



**Факултет инжењерских наука
Универзитета у Крагујевцу**

Назив студијског програма: Машинско инжењерство

Ниво студија: Дипломске академске студије

Модул: Информатика у инжењерству

Предмет: Рачунарски подржано мерење и управљање

Број индекса: 42/2000

Иван М Вељовић

Дипломски рад

**Примена SIEMENS-овог TIA11 софтверског
пакета у програмирању PLC/HMI уређаја и
индустријских рачунарских система**

Комисија за преглед и одбрану:

1. Др Милан Матијевић – ментор

2. _____

3. _____

Датум одбране: _____

Оцена: _____

*Захваљујем се мом ментору проф.др.Милану Матијевићу
на избору занимљиве теме, и вођењу овог дипломског рада.*

*Захваљујем се дипл.инг.Марку Николићу и дипл.инг Срђану
Стојковићу на посвећеном времену и саветима који су ми помогли
да брже, и са мање тешкоћа, решим постављени задатак.*

Поставка дипломског (мастер) рада

Решава се управљање и надзор над два дистрибуирана процеса:

- 1) промена/одржавање нивоа течности у резервоару,
- 2) промена/одржавање брзине и смера окретања корачног мотора.

Оба објекта су повезана са два различита **PLC Siemens S7-1200**, док су они путем рачунарске мреже повезани са **НМИ** и персоналним рачунаром.

НМИ даје приказ и визуелизацију неких одабраних величина и стања у систему, затим, омогућава оператеру да изабере задату висину течности и задати смер и брзину окретања мотора.

Са персоналног рачунара се остварује контрола и надзор свих уређаја и процеса на нижим хијерархијским нивоима и упис у базу/фајл свих процесних величина од интереса са задатом периодом одабирања *T*. Направити Siemens **TIA11** софтверску апликацију која ће на централном персоналном рачунару реализовати приказ промене свих процесних величина од интереса (дијаграмски приказ, приказ индикатора, визуелизација промена величина и сл.), задавање жељених излазних величина (задати ниво течности у резервоару, задата брзина и смер обртања мотора) или омогућити директан утицај на управљачке величине у ручном режиму рада, док у аутоматском режиму рада треба остварити и могућност промене параметара регулатора (алгоритма управљања).

Кандидат треба да направи листу функционалности на основу које ће се после коментарисати и квалитет урађене апликације. Рад је фокусиран на софтверску апликацију коју треба урадити и објаснити. На располагању је опрема која је у лабораторијским условима већ повезана осим лабораторијског модела корачног мотора који није повезан са **PLC Siemens S7-1200**.

Форма рада треба да поштује упутства која су дата на веб страна Факултета инжењерских наука:

http://www.mfkg.rs/index.php?option=com_docman&Itemid=436 (ФИН Мастер рад форма), и

http://www.mfkg.rs/index.php?option=com_content&view=article&id=506&Itemid=355 (Општа упутства за писање рада (извештаја) са елементима математичко техничког правописа)

Препоручена литература:

1. Милан Матијевић, Горан Јакуповић, Јелена Цар, *Рачунарски подржано мерење и управљање*, универзитетски уџбеник, Машински факултет у Крагујевцу, 2008
2. Siemens AG, „*S7-1200_System_Manual_enUS*”, 2009.
3. Н. Матић, „*Увод у индустријске PLC контролере*“, Микроелектроника Београд, 2007.

Примена SIEMENS-овог TIA11 софтверског пакета у програмирању PLC/HMI уређаја и индустријских рачунарских система

Резиме рада

У овом раду треба решити управљање и надзор два дистрибуирана процеса. Један процес је управљање корачним мотором а у другом је управљање нивоом воде у резервоару. Два независна процеса биће контролисана са два SIEMENS-ова **PLC** контролера. Надзор и управљање се обавља преко **HMI KTP 600 PN Panel**-а и **PC Station**-а на PC рачунару. Задатак је да се реализује укључивање и искључивање мотора, промена смера окретања и брзине окретања, контрола нивоа воде у резервоару према жељеној вредности и ручна контрола вредности напона мотора. Биће реализован упис у базу/фајл промена процесних вредности из система.

Кључне речи: Програмабилни логички контролери, **HMI** интерфејси, индустријски рачунарске системи, управљање корачним мотором, **PID** управљање, **SCADA**

Application of SIEMENS TIA11 Software in PLC/HMI Devices and Industrial Computer Systems Programming

Abstract

In this work we shall solve the management and supervision of two distributed processes. One process is the control of a stepper motor and the second is to manage the water level in the tank. For two independent processes will be use two **SIEMENS's PLC** controllers. Control and monitoring is done through the **KTP HMI Panel 600 PN** and **PC-Station** on a PC. The task is to implement the motor on and off, change the direction of rotation and rotation speed, controls the water level in the tank to the desired value and manually control the value of the motor voltage. Registration will be implemented in the database / file changes of process value from the system.

Key words: **PLC** (Programmable Logic Controller), **HMI**, Industrial Computer Networks and Systems, Step Motor Control, **PID** Control, **SCADA**

САДРЖАЈ:

1. Уводна реч	6
2. Структура система и коришћена опрема.....	7
2.1 Структура система.....	7
2.2 Коришћена опрема.....	7
2.3 Лабораторијски модел спрегнутих резервоара.....	8
2.4 Појачавач снаге IPS 2M4SA/3.....	9
2.5 Лабораторијски модел корачног мотора. Корачни мотор.....	9
2.6 Siemens PLC S7-1200.....	11
2.6.1 Основна конфигурација I/O PLC SIMATIC S7-1200.....	12
2.7 SIMATIC HMI KTP 600 PN.....	17
2.8 Повезивање опреме.....	18
2.8.1 Повезивање PLC контролера и корачног мотора.....	18
2.8.2 Повезивање PLC контролера и резервоара.....	20
3. SIEMENS TIA11 софтверски пакет.....	22
3.1 Потребан хардвер.....	24
3.2 Прављење првог пројекта.....	24
3.3 Додавање нових уређаја.....	25
3.4 Додељивање IP адресе PLC-у.....	25
3.5 Extended download to device.....	26
3.6 Примена TIA11 у решавању инжењерских апликација. Поставка листе функције апликације.....	27
3.7 Примена TIA11 у програмирању PLC/HMI ради управљања брзине корачног мотора.....	29
3.7.1 Повезивање уређаја.....	29
3.7.2 Програмирање PLC-а	29
3.7.3 Програмирање HMI-а.....	35
3.8 Примена TIA11 у програмирању PLC/HMI ради управљања нивоа воде у резервоару.....	39
3.8.1 Повезивање уређаја.....	40
3.8.2 Програмирање PLC-а.....	40
3.8.3 Програмирање HMI-а.....	45
4. SCADA и једна њена TIA11 имплементација	49
4.1 Дефиниција SCADA система.....	49
4.2 TIA11 и пример имплементације SCADA апликације. PC Station.....	49
4.2.1 Креирање и подешавање.....	49
4.2.2 Израда HMI екрана на PC Station-у.....	51
4.2.3 Параметри PID регулатора.....	56
4.2.4 Прикупљање и меморисање података.....	57
5. Закључак.....	59
6. Додатци.....	60

6.1 Целокупан LAD дијаграм PLC-а за контролу мотора.....	60
6.2 Целокупан LAD дијаграм PLC-а за контролу резервоара.....	62
6.2.1 PID Рутина.....	62
6.3 Изглед главног HMI екрана.....	63
6.4 Изглед HMI екрана за контролу мотора.....	63
6.5 Изглед HMI екрана за аутоматску контролу резервоара.....	64
6.6 Изглед HMI екрана за ручну контролу резервоара.....	64
6.7 Изглед RunTime екрана на PC Staion-у.....	65
6.8 Меморијска мапа PLC-а.....	66
6.9 Техничке карактеристике PLC-а.....	67
7. Литература.....	70

1. Уводна реч

Предмет овог дипломског рада је примена програмабиле **SIEMENS** -ове опреме и софтвера за аутоматизацију, управљање и надзор система. Развојем информационих технологија и програмабилне опреме се постижу врхунски резултати при изградњи данашњих индустријских система. Аутоматизацијом система се постиже већа продуктивност и енергетска ефикасност система, као и бољи квалитет производа. Грешке настале људским фактором се смањују на минималну меру а тиме се постиже мања производна цена и потискује се конкуренција на тржишту, што је изузетно важна чињеница у савременом пословању.

Циљ рада је да се презентује повезивање и програмирања опреме, и употреба **SIEMENS-овог TIA11 софтверског пакета у програмирању PLC/HMI уређаја и индустријских рачунарских система**. Практично ће на примеру одабране апликације бити демонстрирана примена опреме и софтвера за надзор и управљање дистрибуираног система.

У овом раду, решава се управљање и надзор два дистрибуирана процеса: 1) промена/одржавање нивоа течности у резервоару, 2) промена/одржавање брзине и смера окретања корачног мотора. Оба објекта су повезана са два различита **PLC Siemens S7-1200**, док су они путем рачунарске мреже повезани са **HMI (Touch Screen Siemens KT600 Basic Color PN)** и персоналним рачунаром. **HMI** даје приказ и визуелизацију неких одабраних величина и стања у систему, затим, омогућава оператеру да изабере задату висину течности, и задати смер и брзину окретања мотора. Са персоналног рачунара је могуће контролисати све уређаје на нижим хијерархијским нивоима (**HMI** и оба **PLC**-а), уписивати у базу/фајл све процесне величине од интереса, имати приказ промене свих процесних величина од интереса (дијаграмски приказ, приказ индикатора, визуелизација промена величина и сл.), задавати жељене излазне величине и мењати параметре регулатора.

2. Структура система и коришћена опрема

2.1 Структура система



Слика 1. Структура система

На слици 1 је уопштено приказана структура система која је предмет овог рада. Систем се састоји од централног рачунара, два **PLC** контролера и **HMI** уређаја. Један контролер се користи за управљање корачним мотором а други за пријем вредности сензора и покретање мотора (пумпе за воду), при чему се регулише ниво течности у резервоару (било у отвореној или затвореној повратној спрези).

2.2. Коришћена опрема

Коришћену опрему чине: лабораторијски модел спрегнутих резервоара, појачавач снаге **IPS 2M4SA/3**, лабораторијски модел корачног мотора (тј. корачни мотор са напајањем и интерфејсом), **Siemens PLC S7-1200**, **HMI - Touch Screen Siemens KT600 Basic PN**, и персонални рачунар **PC Pentium 4, 1.7GHz,XP (SP3), 4GB RAM, 20GB HDD** слободног простора .

2.3. Лабораторијски модел спрегнутих резервоара

У циљу управљања нивоом течности у једном резервоару, коришћен је лабораторијски модел 4 спрегнута резервоара који је описан у [1]. Ради се о производу фирме “*Majk Elektronik*” из Младеновца, који изгледа као на слици 2.



Слика 2 Модел резервоара

Лабораторијски модел се састоји од четири издужена резервоара унутрашњих димензија 40x40x300mm окачена изнад сабирног резервоара димензија 14x37x10cm који може да прихвати запремину од 5L течности (служи за скупљање воде). Из сабирног резервоара се вода пумпа са две независне пумпе центрифугалног типа улазног једносмерног напона у опсегу од 0V до 12V. Сваки од резервоара на дну има отвор. Вода из горњих резервоара отиче у резервоар на доњем нивоу, а из доњих резервоара вода отиче у сабирни резервоар. На тај начин, резервоари су међусобно спрегнути, а пумпе зависно од позиције ручних вентила пумпају воду у сва 4 резервоара са различитим протоком. У специјалном случају (позиције вентиле и употребе пумпи), који се овде користио, једна пумпа убацује воду у само један резервоар из кога вода отиче у сабирни резервоар [1].

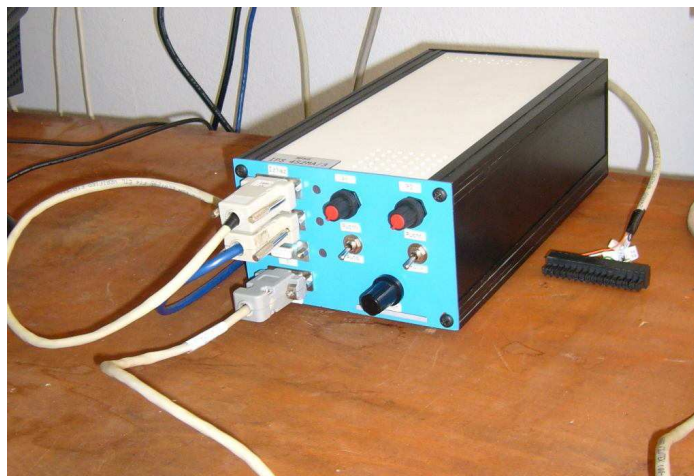
Улазни напон једносмерни пумпе је у опсегу од 0-12 V и као резултат даје опсег протока 0-20 cm³/s (улазно излазни опсег пумпе је део њене статичке карактеристике). Унутрашње димензије резервоара су 40x40x300 mm. Давач нивоа у сваком од резервоара је давач хидростатичког притиска у резервоару који на свом излазу даје напонски сигнал у опсегу од 1-5 V. По спецификацији се ради о мерном претварачу линеарног типа коефицијента појачања 0.1 V/cm [1].

Можемо рећи да је улазни сигнал опсега 0-12 V, а излазни 1-5 V (што одговара опсегу нивоа течности у резервоару од 0-30 cm). Улазни или управљачки сигнал може долазити са D/A интефејса који продукује аналогни напонски сигнал

различитог опсега 0-5 V (попут NI USB 6008 или 6009 картице) или 0-10 V (попут Siemens PLC S7-1200 D/A излаза). Отуда се у лабораторији користи и посебно пројектовани уређај **IPS 2M4SA/3** чија је улога да сигнал са дигитално аналогног конвертора рачунарског подсистема опсега 0-5 V или 0-10 V конвертује у опсег 0-12 V који најбоље одговара карактеристикама пумпе.

2.4. Појачавач снаге IPS 2M4SA/3

IPS 2M4SA/3 је појачавач снаге и електрични прекидач који је посебно пројектован за потребе експерименталних истраживања у Центру за примењену аутоматику, детаљно је описан у [2], и приказан је на слици 3.

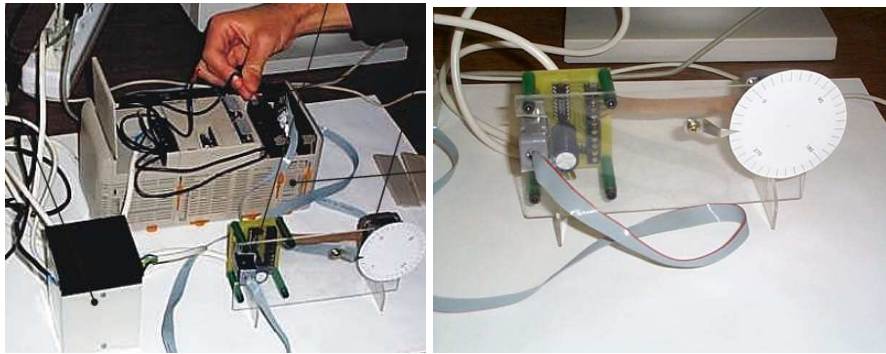


Слика 3. *IPS 2M4SA/3 (има улогу појачавача снаге и мултиплексера)*

Овај уређај има могућност хардверског преласка на ручно управљање, пребацивањем положаја прекидача из аутоматског у ручни мод, и тада се аналогни излаз (ка пумпи) мења потенциометром.

2.5. Лабораторијски модел корачног мотора. Корачни мотор.

Користи се лабораторијски модел корачног мотора који се састоји из трансформатора, појачавача снаге, корачног мотора, и интерфејса за комуникацију са управљачким уређајем који шаље управљачке сигнале. Лабораторијски модел је направљен у оквиру семинарског рада [3], у коме је описана и његова употреба и принципи управљања корачним мотором.



Слика 4. Лабораторијски модел корачног мотора повезан са PLC-ом (у случају слике PLC Omron SYSMAC са процесором CQM1 CPU44)

Приказана експериментална опрема са слике 4 обухвата:

- трансформатор – трансформише наизменични напон 220V у напон потребан за напајање појачавача снаге
- појачавач снаге – појачава управљачке импулсе тако да могу да покрећу корачни мотор
- корачни мотор – којим се управља
- PLC или неки други управљачки уређај који шаље управљачке секвенце

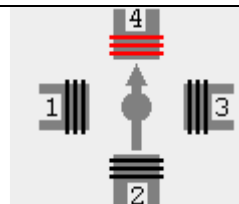
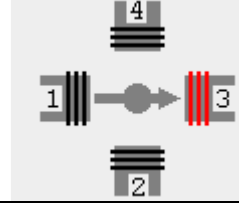
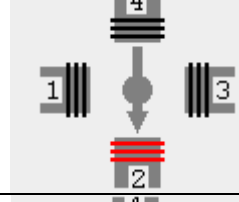
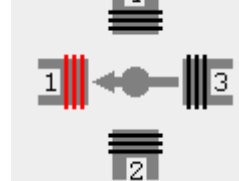
Генерално, корачни мотори су електромеханички претварачи енергије, који импулсну електричну побуду претварају у корачни механички помак. Израђују се као ротацијски и транслацијски (преовлађују ротацијски). На малим ротацијским брзинама ротор се зауставља на сваком корачном положају. На средњим брзинама нема заустављања ротора на сваком корачном положају али угаона брзина осцилира зависно од положаја. Што се корачна брзина више повећава, осцилације угаоне брзине постају све мање, тако да на великим корачним брзинама угаона брзина тежи константној брзини. Појам „велика корачна брзина“ је релативан, зависи од конструкције, код комерцијалних мотора се креће од 102 до 104 корака у секунди [к/с]. Корачни мотор је електрични мотор без комутатора. Сви намотаји су смештени на статору а ротор је перманентни магнет.

Корачни мотори се деле у две групе и то: униполарне и биполарне корачне моторе. Код оба типа мотора један доведени импулс узрокује померање за један корак чија величина у степенима зависи од карактеристика (резолације) примењеног корачног мотора.

угао за 1 корак	број корака за 1 обрт или 360°
0.9	400
1.8	200
3.6	100
7.5	48
15	24

Табела 1. Најчешће вредности које карактеришу резолуцију корачног мотора

Мотор који имамо је корачни мотор са четири намотаја. У монофазном моду за сваки корак доводи се напон само на један намотај што је дато у следећој табели. Та секвенца се понавља онолико пута колико желимо да имамо корака. Будући да се напаја само један намотај то је и мали окретни момент мотора. Поред оваквог начина, могуће је истовремено довести напон на два намотаја. На тај начин би се ротор мотора померио за половину угла, односно дошао у положај између два пола, и на тај начин би начинио међукорак. У овом конкретном раду ћемо користити само положаје када се на један намотај доводи напон и на тај начин ће се мотор окретати увек за исти угао, односно за по један корак. У следећој табели је дат графички приказ начина на који се ротор окреће у зависности од промене напона на половима.

Korak	Namotaj 4	Namotaj 3	Namotaj 2	Namotaj 1	
a.1	1	0	0	0	
a.2	0	1	0	0	
a.3	0	0	1	0	
a.4	0	0	0	1	

Табела 2. Илустрација принципа управљања корачног мотора

2.6. Siemens PLC S7-1200

PLC (Programmable Logic Controller) – програмабилни логички контролер или индустријски контролер је дигитални електронски уређај који поседује програмабилну меморију за смештање инструкција којима се реализују специфичне функције, као што су логичке операције и секвенцирање, тајминг, бројање и аритметичке операције, у циљу управљања различитим машинама и процесима путем дигиталних или аналогних улазно/излазних (I/O) модула. Индустријски контролер (**PLC**) се може дефинисати и као индустријски рачунар чију су хардвер и софтвер посебно прилагођени раду у индустријским условима,

а који се може лако програмирати и уграђивати у нове и постојеће индустријске системе.

Једно од нових технолошко достигнућа **SIEMENS**-а јесте **SIMATIC S7-1200 PLC**, који је заснован на модуларном концепту и лако се прилагођава свим једноставним и оним мало захтевнијим апликацијама. **SIMATIC S7-1200** је **PLC** који је, како сам произвођач каже, безбедна инвестиција, контролер који је модуларан и компактан, разноврстан, па се самим тим савршено уклапа у мноштво разних апликација. Скалабилан и флексибилан дизајн, комуникацијски интерфејсе који употпуњује највише стандарде у индустријској комуникацији, као и мноштво напредних функција чине овај контролер саставним делом комплетог и свеобухватног управљачког система [4].

Код новог **SIMATIC**-а (**PLC S7-1200**), **SIEMENS** је развио посебно добру интеграцију између **PLC**-а, **HMI**-а и софтвера. Резултат тога је флексибилност и скалабилност **S7-1200 PLC**-а, са којим се могу решавати и неки од најзахтевнијих процеса на машинама и производним линијама. За **SIMATIC S7-1200 PLC** и **HMI** постоји један софтвер за једноставно и брзо програмирање, умрежавање и конфигурација, **STEP7v10**. **SIMATIC S7-1200 PLC CPU** јединице подржавају мноштво различитих улазно/излазних модула и комуникацијских картица. **S7-1200 CPU** се састоји од интегрираног **PROFINET** интрфејса чиме се постиже једноставно умрежавање између **PC**-софтвера, контролера и **HMI**-а [4].

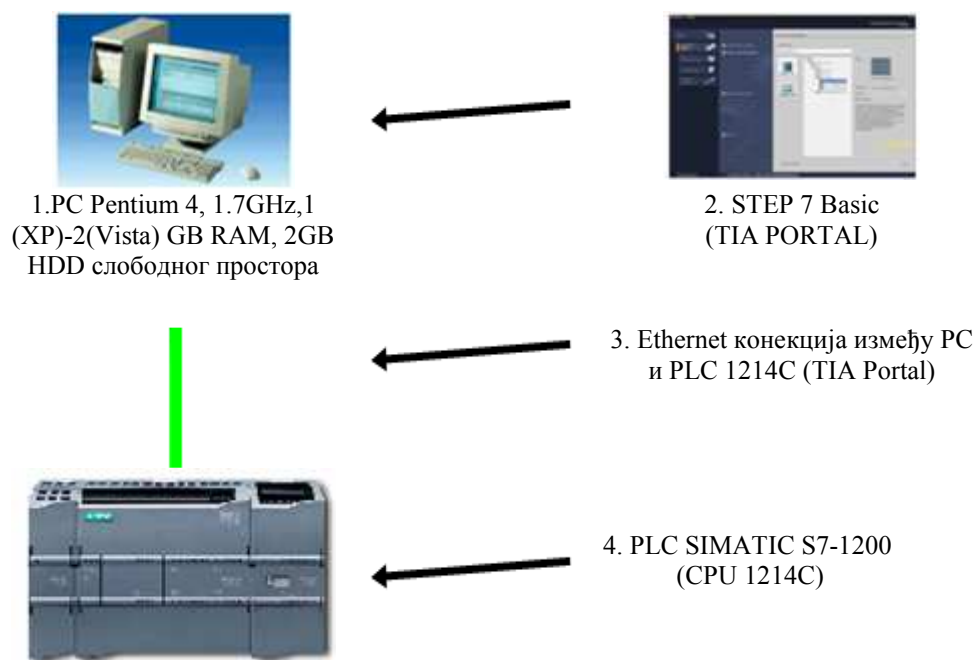


Слика 5 *PLC S7-1200*

- Једноставан и ефикасан софтвер
- Скалабилан и флексибилан дизајн
- Интегрисани **PROFINET** интерфејс
- Једноставно умрежавање
- Комуникација са екстерним уређајима
- Интегрисане многе потраживане технологије
- 100kHz улази за бројање и мерење
- 100kHz излази за позиционирање, регулацију брзине и цикличке операције
- **PLC**- готови функцијски блокови за управљање кретањем
- **PID** инструкције у затвореној петљи
- Максимум 284 I/O

SIMATIC S7-1200 може да садржи неку од три различите централне процесорске јединице **CPU 1211C**, **CPU 1212C** и **CPU 1214C**. У најкомплекснијој конфигурацији коју обезбеђује процесорска јединица **CPU 1214C** може се надоградити 8 сигналних модула и три комуникациона модула, чиме се добија конфигурација са укупно максималних 284 дигитална улаза/излаза (I/O), као и са укупно максималним 51-ним аналогним улазом/излазом (I/O). Оно што представља новину код архитектуре **SIMATIC S7-1200** је мали хардверски додатак под називом сигнал боард, помоћу кога се на једноставан и брз начин било која процесорска јединица може проширити мањим бројем дигиталних и аналогних улаза и излаза [4].

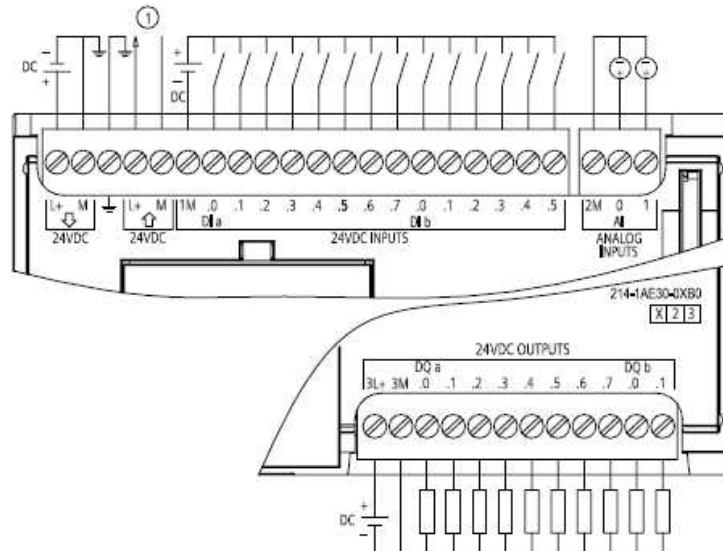
При имплементацији, софтвер **SIMATIC STEP 7 Basic** одао је нове „тајне“ контролера **S7-1200**. Приступ програмирању својствен серијама **S7-300** и **S7-400** у потпуности је задовољио потребе објектно оријентисаног кода. У односу на претходника, серију **S7-200**, меморија је организована по принципу дата блокова што омогућује додатну флексибилност у параметризацији склопова који се понављају [4]. Принципи програмирања су објашњени у [4].



Слика 6 Минимални хардверски и софтверски захтеви за програмирање PLC S7-1200

2.6.1. Основна конфигурација I/O PLC SIMATIC S7-1200

Основна конфигурација улаза и излаза SIEMENS-овог PLC SIMATIC S7-1200 је дата на слици 7.



Слика 7а. Шема повезивања улаза и излаза

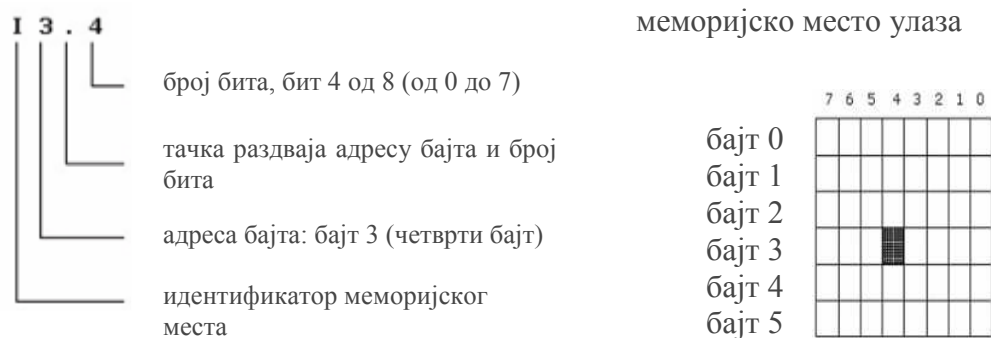
На врху слике, са леве стране се могу видети шематски прикази за напајање контролера. На његове напонске пинове је потребно довести 24V једносмерне струје (20.4 до 28.8 VDC је толеранција напајања PLCа према техничкој документацији). Десно од њих се налазе пинови дигиталних улаза. Дигитални улаз по наслеђеној „PLC“ терминологији може имати вредности 0 или 1 које се уписују у бит улазног регистра. Сваки бит улазног регистра је повезан са одговарајућим физичким дигиталним улазом и ради се о томе да ли је напон од 24 VDC регистрован или није регистрован на улазу. Тачније, логичка јединица одговара стању улаза од 15 VDC на 2.5 mA – **минимално**, па до 30 VDC **максимално**, док је 24 VDC на 4 mA – **номинално**. Логичка нула подразумева до 5 VDC на 1 mA – **максимално**. Међустања не мењају вредност улазног бита. Време у коме се деси читање свих улаза (уписивање физичких стања на одговарајуће меморијске локације), извршење програмског кода, и ажурирање свих излаза је време једног скен циклуса (ово време је у принципу варијабилно). Потребно је напоменути да контролер има 14 дигиталних улаза који номинално треба да региструју напонске нивое 24V или 0V. Ових 14 улаза имају адресе од I0.0 (први улазни бит), I0.1, ..., I0.7 (осми улазни бит, или последњи улазни бит првог улазног бајта), I1.0 (први улазни бит другог улазног бајта, или девети PLC дигитални улаз)... I1.5 (шести улазни бит другог улазног бајта, или четрнаести PLC дигитални улаз). Практично, дигитални улази могу да мере дискретна стања у спољном свету на нивоу укључено/искључено, објекат јесте/није у одговарајућем положају или стању (напр. рампа за пакринг и слично). Такође, у горњем десном углу слике 7 означена су и 2 аналогна улаза које има овај PLC који могу мерити аналогне улазе опсега од 0 до 10 VDC. Вредности аналогних улаза се уписују у 16-битне регистре AI0 и AI1. Међутим, за упис аналогне вредности (0-10V) у 16-битне улазне регистре AI0 и AI1 се не користи целокупан меморијски капацитет регистра већ само део, и тако (како је већ дато у техничкој спецификацији) се уписују вредности од 0 до 27648 у те регистре а

које одговарају стањима од 0-10V на аналогном улазу. Треба запазити да је присутно квантовање по нивоу (уноси се извештај тзв. шум квантовања услед АД конверзије аналогне вредности улаза).

У доњем делу слике се налазе дигитални излази којих има 10. повезани су са меморијским локацијама на адресама Q0.0...Q0.7, Q1.0 и Q1.1. Упис логичке јединице у рецимо бит Q0.0 (први бит излазног регистра) значи да се на дигиталном излазу реализује напонско стање од 24V. Практично, преко дигиталних излаза можемо укључити или искључити електрична кола која контролишу рад предметних објеката управљања (подићи или спусти рампу, и слично). PLC има (лево од дигиталних излаза приказаних на слици) и један аналогни излаз опсега од 0V до 10V једносмерне струје, а који је адресиран са AQ0. Супротно аналогном улазу, овде уписом вредности од 0 до 27648 у регистар са адресом AQ0, на аналогном излазу се појављује напон од 0V до 10V.

Горе је описана основна улазно/излазна конфигурација **PLC SIMATIC S7-1200**, а која може бити проширена додатним модулима. Зависно од изабране централне процесорске јединице (у овом случају CPU 1214C) различити број улазно/излазних периферија може бити придодат основној конфигурацији.

Генерално, навођењем меморијске адресе омогућен је директан приступ подацима. Да би се приступило биту у делу меморије, наводи се адреса, која садржи идентификатор меморијског места, адресу бајта и број бита. Приступање биту назива се и 'byte.bit' адресирање. У следећем примеру меморијско место и адресу бајта (I=input, 3=byte 3) од адресе бита (bit 4) одваја тачка ('.').



Слика 7б. Опис адресирања улаза/излаза

Сваком улазном/излазном уређају се додељује адреса. На овај начин се резервише место где ће бити похрањена логичка вредност која дефинише стање тог уређаја (1 када је уређај укључен, а 0 када је искључен).

На почетку сваког циклуса скенирања **S7-1200** узоркује физичко место улаза и записује ове вредности у улазни регистар. Улазном регистру се приписује словна ознака 'I' и приступа са: I[адреса бајта].[адреса бита], нпр. **I0.4**.

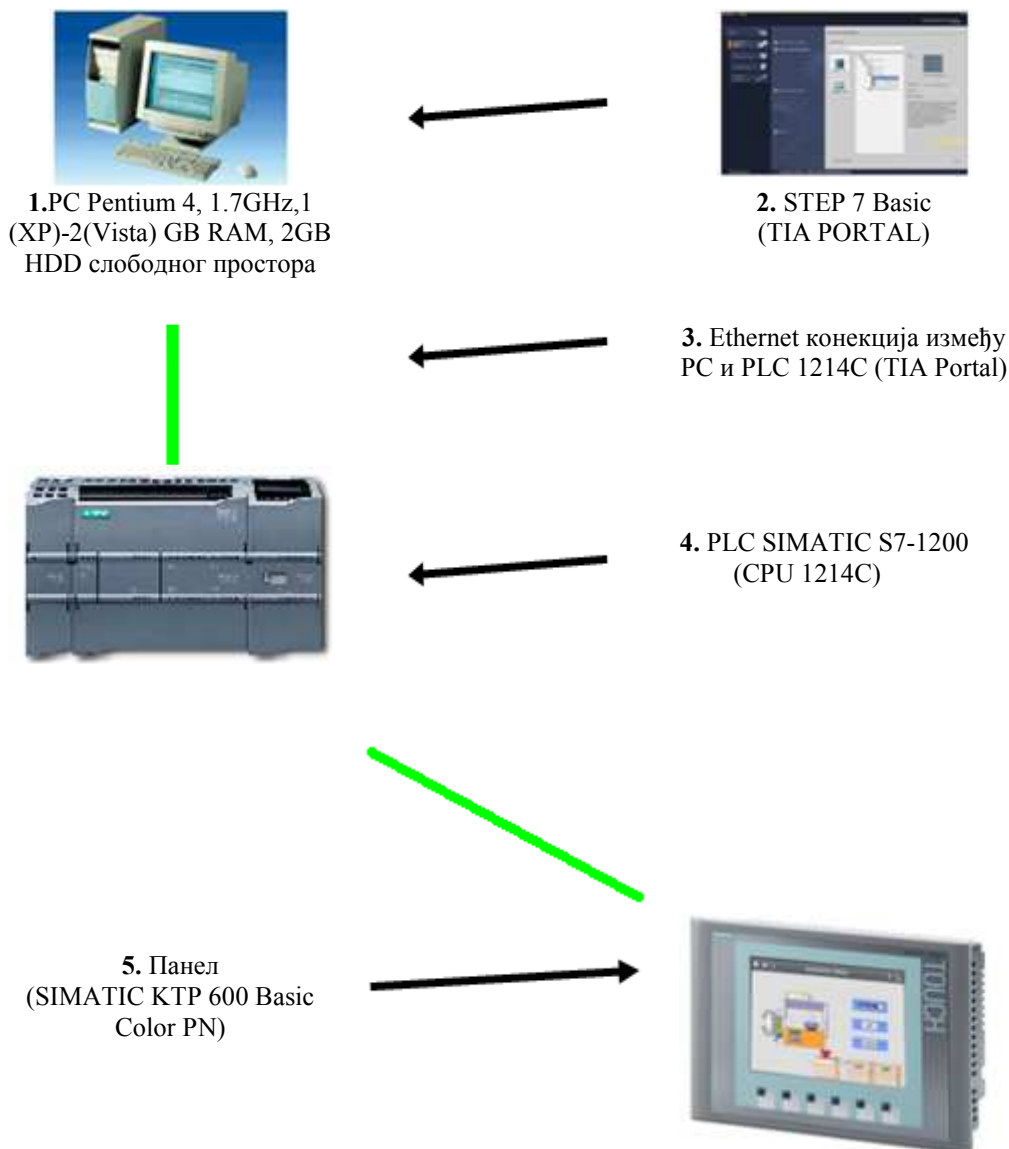
На крају овог циклуса скенирања, **S7-1200** копира вредности смештене у излазном регистру на физичко место излаза. Излазном регистру се приписује словна ознака 'Q' и приступа са: Q[адреса бајта].[адреса бита], нпр. **Q1.1**.

Словне ознаке за остале меморијске локације су:

- **V** за променљиву меморију; складишти посредно податке операција самог програма на контролеру
- **M** за бит меморију; користи се као контролни релеј за складиштење посредног стања операција или других контролних информација
- **T** за меморију тајмера; **S7-1200** обезбеђује тајмере који броје у инкрементима времена у резолуцијама од 1ms, 10ms или 100ms. Са тајмером су повезане две променљиве: тренутна вредност (16-то битна вредност која чува одбројано време) и бит тајмера (овај се бит сетује или чисти као резултат поређења тренутне вредности и унапред задате вредности)
- **C** за меморију бројача; **S7-1200** обезбеђује три врсте бројача који броје сваки ниже-на-ниже прелазни догађај на улазу бројача: један броји само на више, други на ниже, и трећи броји и на више и на ниже. Постоје две променљиве, а аналогија је иста као и код тајмера
- **HC** за брзи бројач: броји брзе догађаје независно од циклуса скенирања
- **AC** за акумулаторе: акумулатори су читај/пиши уређаји који се могу користити као меморија
- **L** за локалну меморију: локална меморија је слична **V** меморији са разликом да је **V** меморија глобална (може да јој се приступи из било ког дела програма: главног програма, подпрограма или рутине прекида)
- **AI** за аналогне улазе: аналогне улазе (температура, напон,...) **S7-1200** претвара у 16-тобитне дигиталне речи
- **AQ** за аналогне излазе: **S7-1200** претвара 16-тобитне дигиталне речи у излазни напон одговарајуће вредност, пропорционално дигиталној вредности
- **S** за меморију секвенцијалног контролног релеја (**SCR**); **SCR** или **S** битови се користе да организују операције машине или кораке у једнаке програмске сегменте

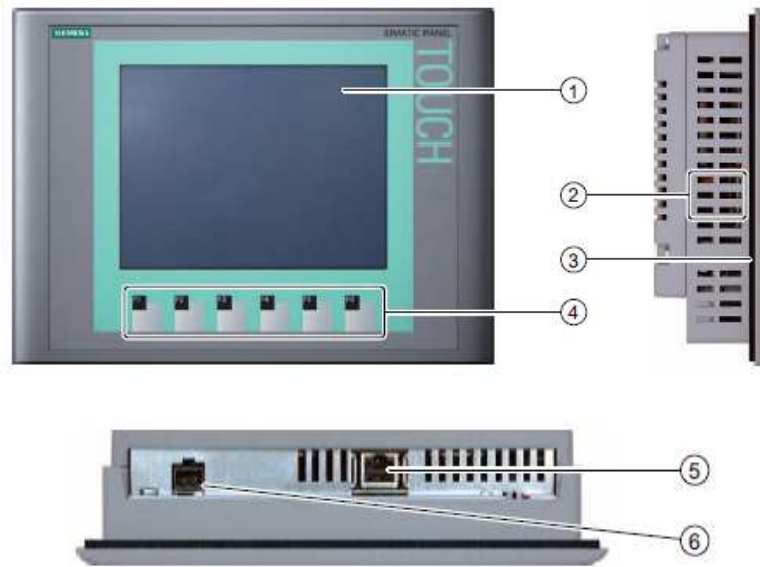
Детаљне карактеристике овог контролера су дате у додатку.

2.7 SIMATIC HMI KTP 600 PN



Слика 8 Минимални хардверски и софтверски захтеви за придруживање HMI уређаја

Изабрани **HMI** уређај јесте **HMI KTP 600 PN** који је један од стандардних панела у **SIMATIC S7** понуди. Функција **HMI** уређаја јесте да обезбеде комуникацију између оператора у пољу аутоматизације и процеса који се управља и надзире. Он омогућава унос одређених задатих вредности које оператор бира или уноси на основу свог експертског знања и информација које управо добија са панела. Преглед информација мора да буде што јаснији за оператора, тако да модерни **HMI** уређаји имају све боље могућности за визуелизацију процесних параметара и функционалности апликације надзора и управљања у целини.



Слика 9 HMI KTP600 PN - 1.Дисплеј(Touch Screen), 2.Монтажни елементи, 3.Заптивак, 4.Функцијски тастери, 5.RJ45 PROFINET конектор, 6.напонски конектор

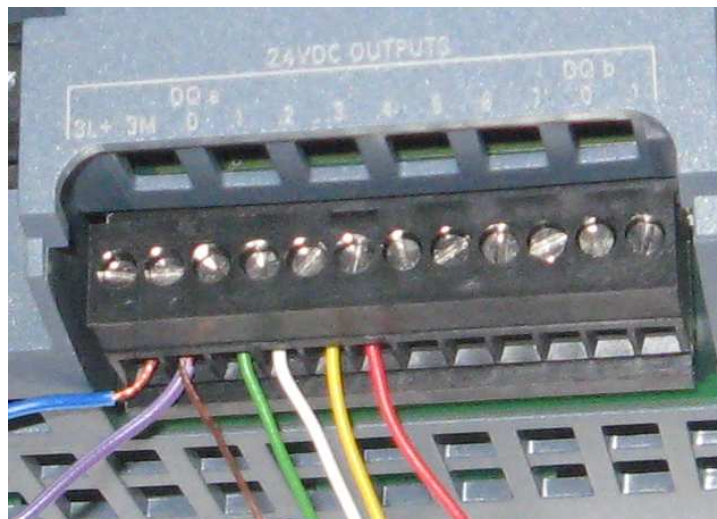
Уређај има 5.7inch-ни дисплеј и 6 функцијских тастера. Конектор, на слици 9 обележен бројем 5, се користи за повезивање овог уређаја у мрежу. Једноставно коришћењем конектора **RJ45** овај уређај постаје део мреже и видљив је на мрежи. Важно је напоменути да је за рад овог уређаја потребно обезбедити електрично напајање од 24V једносмерне струје (конектор је на слици обележен бројем 6). Уређај подржава рад са 265 боја, у резолуцији 320x240 пиксела, а сам произвођач истиче да је предвиђен за рад са контролерима **S7-1200**. Програмирање **HMI** уређаја је објашњено у наставку.

2.8. Повезивање опреме

После обезбеђивања све потребне опреме, избора начина управљања, селекције потребних сигнала, потребно је правилно повезати све улазе и излазе објеката управљања са одговарајућим улазима и излазима **PLC**-а, затим довести све потребне сигнале напајања са одговарајућих извора напајања, и обезбедити неопходну **Ethernet** комуникацију.

2.8.1 Повезивање PLC контролера и корачног мотора

Повезивање излазних пинова **PLC** контролера и мотора је обављено као на следећој слици.



Слика 10 Повезивање излаза PLC-а са мотором

Значење жица, по бојама, је дато у следећој табели.

Име сигнала	Тип	Опис	Опсег	Веза са PLC-ом	PLC I/O
1. намотај мотора	дигитални сигнал	пренос струје у PLC	24V 0.25mA	зелена	Q0.0
2. намотај мотора	дигитални сигнал	пренос струје у PLC	24V 0.25mA	бела	Q0.1
3. намотај мотора	дигитални сигнал	пренос струје у PLC	24V 0.25mA	жута	Q0.2
4. намотај мотора	дигитални сигнал	пренос струје у PLC	24V 0.25mA	црвена	Q0.3
напон излаза (напајање)	дигитални сигнал	напајање излазног регистра	24v 1A	плава	3L+
- (минус) кола („маса“)	0V	заједнички – (минус) кола	0V	љубичаста и не изолована	3M

Табела 3 Излазни и напонски сигнали и повезивање PLC-а

2.8.2 Повезивање PLC контролера и резервоара

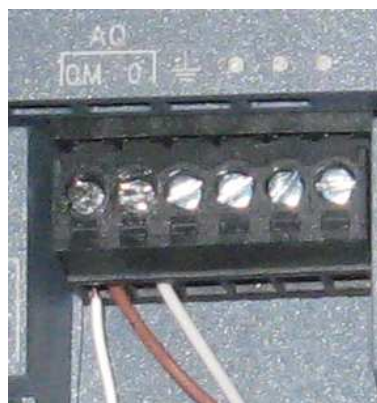
Повезивање сензора за висину воде у резервоару и аналогног улаза на PLC-у је представљено на слици 11.



Слика 11 Повезивање аналогног улаза са сензором (црна-зелена дају разлику потенцијала 0-5VDC са мерача новео)

На слици се види и бела жица на пину **AI1** која је истовремено коришћена за потребе другог пројекта у лабораторији.

Аналогни излаз овог контролера, **AQ**, је повезан са пумпом за воду према следећој слици 12.



Слика 12 Повезивање аналогног излаза са пумпом (тј. Напон од 0-10 VDC се шаље од контролера ка појачивачу снаге)

На слици се види бела жица, на пину **QM** (заједничка маса), која је истовремено коришћена за потребу другог пројекта.

Име сигнала	Тип	Опис	Опсег	Веза са PLC-ом	PLC I/O
сензорски улаз	аналогни сигнал	аналогни сигнал који представља ниво воде у резервоару	од 0V до 5V	зелена	AI0
- (минус) кола	0V	- (минус) кола	0V	црна	2M
излаз за пумпу	аналогни сигнал	аналогни сигнал којим се контролише напон пумпе	Од 0V до 10V	бела	AQ0
- (минус) кола	0V	- (минус) кола	0V	црвена	QM

Табела 4 Излазни и напонски сигнали и повезивање PLC-а

3. SIEMENS TIA11 софтверски пакет

Термин **ТОТАЛНО ИНТЕГРИСАНА АУТОМАТИЗАЦИЈА (TIA)** је Siemens први пут презентирао у Ротердаму у новембру 1996. године [5]. Идеја је да се комплетна аутоматизација целокупног постројења изведе интегрално са хардверским и софтверским модуларним компонентама једног система, у овом случају **SIMATIC**-а [5]. Да би се истакла компатибилност, тј. јединственост у конфигурацији, обради података и комуникацији, појмом **SIMATIC** су обухваћене (сем **PLC**-ева) и остале типичне компоненте једног система за управљање индустријским процесима. Тако је термин „тотално интегрисана аутоматизација“ покрио

- **PLC-ове (SIMATIC Controller)**
- индустријске рачунарске мреже (**SIMATIC NET**)
- индустријске персоналне рачунаре (**SIMATIC PC**)
- уређаји за визуелизацију и комуникацију оператера у пољу аутоматизације и управљаних процеса (**SIMATIC HMI**)
- децентрализоване периферије (**SIMATIC DP**)
- софтвер (**SIMATIC INDUSTRIAL SOFTWARE**)
- систем за вођење процеса (**SIMATIC PCS7**)

Језгро „ТОТАЛНО ИНТЕГРИСАНЕ АУТОМАТИЗАЦИЈЕ“ је софтвер [5]. Комплетан савремен индустријски софтвер пре свега мора да буде у складу са светским стандардима (**SIEMENS** -ов индустријски софтвер задовољава нпр. IEC 1131 и DIN IEC 6.1131-3, а **SIEMENS** је и члан удружења **PLCopen**).

TIA („Тотално интегрисана аутоматизација“) је **Siemens**-ов софтверски пакет који интегрише различите софтверске алате и целине, и –**TIA11** је најновија Siemens-ова генерација инжењерског софтвера за индустријску аутоматизацију. То је први индустријски софтвер са „једним инжењерским окружењем“ за све задатке аутоматизације. То значи да уједињује све софтверске алате потребне за аутоматизацију је само једном развојном окружењу.

TIA је софтвер који омогућава програмирање и **PLC**-а, и **HMI**-а, и конфигурацију мреже, и подешавање параметара, и on-line мониторинг и многе друге функције потребне за брзо и ефикасно решавање захтева аутоматизације. **TIA** пружа алате за управљање и конфигурирање свих уређаја у одређеном пројекту као што су **PLC** и **HMI** уређаји. Пружа кориснику прилагођено окружење за развој, уређивање и праћење логике потребне за управљање апликацијом.

TIA нуди софтверске технологије у интуитивном корисничком окружењу и омогућава кориснику да се концентрише на инжењеринг без успоравања учењем нових корисничких програма. Његов интуитиван изглед и навигација омогућује брзо препознавање жељених функција програмирања и уређивања.

Како би се помогло при повећању продуктивности **TIA** нуди два различита погледа алата:

1. задатку оријентисани поглед (**Portal view**), који је организован на начин да се постигне потпуна функционалност алата и
2. поглед усмерен на елементе у пројекту (**Project view**).

Корисник одабира опцију која му највише одговара те се само једним кликом може пребацивати између два наведена погледа.

Portal view пружа функционални преглед пројектних задатака и организује функције алата према задатку које ће се остварити, попут стварања конфигурације хардверских компоненти и мрежа.

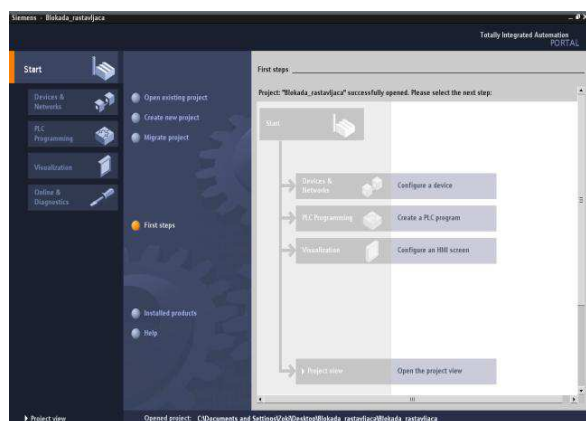
Project view пружа приступ свим компонентама у склопу пројекта. Уз све ове компоненте на једном месту, омогућен је једноставан приступ сваком аспекту реализованог пројекта.

За брзо и ефикасно решавање проблема, **TIA** пружа интелигентну, упитно орјентисану помоћ:

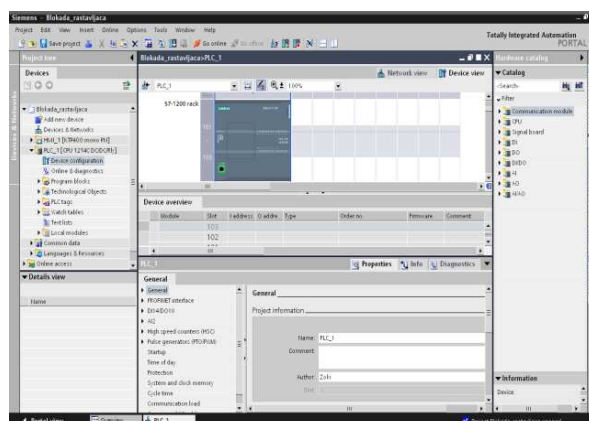
- Поље за унос пружа аутоматску помоћ за унос тачне информације (одговарајући опсег и тип податка) за то поље. На пример, ако није унесена исправна вредност, приказаше се текстуални оквир са распоном важећих вредности.
- Неке помоћи за алате у интерфејсу, како би се пружиле додатне информације.

Осим тога, **TIA** је свеобухватно информационо окружење који у потпуности описује функционалности **SIMATIC** алата.

TIA пружа целокупне информације, односно помоћ која описује све инсталиране **SIMATIC** производе. Информациони интерфејс такође укључује референтне информације и примере.



Portal view



Project view

Слика 13 Изглед екрана у TIA порталу

Као саставни део **TIA** -е, **STEP7** софтвер нуди могућност програмирања **PLC**-а у два основна програмска језика (**LAD** и **FBD**), за практичност и успешност у развоју програма контроле за одређену апликацију. Исто тако се у склопу **TIA**-е налази и **WinCC** софтвер за програмирање и конфигурирање свих

расположивих **SIMATIC S7 HMI** панела. За лакше проналажење потребних информација TIA пружа и велику „он-лине“ помоћ и подршку.

3.1 Потребан хардвер

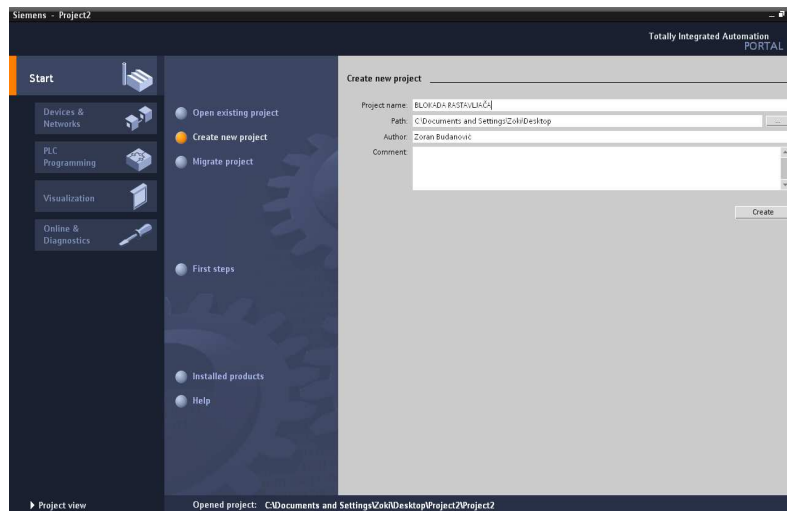
Хардвер потребан за инсталацију **TIA** софтвера је следећи:

- **PC Pentium 4, 1.7GHz** или јачи
- Оперативни систем **Windows XP (Home SP3, Professional SP3)** или **Windows Vista (Home Premium SP1, Business SP1, Ultimate SP1)**
- **1 GB RAM**
- **2GB** слободне меморије на диску (минимум)

3.2 Прављење првог пројекта

За креирање новог пројекта потребно је отворити претходно инсталирану **TIA** - у и у **Portal view** кликнути на „Create new project“ потврдни оквир. У „Project name“ текстуални оквир уноси се име пројекта, те под „Path“ локација на коју ће се пројект снимити(чувати). **TIA** портал аутоматски предлаже има аутора пројекта које је унесено приликом инсталације софтвера. Задња ставка (необавезно) је писање коментара о пројекту (текстуални оквир одмах испод оквира за уношење аутора).

После уношења основних података о пројекту потребно је кликнути на „**Create**“ дугме. Пројекат је након тога успешно створен и снимљен на унету меморијску локацију.



Слика 14 Прављење пројекта

После успешно створеног пројекта, програм се аутоматски пребацује на следеће кораке потребне за успешно извођење пројекта. Следећи корак је додавање уређаја који ће се користити у пројекту.

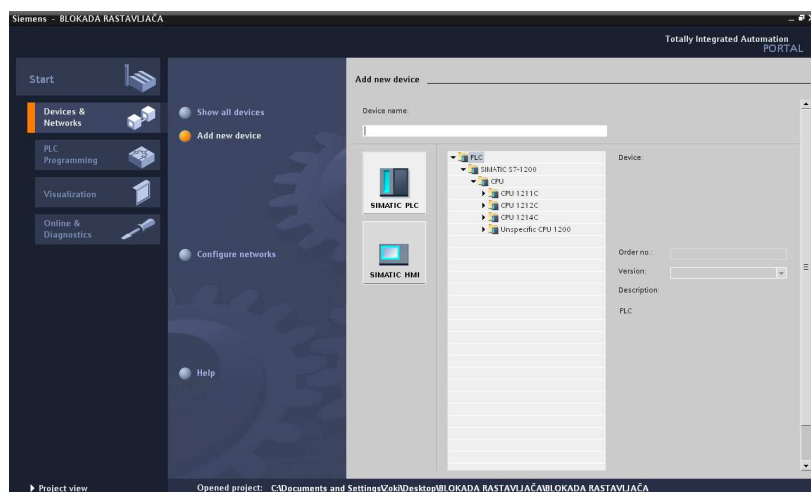
3.3 Додавање нових уређаја

Са десне стране **Project view** потребно је кликнути на „**Devices & Networks**“ икону. Уколико су уређаји већ повезани са конфигурационим рачунаром, под „**Show all devices**“ може се погледати листа доступних уређаја те их додати у пројекат двоструким кликом на икону сваког од њих.

Уколико уређаји нису повезани са конфигурационим рачунаром потребно је кликнути на „**Add new device**“, те са десне стране у „**Device name**“ текстуални оквир уписати име уређаја који се жели додати у пројекат или пронаћи уређај из листе испод. Након што се одговарајући уређај пронађе, двоструким кликом на његову икону додаје се пројекту.

Након успешно доданог уређаја **TIA** се аутоматски пребацује у **Project view** у којем се могу конфигурисати или програмирати додани уређаји. На следећој слици приказано је додавање уређаја у пројект помоћу **Portal view**

У овом раду неће бити представљено додељивање **IP** адресе конфигурацијском рачунару. Та адреса се додељује ручно или аутоматки. У просторијама Центра за примењену аутоматику, Факултета инжењерских наука у Крагујевцу, рачунари су конфигурисани тако да сваки има своју фиксну **IP** адресу коју одређује овлашћено лице, односно, администратор рачунарске мреже.



Слика 15 Избор и додавање уређаја

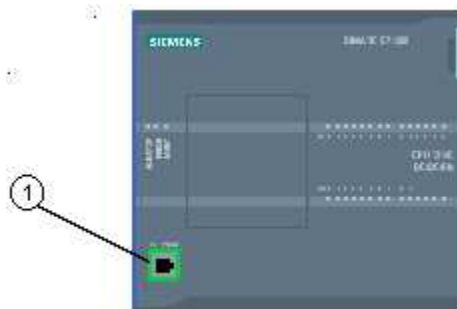
3.4 Додељивање IP адресе PLC-у

За додељивање **IP** адресе централној процесорској јединици може се користити једна од следеће две методе:

- Online додељивање привремене **IP** адресе
- Конфигурисање сталне **IP** адресе

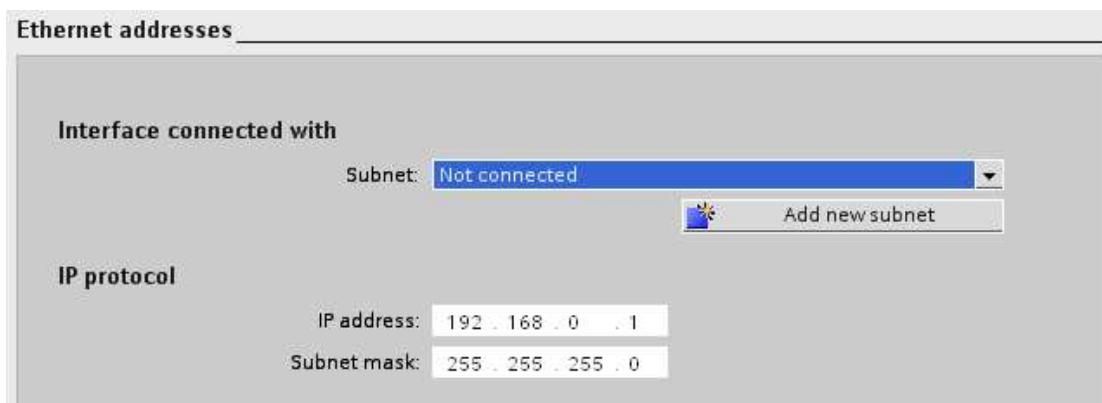
У кабинету су сви уређаји са сталним, односно статичким **IP** адресама. Након што је **PLC** додан пројекту, могу се подесити параметри за **PROFINET**

интерфејс. За подешавање параметара **PROFINET** интерфејса потребно је отићи у **Project view** и кликнути на зелени **PROFINET** оквир на **CPU**-у како би се одабрао **PROFINET** порт.



Слика 16 Мрежни порт – 1. **PROFINET** порт

У „Properties“ прозору потребно је одабрати „Ethernet adress“ конфигурациони улаз. **TIA** портал приказује дијалог са конфигурацијом Ethernet адресе, која повезује софтверски пројект са **IP** адресом **CPU**-а која ће примити тај пројект. Овде се може унети, изменити или прегледати **IP** адреса.



Слика 17 Подешавање мрежних параметара

„**Online**“ додељивање **IP** адресе подразумева њено уношење док је **PLC** у **RUN** начину рада. Дакле, **IP** адреса може се доделити **PLC**-у и у „**On-line**“ начину рада, али ове **IP** адресе су привремене и биће изгубљене уколико **PLC** остане без напајања. За трајно уписивање **IP** адресе потребно је подесити **IP** адресу под „**Device configuration**“ и учитати је у **PLC**. Учитана **IP** адреса, као и део конфигурације **PLC** -а неће бити изгубљени након нестанка напајања.

3.5 Extended download to device

„**Extended download to device**“ функција може приказати све доступне уређаје на мрежи и да ли је јединствена **IP** адреса додељена свим уређајима. За приказ свих доступних уређаја, заједно са њиховим додељеним **MAC** и **IP** адресама потребно је кликнути на „**Show all accessible devices**“ иконицу. Уколико потребни мрежни уређај није на листи, комуникације према том уређају су из неког разлога прекинуте. Мрежа и уређај морају бити истражене за хардверским и/или конфигурацијским грешкама. **IP** адреса за **HMI** панел дођељује се на исти начин као и за **PLC**.

3.6. Примена TIA11 у решавању инжењерских апликација. Поставка листе функционалности апликације

Као што је већ речено, TIA11 ће се применити у решавању проблема управљања и надзора два дистрибуирана процеса:

- 1) промена/одржавање нивоа течности у резервоару,
- 2) промена/одржавање брзине и смера окретања корачног мотора.

Оба објекта су повезана са два различита PLC Siemens S7-1200, док су они путем рачунарске мреже повезани са HMI и персоналним рачунаром. HMI даје приказ и визуелизацију неких одабраних величина и стања у систему, затим, омогућава оператеру да изабере задату висину течности, и задати смер и брзину окретања мотора. Са персоналног рачунара је могуће контролисати све уређаје на нижим хијерархијским нивоима (HMI и оба PLC-а), уписивати у базу/фајл све процесне величине од интереса са задатом периодом одабирања T, имати приказ промене свих процесних величина од интереса (дијаграмски приказ, приказ индикатора, визуелизација промена величина и сл.), задавати жељене излазне величине и мењати параметре регулатора.

Конкретно, листа функционалности апликације је следећа

1. Надзор и управљање брзином и смером обртања корачног мотора

Функционалности омогућене са HMI уређаја:

- укључивање или искључивање мотора
- избор смера окретања мотора (смер казаљке на сату је означен као позитиван, са „+“, и супротно)
- избор брзине окретања, тј. једне од 3 понуђене у падајућем менију: 100 o/min, 10 o/min, или 0.3 o/min
- индикација стања мотора (ON или OFF, + или -, док визуелизација брзине обртања није урађена)

Функционалности омогућене са персоналног рачунара:

- укључивање или искључивање мотора
- избор смера окретања мотора (смер казаљке на сату је означен као позитиван, са „+“, и супротно)
- избор брзине окретања, тј. једне од 3 понуђене у падајућем менију: 100 o/min, 10 o/min, или 0.3 o/min
- индикација стања мотора (ON или OFF, + или -, док визуелизација брзине обртања није урађена)
- SWITCH дугме којим се блокирају сви уноси са HMI-а
- поље за задавање произвољне брзине мотора (од 0 до 99)

2. Надзор и управљање нивоом воде у резервоару

Функционалности омогућене са HMI уређаја:

- укључивање или искључивање пумпе

- избор режима рада пумпе (ручно или аутоматски)
- у ручном режиму рада - избор управљаког напона који **PLC** шаље ка пумпи (0-10 [V])
- у аутоматском режиму рада – избор задате висине воде у резервоару (0-30 [cm])
- графички бар који визуелно приказује висину воде у резервоару (визуелизације измерене висине воде у резервоару)
- Input/Output поља у којима је дата задата и остварена (измерена) висина воде у резервоару, као и напона на пумпи (управљачка величина)

Функционалности омогућене са персоналног рачунара:

- укључивање или искључивање пумпе
- избор режима рада пумпе (ручно или аутоматски)
- у ручном режиму рада - избор управљачког напона који **PLC** шаље ка пумпи (0-10 [V])
- у аутоматском режиму рада – избор задате висине воде у резервоару (0-30 [cm])
- графички бар који визуелно приказује висину воде у резервоару (визуелизације измерене висине воде у резервоару, тј. управљане променљиве)
- графички бар који визуелно приказује задату висину воде у резервоару
- графички бар који визуелно приказује управљачку променљиву (визуелизација напона који **PLC** шаље ка пумпи (0-10 [V])
- Input/Output поља у којима су дата задата, управљана и управљачка променљива
- дијаграмски прикази промене задате, управљане и управљачке променљива
- упис у фајл/базу задате, управљане и управљачке променљиве са задатом периодом одабирања **T** (време у временским тренуцима **κT** се такође уписује у фајл/базу)
- могућност избора периоде одабирања **T**
- могућност избора/промене параметара регулатора **K** (ради се о најједноставнијем **P** регулатору, који има само један подешљив параметар **K**)
- На **PC** екрану постоји **SWITCH** којим се по жељи могу блокирати одређена поља на **HMI** уколико се то жели. Предвиђено за потребу блокирања **HMI** утицаја приликом промене параметара са **PC**-а.

Остварење овог задатка коришћењем **TIA11** софтверског окружења биће представљено у поглављима која следе.

3.7. Примена TIA11 у програмирању PLC/HMI ради управљања брзине корачног мотора

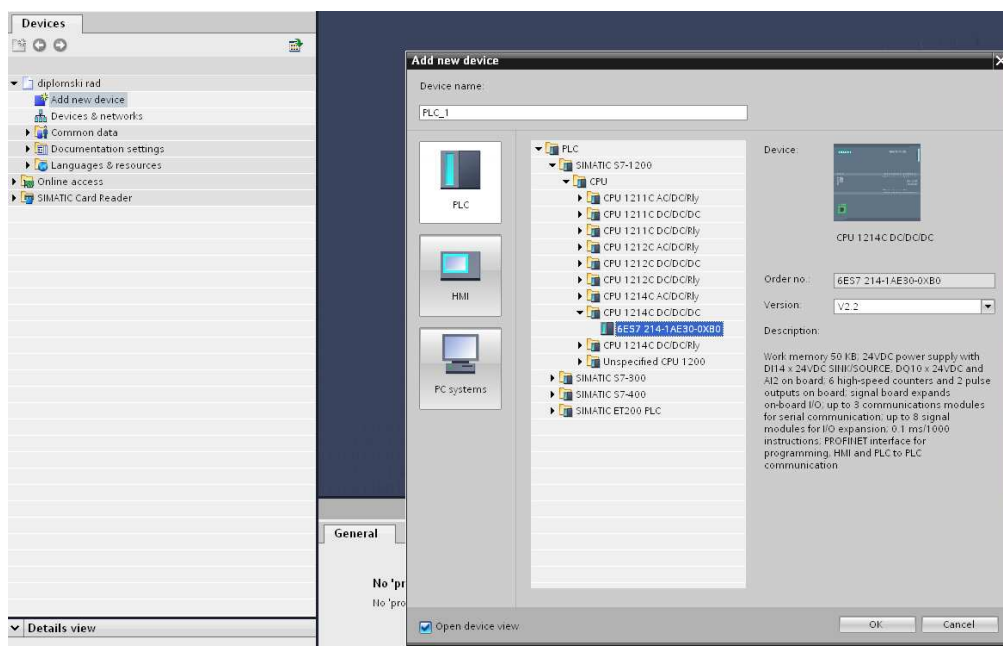
3.7.1. Повезивање уређаја

О повезивању опреме је већ било речи у поглављу 2.8. Сада, у контексту програмирања уређаја: у рачунарској мрежи Факултета су постављене . IP адресе PLC и HMI уређаја планираних за наведену сврху: 147.91.203.131 и 147.91.203.134.

На пиновима дигиталних излаза PLC-а (меморијске адресе: Q0.0,Q0.1,Q0.2,Q0.3) су спојена четири пола мотора преко којих се врши управљање.

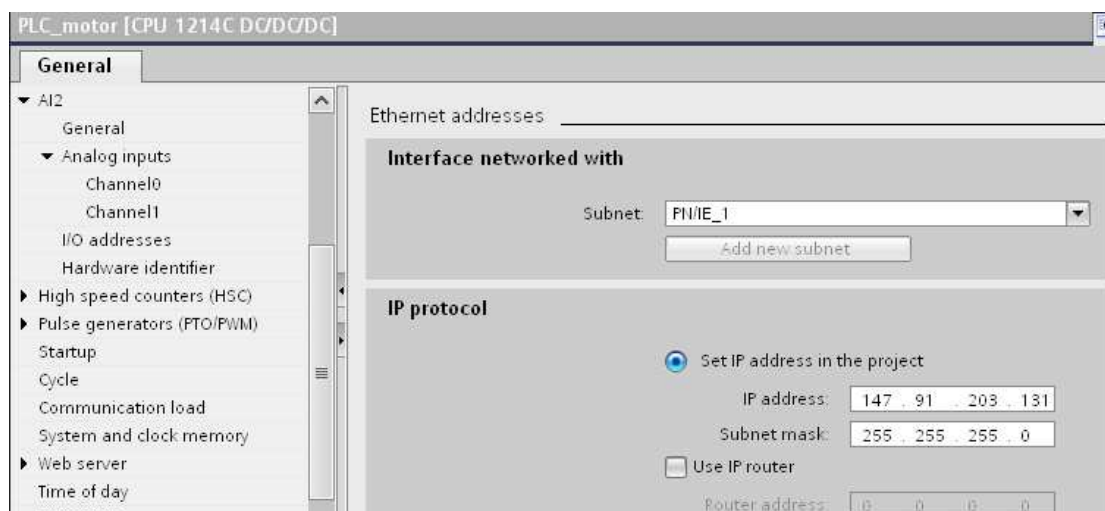
3.7.2 Програмирање PLC-а

После покретања TIA-е, правимо нови пројекат под називом *diplomski rad*, а затим додајемо први уређај, односно PLC који на листи има ознаку CPU 1214C DC/DC/DC.



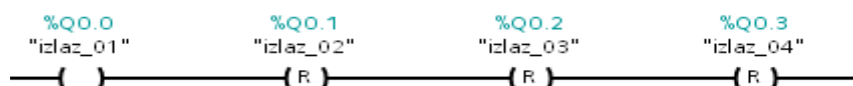
Слика 18 Избор и додавање PLC-а

Двоструким кликом на *Devices & Networks* и двоструким кликом на *Device configuration* (у делу где је приказан нови уређај) отвара се изглед контролера и поља за његово подешавање. Подешавање за IP и интерфејс су приказани на следећој слици.



Слика 19 Мрежна конфигурација PLC-а

Као што је у претходном делу описано, сваки од четири намотаја мотора је везан за један излазни пин PLC -а. Тако ће за покретање мотора бити потребно да се намотаји мотора наизменично активирају, односно да се излазни пинови наизменично побуђују електричним сигналимa. При сетовању једног од излазних пинова на контролеру је такође потребно ресетовати, односно искључити напонски сигнал, на претходном излазном пину. Ако редом, у одређеним временским размацама, сетујемо први излаз, па затим други, па трећи па онда и четврти, уз истовремено ресетовање осталих, покретаћемо ротор мотора у смеру у којем су пинови постављени. Пример једне такве гране у дијаграму је дат на слици 20.



Слика 20 Грана PLC програма

Са слике 20 се јасно види да је један излаз **Q0.0** сетован док се истовремено ресетују остала три излазна пина.

Потребно је сада направити тагове који ће се користити. У **Project tree** са два клика на **Default tag table** се отвара табела. У њој се праве тагови и повезују са одређеним меморијским локацијама, као што је дато на слици 21.

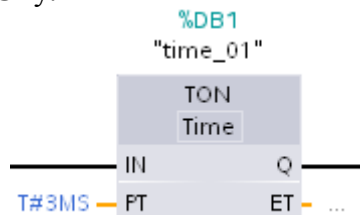
Default tag table							
	Name	Data type	Address	Retain	Visibl...	Acces...	Comment
1	izlaz_01	Bool	%Q0.0		☒	☒	izlaza motor 1
2	izlaz_02	Bool	%Q0.1		☒	☒	izlaza motor 2
3	izlaz_03	Bool	%Q0.2		☒	☒	izlaza motor 3
4	izlaz_04	Bool	%Q0.3		☒	☒	izlaza motor 4
5	motor_ON_OFF	Bool	%M0.1		☒	☒	bit za setovanje ON/OFF motora
6	motor_SMER	Bool	%M0.2		☒	☒	bit za setovanje +/- smeru motora
7	motor_brzina_HMI	Time	%MD2		☒	☒	tag za prijem brzine motora sa HMI-a
8	kontrolni_izlaz	Bool	%Q0.5		☒	☒	izlaz koji daje signal da je motor ukljucen
9	motor_brzina_PC	Time	%MD6		☒	☒	izbor brzine motora sa ekrana
10	switch_brzina	Bool	%M0.3		☒	☒	stop za brzinu sa HMI i unos se brzina sa PC-a
11	motor_brzina	Time	%MD10		☒	☒	direktna promenljiva koja daje brzinu motora
12	motor_brzina_izbor_HMI	Bool	%M0.4		☒	☒	bit koji dozvoljava unos brzine sa HMI

Слика 21 Пороменљиве(тагови) у PLC меморији

Са слике се види да су прва четири тага везана за излаз, излази су везани за коло **ULN2003** које преноси сигнале који се прослеђују мотору.

Ако би се сетовање и ресетовање пинова обављало без временских закашњења онда би струје у намотајима биле истовремене и у намотајима се не би формирала сила која би утицала на ротор и изводила га из равнотежног положаја. Из тог разлога се пре сваког сетовања следећег излазног пина, у низу, уводи временско кашњење помоћу тајмерског блока **TON**.

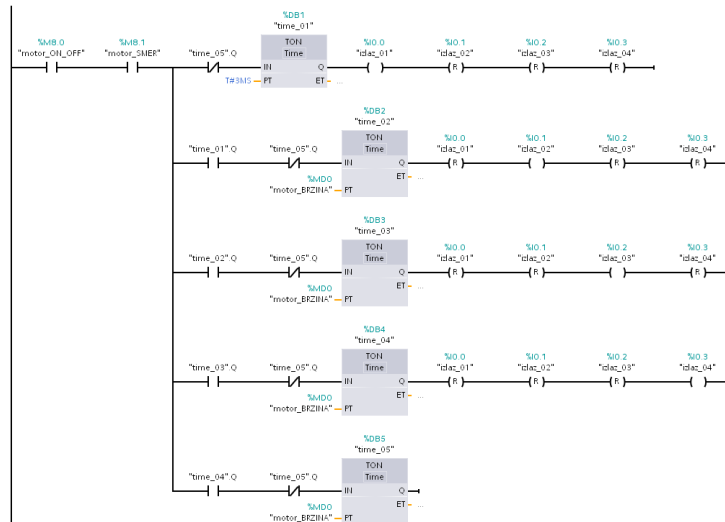
TON инструкција када је улаз (**IN**) укључен. Након задатог времена, по паљењу улаза, пали се излаз (**Q**). Ова инструкција након појаве сигнала на улазу, почиње да броји време које јој се прослеђује кроз вредност **PT**. та вредност може бити изражена бројчано (у милисекундама) или се може доделити вредност са неке меморијске локације у **PLC** -у.



Слика 22 Тајмерска инструкција **TON**

Као што се са слике 22 види, име ове инструкције је “time_01” и смештена је на локацију **DB1** у меморији **PLC**-а. Са слике се такође види да ова инструкција броји по 3 милисекунде, што се види са улаза **PT**. У неким наредним инструкцијама ће се ова вредност узимати са меморијске локације на којој се налази променљива “**motor BRZINA**”. На тај начин се контролише време између сетовања следећег излазног пина, односно време после којег ће се побудити следећи намотај у мотору. Тиме се практично одређује и брзина окретања ротора мотора.

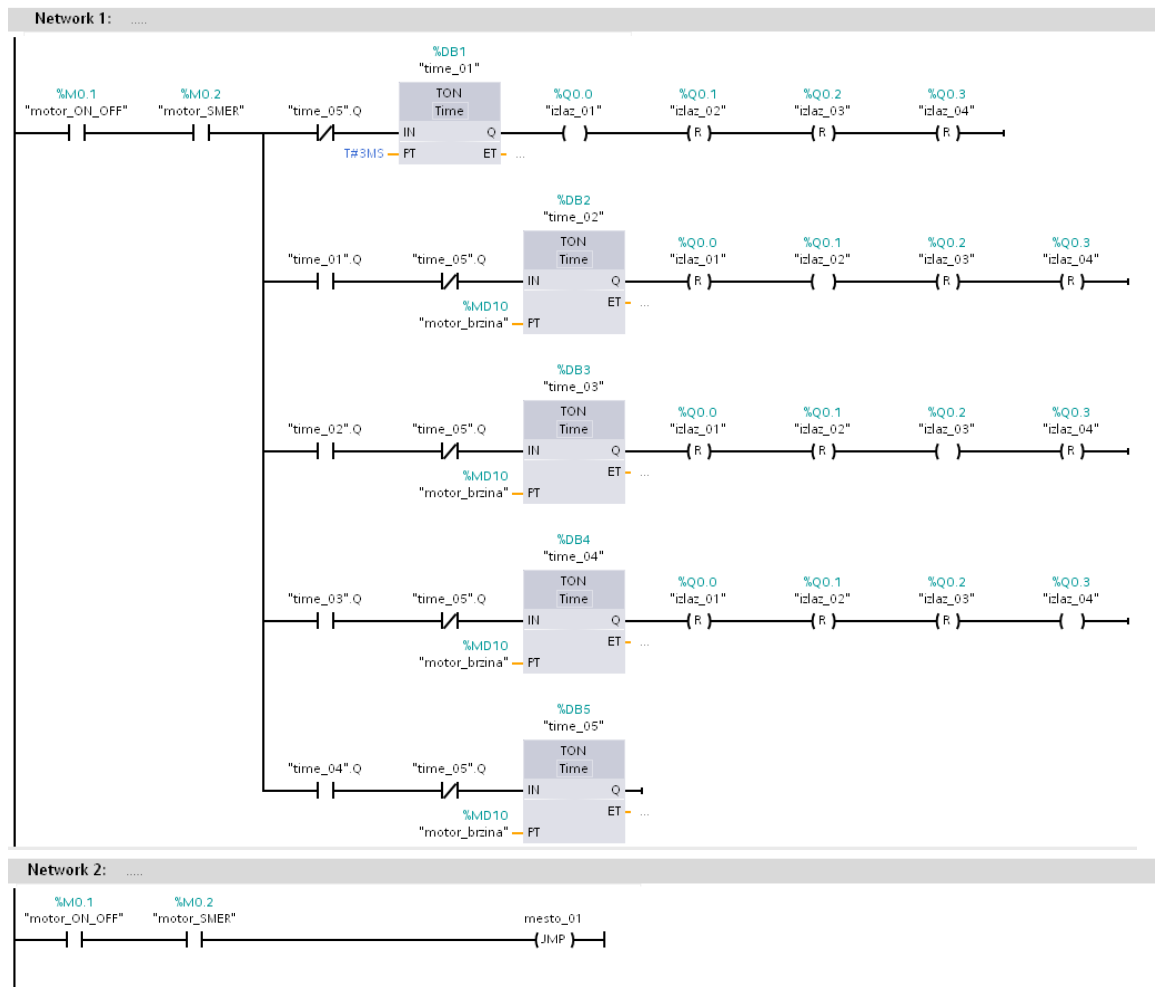
Укључивање и искључивање мотора, као и смер мотора ће се контролисати са променљивима “**motor ON OFF**” и “**motor SMER**” које су смештене на меморијским локацијама **M0.1** и **M0.2**. Ове две променљиве су замена за класичне прекидаче за укључивање и искључивање мотора и за бирање смера мотора. Ове две променљиве ће бити сетоване и ресетоване са **HMI**-а и касније ће бити објашњене. На основу претходног, може се направити дијаграм који ће се мотор окретати у једном смеру након укључивања (сетовања меморијске локације **M0.1** преко променљиве “**motor_ON_OFF**”) и он је представљен на слици 23.



Слика 23 Грана за контролу окретања мотора, из PLC дијаграма

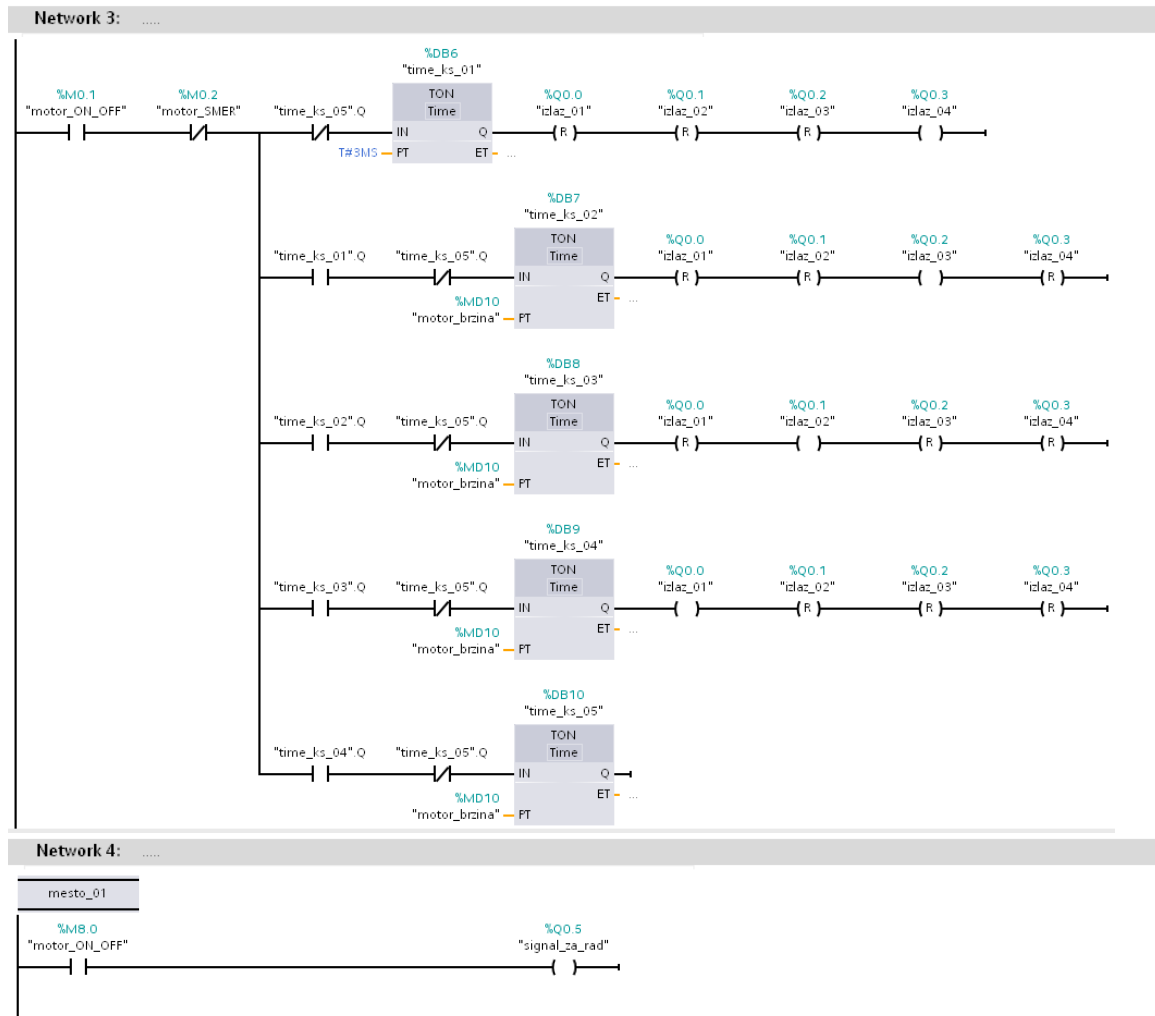
Са слике 23 се види да се на почетку дијаграма прво проверавају локације **M0.1** и **M0.2**, односно да ли је мотор укључен и да ли ће се мотор окретати у смеру казaljке на сату. Ако су те локације сетоване тада може да се извршавају и друге инструкције. У првој грани тајмер инструкција **TON** одброји 3 милсекунде и сетује први пин а остале ресетује. После тога се редом извршавају и остале гране, свака има за циљ да након одређеног времена сетује наредни излазни пин. На крају, када се ресетује и четврти излазни пин, тајмер “*time_05*” ће обезбедити да он остане у активном стању да би се обезбедило исправно окретање ротора. На тај начин је цео циклус окретања мотора, у једном смеру, за 4 корака, успешно изведен.

Како је представљен управљачки дијаграм за окретање мотора у једном смеру, онда није тешко разумети да ће за контра смер бити потребно да се дигитални излази сетују на исти овакав начин само у супротном смеру, од четвртог ка првом излазном пину. Цео дијаграм је представљен на следећој слици 24, у наставку.



Слика 24 Гране PLC програма

У мрежи 2 се види да у колико су сетоване локације **M0.1** и **M0.2** програм ће аутоматски скочити на локацију "**mesto_01**". Ово је битно јер ова два бита у меморији су резервисани за стања да ли је мотор укључен и у којем смеру се окреће. У колико је локација **M0.2** сетована онда нема потребе да се извршава код за окретање мотора у супротном смеру јер се тиме прави грешка у управљању. У даљем делу дијаграма је дата грана којом се мотор окреће у супротном смеру у колико мем. локација **M8.1** није сетована, односно у колико се на **HMI**-у активира команда за окретање у супротном смеру.

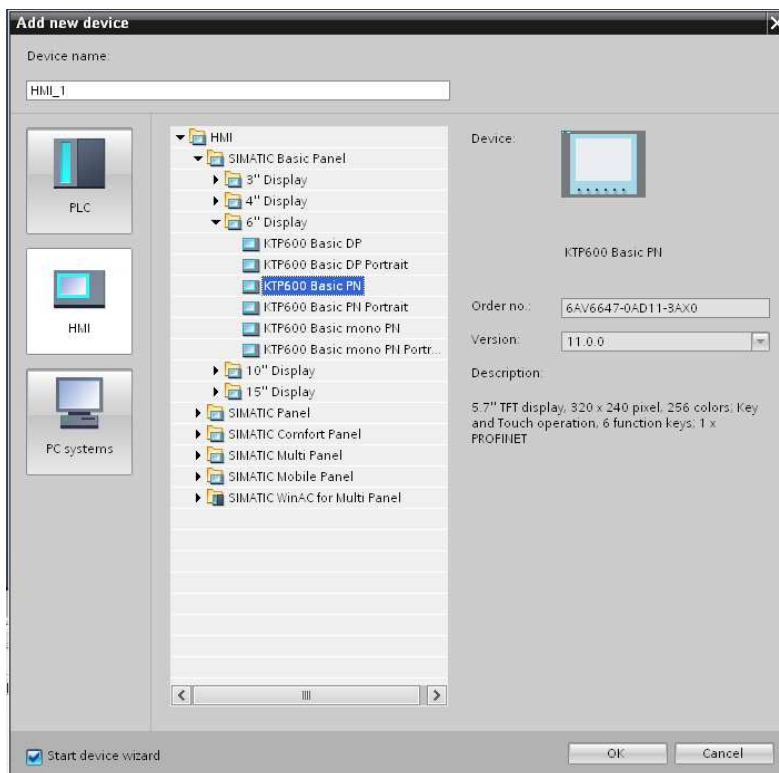


Слика 25 Гране PLC програма

На претходној слици 25 је дата грана у мрежи 3 која окреће мотор у супротном смеру. У мрежи 4 је грана у којој се налази “*mesto_01*” у коју програм долази из мреже 3 ако су испуњени услови. У истој овој мрежи се активира излаз на петом излазном пину у колико је мотор укључен.

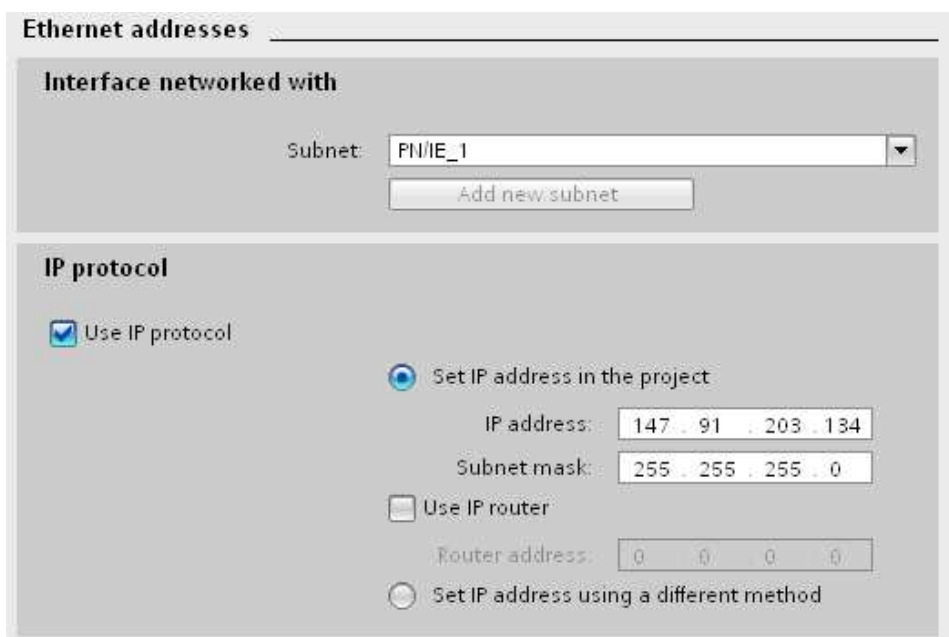
3.7.3 Програмирање HMI-а

Потребно је додати нови уређај, односно **HMI KTP600 PN** као на слици 26.



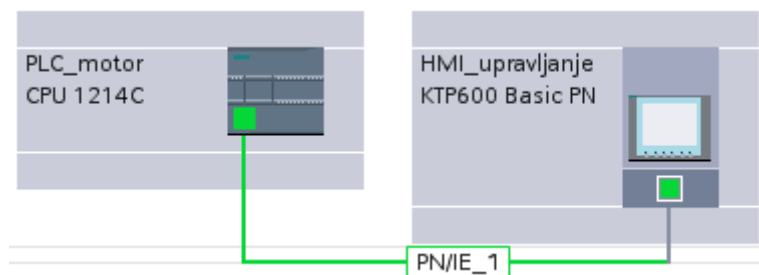
Слика 26 Додавање HMI уређаја

Два клика на **Devices & networks** ће приказати све уређаје, затим два клика на нови уређај ће приказати поља за подешавање мрежних параметара овог уређаја.



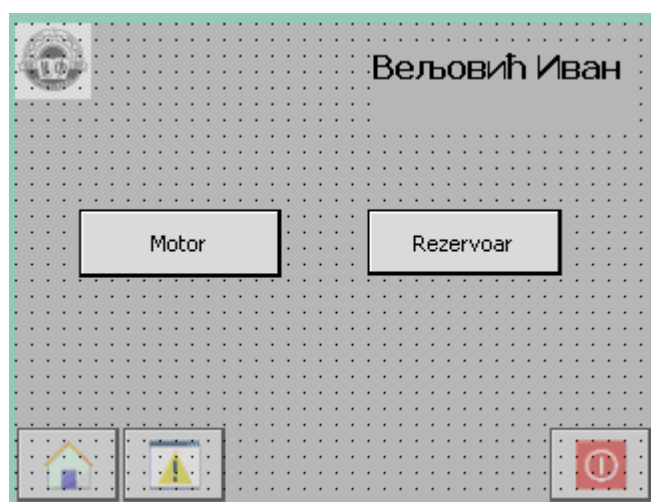
Слика 27 Мрежна конфигурација HMI уређаја

Уређај је успешно конфигурисан, што се види на слици 28, па се може правити апликација.



Слика 28 Мрежа PLC и HMI уређаја

Поред главног изгледа екрана, направљена су и три екрана, “*motor*”, “*rezervoar*” и “*rezervoar_rusni_rezim*”. Сваки од њих ће контролисати по један процес. Изглед главног екрана је дат на слици 29.



Слика 29 Изглед главног екрана HMI-a

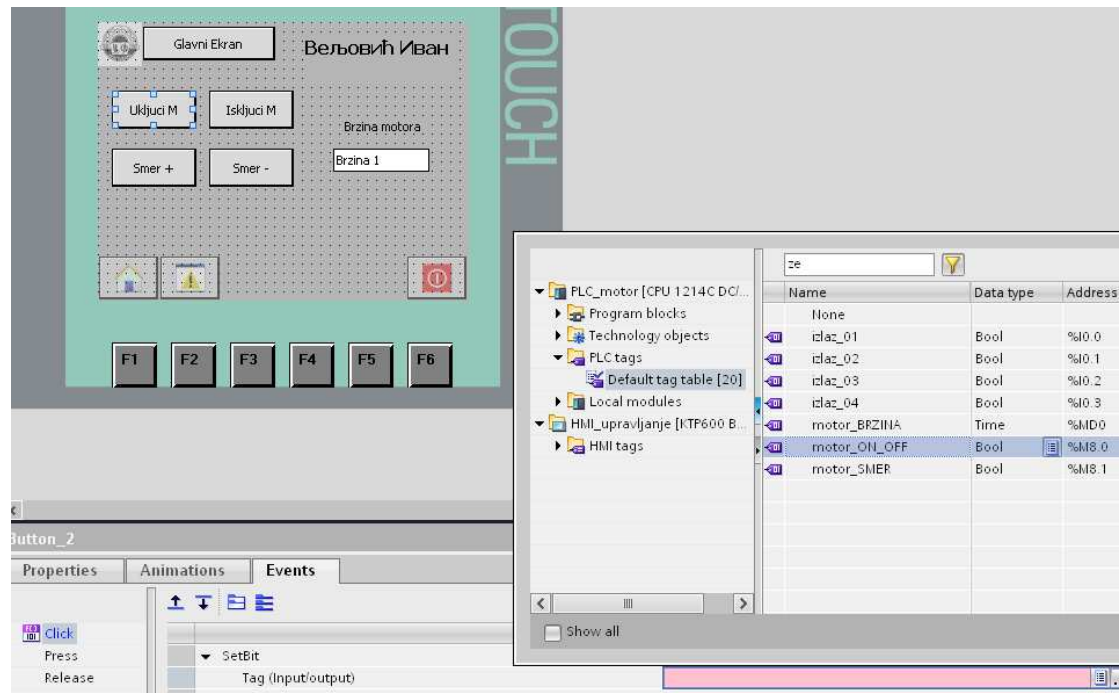
На њему постоје два дугмета ‘*Motor*’ и ‘*Rezervoar*’ којима се прелази у одговарајући екран за контролу једног од ових процеса. Екран за контролу мотора је дат на слици 30.



Слика 30 Изглед HMI екрана за контролу мотора

На врху се налази дугме ‘*Glavni Ekran*’. Оно служи за повратак на почетак, као што његово име и говори.

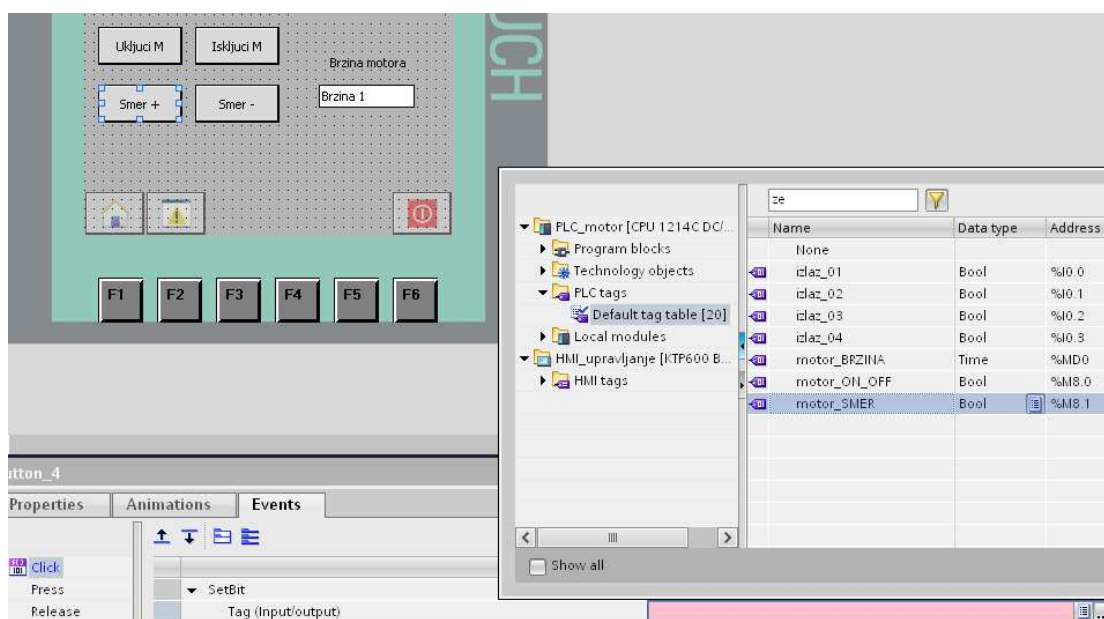
Пуштање мотора у рад и његово заустављање се обавља дугметом ‘*Ukljuci M*’ и ‘*Iskljuci M*’. Два клика на дугме ‘*Ukljuci M*’ ће у доњем делу екрана приказати његове особине. У картици *Events* се подешавају акције које ће се извршити при коришћењу овог објекта. Одабран је догађај *Click* и њему је додељена функција *SetBit*. Ова функција сетује “*motor_ON_OFF*” који се налази на локацији **M0.1** у меморији **PLC**-а што је приказано сликом 31.



Слика 31 Додавање функције графичком објекту на HMI екрану

На исти начин, двокликом на дугме ‘*Iskljuci M*’ је овом дугмету додељена функција која ће ресетовати бит на локацији **M0.1** у меморији **PLC**-а. На тај начин се врши пуштање мотора у рад и његово заустављање.

На екрану се даље уочавају два дугмета, ‘*Smer +*’ и ‘*Smer -*’. Њихова намена је да одређују смер окретања мотора. Усвојено је да позитиван смер буде смер који се поклапа са смером кретања казаљки на часовнику. Ова два дугмета ће, као и претходна два за старт и стоп мотора, бити коришћена за сетовање и ресетовање тага “*motor_SMER*”, односно **M0.1** меморијске локације у **PLC**-у. Пример је илустрован следећом сликом 32 на којој се види одабир локације за функцију *SetBit*.



Слика 32 Додавање функције графичком објекту на HMI екрану

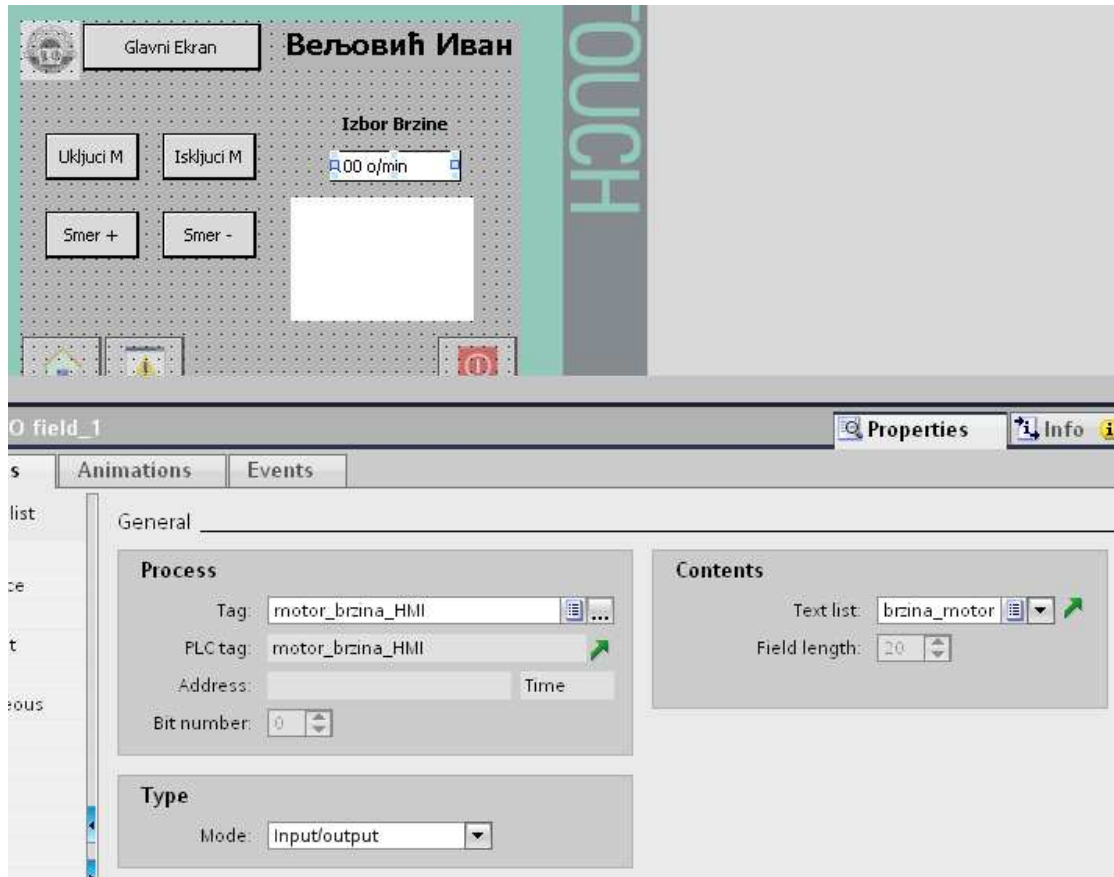
Даље се објектом **Text field** направи текстуални објекат '**Brzina Motora**'. Испод њега, што се види са слике 33, је направљена **Symbolic I/O field** падајућа листа за избор брзине. Даље се у **Project tree-u**, са леве стране, двокликом миша одабере **Text and graphic list** у којој ћемо направити листу '**brzina_motora**'.

Text lists		
	Name ▲	Selection
	brzina_motora	Value/Range
	slike_motora	Value/Range
	stanje_PID	Value/Range
	stanje_PID_rucno	Value/Range
	TextList_ScreenNames	Value/Range
	<Add new>	

Text list entries			
	Default	Value ▲	Text
	☐	3	100 o/min
	☐	28	10 o/min
	☐	1000	0.3 o/min
		<Add new>	

Слика 33 Пример листе и њених вредности

Са слике 33 се види да ова листа има три вредности, редом, од 3, 28 и 1000 милсекунди. Поред сваке вредности је одговарајуће име, 100 o/min, 10 o/min и 0.3 o/min. Ова имена ће бити приказана у падајућој листи док ће се вредности прослеђивати PLC-у.



Слика 34 Повезивање поља за избор вредности и листе у меморији HMI-a

Са слике 34 се види да је ово *input* поље повезано са листом *brzina_motora* и да се његова вредност уписује у *motor_brzina_HMI* који се налази у PLC-у. На тај начин се врши промена брзине мотора јер овај таг се користи као улаз за тајмер који је коришћен у дијаграму PLC-a за управљање мотором.

3.8. Примена TIA11 у програмирању PLC/HMI ради управљања нивоа воде у резервоару

3.8.1. Повезивање уређаја

О повезивању опреме је већ било речи у поглављу 2.8. Сада, у контексту програмирања уређаја: у рачунарску мрежу је додат **PLC** и додељена му је **IP** адреса **134.91.203.132**. За ово управљање је коришћен исти **HMI** као у претходном примеру. Дефиницијом рада је тако предвиђено да један **HMI** уређај управља са два различита процеса.

Аналогни излаз са **PLC**-а је доведен на појачавач снаге, одакле је повезан на пумпу за воду која утиче на ниво воде у резервоару.

Сигнал са сензора који мери висину воде у резервоару се доводи на аналогни улаз **PLC**-а.

3.8.2 Програмирање PLC-а

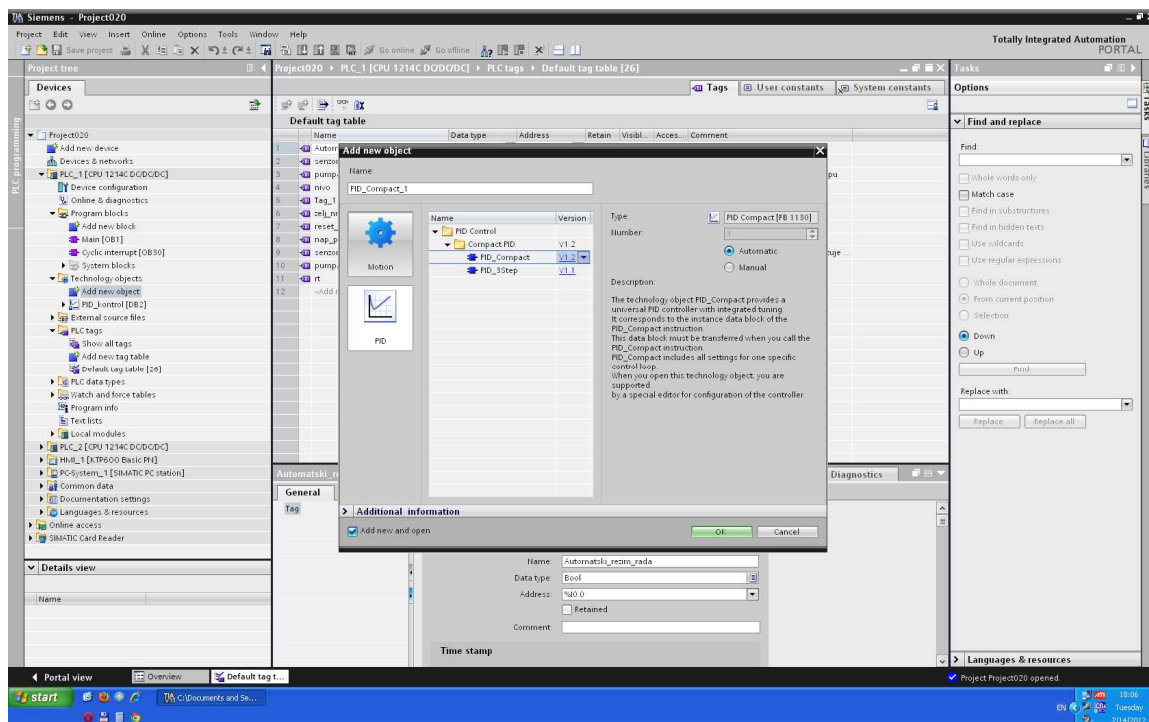
На почетку је прво потребно направити тагове који ће се користити. У *Project tree* са два клика на *Default tag table* се отвара табела и потребно је направити следеће тагове: *'senzor_nivo'* типа *word* на адреси *IW64* и *'pumpa'* типа *word* на адреси *QW80*. први таг је променљива у коју смештамо аналогну вредност напона коју добијамо са сензора из резервоара, док је друга променљива вредност напона који се шаље аналогни излаз.

Default tag table							
	Name	Data type	Address	Retain	Visibl...	Acces...	Comment
1	senzor_nivo	Word	%IW64	☐	☒	☒	vrednost koja se dobija sa senzora
2	pumpa	Word	%QW80	☐	☒	☒	vrednost koja se prosledjuje pumpi
3	zeljeni_nivo	Real	%MD60	☐	☒	☒	zeljeni nivo koji se unosi sa HMI i prosledjuje PID regulatoru
4	pumpa_konvert	Real	%MD70	☐	☒	☒	konvertovana vrednost pumpe koja se skalira i prikazuje na HMI
5	pumpa_konvert_sa_HMI	Real	%MD80	☐	☒	☒	vrednost koja je konvertovana i normalizovana i prosledjuje se HMI-u
6	senzor_nivo_konvert	Real	%MD14	☐	☒	☒	konvertovana vrednost senzora koja se skalira za prikaz na HMI
7	senzor_nivo_konvert_HMI	Real	%MD18	☐	☒	☒	konvertovana vrednost senzora koja se prosledjuje na HMI
8	reset_bit_PID	Bool	%M40.0	☐	☒	☒	bit koji se setuje sa HMI-a a služi za resetovanje PID-a
9	pumpa_unos_rucni_mod_sa_HMI	Real	%MD22	☐	☒	☒	
10	pumpa_unos_rucni_saHMI_konvert	Real	%MD26	☐	☒	☒	
11	manual_enable	Bool	%M30.0	☐	☒	☒	bit koji omogućava rucni režim PID-a
12	manuel_value	Real	%MD32	☐	☒	☒	rucna vrednost koja se pise u PID
13	bit_set_RM_sa_HMI	Bool	%M36.0	☐	☒	☒	ovaj bit se setuje kada se predje dugmetom na ekran za rucni mod
14	vidi_nevidi	Bool	%M44.1	☐	☒	☒	bit kojim se vrši blokiranje nekih polja na HMI

Слика 35 Променљиве (тагови) у PLC меморији

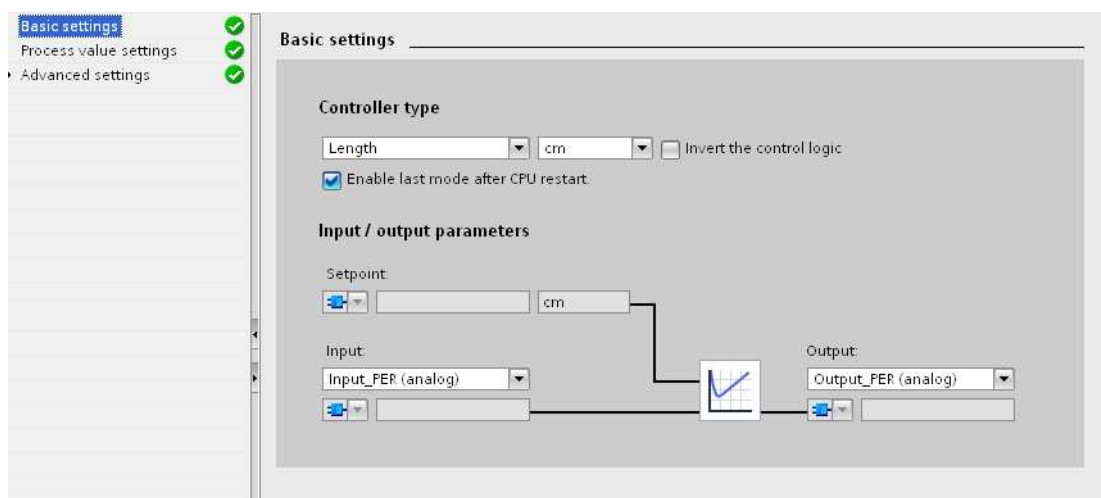
На слици 35 се види да су направљени још неки тагови и о њима ће бити речи касније.

У делу прозора под називом *Project tree* изаберемо опцију *Tehnological Object*, а затим двоструким кликом изаберемо *Add new object*. Отвориће се прозор под називом *Add new object*, притиснемо на иконицу *PID controller*, задамо име новом објекту или оставимо аутоматски генерисан назив. Овај избор је графички приказан сликом 36.



36 Додавање PID регулатора

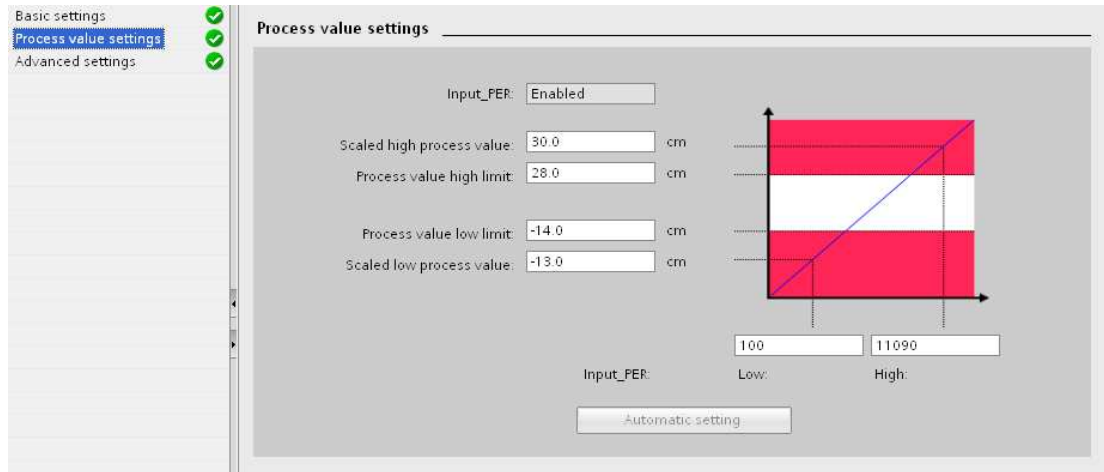
Након тога се подешавају параметри за **PID**.



Слика 37 Конфигурисање основних параметара PID регулатора

У **Basic settings** као мерну јединицу је потребно подесити дужину у центиметрима (**Length, cm**). **Input value** ће бити **Input_PER(analog)** а **Output value** ће бити **Output_PER(analog)**.

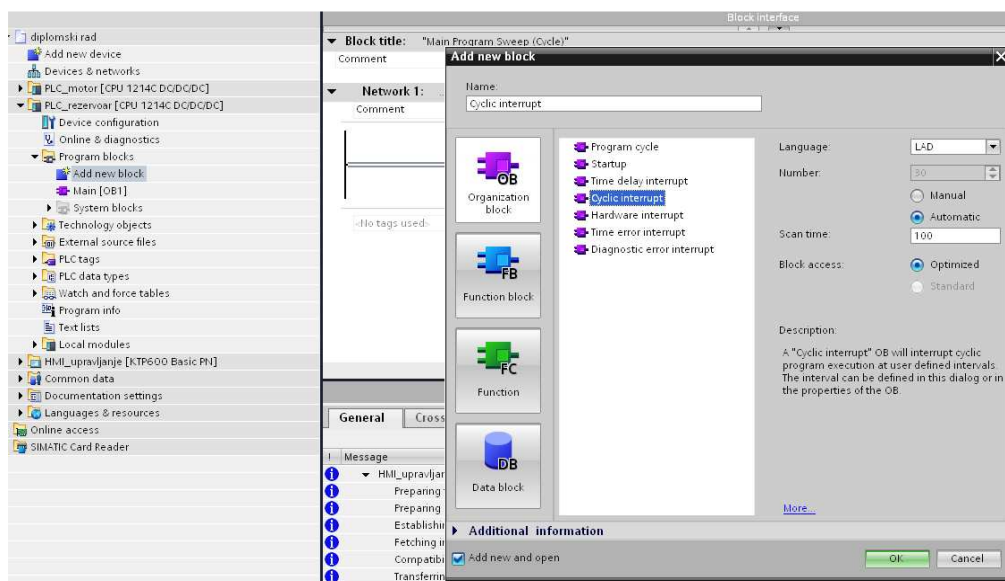
Затим се бира ставка **Process value settings**.



Слика 38 Конфигурација граничних вредности процеса

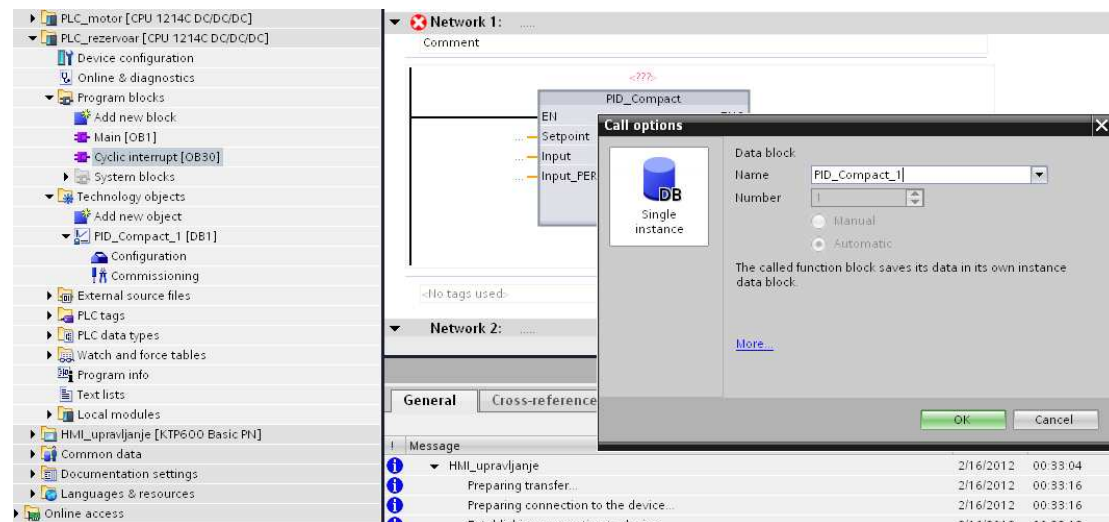
Вредности за **Scaled high process value** и **Process value high limit** уносимо 30.0 и 29cm. Због вредности на сензору, у поља **Process value low limit** и **Scaled low process value** уписујемо вредности -14 и -13cm. Опсег напона сензора је од 0 до 5V. Измерене реалне вредности празног и пуног резервоара одговарају дигиталним вредностима 100 и 11090 и њих уписујемо у поља **Low** и **High**, као што се види на слици 38. У **Advanced settings** се подесе вредности **Warning high limit** и **Warning low limit** на 28cm и -3cm. Тиме су подешавања за **PID** завршена.

Овај **PID** алгоритам треба да сместимо негде. Обично се ставља у прекидну рутину (interrupt) изван главног програма и то на следећи начин. У делу прозора под називом **Project tree** изаберемо опцију **Program block**, а затим двоструким кликом изаберемо **Add new block**. Отвориће се прозор под називом **Add new block**, изаберемо иконицу на којој пише **Organization block (OB)** и у листи са разним врстама блокова изаберемо **Cyclic interrupt. Language** нам је **LAD** – летвичаста дијаграм. Периода одабирања (**Scan time** прекидне рутине) је 100ms. Можемо још дати назив блоку и све потврдимо на **OK**.



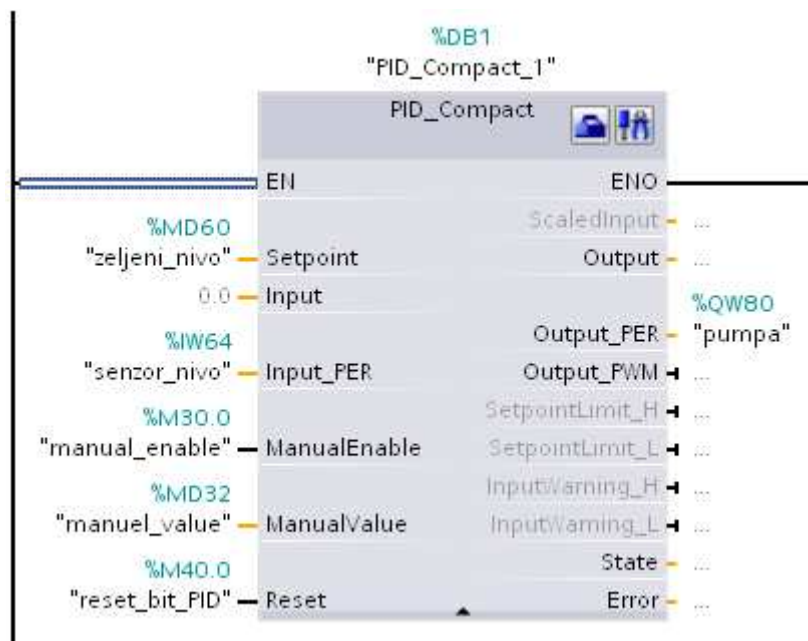
Слика 39 Уметање PID-а у прекидну рутину

Можемо приметити да се у *Program blocks*, поред главног дела програма – *Main[OB1]*, појавио и *Cyclic interrupt[OB30]*. Потребно је да убацимо **PID** алгоритам у интерапт. Отворимо *Cyclic interrupt* и на средини екрана појавиће се **LAD** едитор за писање програма. Са десне стране имамо списак инструкција и под *Technology* изаберемо **PID Control**, па изаберемо *PID_Compact*. Мишем га превучемо и пустимо на грану у делу *Network 1*. Отвориће се прозор *Call options*, где под опцијом *Name* изаберемо *PID_Compact_1*, односно, **PID** објекат који смо направили у *Technological objects* на почетку овог поглавља и потврдимо на **OK**.



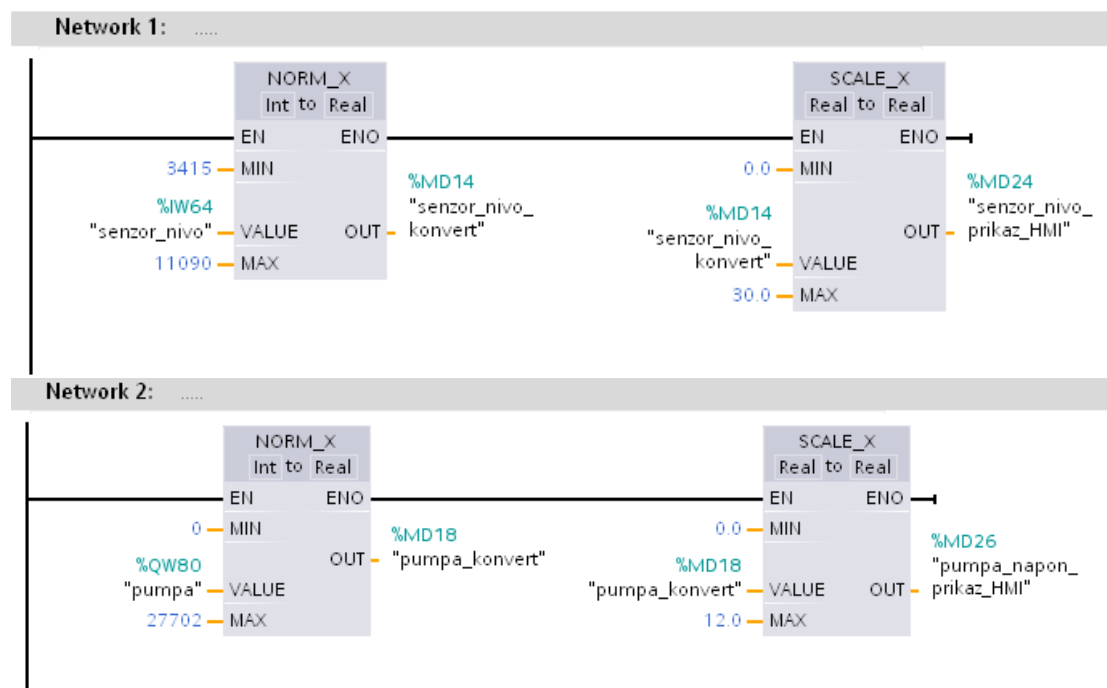
Слика 40 Додавање *PID_Compact* блока

У *Network 1* имамо **PID** инструкцију на којој треба да дефинишемо разне параметре. У *Setpoint* уписујемо *'zeljeni_nivo'*. Вредност ове променљиве се налази у меморији. На ту меморијску локацију се уписује вредност са **HMI**-а, који у свом интерфејсу има поље за унос ове величине. То је вредност на којој ће се одржавати ниво воде у резервоару. Параметри *Input* и *Input_PER* су тренутне вредности које се очитавају на систему. У овом случају то је сензор који очитава тренутну вредност нивоа воде у резервоару и шаље сигнал **PID** инструкцији. Разлика између *Input* и *Input_PER* је у томе што је *Input* тренутна вредност која се задаје у дигиталном облику, а *Input_PER* служи за очитавање тренутне вредности аналогног сигнала. Користиће се *Input_PER* јер сензор шаље аналогни сигнал и у ово поље се уписује *"senzor_nivo"* који је већ тагован, који има адресу **IW64**. Треба напоменути да се никако не сме користити у исто време и *Input* и *Input_PER*. Параметри *Output* и *Output_PER* имају исту аналогију као и *Input* и *Input_PER* само што оне служе за рад са манипулисаним вредностима. Користиће се *Output_PER* јер је он за аналогне сигнале. Уноси се већ таговани излаз *"pumpa"* који има адресу **QW80**. Параметру *Reset* се прослеђује вредност тага *'reset_bit_PID'* који се налази на меморијској локацији **M40.0**. Овај бит се сетује са **HMI**-а и касније ће бити детаљније објашњен.



Слика 41 Изглед PID блока у LAD дијаграму

Аналогне вредности које се добијају са сензора који мери ниво воде у резервоару, као и аналогну вредност напона која покреће пумпу није могуће приказати на **HMI**-у већ се мора извршити нормализација и скалирање ових вредности. Ове операције се уписују у главном програмском блоку (**Main block**).



Слика 42 Нормализација и скалирање вредности у LAD дијаграму

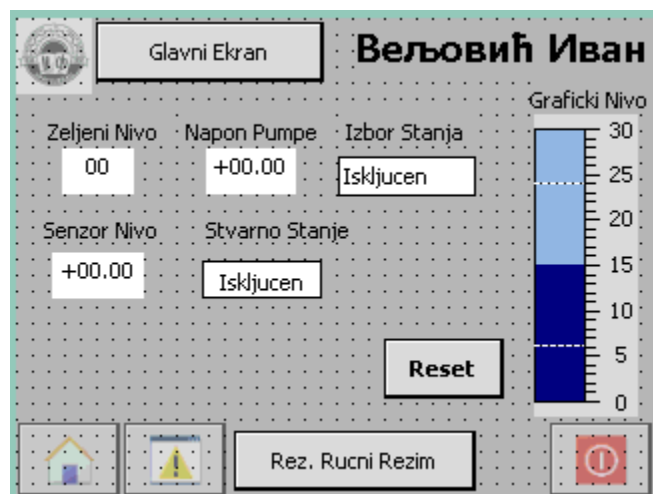
У првој мрежи се нормализује вредност '**senzor_nivo**' чија се вредност креће у границама од 3415 до 11090. Вредност ове операције се уписује у '**senzor_nivo_konvert**'. Ова вредност се даље скалира у опсегу од 0 до 30, што

одговара реалним вредностима висине воде који сензор мери. Та вредност се уписује у *'senzor_nivo_prikaz_HMI'*.

У другој мрежи се нормализује вредност *'pumpa'*. Ова вредност се креће у границама од 0 до 27702. Нормализована вредност се уписује у таг *'pumpa_konvert'* а иста се даље скалира у границама од 0 до 12, што представља реалне вредности напона који се доводи на пумпу, па се уписује у таг *'pumpa_napon_prikaz_HMI'* чија ће вредност бити приказана на **HMI**-у.

3.8.3 Програмирање HMI-а

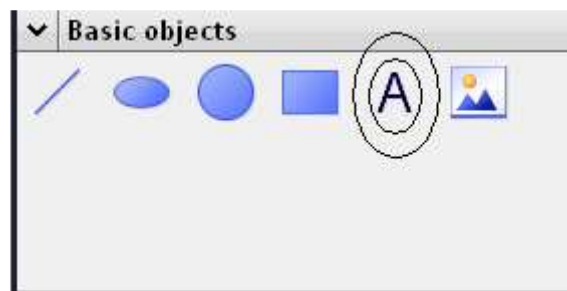
За овај део пројекта се не додаје нови екран већ се користи исти који је коришћен за управљање корачним мотором. Када је обрађиван **HMI** у делу везаном за корачни мотор, наглашено је да су направљена три софтверска екрана од којих је један *Rezervoar*. Сада ће се он користити за управљање резервоаром и компонентама које су укључене у ово управљање. Изглед екрана за ово управљање је дат на следећој слици 43.



Слика 43 HMI екран за аутоматску контролу резервоара

На врху овог екрана се налази дугме *“Glavni Ekran”*. Ово дугме је функцијом *ActivateScreen* повезано са главним екраном и кликом на њега се врши повратак.

Графичким објектом *Text Fild*, који је део главних графичких објеката, су направљена текстуална поља која описују функције поља која се налазе на екрану. Овај алат за текст је приказан на слици 44.



Слика 44 Алат за прављење текста на HMI-у

Поље за унос жељеног нивоа воде у резервоару је *I/O field* објекат. Намена му је да се са **HMI**-а уноси жељени ниво а ту вредност смешта у таг '*zelj_nivo*' а подешавања су представљена сликом 45.

Слика 45 Конфигурација поља за унос жељеног нивоа течности

Naпон pumpe је, слично као и *Zeljeni nivo*, *I/O field* али ово поље има могућност приказивања реалне вредности напона на пумпи без могућности за унос вредности. Вредност за ово поље се добија из тага '*pumpa_konver_za_HMI*' који се налази у **PLC**-у.

Senzor visina је још једно поље које приказује вредност тага ('*senzor_nivo_konvert_HMI*') из **PLC** контролера, али нема могућност уноса величине.

Поља која приказују *Izbor stanja* и *Stvarno stanje* су типа *Symbolic I/O field*. За потребе ових поља је направљена листа која је претстављена на слици 46.

Слика 46 Креирање листе и њених вредности

У *Text and graphic list* опцији, **HMI** уређаја, је направљена *stanje_PID*. Она има две опције, као што се види на претходној слици. Текстуална имена ће бити приказана оператеру а нумеричке вредности се прослеђују **PID** регулатору који

на основу тих вредности контролише процес, искључује се или препушта оператеру да управља процесом.

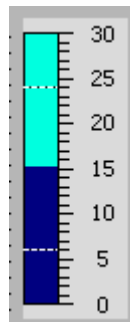
Izlazno stanje је поље које служи за контролу и оно има функцију приказивања тренутног стања **PID** регулатора.

Ова два поља су повезана за функције интерфејса и приказ је дат на слици 47.

Izbor Stanja Iskljucen	General		
	Bit number	0	
	Field length	20	
	Mode	Input/output	
	Process value	0	"PID_Compact_1_sRet.i_Mode"
Stvarno Stanje Iskljucen	Text list	lista_stanja	
	General		
	Bit number	0	
	Field length	20	
	Mode	Input/output	
	Process value	0	PID_Compact_1_State
	Text list	lista_stanja	

Слика 47 Повезивање поља за избор и приказ стања **PID**-а са функцијама интерфејса

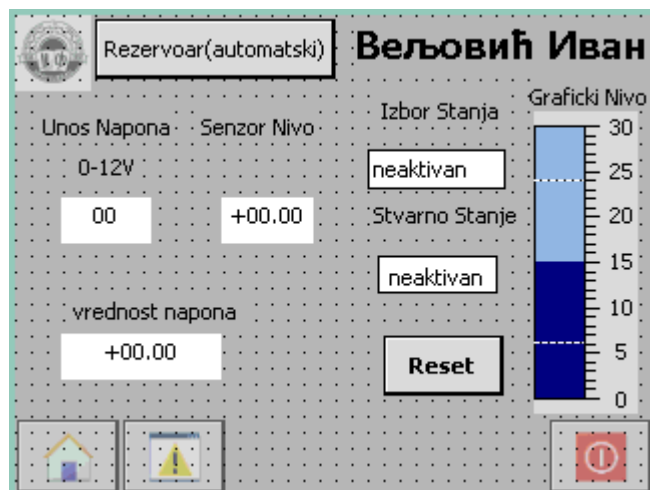
Графички објекат **Bar** има могућност визуелног приказа нивоа воде у резервоару. Границе овог објекта су 0 и 30cm што одговара висини резервоара. Он прима вредност тага **'senzor_nivo_konver_HMI'** са **PLC** контролера и ту вредност графички приказује.



Слика 48 Графички приказ вредности барометром

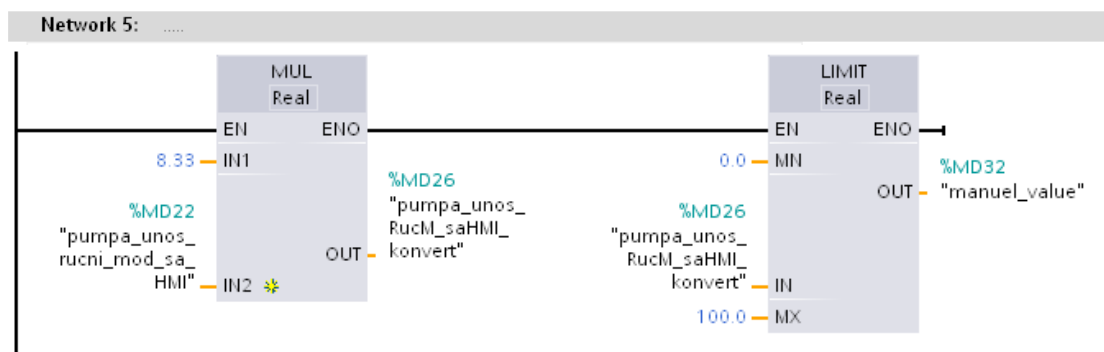
У колико дође до проблема у раду **PID** -а, проблем се решава ресетовањем. Дугме **'Reset'**, када се притисне, сетује бит **M40.0** у меморији **PLC**-а. Вредност овог бита се прослеђује **PID** регулатору на његовом **Reset** улазу. На тај начин **PID** регулатор буде ресетован и започиње поново анализу параметара пред започињање управљања.

Са овог екрана се кликом на дугме **'Rez. Rucni Rezim'** прелази на екран који има могућност ручне контроле пумпе чији је изглед дат на слици 49.



Слика 49 Екран HMI-а за ручну контролу резервоара

Новина на овом екрану је поље за унос вредности напона, којим се ручно управља пумпом. Вредност се из овог поља, које се види на слици, преноси у таг *'pumpa_unos_rucni_mod_sa_HMI'* који се даље преноси и лимитира у процентуалној вредности од 0 до 100% што је приказано сликом 50.



Слика 50 Грана LAD дијаграма за множење и лимитирање вредности напона за ручну контролу пумпе

Дакле, два различита PLC-а су повезана са HMI уређајем и својим објектима управљања у пољу аутоматизације. Остаје повезивање са хијерархијским нивоом изнад – персоналним рачунаром, уз реализацију преосталих задатих функционалности инжењерске апликације коју решава, при чему се практично реализује SCADA систем [7].

4. SCADA и једна њена TIA11 имплементација

4.1 Дефиниција SCADA система

SCADA је акроним од Supervisory Control And Data Acquisition (Прикупљање података, надзор, праћење и управљање) – значи систем за аквизицију података и надзорно управљање – и представља комбинацију система телеметрије и даљинског управљања, односно, састоји се од система за аквизицију података, система за пренос података до централног рачунара и система за приказ и анализу прикупљених података [7].

У [7] је дато по главље о **SCADA системима** (које се овде неће препричавати изнова). Често, мада погрешно, **SCADA се** поистовећује са крајњим презентационим нивоом софтвера намењеним крајњем кориснику – тј. посматра се као чисто софтверски производ. У стварности **SCADA** систем чини целина од више хардверских и софтверских компоненти. Све те компоненте заједно називамо **SCADA системом** [7].

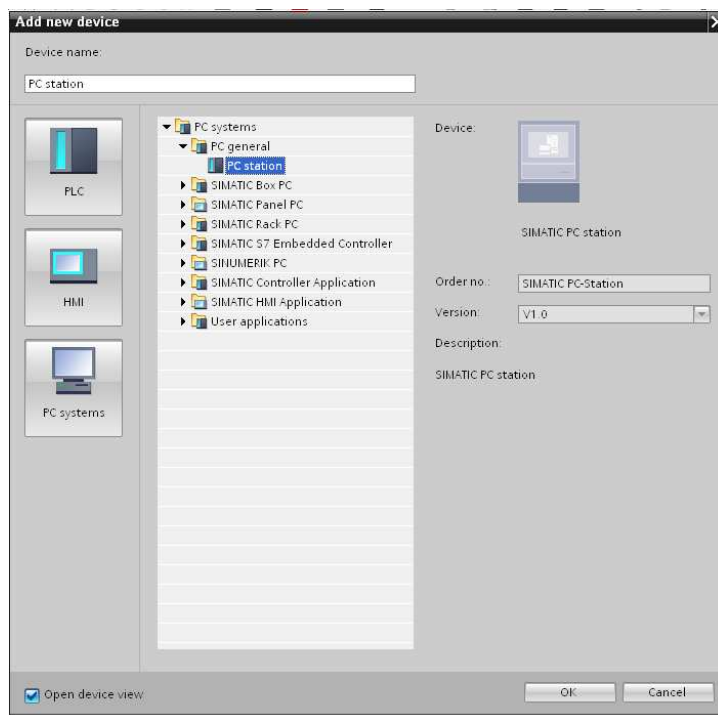


Слика 51. Илустрација једног лабораторијског DCS у SCADA система

4.2 TIA11 и пример имплементације SCADA апликације. PC Station

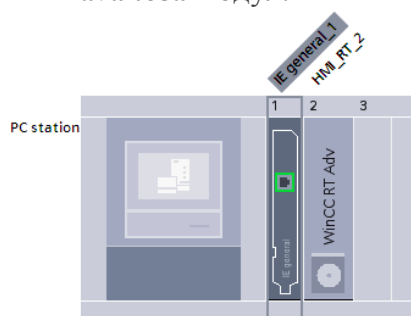
4.2.1 Креирање и подешавање

Прво је потребно додати нови уређај у *Project tree*. Двоклик на *Add new device* па у отвореном прозору изабрати икону *PC Systems* и у листи *PC general* изабрати **PC Station**.



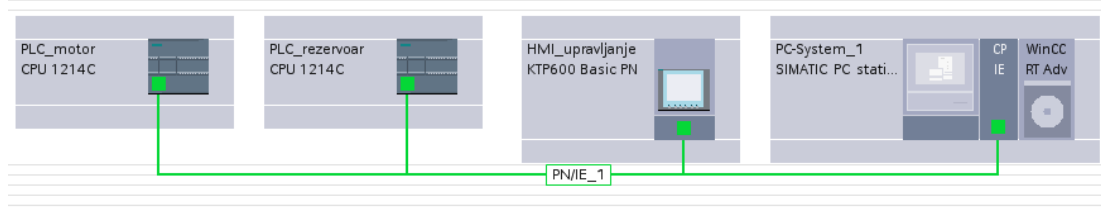
Слика 52 Додавање PC station-а

Из каталожних компонената додати *Communications modules/ PROFINET /Ethernet/IE general* модул. У други слободан слот додати *SIMATIC HMI Application/ WINCC RT Advanced* модул.



Слика 53 Додати модули PC station-у

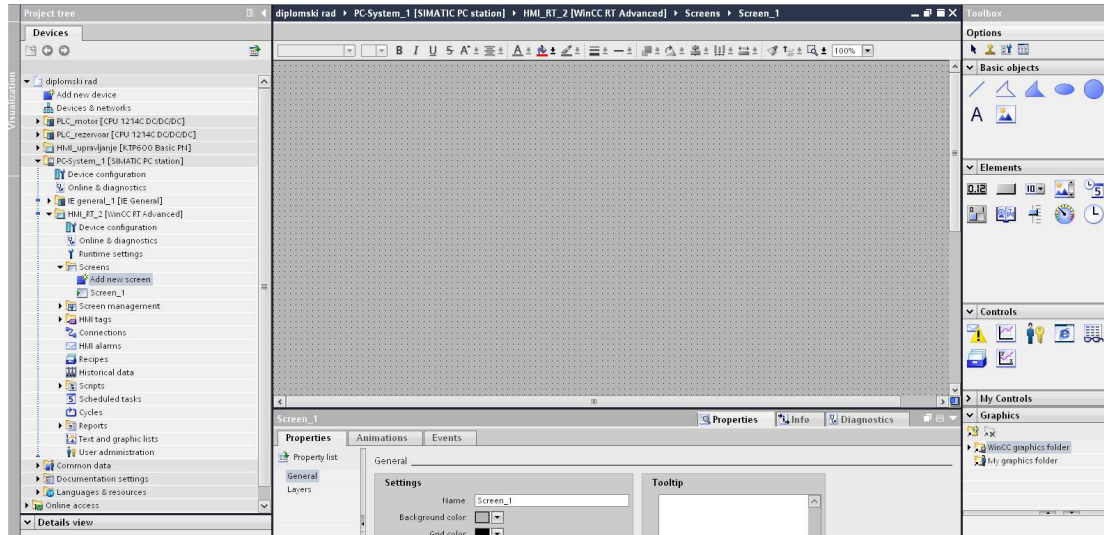
Двоструким кликом на *IE_general_1* модул ће се испод уређаја приказати подешавања за овај модул. Потребно је подесити интерфејс у пољу *Subnet* на *PN/EI_1* и *IP* овог модула (у датом примеру **147.91.203.124**). Сада су оба **PLC**-а, **HMI** и **PC Station** повезани што се најбоље илуструје сликом 54.



Слика 54 Мрежа свих уређаја

4.2.2. Израда HMI екрана на PC Station-y

HMI екран на станици се додаје у *Project tree/ PC-System 1/ HMI RT 2/ Screens/ Add new screen*. Након овога у *Project tree* је додат нови екран а у TIA-и је отворен простор за креирање RT скрина.



Слика 55 Креирање екрана за RunTime

За потребу надгледања мотора је на екрану *PC Station*-а направљен исти графички интерфејс као и на HMI-у у том пројекту.



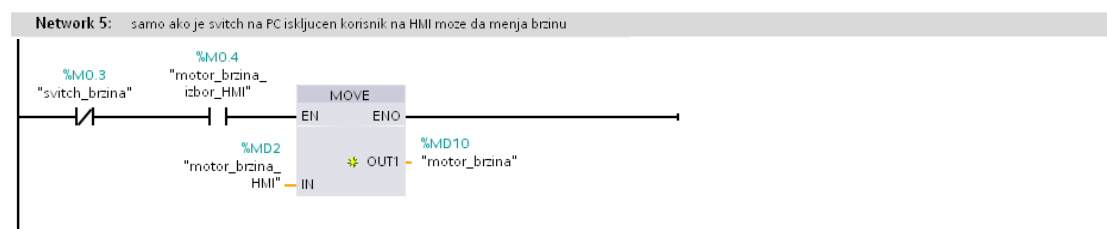
Слика 56 Ф-је конторле мотора на RunTime екрану

Поред ових поља, која су иста као и на HMI, даље се на екрану PC-а може видети *switch* дугме и испод њега је поље за ручни унос брзине. Када се *switch* дугме пребаци у положај 1 тада се брзина мотора користи са поља које се налази испод овог прекидача. На тај начин се онемогућава унос из осталих поља PC-а и HMI-а.

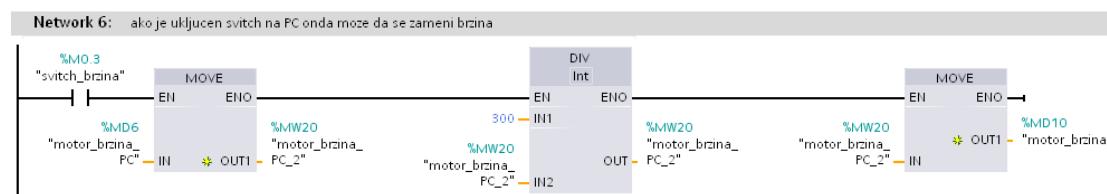


Слика 57 Ручни унос броја обртаја мотора

У PLC-у су направљене две гране. У грани 5 се види да када се овај прекидач активира онда није могуће уносити брзину мотора преко осталих поља. У грани 6 се види да се брзина мотора која се уноси на екрану рачунара прилагођена помоћу **MOVE** и **DIV** блокова и затим прослеђена гранама које управљају мотором. Графичка илустрација је приказана на сликама 58 и 59.



Слика 58 Прослеђивање вредности брзине мотора са HMI-а у колико није активиран switch на екрану RunTime-а



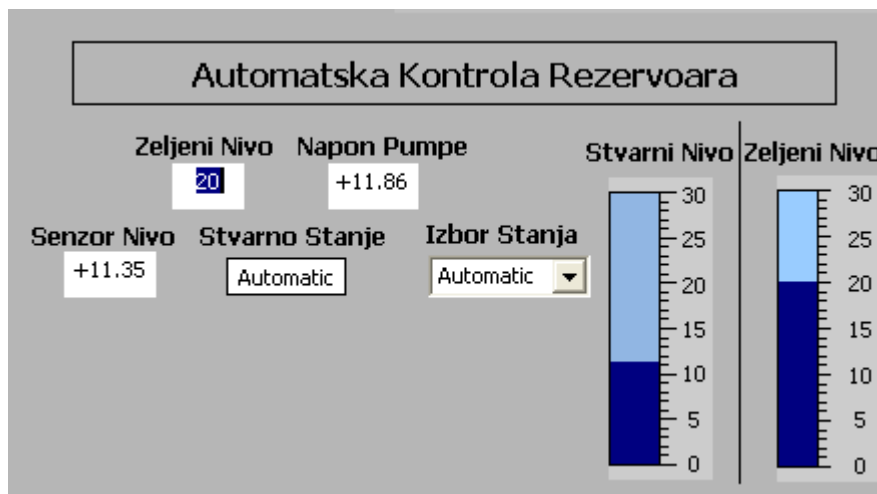
Слика 59 Прослеђивање вредности брзине мотора са RunTime екрана

На екрану РС-а се налази и графичко поље које приказује слику. Када је мотор искључен ово поље приказује чисто белу слику док при покретању мотора појављује се слика која симулира окретање мотора.



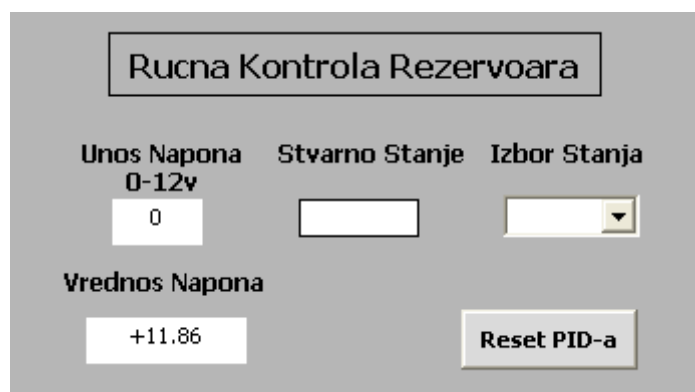
Слика 60 Графички приказ активности мотора на RunTime екрану

За рад са резервоаром је направљена слична апликација као на HMI-у што је приказано следећом сликом 61.



Слика 61 Поља за приказ стања и аутоматску контролу на RunTime екрану

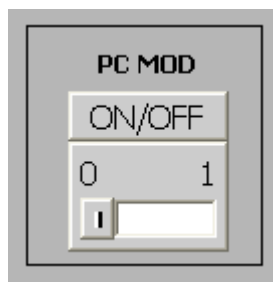
И овде се функције објеката повезују са таговима **PLC**-а који контролише и управља резервоаром.



Слика 62 Поља за ручну контролу резервоара на RunTime екрану

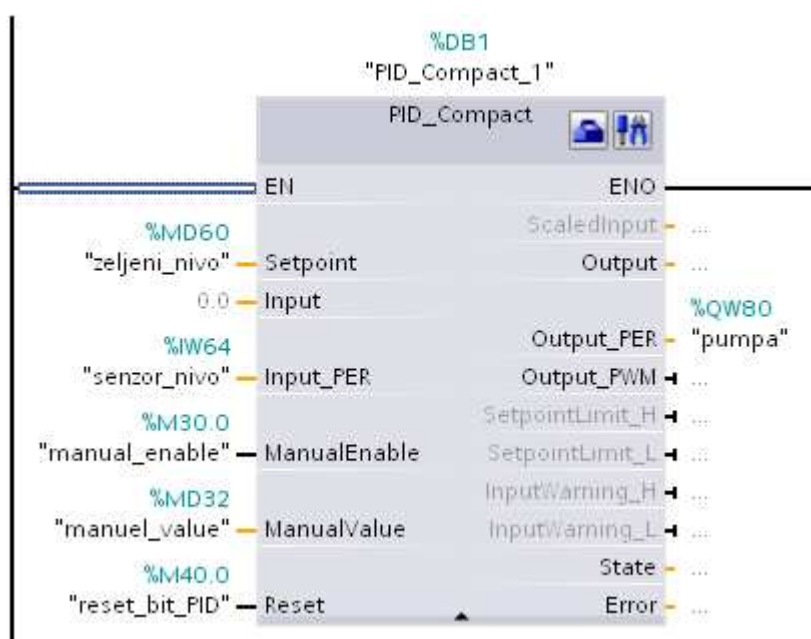
Са претходне слике 62 се виде поља за ручни унос података. Могуће је унети вредност напона, у опсегу од 0 до 12 V, и онда у листи за избор стања одабрати ручни режим. У пољу за стварно стање ће бити приказано стање у којем се **PID** регулатор налази. У колико се појави било какав проблем у раду регулатора, могуће је његово ресетовање на дугмету **Reset PID-a** које је приказано на слици.

При раду са рачунара, у колико корисник жели да онемогући унос података са **HMI**-а из било којег разлога тада се у прекидачу **PC MOD** пребаци ово дугме из позиције 0 у позицију 1. На тај начин ће сва поља за унос података на **HMI**-у да нестану. Сетовањем овог прекидача сетује се таг '*vidi_nevidi*' у меморији контролера. Исти тај таг се проверава са **HMI**-а и када се на тој меморијској локацији појави јединица, помоћу системске функције *visible* ће сва поља, која су обележена, нестати. На тај начин корисник апликације на рачунару има потпуну контролу и спречава свако могуће сетовање са **HMI**-а. Изглед прекидача је дат на следећој слици 63.



Слика 63 Switch дугме за спречавање свих контрола резервоара са HMI екрана

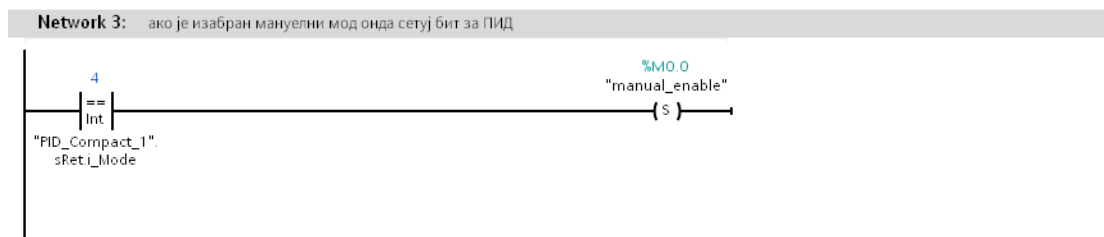
PID контролер има могућност мануелне контроле. Сетовањем његових улаза могу се параметри прослеђивати и ручно.



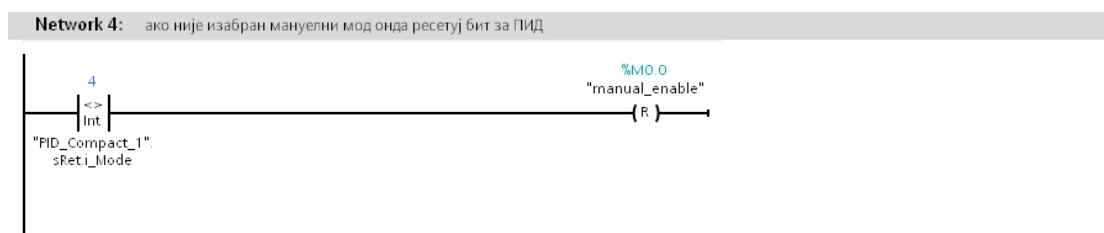
Слика 64 PID блок у LAD дијаграму PLC-а

На улаз **ManualEnable** је прослеђена вредност тага '**manual_enable**'. Овај таг заузима своје место у меморији **PLC**-а а та меморијска локација се сетује и ресетује уколико се на екрану **PC station**-а или **HMI**-а у падајућој листи, којом се бира режим рада регулатора, изабере мануелан режим. Када дође до промене режима рада, **PLC** аутоматски бележи ту промену. Функција која то стање бележи је "**PID_Compact_1**".**sRet_i_Mode**.

У гранама 3 и 4, у главном програму **PLC**-а, се врши поређење вредности ове функције и на основу тога се сетује, или ресетује, бит на локацији тага '**manual_enable**', као што се види на следећим сликама 65 и 66.



Слика 65 Грана LAD дијаграма за сетовање бита на адреси M0.0 у колико је на HMI екрану активираан ручни мод рада PID рутине

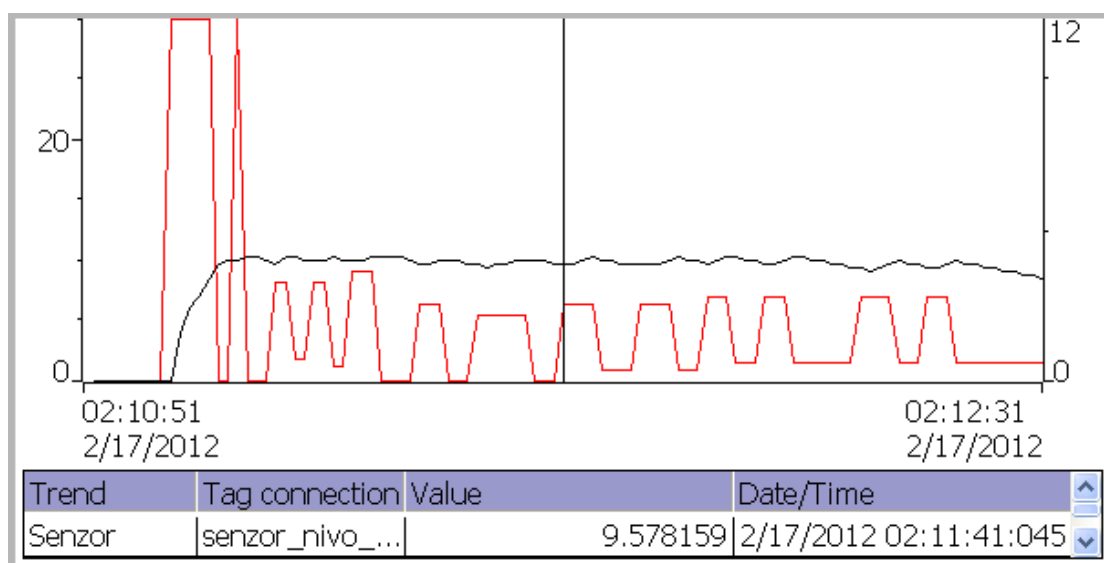


Слика 66 Грана LAD дијаграма за ресетовање бита на адреси M0.0 у колико на HMI екрану није активираан ручни мод рада PID рутине

Када је улаз **'ManualEnable'** активираан, **PID** очекује на свом улазу **'ManualValue'** вредност напона којом се мануелно управља излазним сигналом, односно пумпом на резервоару. **PID** ову вредност једноставно са свог улаза прослеђује на свој излаз, без икакве обраде или појачавања сигнала. Вредност која се доводи на улаз **'ManualValue'** на **PID**-у је вредност тага **'manual_value'**.

Ради бољег и прегледнијег праћења и надгледања овог процеса, на екрану **PC station**-а је додат графички објекат у облику табеле. Овај објекат има могућност приказивања мерених величина у реалном времену.

На слици 67 се са леве стране може уочити скала са опсегом од 0 до 30, а са десне стране скала са опсегом од 0 до 12. Лева скала представља ниво воде у резервоару, док десна представља вредност напона на пумпи.



Слика 67 Дијаграмски приказ напона пумпе и нивоа воде у реалном времену

Црвеном линијом је представљена промена вредности напона а црном ниво воде. На левој страни се може приметити да на почетку **PID** регулатор даје максималних 12V напона пумпи док није достигла жељену вредност од 10cm висине воде. У том тренутку **PID** смањује напон и почиње да шаље импулсне напонске сигнале на пумпу ради одржавања нивоа на жељеној вредности која му је саопштена. У току процеса, **PID** регулатор константно добија вредност од сензора који мери ниво воде и на тај начин сензор, као повратна спрега, враћа вредност из реалног процеса у **PID** регулатор.

4.2.3 Параметри PID регулатора

PID_Compact инструкција пружа **PID** контролер са интегрисаним подешавањима за аутоматски и ручни режим. **PID_Compact** се позива у константним размацима из главног програма. При стартовању CPU-а, **PID_Compact** стартује са параметрима из свог последњег радног мода. **PID_Compact** израчунава вредност излаза према следећем изразу:

$$y = K_p \left[(b \cdot w - x) + \frac{1}{T_i \cdot s} (w - x) + \frac{T_d \cdot s}{a \cdot T_d \cdot s + 1} (c \cdot w - x) \right]$$

y	Вредност излаза
K_p	Коефицијент појачања
s	Лапласов оператор
b	Пропорционални тежински коефицијент
w	Задата вредност
x	Измерена вредност из процеса
T_i	Интегрално време
a	Коефицијент кашњења
c	Тежински коефицијент

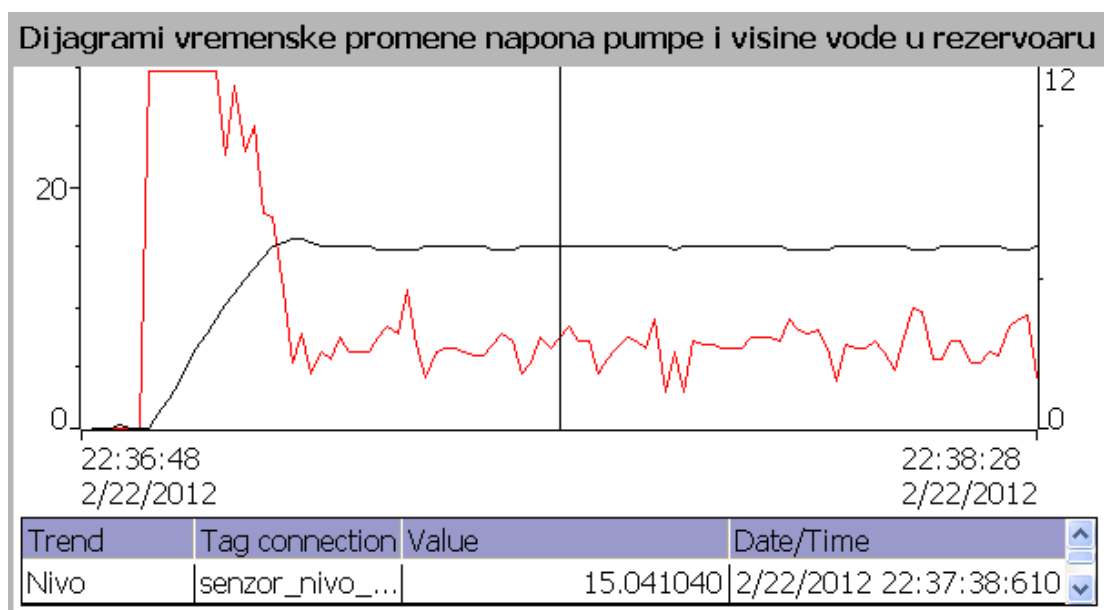
Табела 5 Коефицијенти преносне функције

Вредности ових коефицијената из једног од **PID** процеса су дате на слици 68.

PID Koeficienti		Perioda Odabiranja (Cycle Time)
Ti	2.740	0.100 sec
Td	0.615	
A	0.100	
B	0.577	
C	0.000	
Kp	38.338	

Слика 68 Поља за приказ параметара PID-а на RunTime екрану

За ове вредности коефицијената је добијен дијаграм $u(t)$ и $h(t)$, односно временски дијаграм за напон и висину воде у резервоару, као на слици 69.

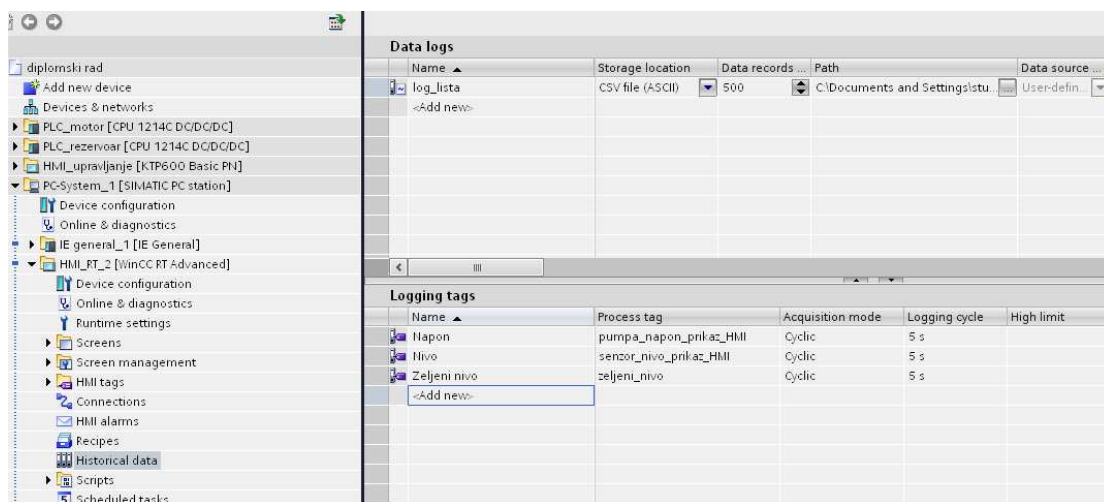


Слика 69 Дијаграмски приказ напона пумпе и нивоа воде у реалном времену

Црвена крива представља промену напона, док је црна промена висине воде, у реалном времену. Са дијаграма се види да је вредност напона максимална док се висина воде не приближи задатој вредности. Након тога **PID** почиње да смањује напон који прослеђује пумпи. Када дође у устаљено стање онда **PID** регулатор одржава ниво воде, у конкретном примеру на 15cm, мењајући незнатно напон око израчунате вредности, у овом случају око 2.1V.

4.2.4 Прикупљање и меморисање података

У *Project tree*, у оквиру *PC station-a*, постоји опција *Historical data*. Ова опција је намењена бележењу и меморисању жељених података из процеса, прављењу евиденције и log фајлова. После двоклика на *Historical data* отвара се картица *Data logs*.



Слика 70 Прављење фајла и одабир вредности које ће се чувати

Са слике 70 се види да је у **Data logs**-у за потребу овог пројекта направљена листа **log_lista**. У опцији **Storage location** постоји могућност снимања података у CSV file (тип података који је погодан за преглед у Office-у), RDB file, Database и TXT file. Поред тога, постоји опција за ограничавање броја записа, избор локације на којој ће фајл бити меморисан, избор да ли ће се записи вршити на одређеним временским интервалима или након неких догађаја. На слици 70 у колони Logging cycle је бирана периода времена са којом се за различите физичке величине врши упис у фајл. У овом случају је то 5 s, а најмања могућа вредност је 1 s. Треба запазити да је изабрана периода одабирања (**Scan time cyclic interrupt-a**) 0.1 s, док је време скен циклуса многоструко мање од 0.1 s. Важно је запазити да се ове три вредности разликују и да имају различито значење. Периода одабирања је време које дефинише тренутке када се активира прекидна рутина у којој је испрограмирано рачунање алогиртма управљања (**PID** регулатора у конкретном случају). Време скенирања је време које дефинише тренутке читања свих улаза, њихове употребе у програмском коду, и на крају, ажурирање свих излаза. Време (Logging cycle) које дефинише упис у базу или фајл би идеално требало да буде изједначено са дефинисаном периодом одабирања. Међутим, оно не може бити мање од 1 s, што је условљено ограничењима која су везана за комуникацију између централног рачунара и уређаја у нижим хијерархијским нивоима (**HMI, PLC**).

У **Logging tags**-у су направљена три тага чије ће вредност и бити меморисане а то су **Napon(pumpa_napon_prikaz_HMI)**, **Nivo(senzor_nivo_prikaz_HMI)**, **Zeljeni nivo(zeljeni_nivo)**. За тагове постоје подешавања везана за праћење и бележење. Могуће је бележити стања у случају промене вредности тага или бележити стање на одређеним временским размацима.

На следећој слици дат је део одговарајућег **Log** фајла за променљиве које су приказане на претходниј слици.

1	VarName	TimeString	VarValue	Validity	Time_ms
2	pumpa_napon_prikaz_HMI	2/16/2012 08:21	1.346834	1	40955347969
3	senzor_nivo_prikaz_HMI	2/16/2012 08:21	14.76557	1	40955347969
4	zeljeni_nivo	2/16/2012 08:21	13	1	40955347969
5	pumpa_napon_prikaz_HMI	2/16/2012 08:21	1.434193	1	40955348027
6	senzor_nivo_prikaz_HMI	2/16/2012 08:21	14.76556	1	40955348027
7	zeljeni_nivo	2/16/2012 08:21	13	1	40955348027
8	pumpa_napon_prikaz_HMI	2/16/2012 08:21	1.786875	1	40955348085
9	senzor_nivo_prikaz_HMI	2/16/2012 08:21	14.70303	1	40955348085
10	zeljeni_nivo	2/16/2012 08:21	13	1	40955348085
11	pumpa_napon_prikaz_HMI	2/16/2012 08:21	0	1	40955348143
12	senzor_nivo_prikaz_HMI	2/16/2012 08:21	14.375	1	40955348143
13	zeljeni_nivo	2/16/2012 08:21	13	1	40955348143
14	pumpa_napon_prikaz_HMI	2/16/2012 08:21	0	1	40955348201
15	senzor_nivo_prikaz_HMI	2/16/2012 08:21	13.625	1	40955348201
16	zeljeni_nivo	2/16/2012 08:21	13	1	40955348201
17	pumpa_napon_prikaz_HMI	2/16/2012 08:21	0	1	40955348260
18	senzor_nivo_prikaz_HMI	2/16/2012 08:21	12.75	1	40955348260
19	zeljeni_nivo	2/16/2012 08:21	13	1	40955348260

Слика 71 Изглед дела фајла у којем су прикупљени подаци

5. Закључак

Описана је структура система коју чини персонални рачунар на коме је инсталиран софтвер **Siemens TIA11**, 1 **HMI** уређај, 2 **PLC Siemens S7-1200** програмабилна логичка контролера, од којих је један својим дигиталним излазима повезан са корачним мотором, а други са лабораторијским моделом 4Т система резервоара са водом. Дате су основне карактеристике сваког од елемената система и објашњено је како су елементи система повезани и како комуницирају међусобно. Објашњена је употреба Siemens-овог **TIA11** софтвера за програмирање **PLC**-а, **HMI**-а и система у целини. Сви елементи система су повезани Ethernet-ом.

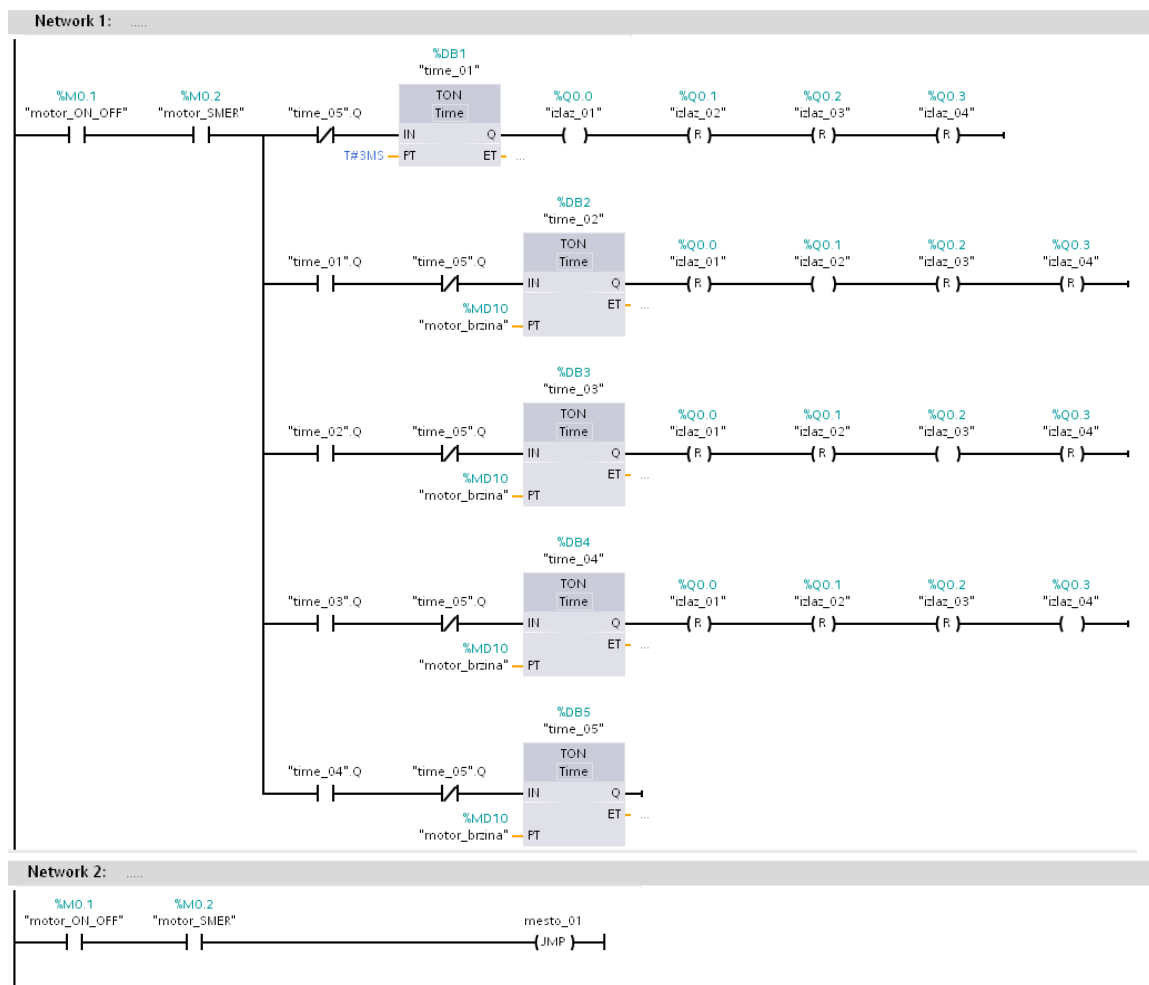
За контролу нивоа течности у резервоару имплементиран је **PID** закон управљања у interrupt (прекидној) рутини **LAD** програма **PLC**-а, а која се извршава са дефинисаном периодом одабирања $T=100[ms]$. Периода одабирања је дефинисан параметар програма који је имплементиран на **PLC**-у, док параметри **PID** алгоритма могу бити мењани са екрана персоналног рачунара. **HMI** и **PC** су омогућили визуелизацију процесних променљивих (надзор), задавање параметара управљања, и разне опције интеракције са оператером/корисником на различитим хијерархијским нивоима, а то укључује опције пуштања у рад, избор радних режима, заустављање процеса, промену параметара регулатора, упис процесних променљивих у фајл или базу. Извршени су експерименти који су поредили ефикасност различитих режима рада система и параметара управљања. Апликација је у свему тестирана и од стране корисника који нису учествовали у њеном програмирању.

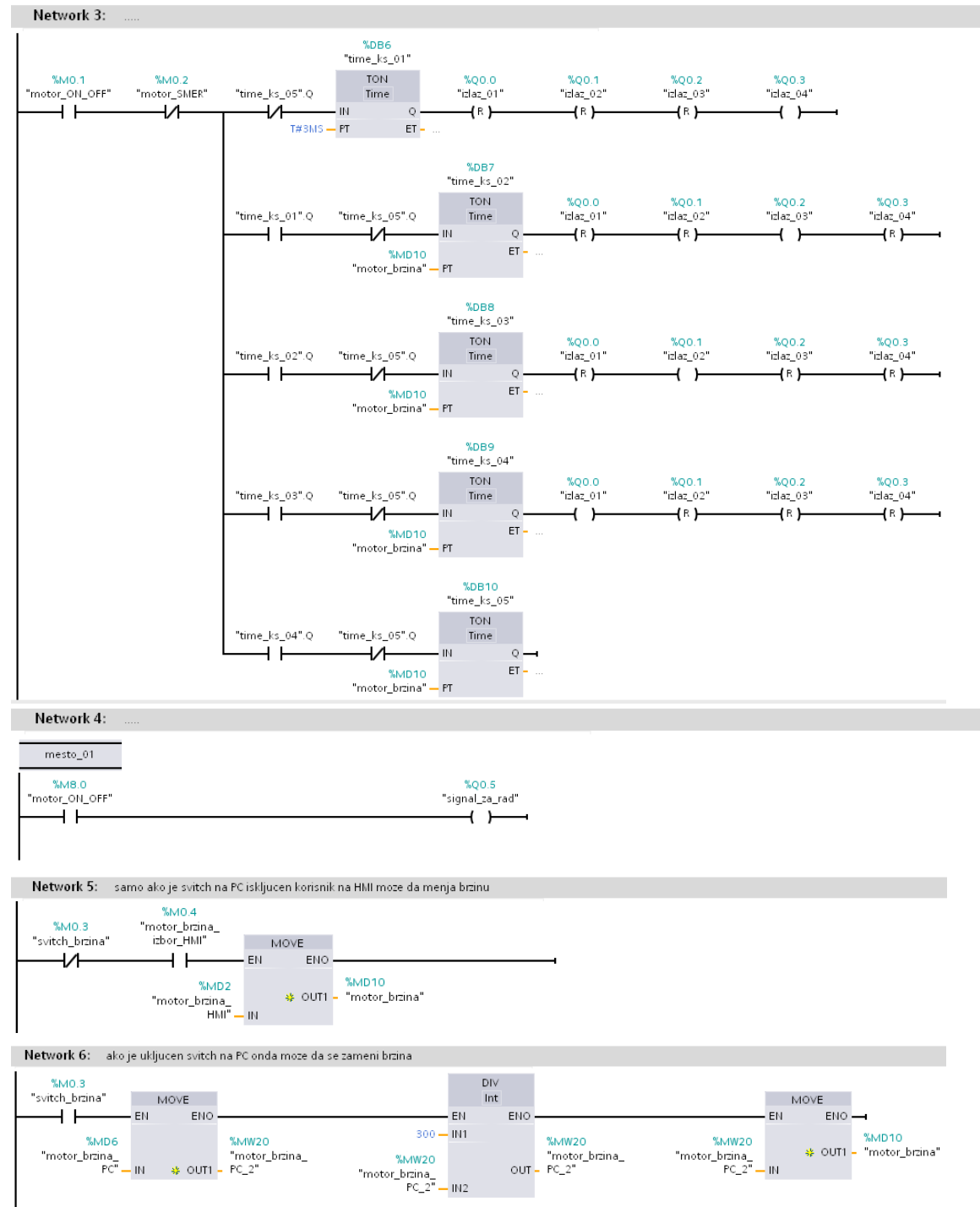
Иако се ради о лабораторијској апликацији и експериментисању, све описано је могуће имплементирати и у индустријским условима.

Поглавља рада су писана стилем који омогућава поновољивост апликације и резултата рада. Отуда, овај рад може бити користан за едукацијске сврхе.

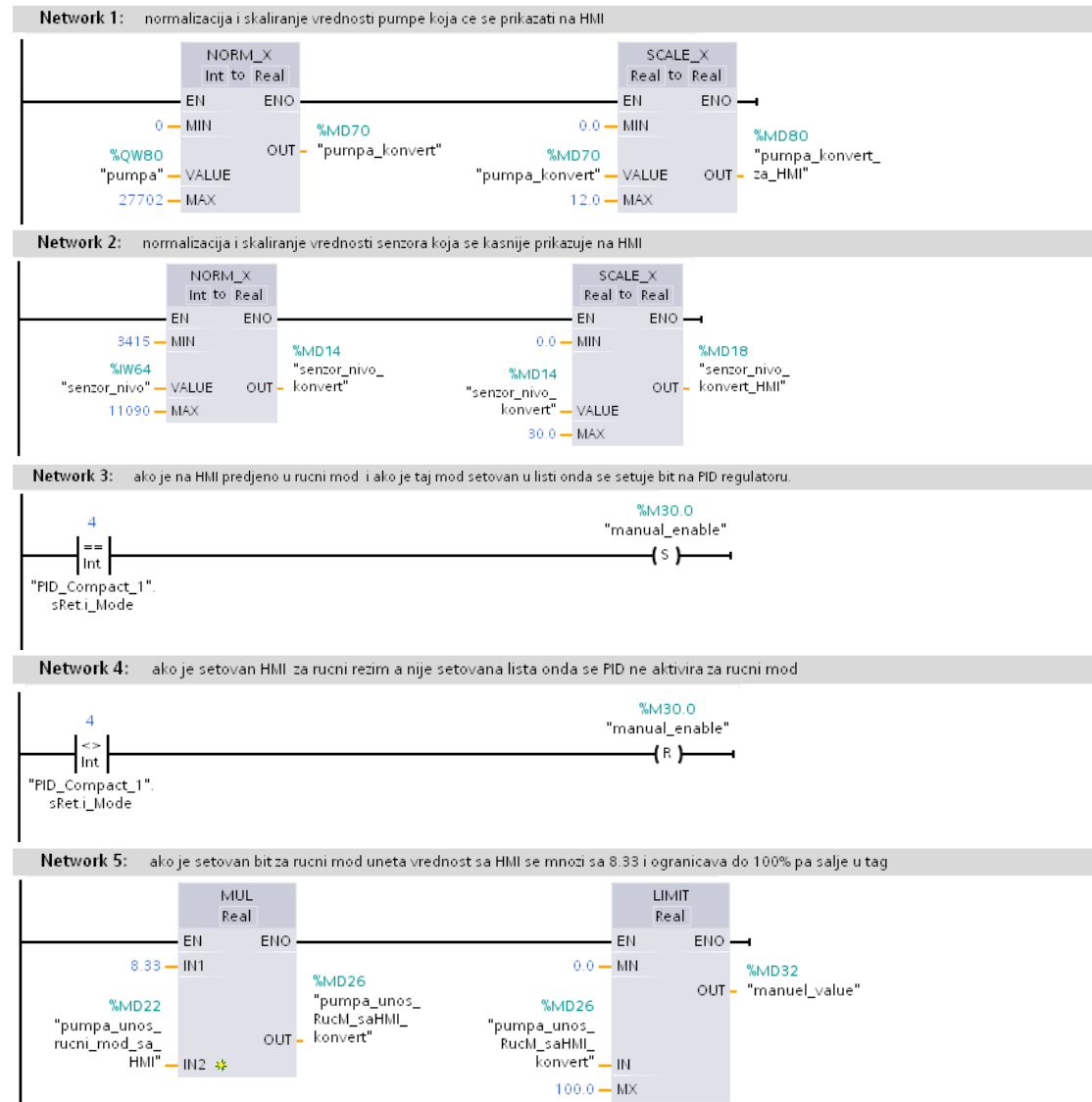
6. Додаци

6.1 LAD дијаграм PLC-а за контролу мотора

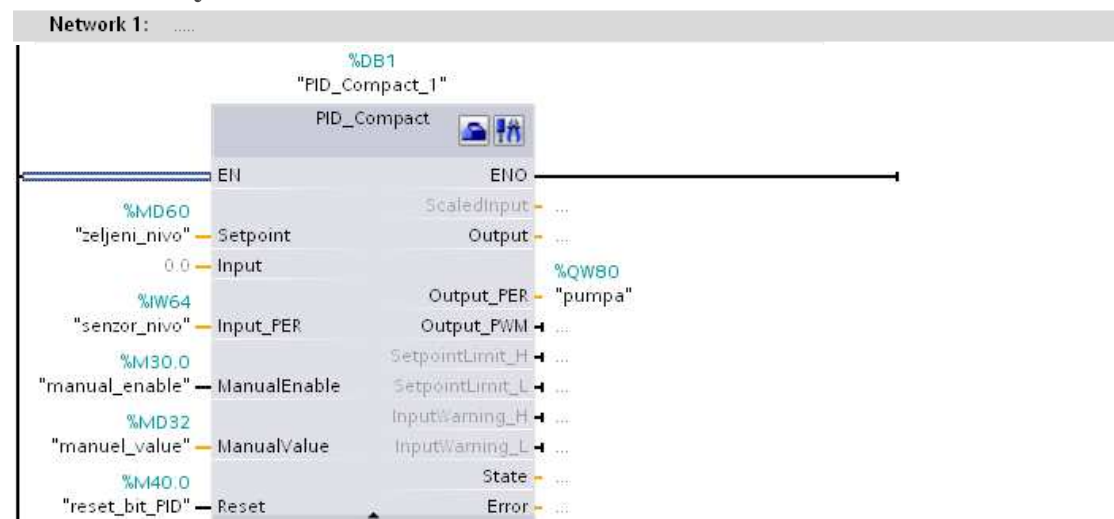




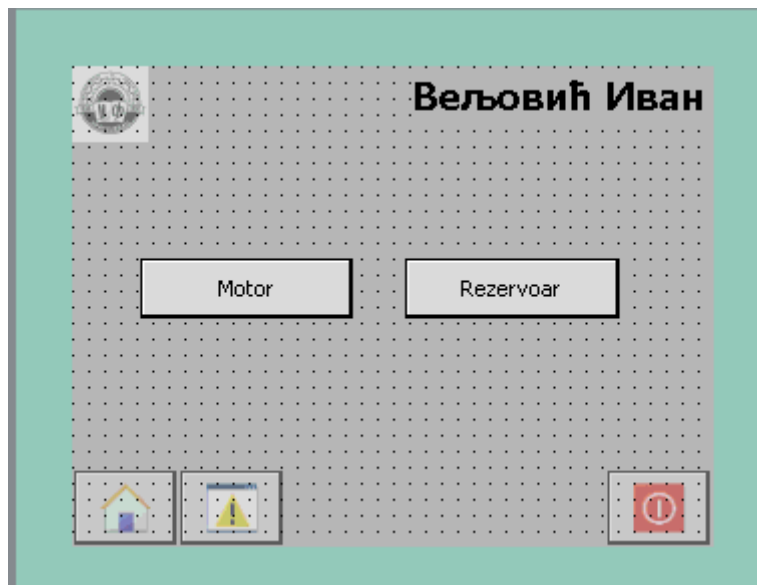
6.2 LAD дијаграм PLC-а за контролу резервоара



6.2.1 PID Рутина

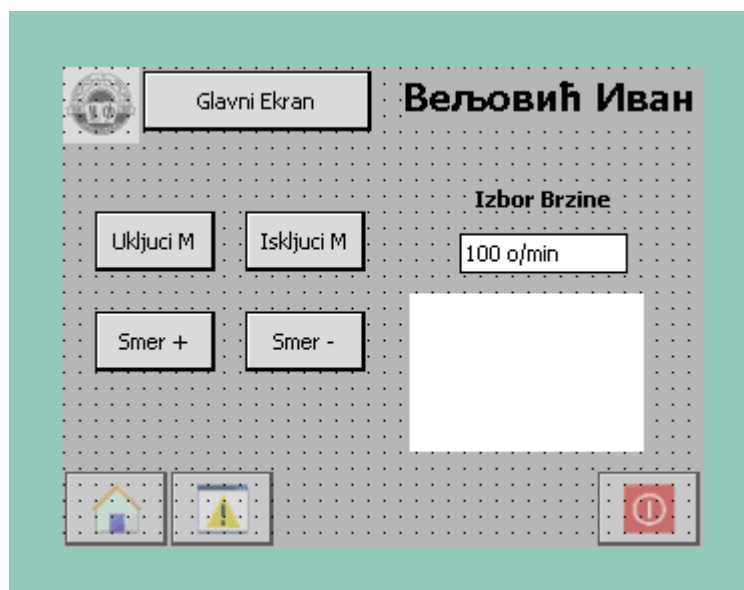


6.3 Изглед главног НМІ екрана



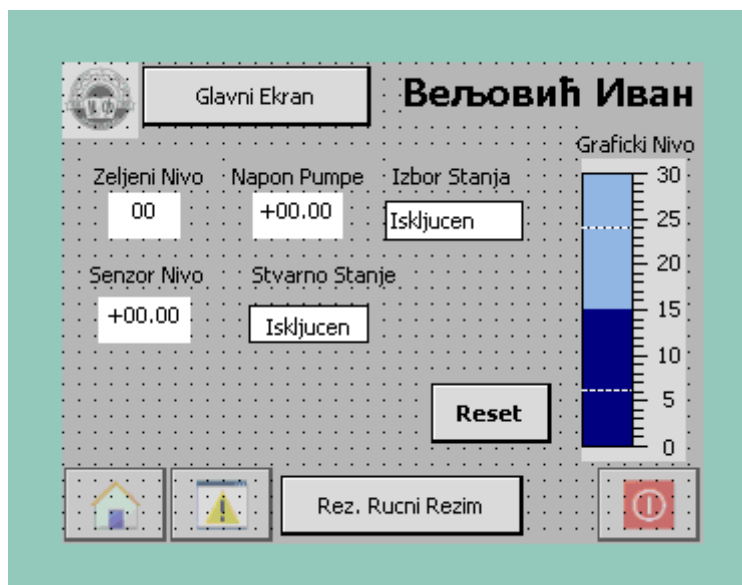
Слика Д1. Главни НМІ екран

6.4 Изглед НМІ екрана за контролу мотора



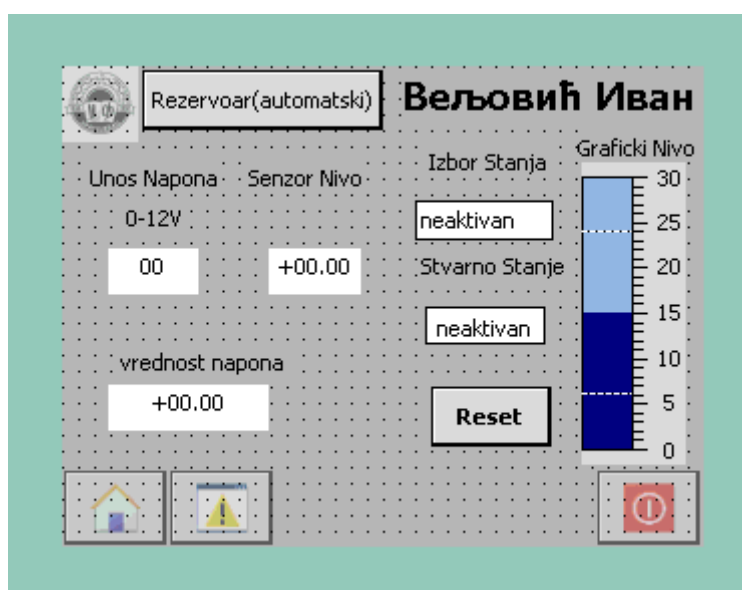
Слика Д2. НМІ екран за контролу мотора

6.5 Изглед НМІ екрана за аутоматску контролу резервоара



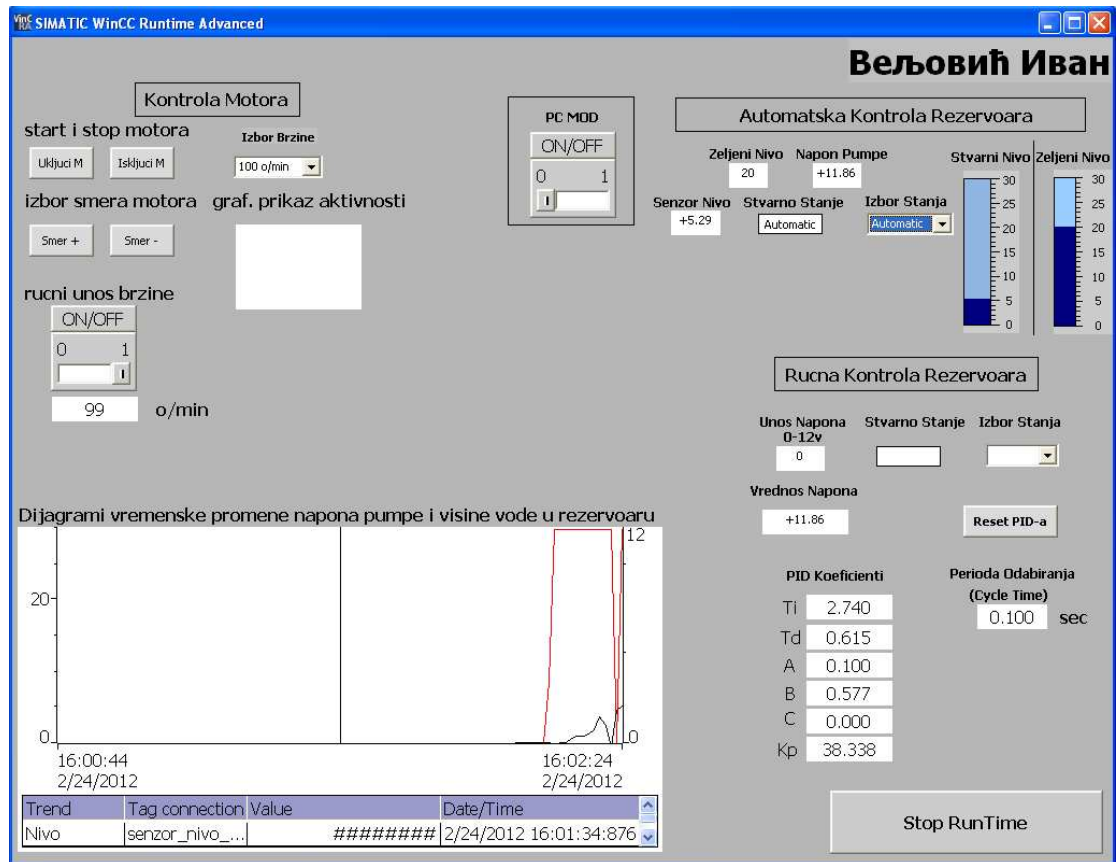
Слика Д3. НМІ екран за аутоматску контролу резервоара

6.6 Изглед НМІ екрана за ручну контролу резервоара



Слика Д4. НМІ екран за ручну контролу резервоара

6.7 Изглед RunTime екрана на PC Station-у



Слика Д5. RunTime екран на PC Station-у

6.8 Меморијска мапа PLC-а

Operand area	Description	Access via units of the following size:	S7 notation
Process image output	The CPU writes the values from the process image output table to the output modules at the start of the cycle.	Output (bit)	Q
		Output byte	QB
		Output word	QW
		Output double word	QD
Process image input	The CPU reads the inputs from the input modules and saves the values to the process image input table at the start of the cycle.	Input (bit)	I
		Input byte	IB
		Input word	IW
		Input double word	ID
Bit memory	This area provides storage for intermediate results calculated in the program.	Bit memory (bit)	M
		Memory byte	MB
		Memory word	MW
		Memory double word	MD
Data block	Data blocks store information for the program. They can either be defined so that all code blocks can access them (global DBs) or assigned to a specific FB or SFB (instance DB). Requirement: The block attribute "Symbolic access only" is not enabled.	Data bit	DBX
		Data byte	DBB
		Data word	DBW
		Data double word	DBD
Local data	This area contains the temporary data of a block while the block is being processed. Requirement: The block attribute "Symbolic access only" is not enabled. Recommendation: Only access local data (temp) symbolically.	Local data bit	L
		Local data byte	LB
		Local data word	LW
		Local data double word	LD
I/O input area	The I/O input and output areas permit direct access to central and distributed input and output modules.	I/O input bit	<tag>:P
		I/O input byte	
		I/O input word	
		I/O input double word	
I/O output area		I/O output bit	
		I/O output byte	
		I/O output word	
		I/O output double word	

Слика Д6. Табела преузета из [8]

6.9 Техничке карактеристике PLC-а

Technical Specifications			
Model	CPU 1214C AC/DC/Relay	CPU 1214C DC/DC/Relay	CPU 1214C DC/DC/DC
Order number (MLFB)	6ES7 214-1BE30-0XB0	6ES7 214-1HE30-0XB0	6ES7 214-1AE30-0XB0
General			
Dimensions W x H x D (mm)	110 x 100 x 75		
Weight	475 grams	435 grams	415 grams
Power dissipation	14 W	12 W	
Current available (SM and CM bus)	1600 mA max. (5 VDC)		
Current available (24 VDC)	400 mA max. (sensor power)		
Digital input current consumption (24VDC)	4 mA/input used		
CPU Features			
User memory	50 Kbytes Work memory / 2 Mbytes Load memory/ 2 Kbytes Retentive memory		
On-board digital I/O	14 inputs/10 outputs		
On-board analog I/O	2 inputs		
Process image size	1024 bytes of inputs/1024 bytes of outputs		
Signal modules expansion	8 SMs max.		
Signal board expansion	1 SB max.		
Communication module expansion	3 CMs max.		
High-speed counters	6 total Single phase: 3 at 100 kHz and 3 at 30 kHz clock rate Quadrature phase: 3 at 80 kHz and 3 at 20 kHz clock rate		
Pulse outputs	2 at 1Hz pulse rate		2 at 100 kHz pulse rate
Pulse catch inputs	14		
Time delay / cyclic interrupts	4 total with 1 ms resolution		
Edge interrupts	12 rising and 12 falling (14 and 14 with optional signal board)		
Memory card	SIMATIC Memory Card (optional)		
Real time clock accuracy	+/- 60 seconds/month		
Real time clock retention time	10 days typ./6 days min. at 40°C (maintenance-free Super Capacitor)		
Performance			
Boolean execution speed	0.1 µs/instruction		
Move Word execution speed	12 µs/instruction		
Real Math execution speed	18 µs/instruction		
Communication			
Number of ports	1		
Type	Ethernet		
Data rates	10/100 Mb/s		
Isolation (external signal to PLC logic)	Transformer isolated, 1500 VDC		
Cable type	CAT5e shielded		

Слика Д7. Табела преузета из [8]

Technical Specifications			
Model	CPU 1214C AC/DC/Relay	CPU 1214C DC/DC/Relay	CPU 1214C DC/DC/DC
Power supply			
Voltage range	85 to 264 VAC	20.4 to 28.8 VDC	
Line frequency	47 to 63 Hz	--	
Input current CPU only at max. load	100 mA at 120 VAC 50 mA at 240 VAC	500 mA at 24 VDC	
CPU with all expansion accessories at max. load	300 mA at 120 VAC 150 mA at 240 VAC	1500 mA at 24 VDC	
Inrush current (max.)	20 A at 264 VAC	12 A at 28.8 VDC	
Isolation (input power to logic)	1500 VAC	Not isolated	
Hold up time (loss of power)	20 ms at 120 VAC 80 ms at 240 VAC	10 ms at 24 VDC	
Internal fuse, not user replaceable	3 A, 250 V, slow blow		
Sensor power			
Voltage range	20.4 to 28.8 VDC	L+ minus 4 VDC min.	
Output current rating (max.)	400 mA (short circuit protected)		
Maximum ripple noise (<10 MHz)	< 1 V peak to peak	Same as input line	
Isolation (CPU logic to sensor power)	Not isolated		
Digital inputs			
Number of inputs	14		
Type	Sink/Source (IEC Type 1 sink)		
Rated voltage	24 VDC at 4 mA, nominal		
Continuous permissible voltage	30 VDC, max.		
Surge voltage	35 VDC for 0.5 sec.		
Logic 1 signal (min.)	15 VDC at 2.5 mA		
Logic 0 signal (max.)	5 VDC at 1 mA		
Isolation (field side to logic)	500 VAC for 1 minute		
Isolation groups	1		
Filter times	0.2, 0.4, 0.8, 1.6, 3.2, 6.4, and 12.8 ms (selectable in groups of 4)		
HSC clock input rates (max.) (Logic 1 Level = 15 to 26 VDC)	Single phase: 100 KHz (Ia.0 to Ia.5) and 30 KHz (Ia.6 to Ib.5) Quadrature phase: 80 KHz (Ia.0 to Ia.5) and 20 KHz (Ia.6 to Ib.5)		
Number of inputs on simultaneously	14		
Cable length (meters)	500 shielded, 300 unshielded, 50 shielded for HSC inputs		
Analog inputs			
Number of inputs	2		
Type	Voltage (single-ended)		
Range	0 to 10 V		
Full-scale range (data word)	0 to 27648 (Refer to Analog input representation for voltage (Page 310))		
Overshoot range (data word)	27,649 to 32,511 (Refer to Analog input representation for voltage (Page 310))		
Overflow (data word)	32,512 to 32767 (Refer to Analog input representation for voltage (Page 310))		
Resolution	10 bits		
Maximum withstand voltage	35 VDC		

Слика Д8. Табела преузета из [8]

Technical Specifications			
Model	CPU 1214C AC/DC/Relay	CPU 1214C DC/DC/Relay	CPU 1214C DC/DC/DC
Smoothing	None, Weak, Medium, or Strong (refer to Analog input response time (Page 310) for step response times)		
Noise rejection	10, 50, or 60 Hz (refer to Analog input response time (Page 310) for sample rates)		
Impedance	≥100 KΩ		
Isolation (field side to logic)	None		
Accuracy (25°C / 0 to 55°C)	3.0% / 3.5% of full-scale		
Common mode rejection	40 dB, DC to 60 Hz		
Operational signal range	Signal plus common mode voltage must be less than +12 V and greater than -12 V		
Cable length (meters)	10 twisted and shielded		
Digital outputs			
Number of outputs	10		
Type	Relay, dry contact		Solid state - MOSFET
Voltage range	5 to 30 VDC or 5 to 250 VAC		20.4 to 28.8 VDC
Logic 1 signal at max. current	--		20 VDC min.
Logic 0 signal with 10 KΩ load	--		0.1 VDC max.
Current (max.)	2.0 A		0.5 A
Lamp load	30 W DC / 200 W AC		5 W
ON state resistance	0.2 Ω max. when new		0.6 Ω max.
Leakage current per point	--		10 μA max.
Surge current	7 A with contacts closed		8 A for 100 ms max.
Overload protection	No		
Isolation (field side to logic)	1500 VAC for 1 minute (coil to contact) None (coil to logic)		500 VAC for 1 minute
Isolation resistance	100 MΩ min. when new		--
Isolation between open contacts	750 VAC for 1 minute		--
Isolation groups	2		1
Inductive clamp voltage	--		L+ minus 48 VDC, 1 W dissipation
Switching delay (Qa.0 to Qa.3)	10 ms max.		1.0 μs max., off to on 3.0 μs max., on to off
Switching delay (Qa.4 to Qb.1)	10 ms max.		50 μs max., off to on 200 μs max., on to off
Pulse Train Output rate (Qa.0 and Qa.2)	1 Hz max.		100 KHz max.
Lifetime mechanical (no load)	10,000,000 open/close cycles		--
Lifetime contacts at rated load	100,000 open/close cycles		--
Behavior on RUN to STOP	Last value or substitute value (default value 0)		
Number of Outputs On simultaneously	10		
Cable length (meters)	500 shielded, 150 unshielded		

Слика Д9. Табела преузета из [8]

7. Литература

- [1] Милан Матијевић, Владимир Цвјетковић, Весна Ранковић, Миладин Стефановић, *Управљање лабораторијским процесима посредством Интернета*, Машински факултет у Крагујевцу, 2007
- [2] М.Равлић, Д.Лазић, М. Ристановић, М.Стефановић, В. Цвјетковић, Н.Бабајић, *„Документација техничког решења – Лабораторијски модел температурног процеса са кашњењем РТ400 са управљањет и надзором преко Интзернета“*, Машински факултет у Крагујевцу, 2009.
- [3] Б. Стојановић, *Управљање кориачним мотором*, семинарски рад из предмета Пројектовање система аутоматског управљања, Машински факултет у Крагујевцу, 2002.
- [4] Видоје Луковић, *Програмирање и примена PLC контролера Siemens SIMATIC S7-1200*, дипломски рад, Машински факултет у Крагујевцу, 2010.
- [5] Иван Дачић, *Програмабилни логички контролер Siemens SIMATIC S7-200*, семинарски рад из Пројектовања САУ, Машински факултет у Крагујевцу, 2008
- [6] Siemens AG, *„TIA_V11_en-US“*, 2011.
- [7] Милан Матијевић, Горан Јакуповић, Јелена Цар, *Рачунарски подржано мерење и управљање*, универзитетски уџбеник, Машински факултет у Крагујевцу, 2008
- [8] Siemens AG, *„S7-1200_System_Manual_enUS“*, 2009.
- [9] Н. Матић, *„Увод у индустријске PLC контролере“*, Микроелектроника Београд, 2007.
- [10] <http://www.automation.siemens.com/>