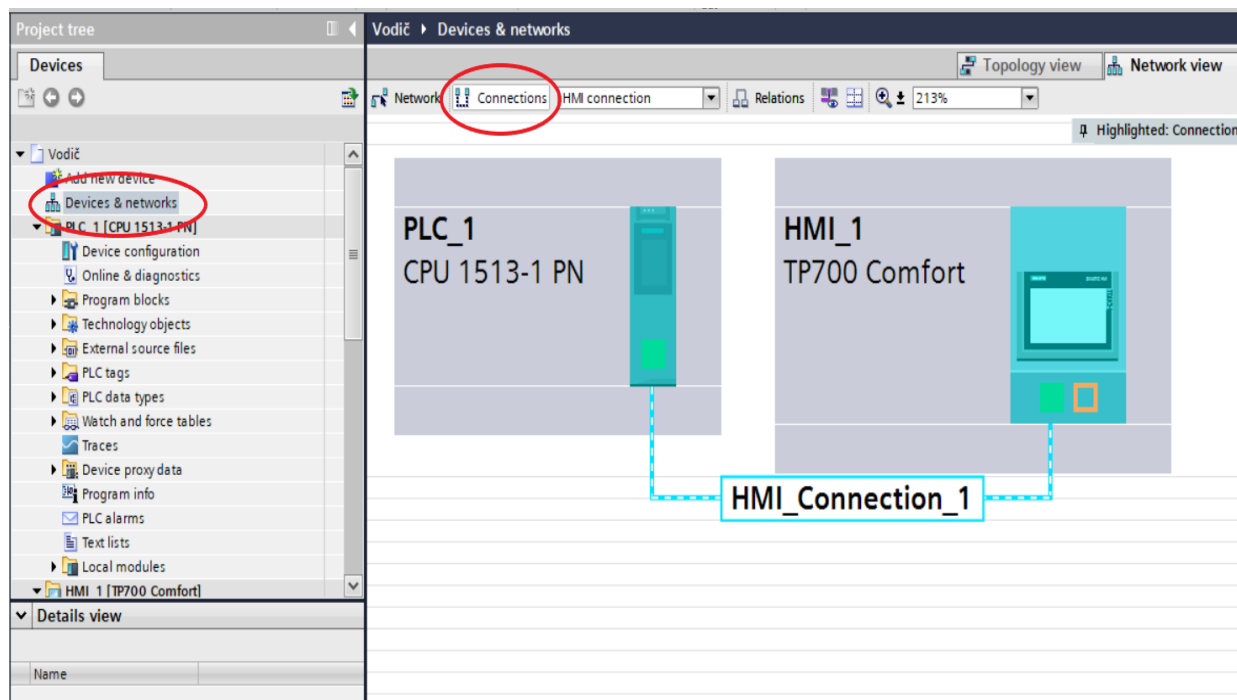
Vodič						
	Name ▲	Data type	Adresse	...	...	Comment
1	Bit	Bool	%M1.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	0, 1
2	Simbol (8 Bit)	Char	%MB2	<input type="checkbox"/>	<input checked="" type="checkbox"/>	A, B, 7 ...
3	Integer (16 Bit)	Int	%MW3	<input type="checkbox"/>	<input checked="" type="checkbox"/>	brojke npr. 1337, -1337
4	Vrijeme (32 Bit)	Time	%MD5	<input type="checkbox"/>	<input checked="" type="checkbox"/>	vrijeme u ms
5	Vrijeme i datum (64 Bit)	LTime_Of_Day	%M9.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	prikaz vremena i datuma
6	<Add new>			<input type="checkbox"/>	<input checked="" type="checkbox"/>	

Slika 3.7. Vrste informacija. Izvorno autor.

### 3.3. HMI

Klikom na „Add new device“, zatim HMI ikonu, bira se odgovarajući uređaj za korisničko sučelje.

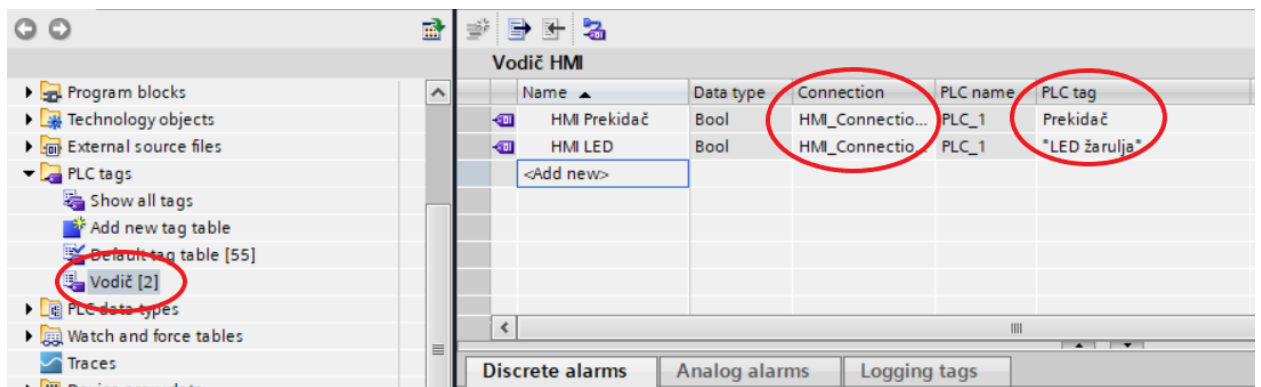
Nakon dodavanja HMI-a potrebno je povezati PLC i HMI kako bi se moglo komunicirati, a upute su prikazane na slici 3.8.



Slika 3.8. Povezivanje PLC i HMI. Izvorno autor.

Zatim se mogu napraviti oznake za HMI koje će se povezati s PLC oznakama.

U HMI listi oznaka daju se željena imena oznaka te se u stupcu Connection odabire poveznica s PLC uređajem. Zatim u stupcu PLC tag poveže se HMI oznaka s PLC oznakom kako je prikazano na slici 3.9.



Slika 3.9. Označavanje i povezivanje HMI varijabli. Izvorno autor.

Klikom na „Screens“ otvara se prozor za kreiranje sadržaja sučelja. Grafičkim objektima dizajniraju se vizualizirani procesi sustava.

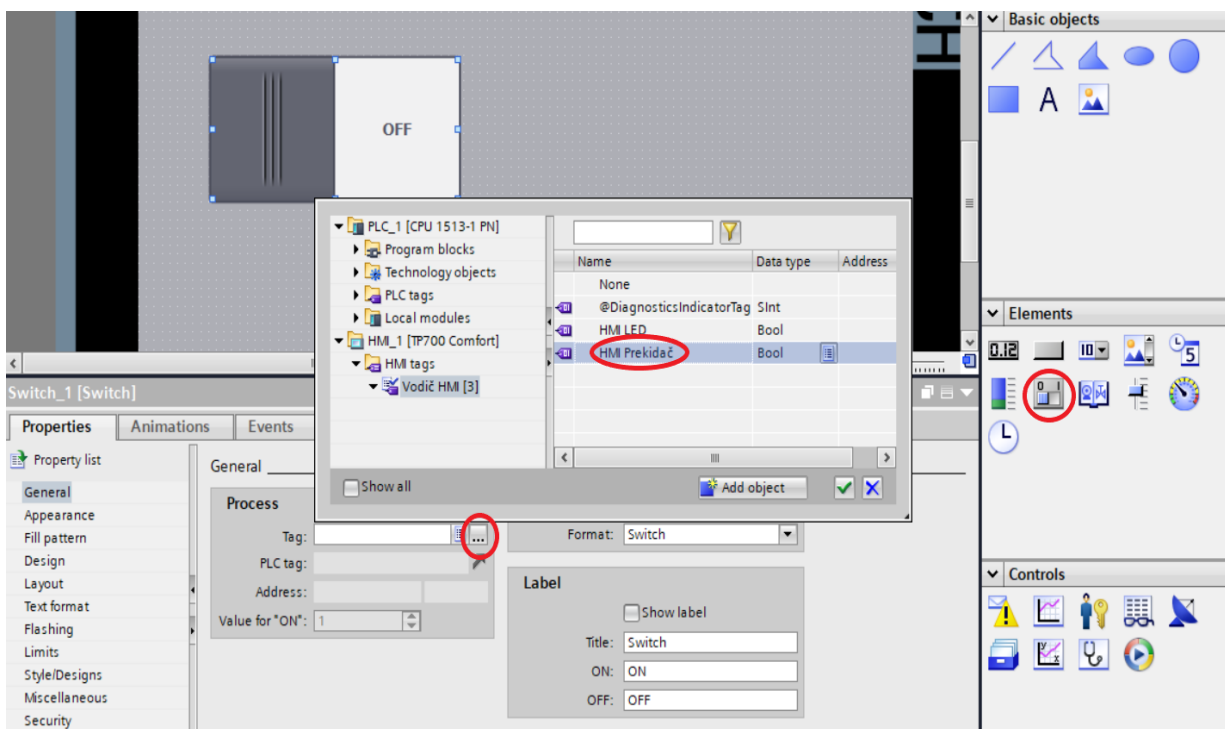
- Osnovni objekti su linije, geometrijski likovi, prostori za tekst i sl.
- Kontrolni objekti služe za prijavu korisnika, prikaz alarma, receptata ili dijagnostiku
- Elementi su objekti koji prikazuju stvarne uređaje kao što su pumpe, senzori, spremnici, pokretne trake, razne elektroničke uređaje itd. Slika 3.10. prikazuje grafički izbor pumpi.



Slika 3.10. Grafički prikaz pumpi. Izvorno autor.

Primjer upravljanja prekidača i žarulje preko HMI-a:

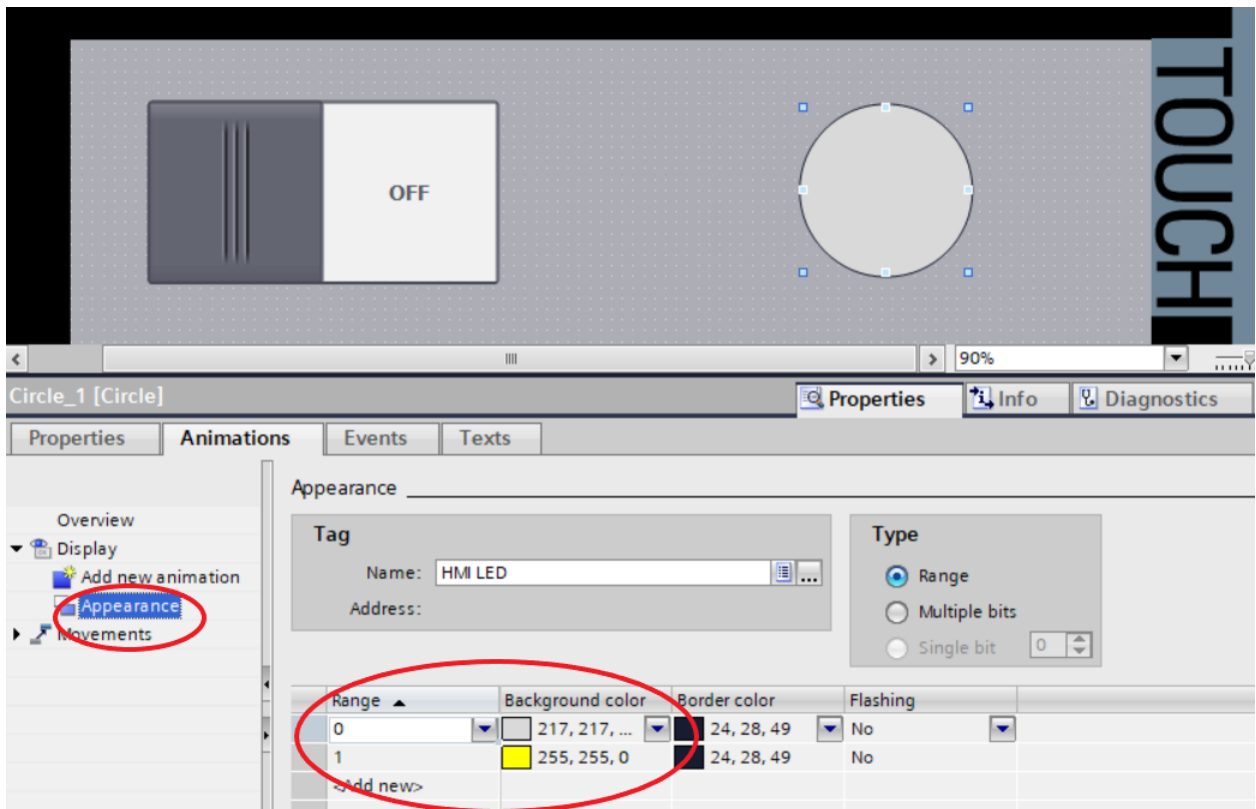
Odabire se „Switch“ element. U svojstvima mu se dodjeli oznaka HMI-a (HMI Prekidač) koja je prethodno spojena s oznakom PLC-a (Prekidač) kao što je prikazano na slici 3.11.



Slika 3.11. Izrada korisničkog sučelja. Izvorno autor.

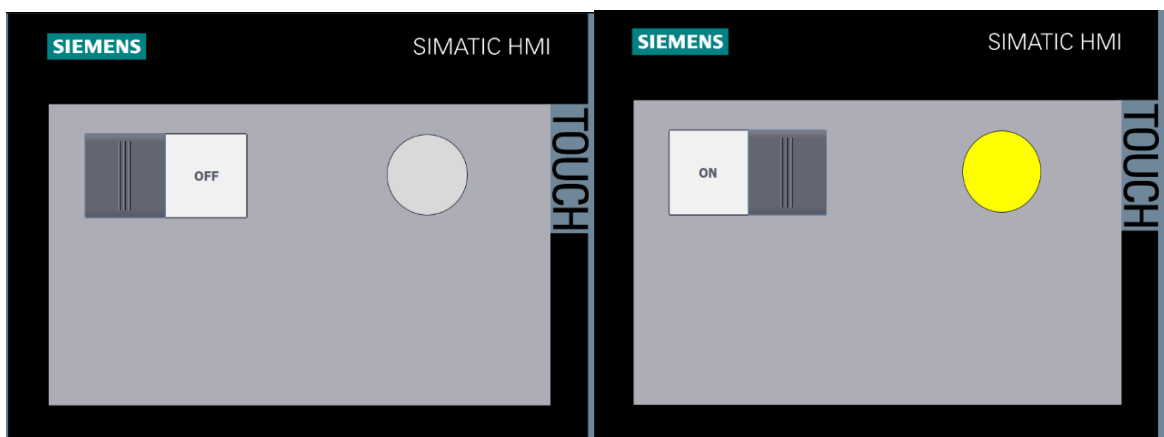
Osim označavanja mogu se mijenjati i izgled prekidača, teksta, dodijeliti se animacije, pokreti, alarmi i drugo.

Za LED žarulju koristit će se simbol kruga iz osnovnih objekata. Klikom na simbol otvara se opcija svojstva i animacije. Kako bi krug pokazao logičku jedinicu žarulje, potrebno je promijeniti boju kruga prilikom promjene iz logičke nule u jedinicu. To se postiže tako da se u opciji „Appearance“ zadaju parametri 0 i 1 kako je prikazano a slici 3.11.



Slika 3.12. Mijenjanje boje kruga s obzirom na bitove. Izvorno autor.

Simulacijom se provjerava ispravnost programa. Oponašanjem fizičke jedinice na ulazu inputa 0.0 krug koji predstavlja žarulju „zasvijetli“.



Slika 3.13. Simulacija programa. Izvorno autor.



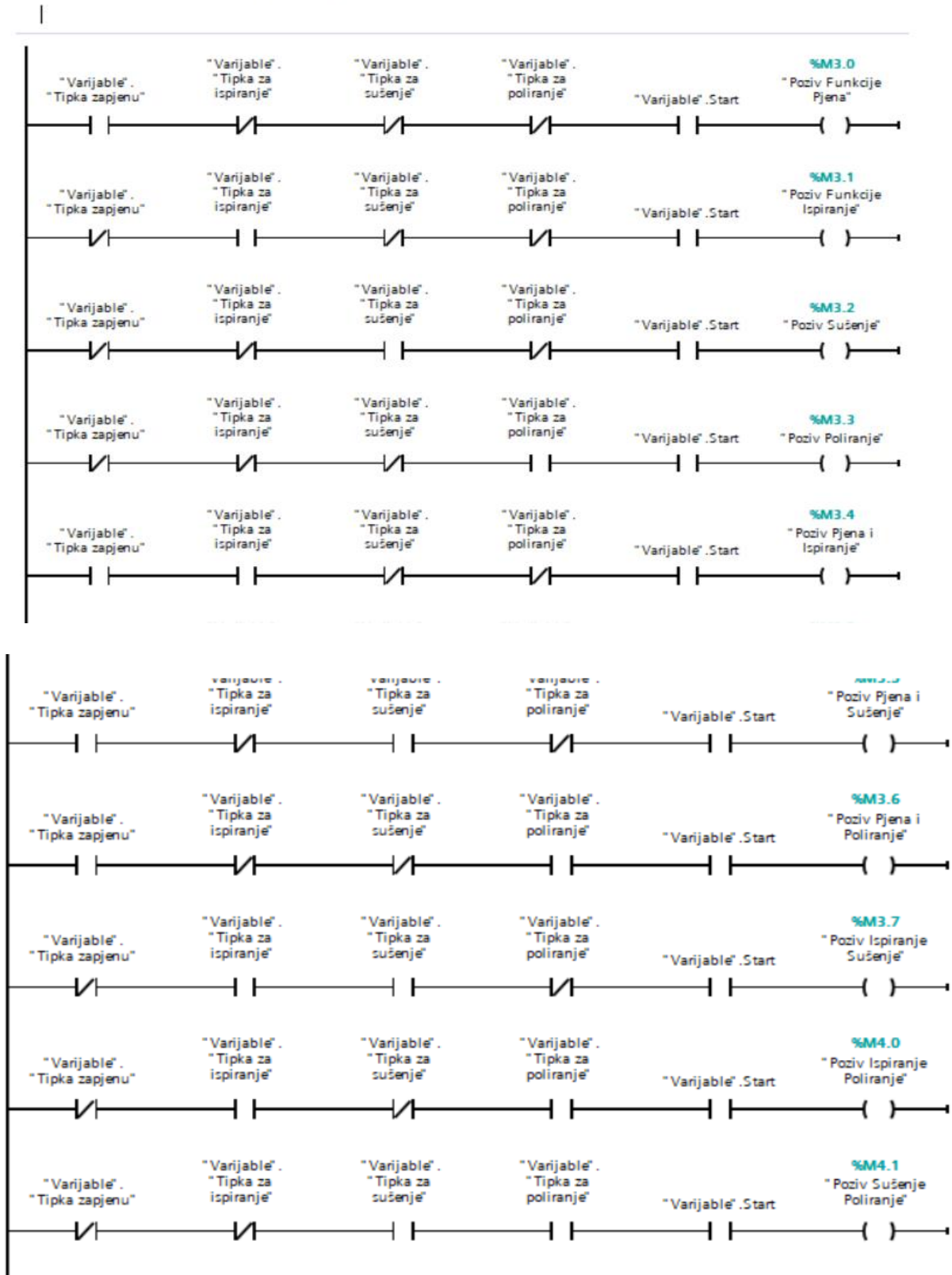
## **EKSPERIMENTALNI DIO**

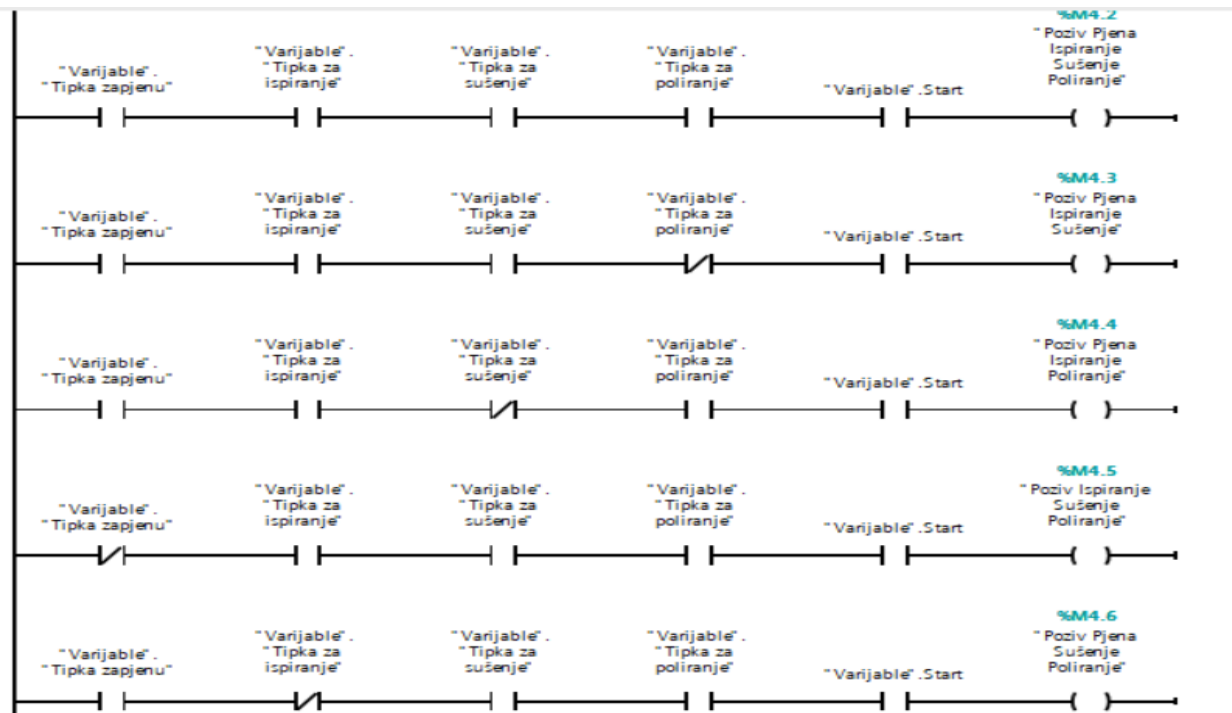
### **4. IZRADA SUSTAVA UPRAVLJANJA I NADZORA AUTOMATSKE AUTOPRANICE**

U četvrtom poglavlju razvijena je upravljačka logika automatske autopraonice i korisničko sučelje za upravljanje iste. Korišten je PLC S7-1500 simuliran S7-PLCSIM i HMI TP700 Comfort simuliran WinCC aplikacijom.

Program je zamišljen tako da operater u korisničkom sučelju može odabrati bilo koju kombinaciju od četiri usluge autopraonice (pjena, ispiranje, sušenje, poliranje). Nakon odabira usluga korisnik može pratiti tijek procesa na HMI-u. Određene kombinacije su isključene (npr. samo pjena ili pjena i sušenje) te korisnik dobiva poruku kako određena usluga nije moguća. Implementirana je i zaštita, pritiskom na STOP svi procesi staju te se pojavljuje nova tipka „Pokreni ponovno“. Mogućih je 15 kombinacija, tako je potrebno za svaku razviti poziv i izvršenje funkcije. Slika 4.1., 4.2.,4.3. prikazuje kompletni kod poziva svake funkcije:

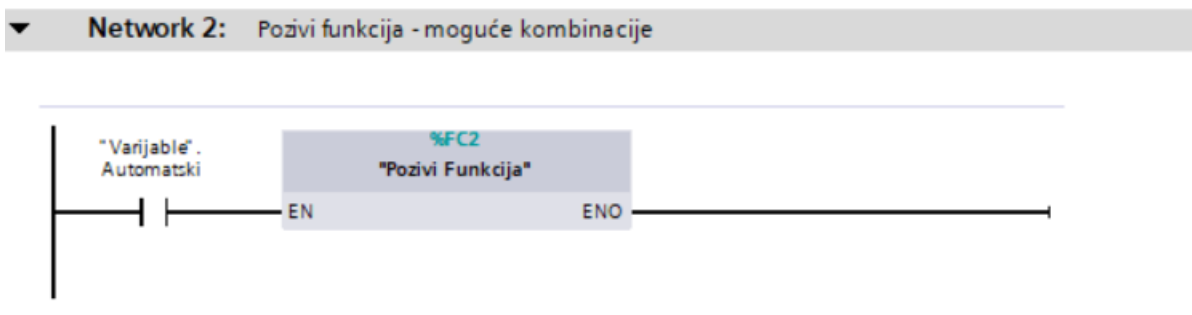
**Network 1: Pozivi funkcija - moguće kombinacije**





Slike 4.1., 4.2. 4.3. Prikaz cijelog koda mogućih kombinacija. Izvorno autor.

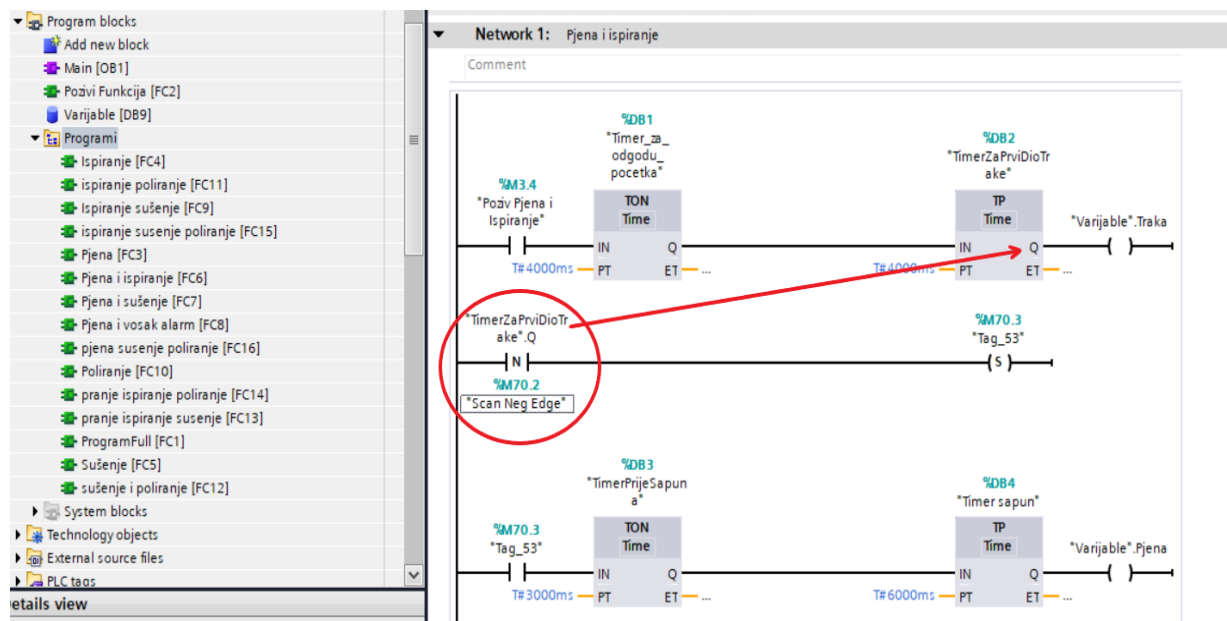
Cijeli ovaj kod odnosi se samo na pozive ostalih funkcija. Kako nam kompletan program ne bi bio nepregledan razvijamo kodove u zasebnim blokovima te ih pozovemo u glavnom bloku. Slika 4.4. prikazuje kako prethodni kod izgleda u glavnom bloku.



Slika 4.4. Prikaz koda u glavnom bloku. Izvorno autor.

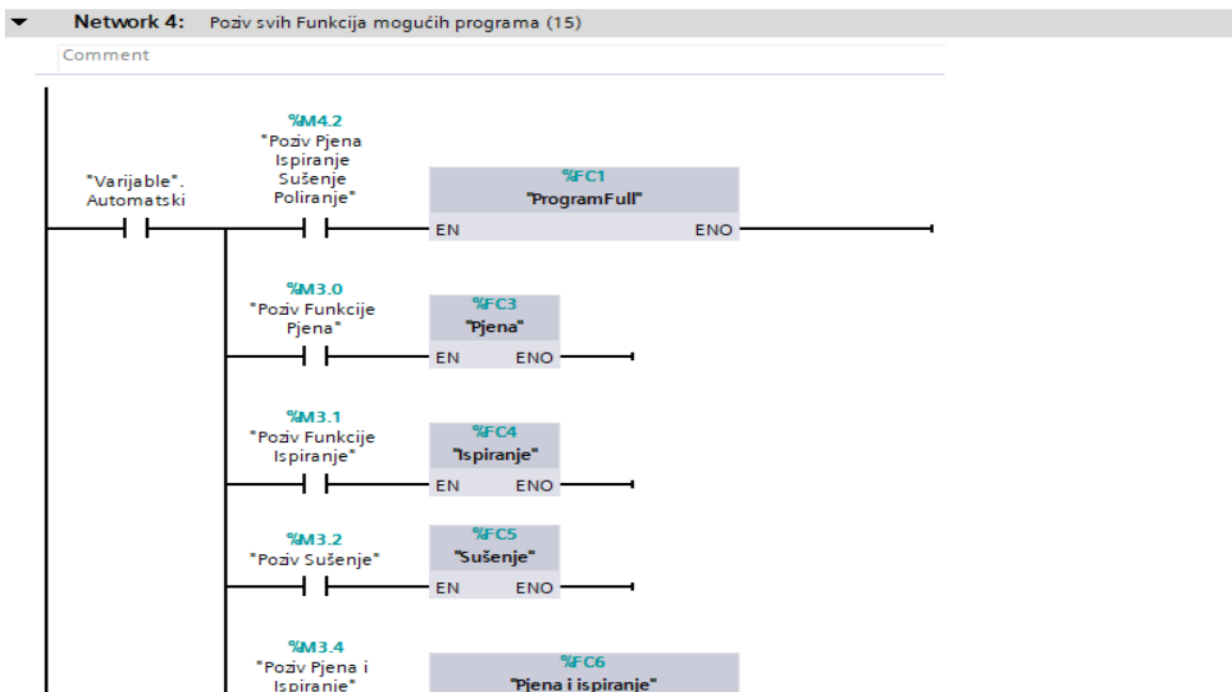
Nakon što je napravljen poziv za svaku od funkcija potrebno je i razviti logiku funkcija. Programi su slični, ali nisu isti tako da je potrebno razviti svih 15 programa zasebno. Slika prikazuje dio programa „Pjena i ispiranje“. Korišteni su TON i TP timer. Timer on Delay služi za odgodu puštanja logičke jedinice, a s Time Pulse timerom dajemo logičku jedinicu

u postavljenom vremenskom intervalu. Naredba „Scan Neg Edge“ daje pozitivan izlaz (samo na jedan impuls) kada odabrani operant prelazi iz stanja logičke jedinice u nulu (na slici 4.5. operant koji se prati je izlaz TP timera).



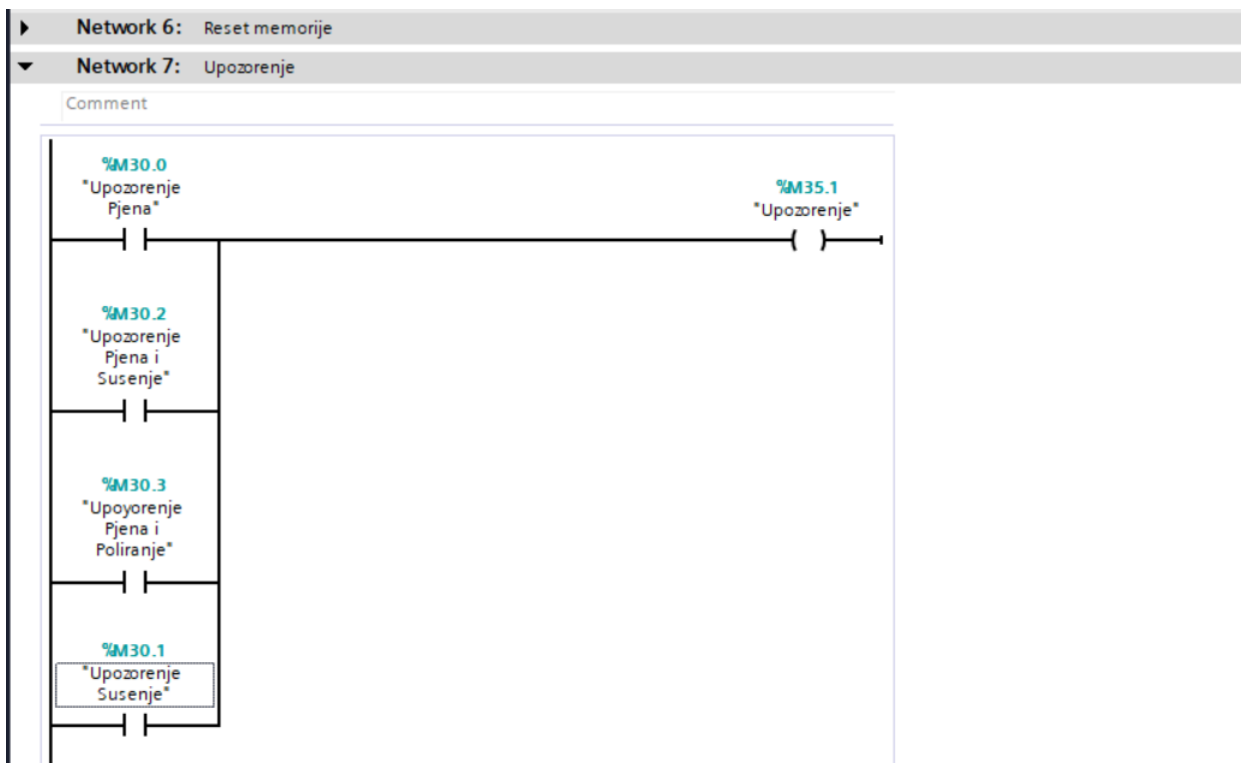
Slika 4.5. Dio logike za program „Pjena i ispiranje“. Izvorno autor.

Nakon izrade svih poziva za programe i sadržaja tih programa potrebno ih je povezati. To činimo u glavnom bloku kako je prikazano na slici:



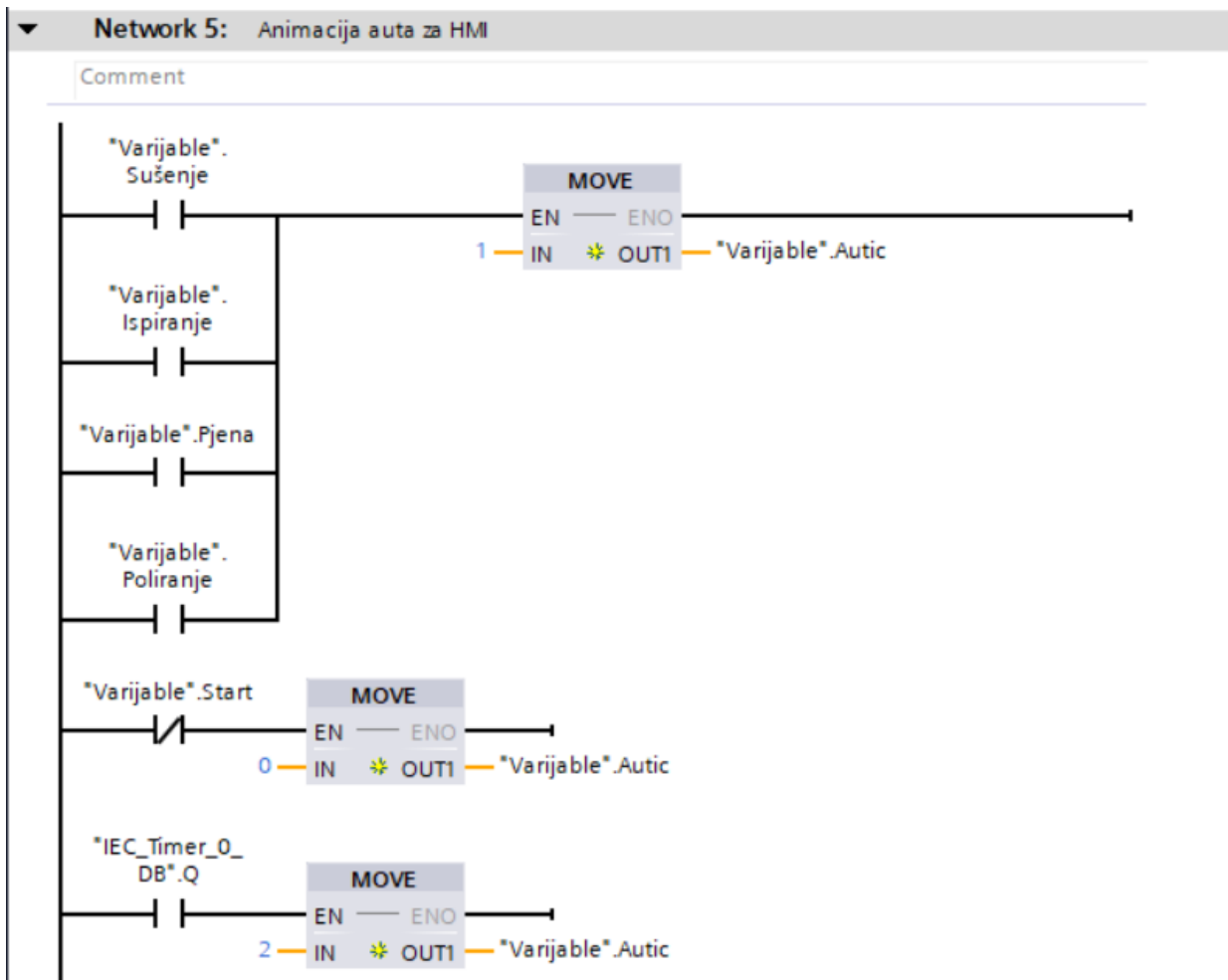
Slika 4.6. Poziv napravljenih programa. Izvorno autor.

Kombinacije koje nisu tako poželje imaju posebno razvijenu logiku. Korisnik dobiva upozorenje kako odabrana kombinacija nije moguća te se s potvrdom vraća na početni izbornik. Slika 4.7. prikazuje te kombinacije.



Slika 4.7. Programi koji nisu mogući. Izvorno autor.

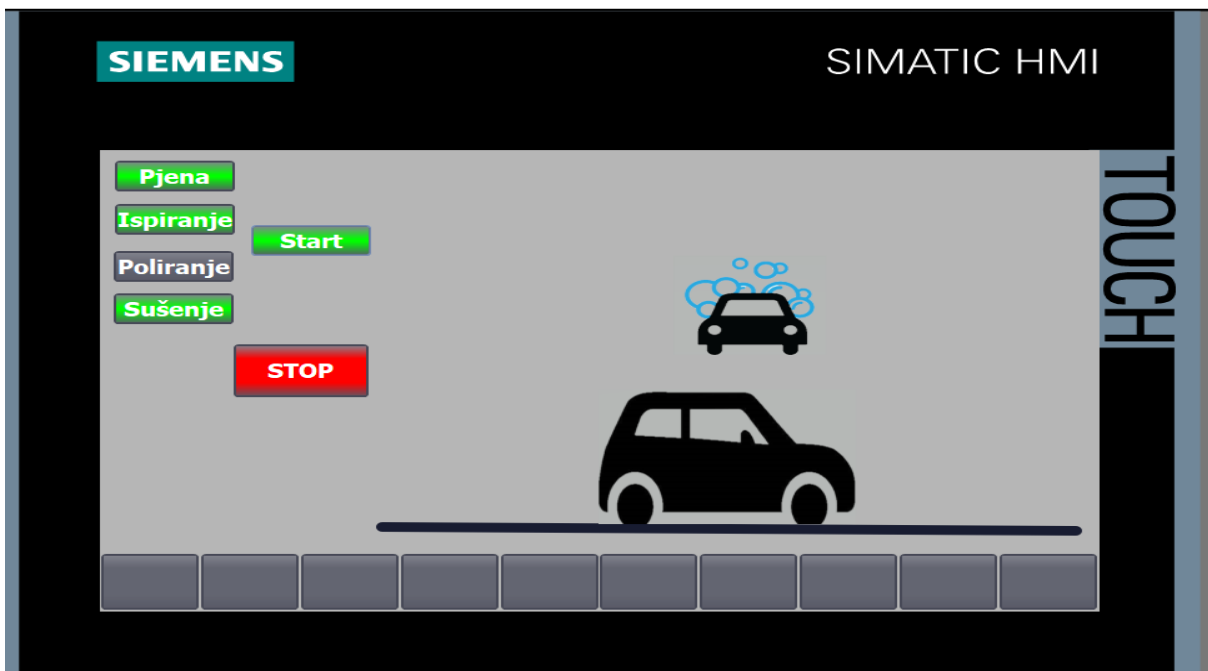
Korisničko sučelje sadrži niz tipki za upravljanje procesima i jednostavnu animaciju automobila koji se giba pokretnom trakom. Za animaciju automobila razvijen je poseban kod prikazan na slici 4.8.



Slika 4.8. Kod za animaciju automobila. Izvorno autor.

Portal za potrebe pokreta ikona koristi *integer* tip datoteke. Potrebno je napraviti posebnu memoriju u sustavu za potrebe pokreta automobila. Prikazani kod pokazuje prijenos određenog broja u napravljenu memoriju. Tako prije odabira kombinacije automobil se nalazi na mjestu 0, za vrijeme radnje aktuatora na poziciji 1, a prilikom završetka programa na poziciji 2.

Pomoću aplikacije WinCC simulira se i ispituje ispravnost projekta Slike 4.9. i 4.10



Slika 4.9. Simuliran program, trenutak kad je program pjene u funkciji. Izvorno autor.



Slika 4.10. Prozor koji upozorava korisnika kako određeni program nije moguć. Izvorno autor.



## 5. ZAKLJUČAK

Svrha ovog rada je pobliže opisati SCADA sustave te uputiti čitatelja u osnove programiranje sustava. SCADA sustavi su odgovorni za vizualizaciju, nadzor procesa, dijagnostiku i upravljanje s određenim dijelovima procesa. Primijenjeni su u gotovo svim složenijim industrijskim procesima današnjice kao što su elektroenergetska postrojenja, naftne platforme, nuklearne elektrane) itd. U industriji SCADA predstavlja viši, napredniji nivo od automatizacije. Sustavi služe kako bi reducirali potrebu za djelovanjem čovjeka te tako olakšavaju i ubrzavaju izvođenje i kontroliranje procesa. Bitno svojstvo ovih sustava je što računalo samostalno može pratiti procese te prilikom nepravilnosti obavijestiti pozivom ili porukom operatora gdje god se on nalazio. Bez ovakvih sustava složene industrijske djelatnosti ne bi bile moguće.

Vodič s generalnim uputama za programiranje sastavljen je tako da čitatelja upozna s osnovnim principima TIA Portal okruženja. Kao što su izbor komponenti, razvoj upravljačke logike te prikaz iste logike na korisničkom sučelju. U radu je opisano projektiranje i izvedba upravljačkog sustava automatske autopraonice. Tokom izrade projekta često se susrećemo sa velikim kodovima stoga nam projekt postaje nepregledan. Prednost TIA Portala je u tome što omogućuje izuzetnu organiziranost sustava, daje nam detaljan uvid u sve fizičke i programske komponente projekta. Osim programskog dijela, Portal omogućuje projektiranje i svih ostalih komponenti sustava, kao što su napajanje, odabir računalne stanice koja prati procese, kontroliranje rada izmjeničnog motora itd. Tijekom procesa projektiranja sustava stečena su znanja i iskustva iz područja automatike, sensorike i mehatronike.

## LITERATURA

1. B.R. Mehta, Y.J. Reddy B.R. Mehta, Y.J. Reddy, Industrial Process Automation Systems Design and Implementation, u: Waltham MA, 2015., 1.
2. Stuart A. Boyer, SCADA Supervisory Control and Data Acquisition 3rd Edition, u: Research Triangle Park NC, 2004., 9.
3. Nicoleta ARGHIRA, Daniela HOSSU, Ioana FĂGĂRĂȘAN, Sergiu Stelian ILIESCU, Daniel Răzvan COSTIANU: MODERN SCADA PHILOSOPHY IN POWER SYSTEM OPERATION 2011,  
[https://www.scientificbulletin.upb.ro/rev\\_docs/arhiva/full87719.pdf](https://www.scientificbulletin.upb.ro/rev_docs/arhiva/full87719.pdf) pristupljeno 1.3.2020. 156 str
4. Gordon Clarke, Deon Reynders, Edwin Wright: Practical Modern SCADA Protocols: DNP3, 60870.5 and Relayted Systems, Oxford, 2004. 6.  
[https://books.google.hr/books?id=ta5dSentTzoC&pg=PA19&lpg=PA19&dq=remote+terminal+unit+hrvatski&source=bl&ots=SG19j\\_3kYU&sig=ACfU3U1cG9wjSwjWr4hwWan05horCF\\_tpw&hl=hr&sa=X&ved=2ahUKEwicxdTz8v7nAhVFQEEAHZRXBDDQ6vAEwA#v=onepage&q&f=false](https://books.google.hr/books?id=ta5dSentTzoC&pg=PA19&lpg=PA19&dq=remote+terminal+unit+hrvatski&source=bl&ots=SG19j_3kYU&sig=ACfU3U1cG9wjSwjWr4hwWan05horCF_tpw&hl=hr&sa=X&ved=2ahUKEwicxdTz8v7nAhVFQEEAHZRXBDDQ6vAEwA#v=onepage&q&f=false)
5. Jeff Hieb Security hardened remote terminal units for SCADA networks, Louisville, Kentucky, 2008  
<https://ir.library.louisville.edu/cgi/viewcontent.cgi?article=1614&context=etd> , STR 9
6. Maršić, D., Malčić, G. & Vlašić, I. (2014) Izrada programskih komponenti u TIA Portal programskom okruženju. U: Petar Biljanović (ur.)Proceedings MIPRO 2014 - 37th International Convention.  
[https://bib.irb.hr/datoteka/711843.CE\\_TIA\\_Portal.pdf](https://bib.irb.hr/datoteka/711843.CE_TIA_Portal.pdf)
7. <https://industryeurope.com/abb-named-global-leader-in-scada-systems-for-the-power-secto/>
8. <http://www.wetcousa.com/clients.html>
9. <https://unindustry.my/wp-content/uploads/2018/09/Siemens-Simatic-S7-1500-Programming.jpg>

10. <https://myzone-kza3sadj.netdna-ssl.com/wp-content/uploads/2015/11/unistream.png>
11. [https://blog.copadata.at/wp/wp-content/uploads/2015/08/Alarm\\_Management\\_figure\\_04\\_EN.jpg](https://blog.copadata.at/wp/wp-content/uploads/2015/08/Alarm_Management_figure_04_EN.jpg)

## POPIS SLIKA

Redni broj	Naslov slike	Stranica
2.1.	Prikaz suvremenog SCADA sustava	2
2.2.	Prikaz prvotnih slanja signala u željeznici	4
2.3.	Struktura SCADA sustava	5
2.4.	RTU uređaj napajan solarno	7
2.5.	PLC S7-1500 s modulima	8
2.6.	HMI uređaj	9
2.7.	Hijerarhija alarma	10
3.1.	Odabir odgovarajućeg hardwarea	13
3.2.	Prikaz konfiguracije analognih ulaza	15
3.3.	Prikaz programskih blokova	16
3.4.	Prikaz instrukcija	17
3.5.	Prikaz označavanja varijabli	18
3.6.	Prikaz upravljačke logike	19
3.7.	Vrste informacija	19
3.8.	Povezivanje PLC i HMI	20
3.9.	Označavanje i povezivanje HMI varijabli	21
3.10.	Grafički prikaz pumpi	21
3.11.	Izrada korisničkog sučelja	22
3.12.	Mijenjanje boje kruga s obzirom na bitove	23

3.13.	Simulacija programa	23
4.1., 4.2. 4.3.	Prikaz cijelog koda mogućih kombinacija	24-25
4.4.	Prikaz koda u glavnom bloku	26
4.5.	Dio logike za program „Pjena i ispiranje“	27
4.6.	Poziv napravljenih programa	27
4.7.	Programi koji nisu mogući	28
4.8.	Kod za animaciju automobila	29
4.9.	Simuliran program, trenutak kad je program pjene u funkciji	30
4.10.	Prozor koji upozorava korisnika kako određeni program nije moguć	30