# Maketa hidrauličke jedinice

**Zubak, Tomislav**

**Master's thesis / Diplomski rad**

**2017**

*Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj:* **Josip Juraj Strossmayer University of Osijek, Faculty of Electrical Engineering, Computer Science and Information Technology Osijek / Sveučilište Josipa Jurja Strossmayera u Osijeku, Fakultet elektrotehnike, računarstva i informacijskih tehnologija Osijek**

*Permanent link / Trajna poveznica:* https://urn.nsk.hr/urn:nbn:hr:200:839144

*Rights / Prava:* In copyright / Zaštićeno autorskim pravom.

*Download date / Datum preuzimanja:* **2024-03-28**

*Repository / Repozitorij:*

Faculty of Electrical Engineering, Computer Science and Information Technology Osijek

zir.nsk.hr

Image not found or type unknown

**SVEUČILIŠTE JOSIPA JURJA STROSSMAYERA U OSIJEKU**
**Fakultet elektrotehnike, računarstva i informacijskih tehnologija**
**Diplomski studij**

# MAKETA HIDRAULIČKE JEDINICE

**Diplomski rad**

**Tomislav Zubak**

Osijek, 2017.

Sadržaj:

1. Uvod ..... 1
1.1. Zadatak diplomskog rada ..... 3
2. Maketa ..... 4
2.1. Mjerenje procesnih veličina ..... 5
2.1.1. Mjerenje razine fluida analognim i digitalnim senzorima razine ..... 6
2.1.2. Mjerenje protoka fluida ..... 8
2.2. Upravljanje maketom ..... 10
2.2.1. Elektronički sklop za formiranje pulsno-širinski moduliranog signala ..... 11
2.3. Automatska regulacija protoka ..... 14
2.3.1. PID regulator ..... 16
3. Sustavi sa programabilnim logičkim kontrolerima ..... 18
3.1. Programabilni logički kontroleri ..... 18
3.2. Hardverska konfiguracija sustava automatskog vođenja ..... 20
3.2.1. Siemens S7 414-3 DP ..... 21
3.2.2. Siemens ET 200S (IM151-1 HF) ..... 22
3.2.3. Digitalni signalni moduli ..... 22
3.2.4. Analogni signalni modul ..... 23
3.3. IEC 61131-3 standard ..... 25
3.3.1. Elementarni tipovi podataka ..... 25
3.3.2. Programske organizacijske jedinice ..... 26
3.3.3. Programski jezici ..... 26
3.4. Industrijske komunikacijske mreže i protokoli ..... 27
3.4.1. PROFINET ..... 28
3.4.2. PROFIBUS ..... 29
3.5. Vizualizacija sustava automatskog vođenja ..... 31
4. Programski alati i programsko rješenje ..... 32
4.1. Siemens SIMATIC STEP 7 i WinCC ..... 32
4.1.1. Povezivanje računala sa PLC-om ... **Pogreška! Knjižna oznaka nije definirana.**
4.1.2. Dodavanje programskih blokova ..... 35
4.1.2. Mjerenje razine ..... 37
4.1.3. Zadavanje referentnih vrijednosti za upravljanje protokom crpki ..... 39
4.1.4. PID regulator ..... 40
4.1.5. Mjerenje protoka ..... 41
4.1.6. Grafičko sučelje WinCC Advanced ..... 43

4.2. CODESYS ..... 45
4.2.1. Izrada sučelja vizualizacije ..... 46
4.2.2. Programske organizacijske jedinice i globalne varijable ..... 48
Zaključak ..... 49
Popis korištene literature ..... **Pogreška! Knjižna oznaka nije definirana.**
Sažetak ..... 51
Abstract ..... 52
Životopis ..... 53
Prilozi ..... 54

# 1. Uvod

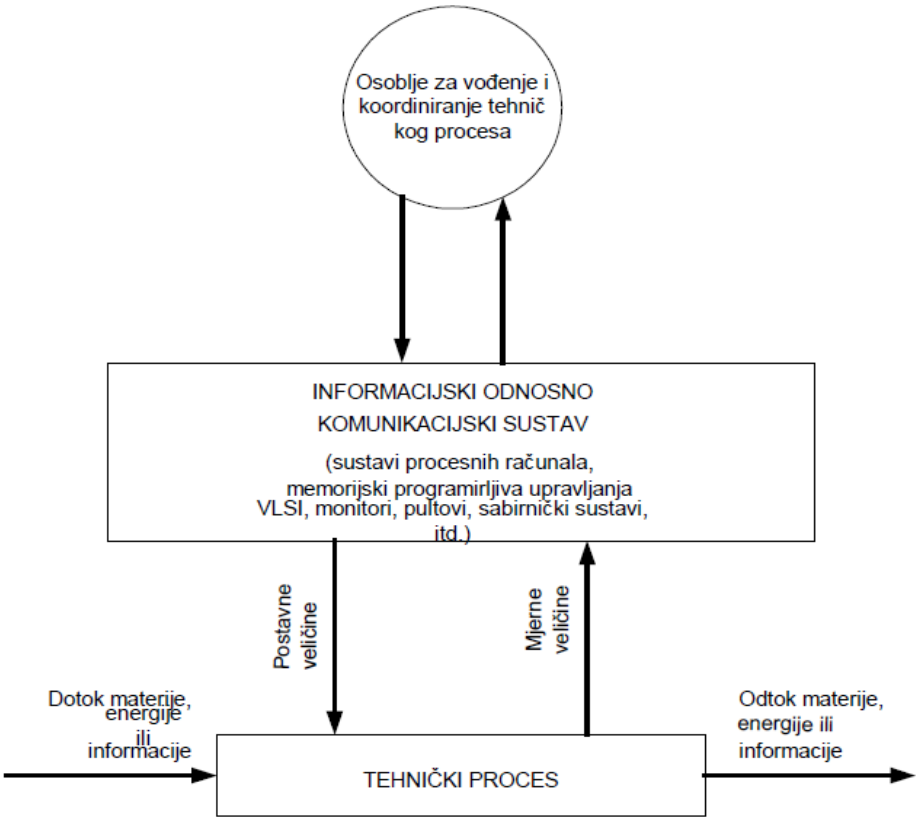
U središtu ovog rada nalazi se maketa hidrauličke jedinice koja predstavlja primjer jednog tehničkog procesa nad kojim se vrši proces automatizacije. Definicija tehničkog procesa glasi:

*Tehnički proces je sveukupnost događanja kod kojih se pomoću tehničkih sredstava upravlja i nadzire fizikalne veličine procesa. [1]*

Da bi se takav sustav kontinuirano održavao u željenom načinu rada njega je potrebno na određeni način **nadzirati**, njime **upravljati**, te ga **regulirati**. Nadzirati tehnički proces znači mjeriti njegove procesne veličine, prilagoditi i normizirati informacije o mjerenju, te na odgovarajući način prikazivati rezultate mjerenja sa ciljem uvida u trenutno stanje procesa. Uređaji koji su u izravnoj vezi s fizikalnim procesom, i koji fizikalne veličine procesa pretvaraju u električne veličine pogodne za procesiranje nazivaju se **mjerni članovi**, a sastoje se od mjernog osjetila (senzora) i mjernog pretvornika. Upravljati tehničkim procesom znači omogućiti tvorbu normiziranih postavnih veličina (upravljačkih signala), te ih prosljeđivati **izvršnim članovima** tehničkog procesa, i na taj način utjecati na njegov tijek, odnosno na rad njegovih elemenata ili podsustava. Izvršni članovi uređaji su koji upravljačke signale pretvaraju u mehanički rad, a sastoje se od postavnog člana i izvršne sprave. Regulacija je pojam koji objedinjuje elemente čija je uloga kompenziranje utjecaja vanjskih i unutrašnjih **poremećaja** (smetnji) koji utječu na sustav, te na taj način održavanje sustava u željenom režimu rada. Regulacija (ovisno o udjelu ljudskih aktivnosti) može biti ručna, poluautomatska i automatska.

Pojam vođenje procesa objedinjuje (između ostalog) navedene aktivnosti (nadzor, upravljanje i regulacija) u jednu cjelinu, a u svrhu stvaranja sustava za vođenje procesa potrebno je u cjelinu povezati osoblje, informacijski (komunikacijski) sustav, te tehnički proces opremljen mjernim i izvršnim članovima (procesnom periferijom), što je blokovski prikazano slikom 1.1. Zbog niza prednosti računala u odnosu na čovjeka (velika brzina obrade velikog broja podataka, višestruko veća i dugotrajna memorija, brza komunikacija s okolinom…) nerijetko se dio (ili većina) operativnih aktivnosti vezanih uz vođenje procesa prepušta računalu, te se na taj način javlja pojam automatsko vođenje procesa ili **automatizacija**. Ovisno o udjelu operativnih aktivnosti koje obavlja računalo i uloge koje ono ima u sustavu automatskog vođenja sustav može biti različitog stupanja automatizacije, a o ulozi računala će više biti rečeno u poglavlju 3.

Hidraulički sustavi ovakve vrste temelje se na međudjelovanju hidrauličkih i elektroenergetskih sustava upravljanih sklpovima energetske elektronike. Glavna im je uloga pretvaranje električne energije u mehaničku poput prijenosa fluida između elemenata hidrauličkog sustava, ili stlačivanje fluida za pretvorbu energije u gibanje elemenata hidrauličkog sustava. Kombinacijom ta dva sustava rezultira brzim , preciznim i ekonomičnim sustavom upravljanja.




*Slika 1.1.. Međudjelovanje osoblja, informacijskog odnosno komunikacijskog sustava i tehničkog procesa [1]*

U današnje vrijeme, automatsko upravljanje je sustavno uređena inženjerska znanost, te se može reći da živimo u doba automatizacije. Više od devedeset posto današnjih sustava ne bi moglo ispravno raditi bez primijenjenog načela automatske regulacije. Iako relativno nerazvikana disciplina, razvoj automatskog upravljanja predstavlja vrlo bitan faktor u tehnološkom napretku. Razvoj automatskog upravljanja i informacijskih tehnologija temelj su doba u kojem živimo, doba automatizacije [1].

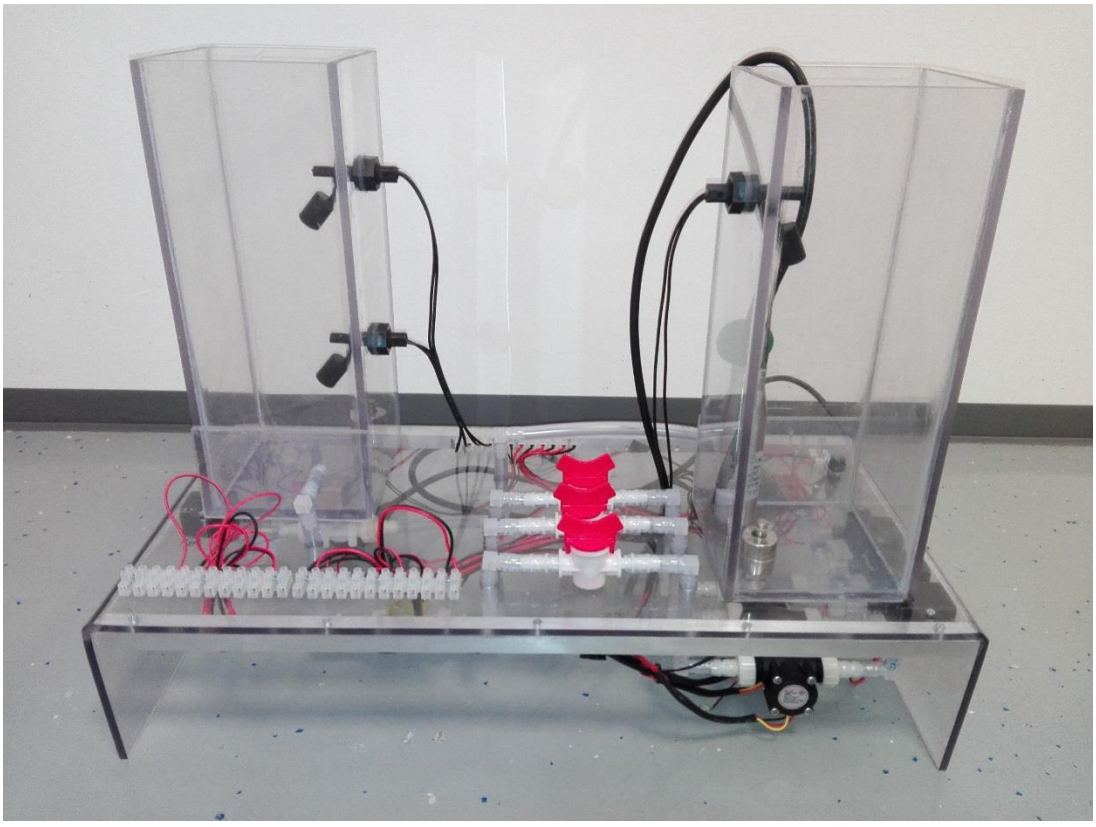
Ovaj rad podijeljen je u ukupno 6 poglavlja, od kojih je prvo poglavlje teorijski uvod u temu rada, i zadatak rada. U drugom poglavlju rada opisana je maketa hidrauličke jedinice. Opisan je način na koji je maketa izrađena, navedene su komponente od kojih se maketa sastoji, kao i princip rada i uloga određenih komponenata. Treće poglavlje bavi se sustavima automatskog vođenja sa programabilnim logičkim kontrolerima (PLC-ovima). Sadrži općenite informacije o PLC-u, princip rada i njegovu primjenu, informacije o IEC standardu za programabilne kontrolere (poput definiranih programskih jezika), informacije o komunikacijskim mrežama koje su korištene kod ovog rada. Osim navedenog treće poglavlje sadrži informacije o konkretnoj hardverskoj konfiguraciji PLC-a korištenoj u ovom radu, te su u zadnjem potpoglavlju opisani sustavi za vizualizaciju sustava automatskog vođenja. Četvrto poglavlje sadrži pregled korištenih programskih okruženja, te prikaz i objašnjenje softverskog rješenja programiranja logike PLC-a, te razvoja grafičkog sučelja tzv. HMI, a peto zaključak rada.

## 1.1. Zadatak diplomskog rada

Zadatak ovog diplomskog rada je napraviti maketu hidrauličke jedinice koja se sastoji od glavnog spremnika fluida koji je opremljen sa tri crpke, i recirkulacijskog opremljenog sa dvije crpke. Spremnici su međusobno povezani tako da se fluid prenosi iz jednog spremnika u drugi. Razina fluida u oba spremnika nadzire se senzorima razine. Potrebno je omogućiti upravljanje i nadzor makete pomoću programabilnog logičkog kontrolera (PLC-a) razvojem programskih rješenja i vizualizacije u programskim okruženjima Siemens STEP 7, WinCC i CoDeSys.

# 2. Maketa

Kao što je navedeno u zadatku diplomskog rada izrađena je maketa hidrauličke jedinice (slika 2.1.) koja predstavlja tehnički proces nad kojim se vrši automatizacija. Sastoji se od dva jednaka spremnika fluida od kojih je jedan glavni, a drugi tzv. recirkulacijski spremnik. Načinjeni su od akrilnog stakla debljine 6 mm unutrašnjih dimenzija 340 mm x 120 mm x 120 mm (VxŠxD). Za prijenos fluida među spremnicima korištene su crpke pogonjene istosmjernim motorom bez četkica (engl. brushless DC motor) nazivnog napona 12 V DC koje su opisane u potpoglavlju 2.2. Glavni spremnik opremljen je sa tri crpke, a recirkulacijski sa dvije, a one su postavljene na dno svakog spremnika. Crpke glavnog spremnika prenose fluid u recirkulacijski i obratno. U krugu glavnog spremnika nakon crpki nalaze se mjerači protoka fluida čija je izvedba i način rada pojašnjen u potpoglavlju 2.1.2. U oba kruga nalazi se po jedan upravljivi ventil temeljen na elektrodinamičkom principu. U krugu glavnog spremnika nalaze se još i tri ručno upravljiva ventila koji služe za simuliranje smetnje pri protoku. Razina fluida u oba spremnika nadzire se krajnjim senzorima razine, konkretno glavni spremnik opremljen je sa dva digitalna senzora, a recirkulacijski spremnik sa tri. Uz navedeno, glavni spremnik sadrži i analogni senzor razine, a sve vezano uz mjerenje procesnih veličina (razine i protoka fluida) objašnjeno je u potpoglavlju 2.1. Osim navedenih elemenata maketa sadrži elektroničke komponente koje služe za tvorbu upravljačkih signala za upravljanje protokom crpki, mikroupravljač Arduino Nano i tri integrirana kruga naziva TB6612FNG opisana u potpoglavlju 2.2.1., te dva izvora napajanja 12 VDC koji su potrebni za napajanje mikroupravljača, napajanje crpki, otvaranje ventila, te releje koji omogućavaju uključivanje crpki i ventila na 12 VDC pomoću standardnog napona 24 VDC s kojim radi PLC.



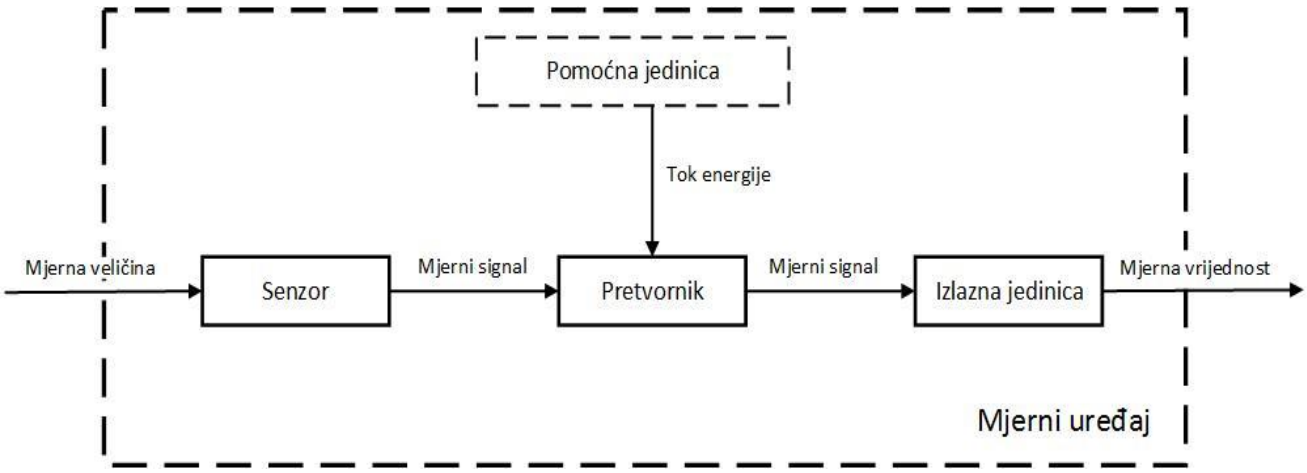
*Slika 2.1. Izgled makete hidrauličke jedinice*

## 2.1. Mjerenje procesnih veličina

Mjerenje procesnih veličina (procesna mjerenja) predstavlja kontinuirano utvrđivanje brojčane vrijednosti procesnih varijabli (fizikalnih veličina) koje su svojstvene tehničkom procesu. U ovom radu procesne veličine koje se mjere su razina fluida u spremnicima, te protok fluida, a opisani su u nastavku ovog potpoglavlja.

**Mjerni uređaji** funkcionalne su cjeline na čijem se ulazu javlja signal mjerne veličine, a na izlazu mjerna vrijednost kao odraz mjerne veličine. Svaki se mjerni uređaj u osnovi sastoji od više **mjernih članova** među kojima teku mjerni signali. Neki od tih članova za svoj rad trebaju dodatnu pomoćnu energiju koju im osiguravaju pomoćne jedinice, a nalaze se izvan toka mjernog signala, poput pretvornika na slici 2.1. Mjerni članovi povezani tokom mjernog signala tvore mjerni lanac ili mjerni slijed. U graničnom slučaju jedan se mjerni uređaj može sastojati od jednog jedinog mjernog člana i naziva se mjernim aparatom ili instrumentom [2].

Dio mjernog uređaja koji je u izravnoj vezi sa procesom, reagira na promjenu **procesne varijable** (fizikalne veličine) i pretvara njenu vrijednost u neki električni signal naziva se **mjerno osjetilo** (senzor). Ovisno o tome je li im potrebno vanjsko (električno) napajanje za rad, osjetila mogu biti pasivna i aktivna. Mjerno osjetilo u izravnoj je vezi sa **mjernim pretvornikom** (engl. Transducer), čija je uloga neobrađeni električni signal dobiven iz mjernog osjetila oblikovati tako da ga pojača, filtrira, digitalizira i sl. sa ciljem da ga oblikuje u neki od standardnih strujnih (4…20 mA, 0…20 mA, 0…1 mA, 0…5 mA, 0…50 mA) ili naponskih (0…10 V, -10…+10 V, -5…+5 V, 0…1 mV, 0…30 mV, 0…100 mV), te da bi kao takav bio pogodan za daljnju obradu.

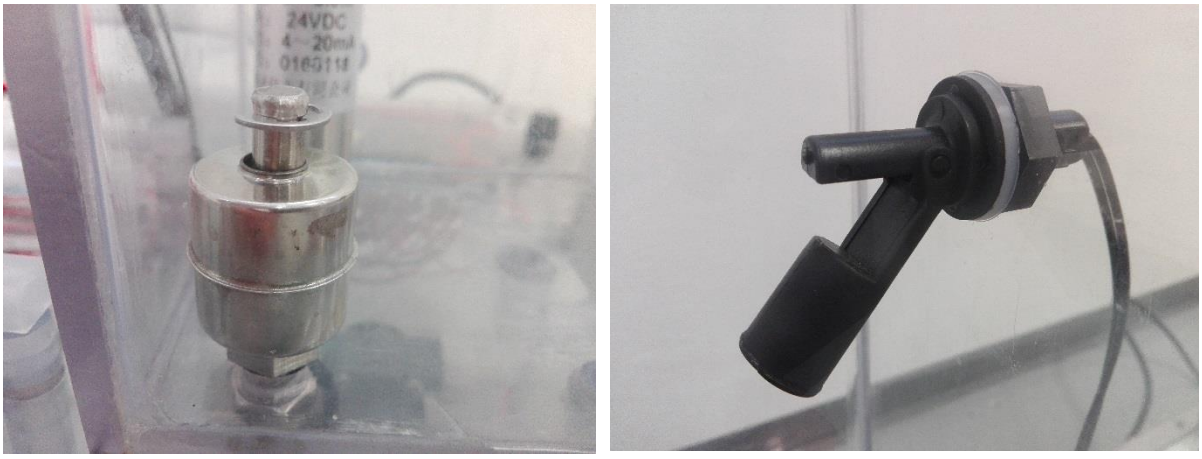



*Slika 2.1. Pojednostavljeni blokovski prikaz mjernog uređaja [2]*

### 2.1.1. Mjerenje razine fluida analognim i digitalnim senzorima razine

Mjerna veličina *razina* fizikalno predstavlja duljinu, odnosno visinu $h$ mjerenu u metrima, ali često i količinu tvari. Količina može biti iskazana kao obujam $V$ mjeren u $m^3$ ili kao masa $m$ mjerena u kg. Obujam se određuje za poznat spremnik, a masa za poznat spremnik i poznatu gustoću mjerne tvari.
Jedni uređaji mjere visinu mjerne tvari izravno, a drugi posredno, kao što je to primjerice kod određivanja razine preko mjerenja hidrostatskog tlaka na dno spremnika. Senzori koji tu nalaze primjenu mogu se podijeliti na one za kontinuirano mjerenje i one za mjerenje graničnih vrijednosti razine, najmanje i najveće. Ovisno o primijenjenom senzoru i dodatnim zahtjevima razvile su se različite izvedbe mjernih uređaja. [2]

Razina u spremnicima makete nadzire se krajnjim senzorima razine (slika 2.2.) postavljenim na dno i vrh spremnika (kod recirkulacijskog spremnika i u sredinu), a mjerenje razine u glavnom spremniku u analognom obliku ostvareno je pomoću senzora postavljenog na dno spremnika.




*Slika 2.2. Izgled krajnjih senzora razine postavljenih na spremnike*

Rad korištenog analognog senzora temelji se na principu mjerenja hidrostatskog tlaka tekućine, odnosno posredno mjerenjem sile kojom stupac tekućine djeluje na mjernu membranu senzora na dubini $h$. Pošto senzor mjeri visinu stupca vode na temelju ukupnog tlaka koji se sastoji od zbroja atmosferskog i hidrostatskog tlaka potrebno je bilo softverski omogućiti mjerenje vrijednosti tlaka kada je spremnik prazan i na taj način otkloniti utjecaj atmosferskog tlaka.

Izraz za hidrostatski tlak dan je izrazima (2-1) i (2-2)**,** gdje je:

$p$ - ukupni tlak na dubini $h$ kao zbroj atmosferskog i hidrostatskog tlaka [Pa]
$p_a$ – atmosferski tlak [Pa]
$\rho_v$ – gustoća vode [$^{kg}/_{m^3}$]
$g$ – ubrzanje Zemljine sile teže [$^{m}/_{s^2}$]
$h$ - dubina [m]

$$p = p_a + \rho_v g h \tag{2-1}$$

$$h = \frac{(p - p_a)}{\rho_v g} \tag{2-2}$$

Mjerni raspon korištenog analognog senzora razine prikazan slikom 2.3. dan je od strane proizvođača i iznosi od 0 do 5 m. Rezultat mjerenja je vrijednost strujnog signala koja se mijenja prema standardnoj strujnoj petlji 4-20 mA proporcionalno visini stupca fluida iznad senzora, gdje 4 mA odgovara vrijednosti 0 m, a 20 mA vrijednosti 5 m. Način na koji se senzor spaja u strujnu petlju (engl. Current loop) i kako se koriste vrijednosti koje on daje opisan je u potpoglavlju 3.2.4.

*Slika 2.3. Analogni senzora razine*

### 2.1.2. Mjerenje protoka fluida

Protok ($Q$) je fizikalna veličina koja definira količinu fluida (tekućine ili plina) koji proteče u određenom vremenu. Količinu fluida možemo definirati prema njegovoj masi ili volumenu, pa analogno tome poznajemo maseni (2-3) i volumni protok (2-4) čiji je međusobni omjer jednak gustoći fluida.

$$q_m = \frac{m}{t} [{}^{kg}/_{s}] \tag{2-3}$$

$$q_v = \frac{V}{t} [{}^{m^3}/_{s}] \tag{2-4}$$

U razmatranoj maketi protok fluida mjeri se pomoću tri jednaka mjerača protoka (volumetra) oznake modela YF-S201 prikazana slikom 2.4., koji su smješteni po jedan u krug svake crpke glavnog spremnika. Mjerači su izvedeni kao tzv. brojila s mjernim krilcima koja su u ovoj izvedbi u kombinaciji sa **senzorom Hallovog efekta**. Njihov se rad temelji na brojanju okretaja mjernog kola, te se uz umnožak poznatog volumena kojeg zahvati jednim okretajem i broja okretaja koje mjeri Hallov senzor dobiva volumen fluida koji je protekao mjeračem, odnosno volumni protok. U tablici 2.1. nalaze se tehničke specifikacije mjerača protoka.

Slika 2.4. Mjerač protoka YF-S201

**Hallov efekt** (Edwin Hall, 1855-1938), prikazan slikom 2.5. je pojava razlike potencijala okomito na smjer električne struje $I_H$ kroz vodljivo tijelo (načinjeno najčešće od indij-arsena ili indij-antimona). Utjecanjem magnetskog polja $\vec{B}$ na vodljivo tijelo protjecano upravljačkom strujom javlja se Lorentzova sila na nosioce naboja, te ih polje grupira na jednu stranu vodljivog tijela i stvara razliku potencijala $V_H$ (Hallov napon). Hallova sonda magnetski je senzor namijenjen mjerenju gustoće magnetskog toka koju, na temelju Hallovog efekta, pretvara u naponski signal. Hallov senzor najčešće se koristi za mjerenje jakosti struje, brzine vrtnje, kod beskontaktnog pozicioniranja, mjerenja brzine vrtnje itd.

*Tablica 2.1. Specifikacije mjerača protoka*

| | |
|---|---|
| Napajanje | 5-24 V DC |
| Radni napon | 5-18 V DC |
| Maksimalna potrošnja struje | 15 mA (na 5V) |
| Raspon mjerenog protoka | Od 1 do 30 l/min |
| Radna temperatura | -25 do 80 °C |
| Točnost | +/- 10% |
| Maksimalan tlak tekućine | 2 MPa |
| Ovisnost frekvencije pulseva i protoka | f (Hz) = 7,5 * protok (l/min) |
| Broj pulseva po litri | 450 |
| Dimenzije | 63,5 x 35,5 x 35,5 mm |
| Output rise time | 0,04 us |
| Output fall time | 0,18 us |

## 2.2. Upravljanje maketom

Upravljanje ovom maketom odnosi se na upravljanje radom crpki, te otvaranje i zatvaranje ventila u odvodima fluida iz spremnika čiji se rad temelji na elektrodinamičkom principu. Upravljanje crpkama (odnosno elektromotornim pogonima) podrazumijeva uključivanje i isključivanje crpke, te podešavanje protoka u granicama od 0% do 100%. Pri izradi ove makete korištene su crpke opremljene istosmjernim motorima bez četkica (eng. Brushless DC motor), a njihovo uključivanje omogućeno je relejima nazivnog ulaznog napona 24 V DC koji odgovara vrijednosti napona kartice PLC-a.

Promjena smjera struje kroz pojedine namote (komutacija) beskolektorskog istosmjernog motora ne odvija se putem mehaničkog komutatora po kojem klize četkice nego elektronički. Više nije potrebna mehanička komutacija sa svim problemima habanja kakvi su kod klasičnog istosmjernog motora. Tako beskolektorske istosmjerne motore gotovo da i ne treba održavati. Veliki nedostatak klasičnih istosmjernih motora uklonjen je pomoću elektroničkih sklopova. [3]

Slika 2.5. prikazuje izgled korištene crpke, a u tablici 2.2. navedene su tehničke karakteristike crpke. Upravljanje iznosom protoka crpke izvedeno je pulsno-širinskom modulacijom (engl. Pulse Width Modulation-PWM) pomoću mikroupravljača Arduino Uno i integriranog kruga TB6612FNG. Hardversko i softversko rješenje za PWM upravljanje crpkama objašnjeno je u potpoglavlju 2.2.1.

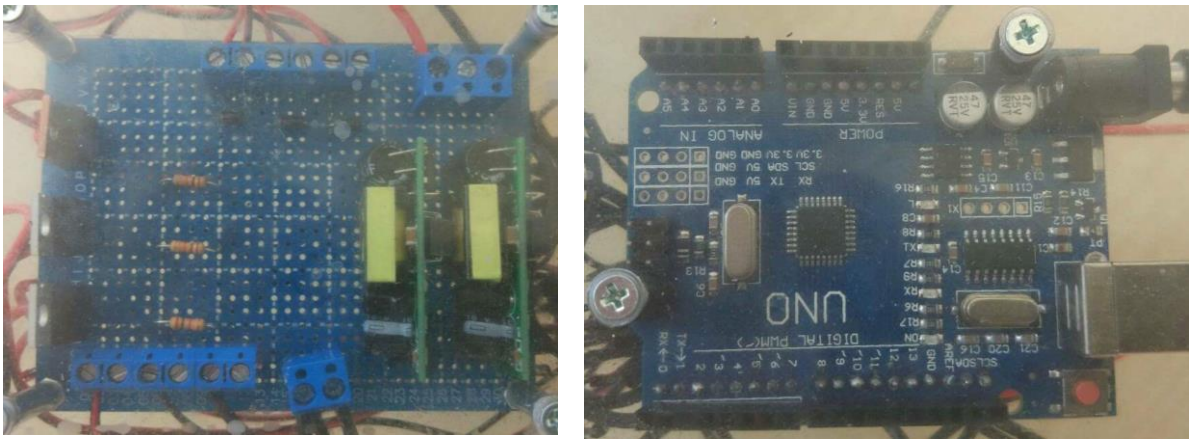
*Slika 2.5. Crpka korištene pri izradi makete*

*Tablica 2.2. Tehničke karakteristike korištene crpke*

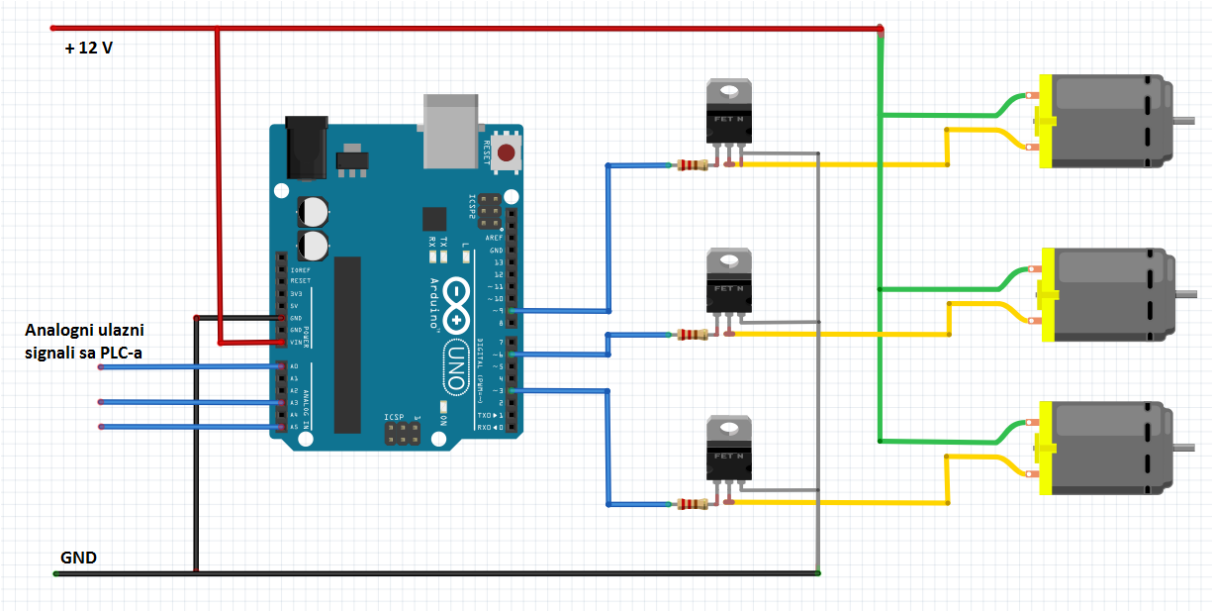
| | Karakteristika | Vrijednost |
|---|---|---|
| 1 | Nazivni napon | 12 V DC |
| 2 | Snaga | 4,2 W |
| 3 | Maksimalni protok | 240 l/h |
| 4 | IP stupanj zaštite | IP 68 |
| 5 | Maksimalna temperatura medija | 60 °C |
| 6 | Priključci | 8 mm |

### 2.2.1. Elektronički sklop za PWM modulaciju sa Arduino Uno mikroupravljačem

Kao što je navedeno u potpoglavlju 2.1. protok crpki potrebno je podešavati u granicama od 0 do 100 % protoka. To se postiže na način da se istosmjerni elektromotor (koji pogoni crpku) napaja promjenjivim iznosom istosmjernog napona, što je u ovom slučaju napon u rasponu od 0 do 12 V. Takav promjenjivi napon ostvaruje se pomoću pulsno-širinske modulacije (engl. Pulse Width Modulation (PWM)). Sklop namijenjen za to nalazi se na tiskanoj pločici prikazanoj slikom 2.6., koja omogućava fizičko i električno povezivanje elementa a način na koji su spojeni signalni vodiči spomenutih elemenata na tiskanoj pločici prikazan je na slici 2.7., gdje tri vodiča sa desne strane predstavljaju dovod analognih signala iz programabilnog logičkog kontrolera.



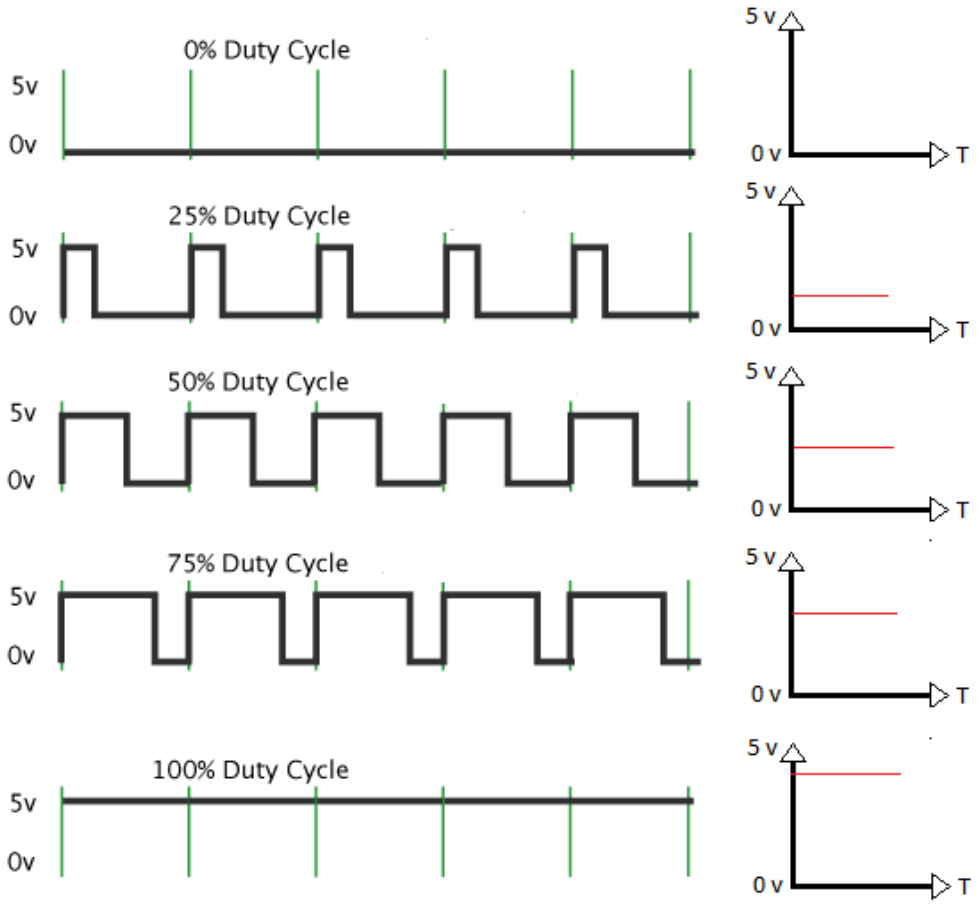
*Slika 2.6. Izgled tiskane pločice*




*Slika 2.7 Način spajanja mikroupravljača, tranzistora i motora*

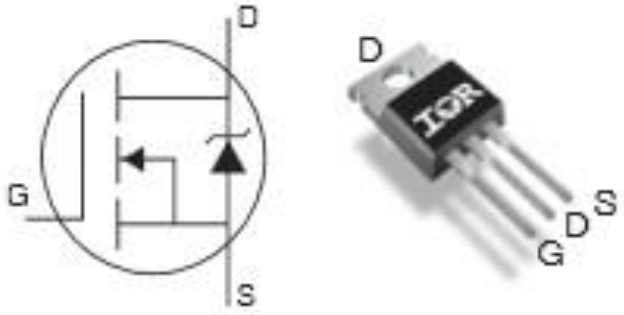
**Pulsno-širinska modulacija** metoda je za dobivanje analognog signala iz digitalnog, ili obratno, i to na način da se mijenja vrijeme trajanja uključenosti (engl. Duty Cycle) ulaznog

signala fiksne vrijednosti napona (kod unipolarne modulacije) ili različitim trajanjem uključenosti pozitivnog i negativnog napona naizmjenično (kod bipolarne modulacije). Različitim trajanjem uključenosti ulaznog napona unutar jedne sklopne periode modulacije postiže se promjena srednje vrijednosti izlaznog naponskog signala, te se na taj način srednja vrijednost napona mijenja u granicama od 0 do 100 % proporcionalno trajanju uključenosti što prikazuje slika 2.8. Struja rotora elektromotora prateći promjene napona raste ili pada (gotovo) linearno, a promjene struje uvijek su glatke (nema skokova) zbog induktiviteta rotora.



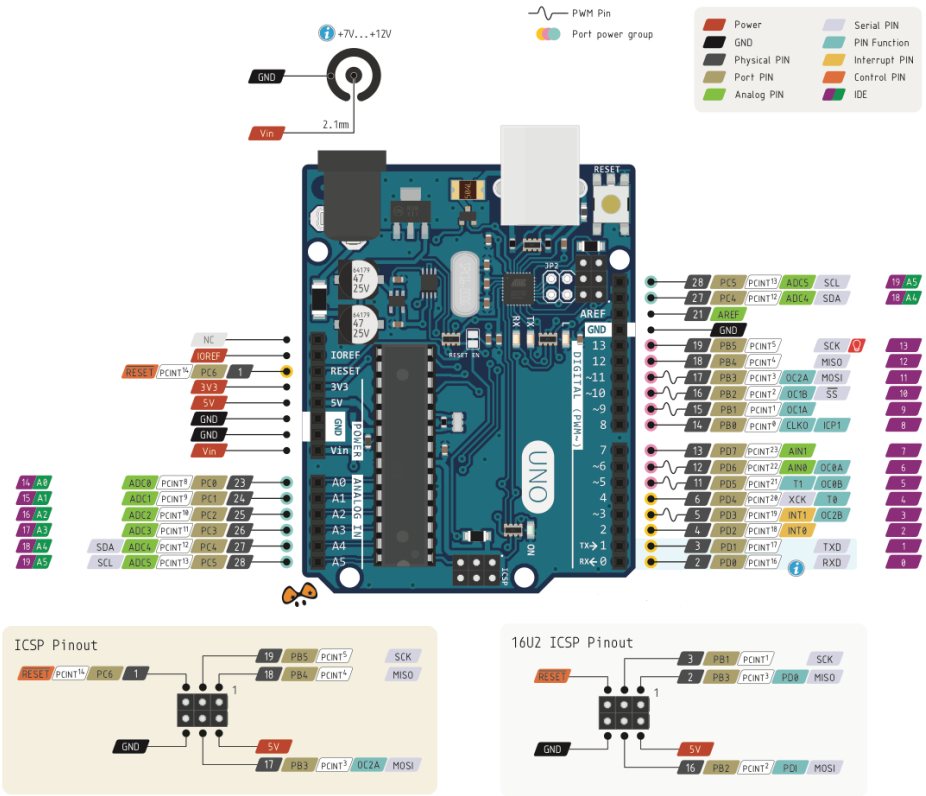

*Slika 2.8. Valni oblice pulsno-širinski moduliranog signala [4]*

Kako bi se formirao PWM naponski signal za željeni način napajanja crpki glavnog kruga korišteni su standardni naponski signali +/-10V dobiveni pomoću analognog signalnog modula PLC-a opisanog u potpoglavlju 3.2.4. Naponski signal šalje se na analogne ulaze mikroupravljača Arduino koji na temelju programa formira pulsno-širinski modulirane upravljačke signale, te omogućuje uključivanje i isključivanje pojedine crpke. a napisani kod za mikroupravljač nalazi se u prilogu. Mikroupravljač može primati analogne vrijednosti u granicama od 0 do 5 V, te je potrebno izlazne analogne signale iz PLC-a ograničiti u tom rasponu. Upravljački modulirani signali šalju se sa mikroupravljača na tranzistorske sklopke sa n-kanalnim MOSFET (engl. Metal oxide semiconductor field effect transistor) tranzistorima (slika 2.9.) čija je uloga da na temelju upravljačkih signala sa mikroupravljača zatvara strujni krug napajanja elektromotora crpki.




*Slika 2.9. Tranzistor IRF530N [5]*

**Arduino Uno** (slika 2.10.) je mikroupravljač čiji se rad temelji na mikroprocesoru ATmega328. Sadrži 14 digitalnih kanala (programski se određuje hoće li biti konfigurirani kao ulazi ili izlazi) od kojih 6 mogu biti konfigurirani kao PWM izlazi, 6 analognih ulaza, te USB-B priključak za spajanje na računalo. Programira se preko Arduino IDE (engl. Integrated Developement Environment) razvojnog okruženja, a operativna naponska razina na kojoj funkcionira je 5 V.

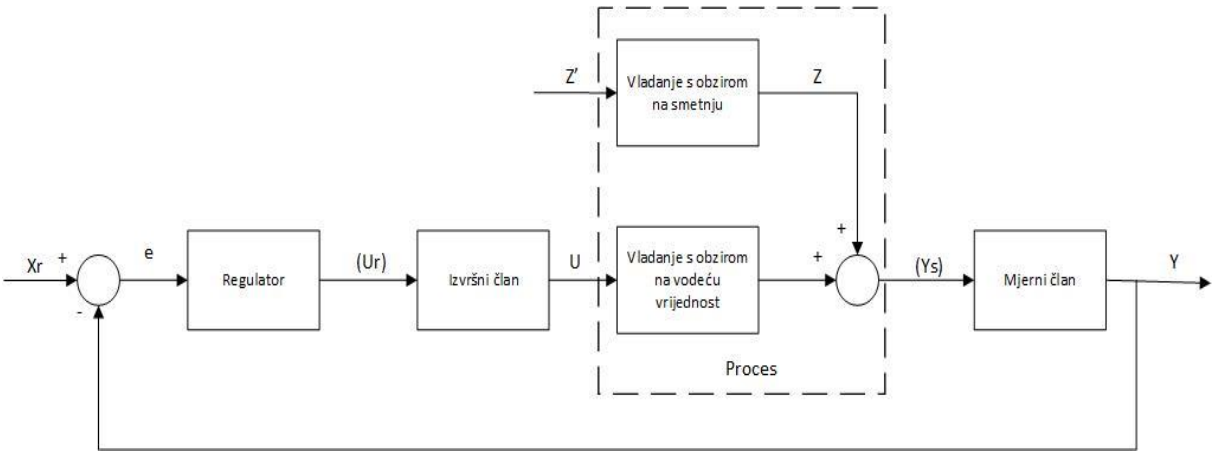



*Slika 2.10. Raspored pinova mikroupravljača Arduino Uno [6]*

## 2.3. Automatska regulacija protoka

Održavati proces u željenom načinu rada (ili u željenoj radnoj točki) neophodno je za kvalitetan i siguran rad sustava (npr. proizvodnog sustava). Primjer toga je održavanje zadane temperature prostorije, zadanog tlaka plina, ili, kao u ovom radu, održavanje protoka fluida na vrijednosti zadanoj od strane korisnika. **Poremećaji** (ili smetnje) nezaobilazne su pojave kada se radi o stvarnim sustavima, a one mogu biti vanjske i unutrašnje. Uloga sustava regulacije je kontinuirano kompenzirati utjecaj poremećaja koji djeluju na sustav tokom rada, te na taj način osiguravati da izlazna veličina (u ovom slučaju protok) što vjernije slijedi zadanu odnosno referentnu vrijednost.

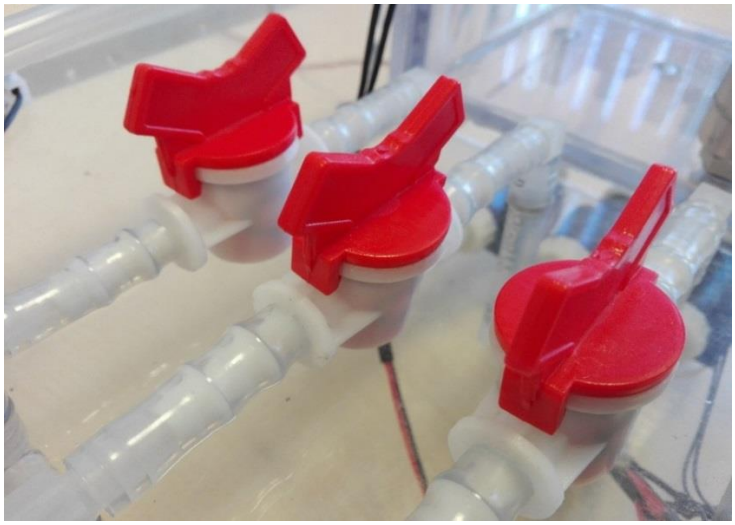
U ovom radu izvedena su tri jednaka regulacijska kruga za regulaciju protoka fluida iz glavnog u recirkulacijski spremnik svake od triju crpki. Svaki regulacijski krug sastoji se od elemenata koji su navedeni na osnovnoj strukturi upravljanja (slika 2.11.) (engl. Control loop), gdje su strelicama označeni tokovi različitih signala, krugovima mjesta njihova zbrajanja, a blokovima su predstavljeni podsustavi čija je uloga ulazni signal (pobudu) na željeni način transformirati u izlazni signal (odziv).



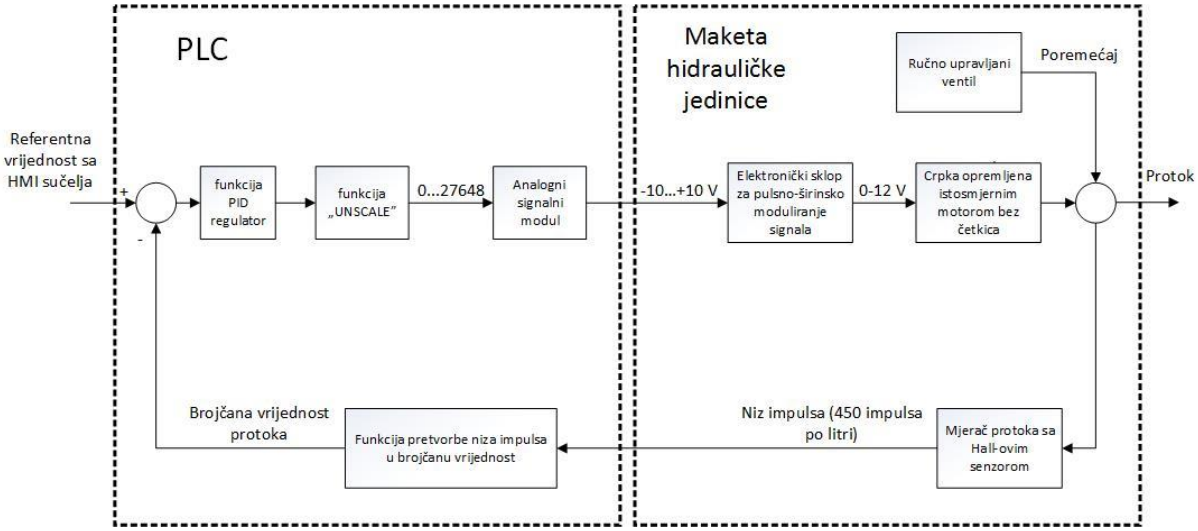

*Slika 2.11. Osnovna struktura sustava upravljanja [7]*

*Xr* predstavlja ulazni signal u regulacijski sustav, a naziva se referentna ili vodeća vrijednost (engl. Setpoint), i služi kao (od strane korisnika zadana) željena radna točka reguliranog sustava. Veličina *Y* predstavlja reguliranu veličinu (engl. Controlled variable) koja je rezultat cjelokupnog međudjelovanja tehničkog procesa i regulatora. Regulirana vrijednost $U_r$ signal je nastao kao rezultat djelovanja regulatora, odnosno kao odziv regulatora na pobudu u obliku regulacijskog odstupanja (engl. Actuating signal) *e*. Regulacijski član, ili kraće samo **regulator** je podsustav sustava automatske regulacije koji na temelju regulacijskog odstupanja generira upravljački signal, kojim se nastoji smanjiti regulacijsko odstupanje. Regulacijsko odstupanje je signal koji nastaje oduzimanjem regulirane veličine od referentne. **Povratna veza** (negativna) pojam je koji definira put mjernog signala od mjernog člana (sa izlaza regulacijskog kruga) do ulaza u regulator kako bi se ostvarilo regulacijsko odstupanje. Regulacijski sustavi koji posjeduju povratnu vezu nazivaju se zatvoreni regulacijski krug. Signal *U* naziva se upravljačka ili postavna veličina (engl. Manipulated variable) i služi za izravno djelovanje na tehnički proces, a signal *Z* predstavlja vanjsku smetnju odnosno poremećajni signal koji dolazi iz okoline procesa.

Kod regulacije protoka ulogu referentne vrijednosti $X_r$ ima vrijednost varijable u granicama od 0 do 100 % protoka svake crpke pojedinačno, a zadaje ih korisnik preko HMI (engl. Human Machine Interface) sučelja. Kao regulator korištena je softverska funkcija **PID regulator** (objašnjen u potpoglavlju 2.3.1.) unutar razvojnog okruženja za programiranje PLC-a, a postupak parametriranja opisan je u poglavlju 5. Izvršni član čine tri crpke glavnog kruga kao objekata regulacije sa pripadajućim elektroničkim sklopom (iz potpoglavlja 2.2.1.). Smetnja koja djeluje na proces regulacije izvedena je pomoću tri ručno upravljana ventila (slika 2.12.) smještena u krug glavnog spremnika koji ograničava protok fluida, te ga je u mogućnosti i u potpunosti onemogućiti, a uloga je regulatora da njegov utjecaju u što većoj mjeri kompenzira. Ulogu mjernog člana ima mjerač protoka opisan u potpoglavlju 2.1.2., a njegova je uloga u regulacijskom krugu davati povratnu informaciju o stvarnoj vrijednosti regulirane veličine $Y$ (protoka). Informacija koju on daje kao rezultat mjerenja šalje se povratnom vezom na „početak“ regulacijskog kruga. Regulacijska struktura makete hidrauličke jedinice koja blokovski prikazuje elemente regulacijskog kruga sa pripadnim signalima između njih prikazana, te granicama uređaja od kojih se sastoji prikazana je slikom 2.13.



*Slika 2.12. Ručno upravljivi ventili za simulaciju poremećaja*

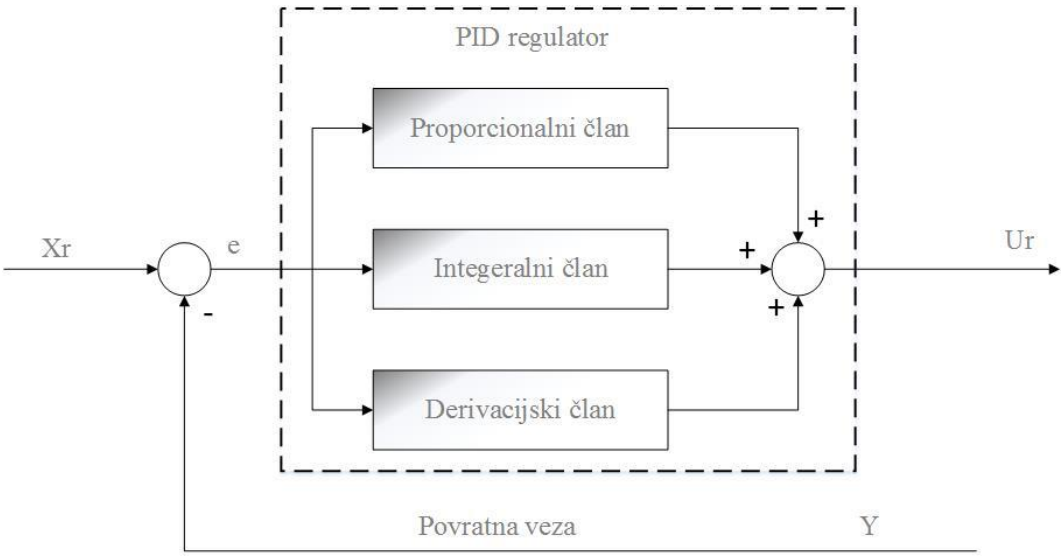



*Slika 2.13.Struktura regulacijskog kruga makete hidrauličke jedinice*

### 2.3.1. PID regulator

Regulator tvori regulacijsko odstupanje $e(t) = x_R(t) - y(t)$, koje se dalje obrađuje kako bi se dobila upravljačka veličina *u(t)*. Upravljačka veličina *u(t)* osigurava, preko izvršnog člana, kontrolirani tok energije (materije) upravljanom procesu i na taj način "držanje" regulirane veličine na određenom iznosu i uz djelovanje poremećajnih veličina. Upravljani proces ne može trenutačno reagirati na promjenu upravljačke veličine *u(t)*, zbog vremenskog zatezanja, odnosno energetski (materijalni) spremnici procesa ne mogu se trenutačno puniti/prazniti. Brzina promjena stanja energetskih (materijalnih) spremnika procesa određena je vremenskim konstantama. Prema tome, struktura i parametri regulatora moraju proizaći iz strukture i parametara matematičkog modela procesa. [7]

Jedan od najčešće korištenih standardnih regulatora je tzv. PID regulator (ili iz njega izvedeni regulatori) (blokovski prikazan slikom 2.14.), odnosno regulator koji se sastoji od tri člana: proporcionalnog (P), integralnog (I) i derivacijskog (D) koji djeluju kao tri neovisne cjeline za odgovarajuću obradu signala, točnije regulacijskog odstupanja *e* čija se kombinacija djelovanja uz određene parametre koristi kao efikasan način za željeno reguliranje procesa.




*Slika 2.14. Blokovski prikaz PID regulatora*

- **Proporcionalni član** (sa pripadnim koeficijentom pojačanja $\boldsymbol{K_P}$ (engl. Proportional Gain)) pojačava ulazni signal te na taj način proporcionalno iznosu regulacijskog odstupanja utječe na upravljačku veličinu kako bi se regulacijsko odstupanje što je više moguće smanjilo. Služi kako bi se smanjilo vrijeme prijelaznih pojava tokom dinamičkog vladanja procesa, no često se lošim odabirom pojačanja mogu javiti nepoželjne oscilacije sustava, kao i nestabilnost.

- **Integralni član** (sa pripadnom integralnom vremenskom konstantom $\boldsymbol{T_I}$ (engl. Integral time)) integrira odnosno vremenski akumulira iznos regulacijskog odstupanja ovisno o prošlim iznosima odstupanja. Takvim načinom djelovanja otklanja se pogreška u stacionarnom stanju (engl. Steady-state error), no odziv samog integralnog člana nije dovoljno brz.

- **Derivacijski član** sa pripadajućom derivacijskom vremenskom konstantom $\boldsymbol{T_D}$ (engl. Derivative time) ima ulogu djelovati na temelju brzine promjene regulacijskog odstupanja, te iz tog razloga, sam po sebi ne može služiti za smanjivanje istog. Idealan derivacijski član u stvarnosti ne postoji, te se kao realan derivacijski član koristi koristi tzv. $DT_1$ član odnosno član sa vremenskim kašnjenjem.

Parametre $K_P$, $T_I$ i $T_D$ moguće je u određenim granicama podešavati, te se na taj način regulator može parametrirati na određen način tako da se prilagodi određenom tehničkom procesu, te se na taj način omogući željeno vladanje sustava.

Upravljački algoritam PID regulatora u vremenskoj domeni dan je matematičkim izrazom (2-5):

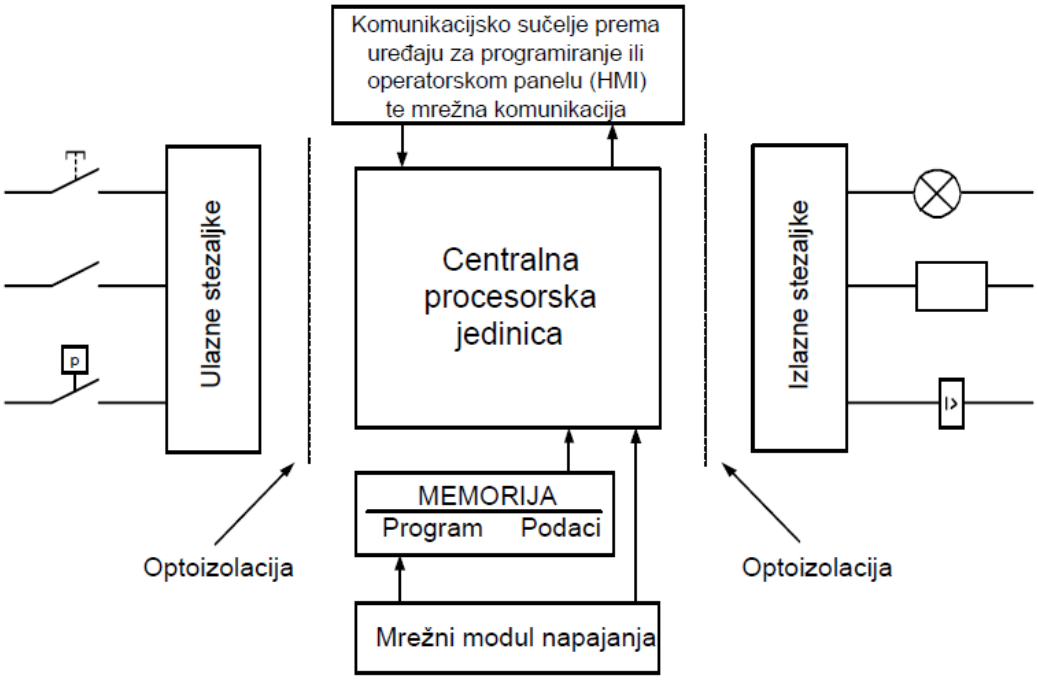
$$U(t) = K_R e(t) + \frac{K_R}{T_I} \int_0^t e(\tau) d\tau + K_R T_D \frac{de(t)}{dt} \tag{2-5}$$

# 3. Sustavi sa programabilnim logičkim kontrolerima

## 3.1. Programabilni logički kontroleri

Programabilni logički kontroleri (engl. *Programmable Logic Controller, PLC* ) **modularna** su i **robusna** računala prilagođena radu u industrijskim postrojenjima za automatizaciju tehničkog procesa, gdje imaju ulogu procesnog računala. Najčešće se koriste za primanje informacija o procesnim veličinama od mjernih članova u polju, prosljeđivanje postavnih veličina izvršnim članovima, upravljanje radom uređaja poput frekvencijskih pretvarača i sl. Pojavili su se krajem 60-ih godina prošlog stoljeća kao alternativa fiksno ožičenim upravljačkim krugovima temeljenim na sklopnicima i relejima (tzv. relejna logika) čiji su glavni nedostatci bili nefleksibilnost kod promjena logike upravljanja, velike dimenzije, te osjetljivost na vanjske utjecaje u postrojenju. Jedna od glavnih značajki PLC-a je njegova modularnost, odnosno mogućnost da se kombinacijom različitih modula (procesorskih jedinica, signalnih modula, komunikacijskih modula, funkcijskih modula itd.) stvori PLC željenih karakteristika za točno određenu primjenu, a glavne prednosti male dimenzije, niska potrošnja energije, lako testiranje i nadzor izvršenja programa, laka instalacije i prepravljanje (sklopovsko i programsko) itd.

**Centralna procesorska jedinica** (engl. central processor unit, CPU) središnja je komponenta PLC-a čija je uloga izvođenje korisničkog programa spremljenog u memoriji, čitanje i pisanje stanja ulaza i izlaza, te komunikacija i upravljanje drugim komponentama sustava upravljanja preko komunikacijskog sučelja, a neizravno, preko izvršnih uređaja i procesom. Slika 3.1. prikazuje osnovne cjeline PLC uređaja u čijem je središtu CPU, te njihovu međusobnu povezanost. Na slici je navedena i optoizolazija (izvedena pomoću optosprežnika (engl. optocoupler)) između ulaznih i izlaznih stezaljki koja služi za galvansko odvajanje unutarnjih strujnih krugova PLC-a od ulaznih i izlaznih signala, te se na taj način omogućava zaštita od potencijalnih kvarova uslijed potencijalnih razlika.




*Slika 3.1. Osnovne cjeline PLC uređaja [8]*

PLC ima ulogu primati informacije o trenutnom stanju procesa, te generirati upravljačke signale, a uređaji i komponente (ulazni ili izlazni) preko kojih se odvija navedena razmjena informacija nazivaju se **procesna periferija**. Pod procesnu periferiju spadaju svi uređaji koji služe za mjerenje procesne veličine i utjecanje na proces, prijenos informacija u obliku signala, analogno-digitalnu i digitalno-analognu pretvorbu, filtriranje signala zbog otklanjanja smetnji itd. Glavna podjela signala s obzirom na tip signala preko kojih PLC razmjenjuje informacije sa procesom preko procesne periferije je na **digitalne** (diskretne) i **analogne** (kontinuirane). Digitalni signal je binarna informacija koja može poprimiti dva stanja; stanje logičke nule (naponska razina 0-5 VDC), ili stanje logičke jedinice (naponska razina 14-30VDC), odnosno neaktivno (nula) i aktivno (jedan) stanje. Primjer su digitalni senzori razine korišteni u ovom radu, a opisani su u potpoglavlju 2.1.1. koji su u mogućnosti PLC-u dati informaciju o tome ima li fluida na razini na kojoj su postavljeni ili ne. Analogni mjerni signali, za razliku od digitalnih, daju informaciju o iznosu mjerenja fizikalne veličine, i to u nekom od standardnih strujnih (npr. 0…20mA,4…20mA...) ili naponskih (npr. 0…10V, -10…+10V, 0…+5V, 1…5V ...) signala. Nakon prihvaćanja analognog signala, PLC ga pojačava, filtrira, te ga, pomoću A/D pretvornika digitalizira kako bi ga kao takvog mogao dalje obrađivati. Primjer korištenja analognih signala u ovom radu je slanje informacije o razini fluida u spremniku, i formiranje signala za zadavanje referentnih vrijednosti protoka za upravljanje protokom crpki.

**Ulazne stezaljke** nalaze se na ulaznom signalnom modulu (u slučaju modularne izvedbe PLC-a) ili su sastavni dio PLC-a (kod kompaktne izvedbe). One služe da bi se mjerni signali koji putuju od procesa kao rezultati mjerenja fizikalne veličine, preko mjernih članova (senzora, mjerača) doveli u PLC na određenoj i njemu prilagođenoj naponskoj razini kako bi ih on dalje procesirao**. Izlazne stezaljke** veza su između PLC-a i izvršnih članova preko kojih se upravljački signal formiran u PLC-u šalje procesnoj periferiji i tako utječe na rad sustava (ventili, motorne sklopke, sklopnici, svjetlosna signalizacija…). Izvršni član sastoji se od postavnog člana (pretvornika upravljačkog signala) i izvršne sprave (aktuatora). Kontakti izlaznih stezaljki mogu biti tranzistorski (imaju veliku frekvenciju sklapanja, ali su ograničeni na 0.5 A jakosti struje tereta) i relejni (sklapaju veće struje, ali su sporiji).
PLC upravlja procesima koji se odvijaju u stvarnom vremenu, te se informacije o stanjima ulaza, i postavljanje stanja izlaza trebaju osvježavati dovoljno velikom frekvencijom. Iz tog se razloga korisnički program u PLC-u ciklički izvršava u beskonačnoj petlji, što znači da nakon izvršenja jednog ciklusa odmah počinje sa novim ciklusom. Trajanje jednog ciklusa može varirati od nekoliko ms do nekoliko desetaka ms, a duljina ciklusa ovisi o korištenom CPU, broju korištenih ulaza i izlaza, te zahtjevnosti korisničkog programa.

Jedan takav ciklus sastoji se od četiri koraka:

1) Čitanje stanja fizičkih ulaza i spremanje stanja u memorijsko područje pod nazivom slika procesnih ulaza (eng. process-image input (PII))
2) Izvršavanje korisničkog programa pri čemu se na temelju stanja ulaznih veličina (PII) i korisničkog programa vrši upisivanje rezultata logičkih operacija u sliku procesnih izlaza (eng. process-image output (PIQ))
3) Preslikavanje slike procesnih izlaza na fizičke izlaze PLC-a
4) Izvršavanje ostalih funkcija poput kontrole trajanja ciklusa, pregled ima li sklopovskih ili programskih grešaka, izvršavanje potrebne razmjene podataka itd.

## 3.2. Hardverska konfiguracija sustava automatskog vođenja

Jedan PLC sastoji se od više komponenata, a jedna takva cjelina naziva se i jedinicom. Središnji dio jedinice predstavlja centralna procesorska jedinica (eng. central processor unit, CPU). Na CPU je priključen izvor napajanja (eng. power supply, PS) koji osigurava odgovarajući ulazni napon, u industriji je to najčešće 24 V DC. Okvir (eng. rack) ima ulogu povezati sve komponente jedne jedinice u cjelinu i omogućiti mehaničku postojanost. Kao okvir često se koristi standardna DIN šina (engl. DIN rail) od 35 mm (npr. kod SIMATIC S7-300 i S7-1200 kontrolera), a kod ove konfiguracije korišten je okvir za S7-400 kontrolere pod nazivom (UR2) koji se sastoji od ukupno 9 slotova.

Osim navedenog, jedinica se može sastojati i od modula proširenja:

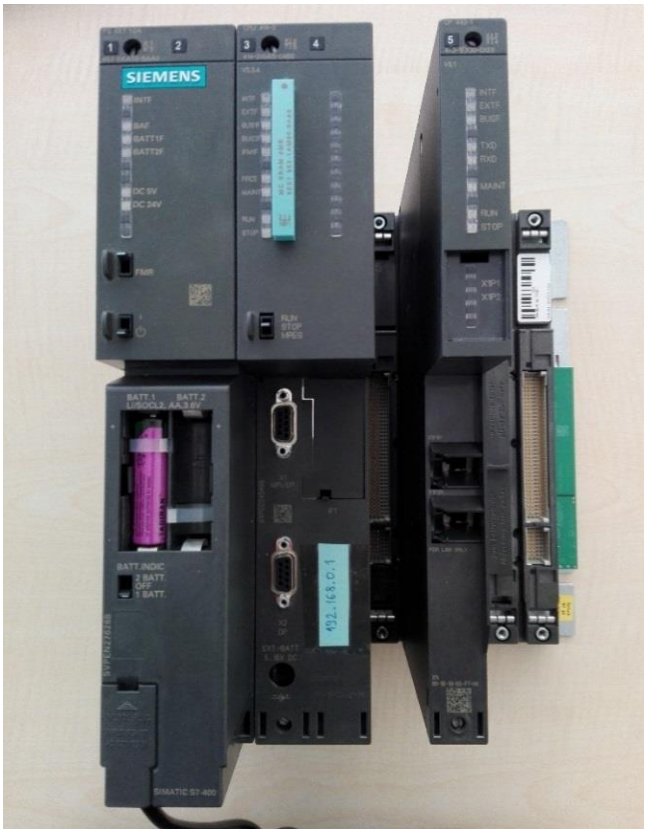
- Moduli sučelja (engl. interface module, IM) služe za međusobno povezivanje glavnog okvira sa okvirima proširenja
- Signalni moduli (engl. signal modules, SM) služe za povećanje broja digitalnih ili analognih ulaza i izlaza signala kada je to potrebno
- Funkcijski moduli (engl. function modules, FM) služe za predprocesiranje i pripremu provedbu zahtjevnijih i bržih zadataka pri obradi signala
- Komunikacijski procesori (engl. communications processors, CP) služe za ostvarivanje komunikacije PLC-a sa podmrežama u sustavu

Hardverska konfiguracija korištena pri izradi ovog rada sastoji se od glavnog (centralnog) okvira i okvira za proširenje međusobno povezanih PROFIBUS industrijskom sabirnicom (komunikacijski protokol opisan u potpoglavlju 3.4.2.). Središte glavnog okvira je PLC Siemens SIMATIC S7 414-3 DP, a uz njega glavni se okvir sastoji i od modula napajanja (PS 407 10A), te komunikacijskog procesora, odnosno signalnog modula proširenja CP 443-1 koji služi za povezivanje putem PROFINET komunikacijskog standarda za industrijsku komunikacijsku (opisanog u potpoglavlju 3.4.1.) koji u ovom slučaju služi za povezivanje PLC-a sa osobnim računalom. Središnji dio okvira proširenja je uređaj za distribuirane ulaze i izlaze Siemens ET 200S.

### 3.2.1. Siemens S7 414-3 DP

CPU 400 može se konfigurirati i programirati s punim funkcionalnim opsegom primjene (čak i za starije module) korištenjem STEP 7 V5.x. Programski jezici za korisničke programe LAD, FBD te STL integrirani su u STEP 7. Kontinuirani funkcijski dijagram (CFC) grafički programski urednik, program GRAPH za slijedno upravljanje i HiGraph za upravljanje stanjima dostupni su kao opcijski paketi za programski jezik SCL. Konfiguriranje i programiranje pomoću STEP 7 Professional unutar TIA Portala obuhvaća module koji se trenutačno isporučuju. Programski jezici za korisnički program LAD, FBD, STL, SCL, te GRAPH za slijedno upravljanje integrirani su u STEP 7 Professional V11. [9]

Izgled glavnog okvira skupa sa rack-on i modulima na njemu prikazan je slikom 3.2., a tehničke specifikacije PLC-a navedene su u tablici




*Slika 3.2. Glavni okvir PLC-a sa CPU-om S7 414-3 DP*

*Tablica 3.1. Tehničke karakteristike PLC-a SIMATIC S7-414-3 PN/DP*

| | |
|---|---|
| Dimenzije (mm) | 50 x 290 x 219 |
| Slotova koje zauzima u okviru | 2 |
| Integrirana radna memorija | 2.8 MB |
| Vrijeme procesiranja bit operacije | 0.045 µs |
| Vrijeme procesiranja word operacije | 0.045 µs |
| Vrijeme procesiranja int operacije | 0.045 µs |
| Vrijeme procesiranja real operacije | 0.135 µs |
| Broj SIMATIC timera | 2048 |
| Broj SIMATIC brojača | 2048 |

### 3.2.2. Siemens ET 200S (IM151-1 HF)

Siemens ET 200 (slika 3.3.) distribuirani je ulazno/izlazni uređaj čija je uloga razmjena informacija o stanjima procesnih mjernih i upravljačkih signala procesne periferije sa PLC-om putem PROFIBUS DP-a i PROFINET IO-a. IM151-1 HF je modul sučelja čija je uloga povezati ET200S sa ostalim uređajima (u ovom slučaju sa PLC-om) na PROFIBUS DP sabirnici preko RS485 sučelja, gdje ima ulogu DP slave-a.

Slika 3.3. Okvir proširenja PLC-a (ET200S)

**ET200S** prilagodljiv je I/O sustav sa stupnjem zaštite IP 20 čiji bit-modularni dizajn omogućuje točnu prilagodbu automatizacijskom zadatku. Dostupni su digitalni I/O moduli, analogni I/O moduli, tehnološki moduli te pokretači za motore i frekvencijski pretvarači. Može se spojiti do 63 I/O modula na sučelje ET 200S modula. I/O moduli mogu se zamijeniti tijekom rada pogona Oni se prikače na terminalni modul koji sadržava ožičenje. ET 200S dostupan je s PROFIBUS DP sučeljem (maksimalna je brzina prijenosa podataka 12 Mbit/s) ili PROFINET IO sučeljem (maksimalna je brzina prijenosa podataka 100 Mbit/s). [9]

### 3.2.3. Digitalni signalni moduli

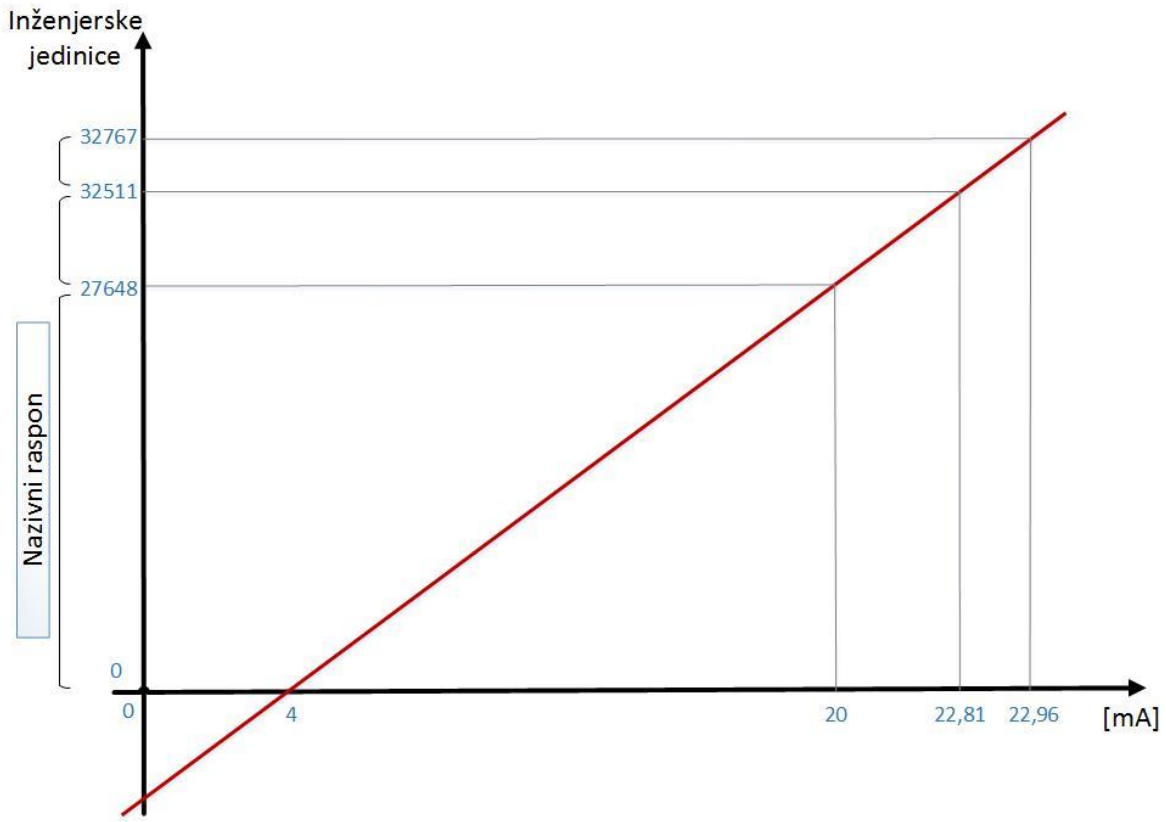
Digitalni signalni moduli služe za razmjenu digitalnih (binarnih) procesnih signala između PLC-a i tehničkog procesa, a mogu biti ulazni i izlazni. Uloga im je napon vanjskog signala koji je najčešće naponske razine 24 VDC (ili 230 VAC) sniziti na razinu na kojoj CPU procesira signale (ulazni digitalni moduli) i obratno (izlazni digitalni moduli). Kod ove maketa korišteni su digitalni ulazni i izlazni moduli 4DI DC24V ST, i 4DO DC24V/0,5A ST korišteni za razmjenu digitalnih procesnih signala sa maketom. Pomoću njih prihvaćaju se binarne informacije sa krajnjih senzora razine, niz impulsa sa mjerača protoka, te se zadaju binarne komentare za uključivanje crpki, te otvaranje ventila.

### 3.2.4. Analogni signalni modul

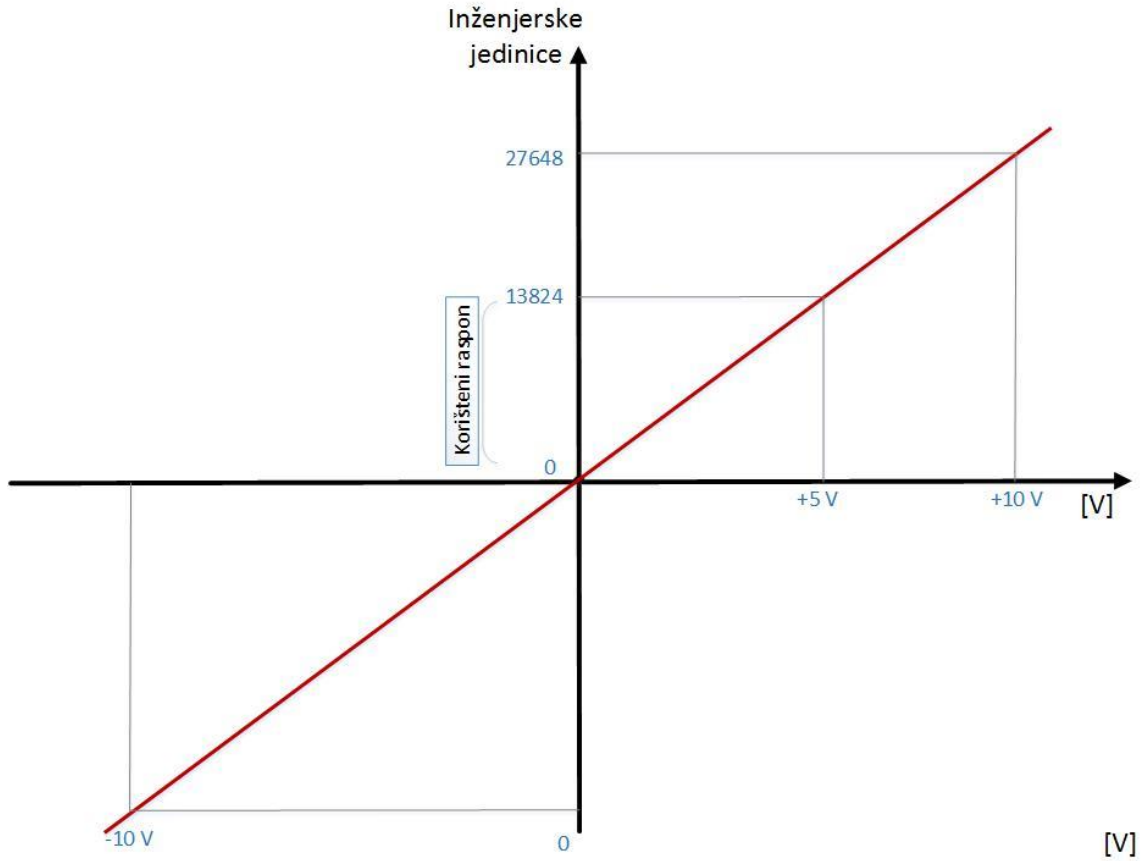
Analogni signalni moduli služe za razmjenu analognih procesnih signala između PLC-a i tehničkog procesa. Imaju ulogu digitalizirati analogne mjerne signale (A/D pretvornikom) dobivene od mjernih uređaja u polju koji se pretvaraju u 16-bitne podatke kako bi ih PLC mogao procesirati, ili digitalni signal formiran u PLC-u pretvoriti u analogni postavni signal (D/A pretvornikom) pogodan za slanje na postavne članove (aktuatore). Jedan procesni signal zauzima jedna kanal modula, a modul može imati više kanala. Vrijednost digitaliziranog signala izražena bezdimenzionalnom veličinom tzv. **inženjerskim jedinicama** zapisuje se kao 16-bitna cjelobrojna varijabla, odnosno INT tip podatka, a raspon vrijednosti digitaliziranog signala kod Siemens proizvođača kreće se (za unipolarni opseg) od 0 do 27648 za pripadajući raspon procesnog signala. Kao posljedica utjecaja vanjskih utjecaja (elektromagnetskih smetnji) na vodič kojim se prenosi mjerni signal javljaju se superponirane vrijednosti mjernom signalu, te on tada nadilazi nazivno mjerno područje. U tom slučaju vrijednosti digitaliziranog signala također mogu biti povišene, a na slikama 3.4. i 3.5. prikazane su ovisnosti digitaliziranog signala o mjernom signalu. Pri izradi ovog rada analognim signalnim modulima mjereni su procesni signali za mjerenje razine fluida u glavnom spremniku, te za zadavanje referentnih vrijednosti protoka crpki glavnog kruga. Za mjerenje razine korištena je standardna strujna petlja (engl. Current loop) **4…20 mA**, a za zadavanje referentnih vijednosti naponski signal **-10 V…+10 V.**

Ovisnost ovih dvaju veličina linearna je funkcija, a izračunati vrijednosti digitaliziranog signala iz poznate vrijednosti struje procesnog signala moguće je preko jednostavnog izraza (3-1) gdje je $X_1$ najmanja vrijednost električnog signala (4 mA ili -10 V), $X_2$ najveća vrijednost električnog signala (20 mA ili +10 V) $Y_1$ 0 inženjerskih jedinica, $Y_2$ 27648 inženjerskih jedinica, $X$ poznata vrijednost električnog signala, a $Y$ vrijednost digitaliziranog signala.

$$\frac{X - X_1}{X_1 - X_2} = \frac{Y - Y_1}{Y_1 - Y_2} \qquad (3\text{-}1)$$




*Slika 3.4. Ovisnost digitaliziranog signala o strujnom signalu 4…20 mA*




*Slika 3.5. Ovisnost digitaliziranog signala o naponskom signalu -10…+10 V*

## 3.3. IEC 61131-3 standard

IEC (engl. *International Electrotechnical Commission*) 61131-3 globalno je prihvaćen standard za programiranje programabilnih kontrolera, koji u praksi funkcionira kao skup smjernica (ne pravila) čiji je cilj standardizirati proizvodnju i način rada sa programabilnim kontrolerima, te na taj način unificirati rad sa istima. Standard je uspostavila grupa sastavljena od strane IEC (engl. International Electrotechnical Commission) koja se sastoji od predstavnika različitih PLC proizvođača, programerskih tvrtki i korisnika, a definira stvari poput programskih jezika, strukture programiranja, tipove podataka, standardne funkcije, način adresiranja varijabli itd.

### 3.3.1. Elementarni tipovi podataka

Varijable služe za pohranu podataka vezanih uz korisnički program. Tipovi podataka način su standardiziranja oblika zapisa, i to raspona podataka koji se u nju spremaju, veličine koju zauzimaju u memoriji, načina na koji se vrijednost prikazuje itd. **Elementarni tipovi podataka** (navedeni u tablici 3.2.) definirani standardnom IEC 61131-3 dogovoreni su tipovi podataka koji se koriste kod programiranja PLC-ova.

*Tablica 3.2. Pregled elementarnih tipova podataka [**9**]*

| Ime | Tip podatka | Bitova |
|---|---|---|
| | | |
| BOOL | 1-bitna binarna vrijednost | 1 |
| SINT | 8-bitni broj s predznakom prikazan s nepomičnom decimalnom točkom | 8 |
| INT | 16-bitni broj s predznakom prikazan s nepomičnom decimalnom točkom | 16 |
| DINT | 32-bitni broj s predznakom prikazan s nepomičnom decimalnom točkom | 32 |
| USINT | 8-bitni broj bez predznaka prikazan s nepomičnom decimalnom točkom | 8 |
| UINT | 16-bitni broj bez predznaka prikazan s nepomičnom decimalnom točkom | 16 |
| UDINT | 32-bitni broj bez predznaka prikazan s nepomičnom točkom | 32 |
| REAL | 32-bitni broj prikazan s pomičnom decimalnom točkom | 32 |
| LREAL | 64-bitni broj prikazan s pomičnom decimalnom točkom | 64 |
| TIME | Vrijeme zapisano u IEC formatu | / |
| DATE | Datum | / |
| TIME_OF_DAY,TOD | Vrijeme u danu | / |
| DATE_AND_TIME,DT | Datum i vrijeme | / |
| BYTE | 8-bitna binarna vrijednost | 8 |
| WORD | 16-bitna binarna vrijednost | 16 |
| DWORD | 32-bitna binarna vrijednost | 32 |

### 3.3.2. Programske organizacijske jedinice

**Programske organizacijske jedinice** (eng. Program Organization Unit, POU) dio su korisničkog programa i predstavljaju programske blokove, ili manje zasebne cjeline programa od kojih je sačinjeno softversko rješenje, odnosno korisnički program PLC-a. Služe za stvaranje programskog rješenja željene strukture koja je prilagođena procesu. Često se koriste unaprijed definirane organizacijske jedinice programirane od strane proizvođača, djelatnika unutar tvrtke ili drugih korisnika koje su napravljene za univerzalne radnje koje se često koriste ili za radnje koje su unutar tvrtke dogovorene standardom kako bi se skratilo vrijeme programiranja.

Tri su glavne programske organizacijske jedinice:

- **Program** je POU koja predstavlja vezu između operacijskog sustava i korisničkog programa. Funkcionira kao veza između fizičkih adresa PLC-a i ostalih organizacijskih jedinica i ciklički se izvršava.
- **Funkcija** je POU u koju se sprema određeni dio cjelokupnog programskog rješenja, te se jednom programirana funkcija može koristiti više puta. Unutar funkcije moguće je pozivati druge funkcije ili funkcijske blokove.
- **Funkcijski blok** je jedinica koja ima mogućnosti kao i funkcija, no razlika je u tome što funkcijski blok posjeduje svoje vlastito memorijsko područje unutar memorije PLC-a u koje funkcijski blok trajno sprema podatke tokom izvođenja programa.

### 3.3.3. Programski jezici

Prema standardu IEC 61131-3 definirano je pet programskih jezika za programiranje PLC-ova od kojih, ovisno o vrsti problema kojem pristupa, te njegovim afinitetima, korisnik (programer) sam odabire kojim jezikom će pisati korisnički program, a razni proizvođači koriste vlastite verzije osnovnih programskih jezika. Jezici definirani standardnom su:

- **Ljestvičasti dijagram** (engl. Ladder Diagram (LD))
- **Lista instrukcija** (engl. Instruction List (IL))
- **Strukturirani tekst** (engl. Structured text (ST))
- **Funkcijski blok dijagram** (engl. Function Block Diagram (FBD))
- **Sekvencijalni funkcijski dijagram** (engl. Sequential Function Chart (SFC))

## 3.4. Industrijske komunikacijske mreže i protokoli

Informacijsko-komnikacijski sustav (informacijska mreža) pojam je koji definira skup međusobno spojenih hardverskih (sklopovskih) i softverskih (programskih) komponenata koje čine međusobno povezane podmreže, te izvršavaju operacije **prijenosa**, **usmjeravanja**, **procesiranja** i **skladištenja** informacija.

**Industrijske komunikacijske mreže** imaju ulogu međusobnog umrežavanja uređaja sustava automatskog vođenja procesa „vertikalno“ (povezivanje podsustava različitih hijerarhijskih upravljačkih razina) i „horizontalno“ (povezivanje podsustava iste hijerarhijske upravljačke razine) kako bi bili u stanju razmjenjivati informacije potrebne za tijek tehničkog procesa, ali i informacije o radu procesa i proizvodima potrebnih za analizu rada postrojenja. Kako bi takav relativno složen sustav funkcionirao na optimalan način potrebno je ispunjavati određene zahtjeve poput zahtijevane brzine prijenosa podataka, mogućnost povezivanja više različitih komunikacijskih podustava, mogućnost jednostavnog proširivanja ili izmjena strukture unutar postojećeg sustava, pouzdanost sustava u vidu otpornosti na vanjske utjecaje i prijetnje za sigurnost informacija koje se sustavom prenose.

**ISO/OSI** (engl. Open System Interconnection) referentni je komunikacijski model razvijen od strane međunarodne organizacije za standardizaciju (engl. International Standardization Organisation, ISO) koji služi kao univerzalni protokol definiran pravilima za razmjenu podatkovnih paketa (tzv. telegrama) između umreženih računala ili drugih uređaja. Definiran je na temelju 7 slojeva arhitekture hijerarhijski poredanih od najvišeg do najnižeg od kojih su gornja tri definiraju interakciju korisnika sa računalom, a donja četiri način na koji se informacije razmjenjuju između računala:

- Aplikacijski sloj
- Prezentacijski sloj
- Pristupni sloj
- Prijenosni sloj
- Mrežni sloj
- Podatkovni sloj
- Fizički sloj

**TCP/IP protokol** (engl. Transmission Control Protocol/Internet protocol) namijenjen je komunikaciji različitih uređaja preko različitih (međusobno povezanih) komunikacijskih mreža, te ne ovisi o mediju prijenosa podataka. Temelj je za većinu današnjih LAN mreža, te za najveću komunikacijsku mrežu danas, Internet.

**Industrijski *ethernet*** podrazumijeva korištenje klasičnog *ethernet* protokola uz robusne priključnice otpornije na zahtjevne industrijske uvijete (otpornost na vibracije, vlagu, visoku temperaturu itd.). Na njemu se temelji komunikacijski standard PROFINET opisan u idućem potpoglavlju.

U modernim strojevima i sustavima rješenja automatizacije imaju sljedeće značajke:

- povećani broj senzora i aktuatora

- decentralizacija komponenata i

-povećana „inteligencija“ komponenata

U klasičnim rješenjima automatizacije postoji središnje upravljanje pojedine „inteligentne“ komponente na koje se dovode svi statusi signala i stvarne vrijednosti te iz kojeg se odvode svi upravljački signali i postavne vrijednosti. U današnjim je rješenjima „inteligencija“ raspodijeljena te je često smještena na lokalnoj razini unutar stroja izvan središnjeg ormara, npr. u decentraliziranim pogonima. S tim su razvojem povezani kvantitativno i kvalitativno povećanje količine razmijenjenih signala. To posebno dolazi do izražaja kada se uzme u obzir velik broj dijagnostičkih informacija koje su potrebne za praćenje takvog rješenja automatizacije. [7]

### 3.4.1. PROFINET

PROFINET (engl. *Process field network*) je komunikacijski standard namijenjen razmjeni podataka u industrijskim sustavima temeljen na industrijskom *ethernet*-u. Za razmjenu podataka definirana su prvi, treći i četvrti sloj ISO/OSI referentnog modela (fizički, mrežni i prijenosi sloj). Postoje dvije vrste standarda:

- PROFINET I/O (Input/Output) koji služi za povezivanje decentralizirane periferije sa središnjim upravljačkim sustavom
- PROFINET CBA (engl. Component Based Automation) koji služi za povezivanje strojeva, uređaja ili podsustava u zajedničku mrežu

Kod hardverske konfiguracije u ovom radu korišten je PROFINET I/O za uspostavljanje veze između osobnog računala i glavnog okvira PLC-a, a fizička veza između njih uspostavljena je pomoću kabela prikazanog slikom 3.6.

*Tablica 3.3. Karakteristike PROFINET-a [3]*

| **Karakteristike** | **Opis** |
|---|---|
| Topologija | Zvjezdasta, prstenasta, linijska, stablasta |
| Spoj | Aktivan |
| Produženje | Do 100 m svaki segment |
| Sudionici | Ograničeno samo MAC adresnim prostorom |
| Pristup | Full *duplex*, bez kolizija, brzi *ethernet* 100 Mbit/s s tehnologijom mrežnih preklopnika |
| Signali | Binarni signal, analogni signali, parametri, dijagnostičke informacije |
| Kabel | Upletena parica, 4-žična linija, oklopljena, kategorija 5 (IEC 11801), moguća je primjena optičkih vlakana |
| Priključna tehnika, montaža | RJ45 konektor |
| Sigurnosni signali | S PROFIsafe profilom |
| Norme | *Ethernet:* IEEE 802.3, PROFINET: IEC 61158 |
| Korisnička organizacija | www.profibus.com |

*Slika 3.6. Kabel korišten za povezivanje uređaja putem PROFINET-a*

### 3.4.2. PROFIBUS

PROFIBUS (engl. Process Field Bus) je standardni komunikacijski protokol iz skupine protokola za industrijsku komunikaciju u stvarnom vremenu tzv. *Fieldbus,* a korišten je kod sustava proizvođača Siemens. Služi za uspostavljanje međusobne komunikacije između uređaja sustava nad kojim se provodi automatizacija, konkretno za razmjenu (većinom binarnih) podataka upravljačke jedinice sustava sa nadzornom jedinicom i mjernim i izvršnim članovima sustava koji se nalaze u „polju“.

Kao sredstvo prijenosa koristi se oklopljeni dvožični vod ili optički kabel od stakla ili plastike. Duljina kabela u sabirničkom segmentu ovisi o brzini prijenosa podataka, a to je 100 m za najveću brzinu prijenosa (12 Mbit / s) i 100 m za najnižu brzinu prijenosa (9,6 Kbit / s). Raspon mreže može biti povećan s pomoću obnavljača ili modula za optičku vezu. Maksimalan je broj čvorova 127, a čvorovi mogu biti aktivni ili pasivni. Aktivni čvor može pristupiti sabirnici u određenom razdoblju i može slati podatke. Nakon tog razdoblja čvor prosljeđuje *token* (pravo pristupa) sljedećoj aktivnoj jedinici (pristupna metoda prosljeđivanja *tokena*). Ako su pasivni čvorovi (*slave*-jedinice) dodijeljeni aktivnom čvoru (master), master će, dok ima *token*, izvršiti razmjenu podataka s dodijeljenim *slave*-jedinicama. Pasivni čvor nema odobrenje za pristup sabirnici. [9]

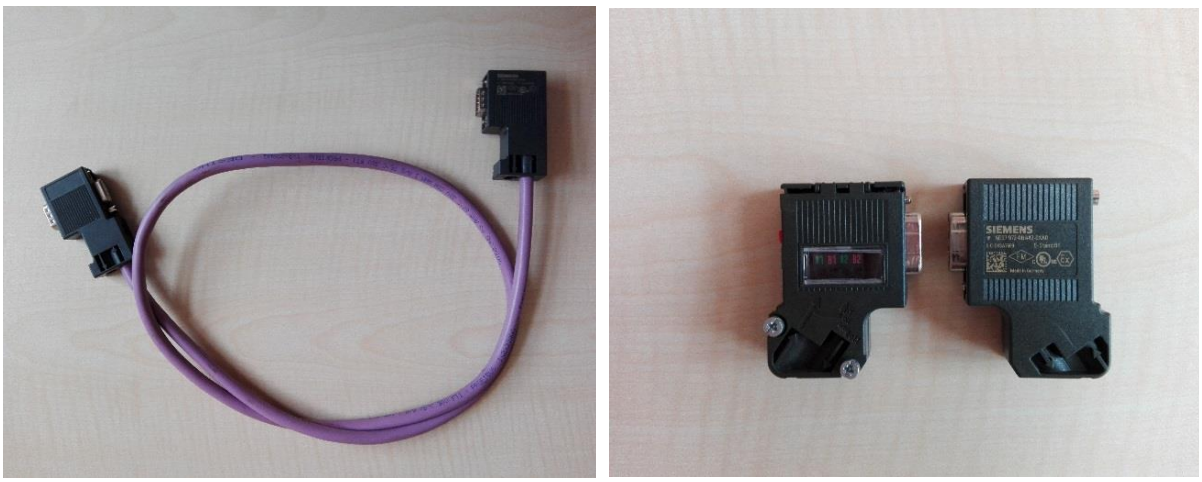
*Tablica 3.4. Karakteristike PROFIBUS DP-a* [3]

| **Karakteristike** | **Opis** |
|---|---|
| Topologija | Linijska struktura, sa zaključanim otpornicima |
| Spoj | Pasivan |
| Produženje | Do 1,2 km svaki segment (električna izvedba s dvjema žicama na liniji) do 9,6 kBit/s |
| Sudionici | Maksimalno 124 |
| Pristup | Prosljeđivanje *tokena* za rad s više mastera, s prozivanjem |
| Signali | Binarni signal, analogni signali, parametri, dijagnostičke informacije |
| Kabel | Upleteni dvožilni kabel, oklopljeni (optička vlakna također moguća) |
| Priključna tehnika, montaža | Konektor |
| Sigurnosni signali | Mogu se prenositi s pomoću posebnog profila (PROFIsafe) |
| Norme | EN 50170, IEC 61158/IEC 61784 |
| Korisnička organizacija | www.profibus.com |

Za različite primjene postoje različite varijante protokola:

- PROFIBUS DP (engl. Decentralized Peripherals)
- PROFIBUS PA (engl. Process Automation)
- PROFIBUS FMS (Fieldbus Message Specification)

PROFIBUS DP je naziv za vrstu sabirnice u industrijskim komunikacijskim mrežama korištenu za komunikaciju PLC-a kao nadređenog sudionika (engl. Master) sa procesnom periferijom kao podređenim sudionicima (engl. Slave), te je korišten u ovom radu za povezivanje glavnog okvira i okvira proširenja. Temelji se na RS-485 tehnologiji, a za prijenos podataka koristi oklopljeni dvožilni vod (slika 3.7. (lijevo)) i 9-pinske D-sub konektore (slika 3.7.(desno)) . Glavna mu je prednost što jednim vodom može prenositi više signala, i velika brzina prijenosa signala (12 Mbit/s za kabele kraće od 100 m). Kod hardverske konfiguracije u ovom radu PROFIBUS DP komunikaciju podržava CPU, i na temelju te vrste komunikacije CPU komunicira sa čvorom distribuirane periferije.




*Slika 3.7. Oklopljeni dvožični vod (lijevo) i D-sub konektor (desno)*

## 3.5. Vizualizacija sustava automatskog vođenja

Kako bi se ostvarila struktura automatskog upravljanja tehničkim procesom kao na slici 1.1. iz uvoda potrebno je omogućiti sučelje između čovjeka (operatera) i informacijsko-komunikacijskog sustava kao posrednika između tehničkog procesa i operatera. Zadaće takvog sustava obuhvaćaju aktivnosti kao što su nadzor trenutnih stanja u obliku prikaza vrijednosti procesnih veličina, prikaza tijeka procesa i stvaranja baze mjerenih (i drugih) podataka, opsluga sustava poput uključivanja (dijelova) postrojenja, zadavanja referentnih vrijednosti, prijelaz između sekvenci procesa i sl., bilježenje i obavještavanje o događajima i alarmantnim stanjima i drugo. Zajednički naziv za skupinu takvih računalno sklopovskih sustava je SCADA (engl. *Supervisory Control And Dana Acquisition*) sustavi. **HMI** (engl. Human Machine Interface) pojam je koji predstavlja sučelje između čovjeka i stroja, odnosno obuhvaća dio zadaća SCADA sustava vezanih uz grafički prikaz podataka.

U sklopu ovog rada izrađena je vizualizacija procesa makete u programskim okruženjima Siemens SIMATIC WinCC te Codesys, a opis postupka izrade vizualizacije prikazan je u poglavlju 4. Vizualizacija se sastoji od simboličkog izgleda makete sa odgovarajućim funkcijama uključivanja svake od pet crpki i prikaza trenutnog stanja crpki (stanje uključenosti i isključenosti, te prikaza koliko je vremena radila pojedina crpka), zadavanje referentnih vrijednosti protoka crpki glavnog kruga za upravljanje protokom, praćenje iznosa protoka koje daju mjerači protoka u krugu svake crpke glavnog kruga, otvaranje i zatvaranje vantila, prikaz iznosa analogne vrijednosti razine tekućine u glavnom spremniku, i ostale pomoćne funkcije.

# 4. Programski alati i programsko rješenje

Kako bi rješavanje problema (zadatka) bilo moguće softverski izvesti potrebno je bilo izabrati odgovarajuće funkcije, pregledno organizirati globalne varijable, odrediti odgovarajuće tipove podataka za korištene varijable, omogućiti komunikaciju između uređaja hardverske konfiguracije, na odgovarajući način podesiti i spojiti module PLC-a itd.

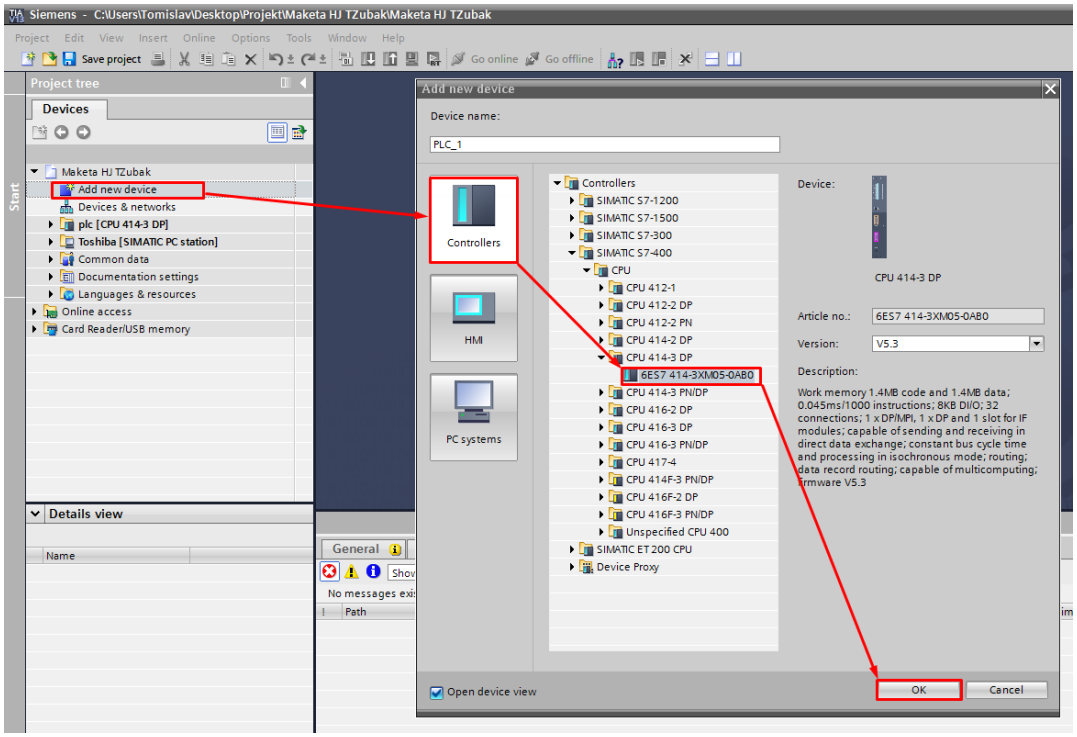
U ovom poglavlju bit će opisana programska okruženja korištena za razvoj programskog rješenja, te će biti prikazani i opisani dijelovi programskog rješenja, kao i izgled korisničkog sučelja i način na koji se određene funkcije okruženja koriste.

## 4.1. Siemens SIMATIC STEP 7 i WinCC

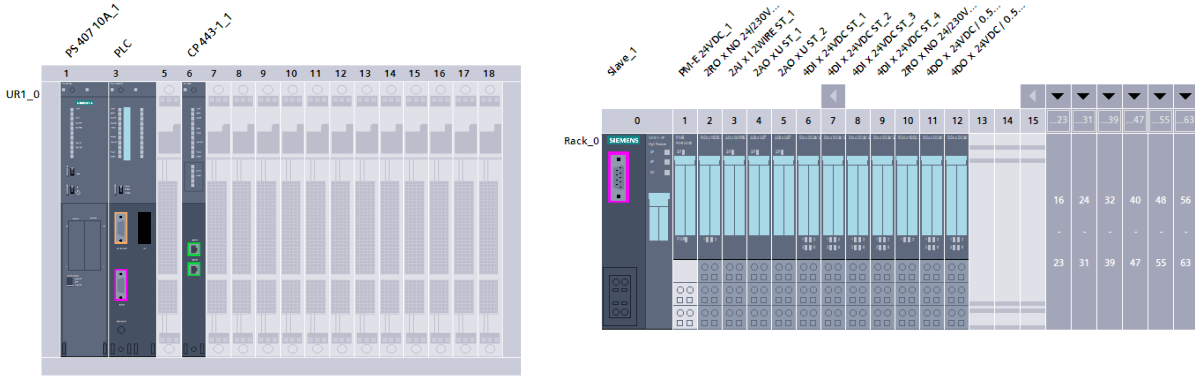
STEP 7 naziv je za inženjerski alat njemačkog proizvođača Siemens namijenjenog za programsko i sklopovsko konfiguriranje komponenata tzv. SIMATIC automatizacijskih sustava. Glavne zadaće koje se njime izvršavaju su upravljanje sa projektima, stvaranje sklopovske konfiguracije (koja se sastoji od PLC-ova i njima pridruženih HMI uređaja za vizualizaciju), stvaranje programskog rješenja (korisničkog programa), *online* testiranje programa, definiranje komunikacijske konfiguracije za povezivanje uređaja, upravljanje memorijom PLC-a i adresiranjem varijabli itd. Osnovna podjela programskog alata STEP 7 je na dvije generacije: prva je SIMATIC Manager i obuhvaća verzije programa do V5.5, te druga tzv. STEP 7 unutar TIA Portala u kojem je razvijeno programsko rješenje zadatka.
**TIA Portal** (engl. Totally Integrated Automation Portal) naziv je za programsko okruženje koje objedinjuje više programskih alata iz različitih porodica Siemens-ovih uređaja (STEP 7, WinCC, Startdrive, Scout TIA, SIMOCODE) u jedan projekt kako bi se proces konfiguriranja cjelovitog sklopovskog i programskog rješenja pojednostavio i ubrzao.

### 4.1.1. Sklpovska konfiguracija sustava

Kako bi rad s PLC-om bio moguć potrebno je bilo konfigurirati sklopovsku konfiguraciju uređaja. Unutar projekta dodane su hardverske komponente sustava koje su međusobno povezane na odgovarajuće komunikacijske podmreže. Klikom na „Add new device“ unutar projektnog stabla otvara se prozor unutar kojeg se odabiru uređaji (PLC, HMI panel, sustav baziran na osobnom računalu, frekvencijski pretvarač itd.), a ponuđeni uređaji ovise o instaliranim softverima unutar TIA portala. Dodavanje PLC-a prikazano je slikom 4.1. U ovom slučaju odabrani su PLC 414-3 DP (slika 4.2. lijevo) i distribuirani ulazno/izlazni uređaj ET200S (slika 4.2. desno) sa pripadajućim modulima proširenja. Potrebno je bilo izabrati odgovarajući okvir (UR1), modul napajanja, komunikacijski procesor, središte ET200S uređaja, te njemu pripadne module (modul napajanja, brzi brojač, te analogne i digitalne ulazne i izlazne module).

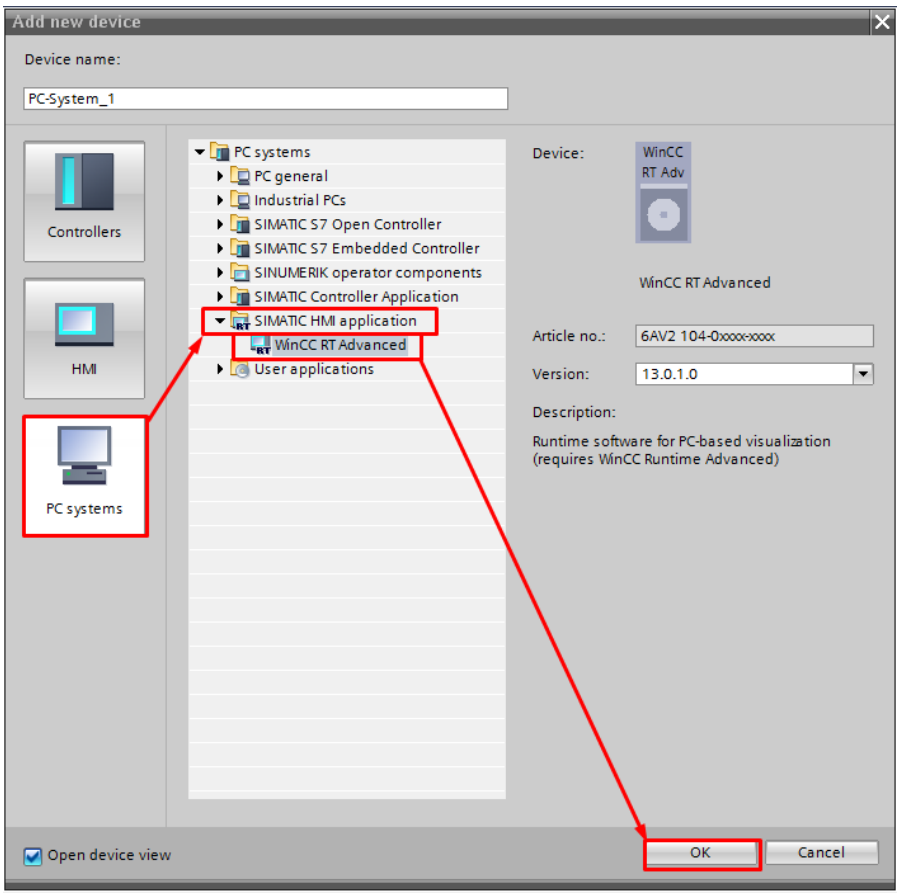



*Slika 4.1. Dodavanje PLC-a*

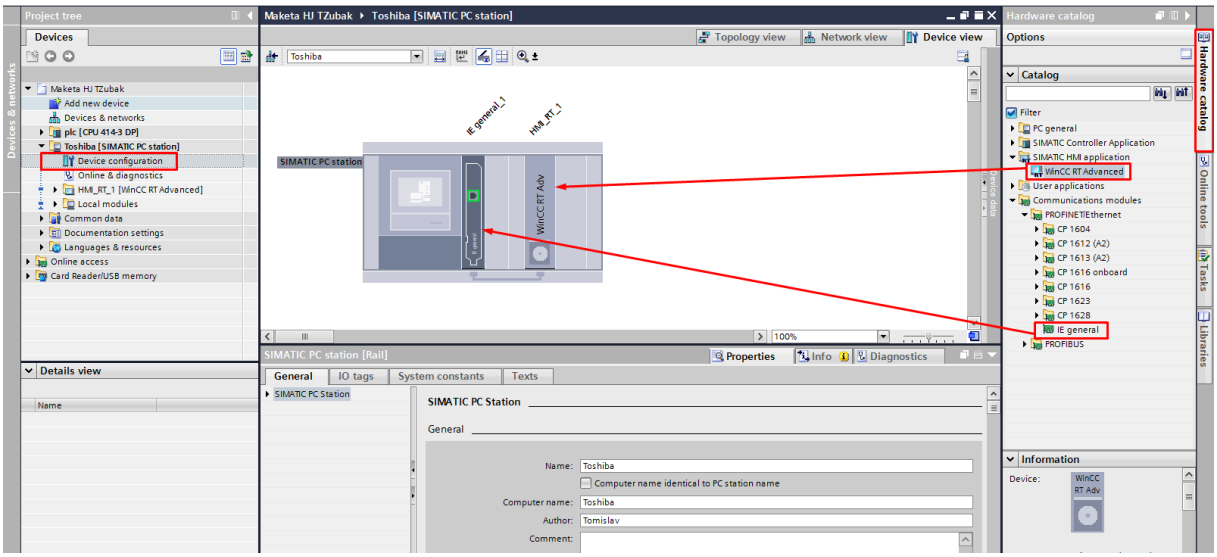



*Slika 4.2. Prikaz hardverske konfiguracije PLC-a (lijevo) i distribuiranog U/I uređaja (desno)*

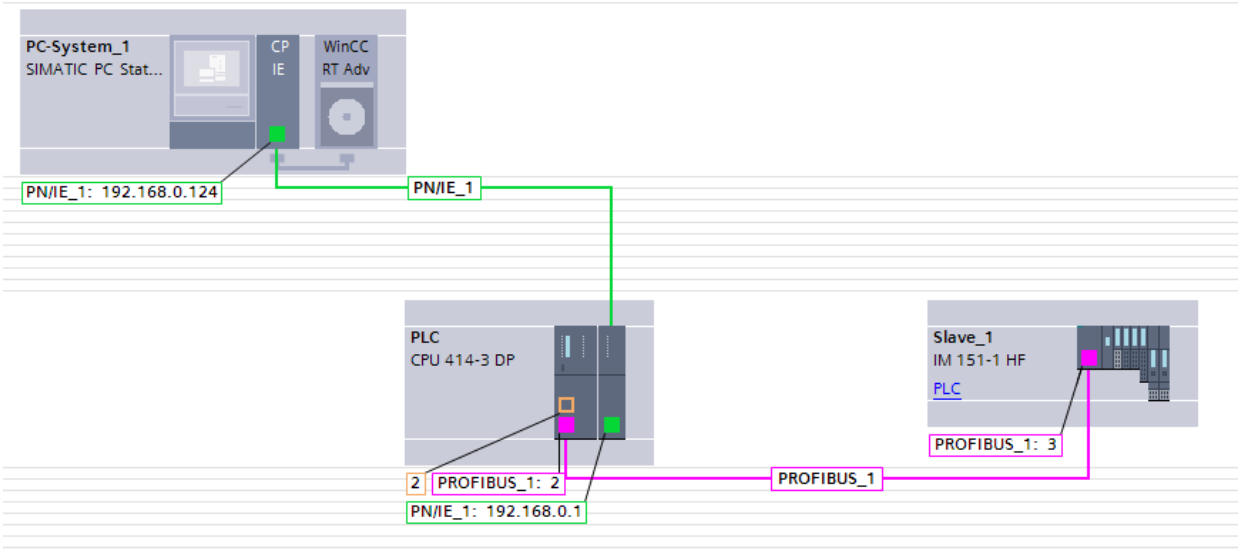
Također je bilo potrebno dodati sustav baziran na osobnom računalu (engl. PC system) (slika 4.3.) za SIMATIC HMI sučelje, odnosno WinCC RT Advanced, te mu pridružiti komunikacijski modul za komunikaciju sa PLC-om preko PROFINET-a (slika 4.4.). Klikom na „Devices & network“ otvara se prozor u kojem se konfigurirani uređaji međusobno povezuju na odgovarajuće podmreže, te je na taj način konfigurirano sklopovsko rješenje i mrežna topologija unutar razvojnog okruženja što je prikazano slikom 4.5. Valja napomenuti da je bilo potrebno podesiti IP adrese uređaja tako da se nalaze na istoj podmreži, te podesiti odgovarajuću podmrežu za sustav grafičkog sučelja.




*Slika 4.3. Dodavanje uređaja za HMI sučelje*




*Slika 4.4. Dodavanje modula uređaju za HMI sučelje*

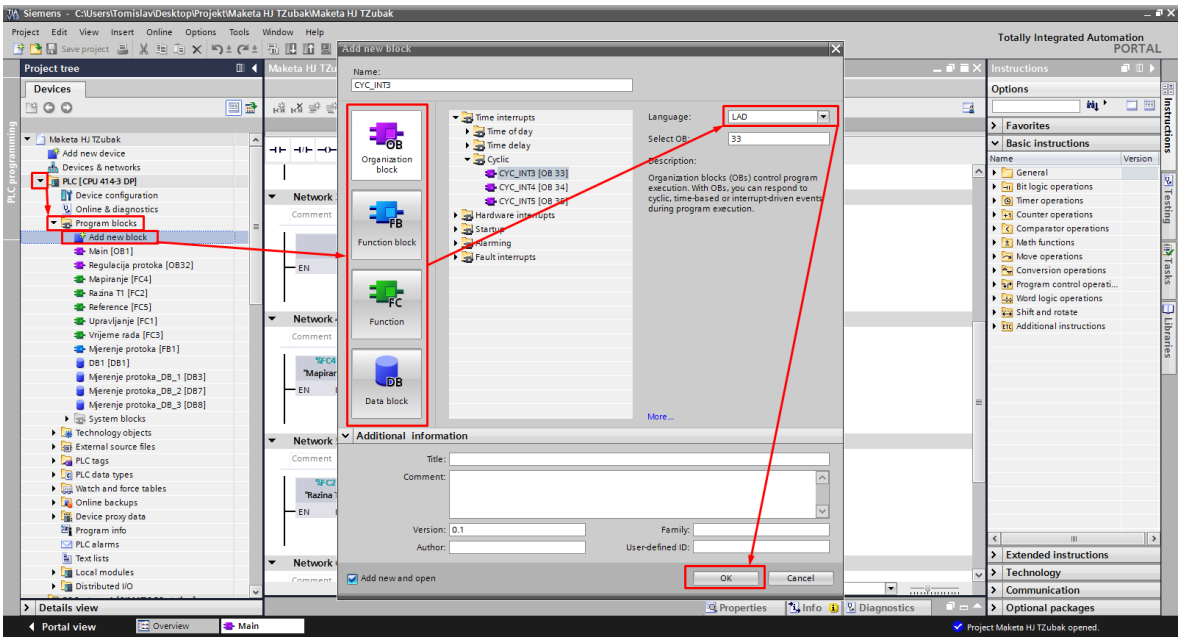



*Slika 4.5. Mrežna topologija korištenih uređaja*

### 4.1.2. Dodavanje programskih blokova

Nakon što je izrađena hrdverska konfiguracija pod izabranim PLC-om moguće je, klikom na „Program blocks“ > „Add new block“ dodati željene programske blokove. Na raspolaganju su organizacijski blokovi koji služe kao prekidi (prekidi u zadano vrijeme, prekidi s vremenskom odgodom, ciklički prekidi, sklopovski prekidi itd.), funkcijske blokove i funkcije opisane u 3.3.2., te podatkovne blokove (engl. Data block) u kojima su organizirani i strukturirani podaci potrebni za izvršavanje korisničkog programa. Organizacijski blok OB1 predefiniran je sustavom, i ima ulogu glavnog programa, odnosno programa koji se ciklički izvršava. Njegova je uloga povezivati korisnički program s operacijskim sustavom CPU-a, te se iz njega pozivaju ostale programske organizacijske jedinice. Slikom 4.6. prikazan je postupak dodavanja novog programskog bloka, i to cikličkog prekida kakav je korišten za regulaciju protoka pomoću sustavnih PID regulatora. Potrebno je, nakon odabira bloka odabrati i željeni programski jezik kojim će se programirati unutar dodanog bloka, što je u ovom slučaju (a i u svim blokovima ovog rada) ljestvičasti dijagram (LAD). U tzv. programskom stablu (lijevo) vidljivi su

programski blokovi pomoću kojih je strukturirano programsko rješenje ovog rada. OB1 i OB32 („Regulacija protoka“) spomenuti su ranije. DB1 služi kao memorijski prostor u kojem su organizirane varijable potrebne za rad programskog rješenja. FC4 („Mapiranje“) funkcija je u kojoj se povezuju varijable iz podatkovnog bloka DB1 sa adresama fizičkih ulaza i izlaza. FC2 („Razina T1“) funkcija je u kojoj je rješenje za prilagodbu mjernog signala dobivenog od senzora razine u glavnom spremniku. FC5 („Reference“) formira upravljačke signale za upravljanje protokom crpki na temelju vrijednosti zadane na HMI sučelju. U funkciji FC1 („Upravljanje“) nalazi se logika potrebna za odgovarajući način rada makete ovisno o izabranom načinu rada (automatskom ili ručnom), a to se odnosi na uključivanje i isključivanje crpki, otvaranje i zatvaranje ventila, uz ostvarenje uvijeta pod kojim se elementima upravlja. Unutar funkcije FC1 pozivan je i funkcijski blok FB1 („Mjerenje protoka“) tri puta za svaki mjerač protoka, a logika podatkovnog bloka opisana je u potpoglavlju 4.1.5., a svako pozivanje funkcijskog bloka stvoren je tzv. pridruženi podatkovni blok (engl. Instance data block) („Mjerenje protoka_DB“).

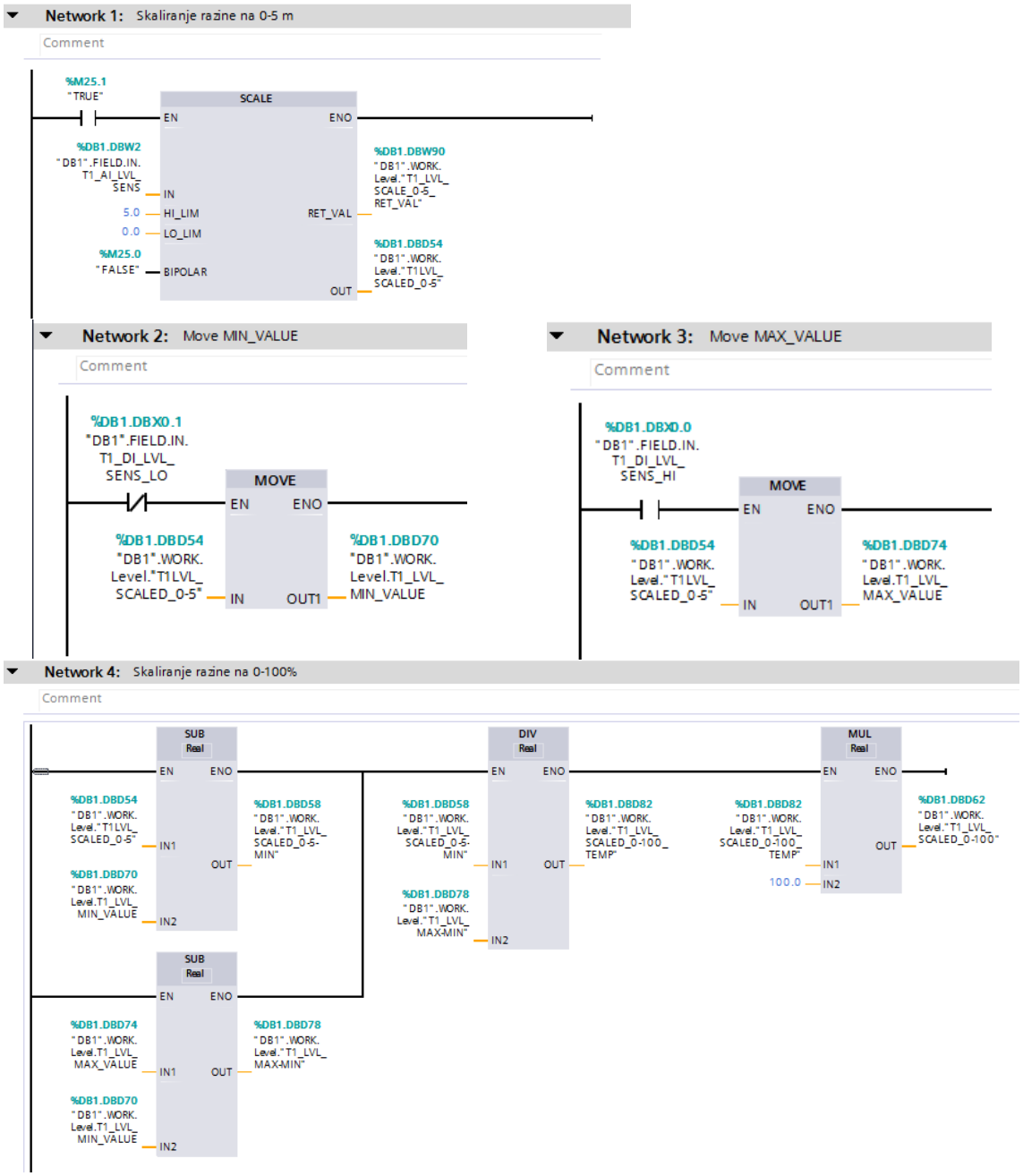



*Slika 4.6. Dodavanje programskih organizacijskih jedinica*

### 4.1.2. Mjerenje razine

Analogni mjerač razine ima od proizvođača zadan nazivni raspon mjerenja, a on iznosi 0 do 5 m. Kao što je spomenuto u potpoglavlju 3.2.4. njegov se mjerni signal šalje analognoj mjernoj kartici PLC-a strujnim signalom 4-20 mA. Digitalizacija tog analognog signala izvršena je u mjernoj kartici, a skaliranje te vrijednosti u inženjerskim jedinicama omogućeno je funkcijom „SCALE“. Na ulaz (IN) funkcije dovodi se vrijednost signala spremljena u varijablu tipa int iz podatkovnog bloka DB1 (varijabla T1_AI_LVL_SENS), te se, koristeći navedenu funkciju skalira na raspon od 0 do 5 (HI_LIM=5, LO_LIM=0) što odgovara mjernom rasponu senzora i konačno, kao skalirana sprema u varijablu T1 LVL_SCALED_0-5.

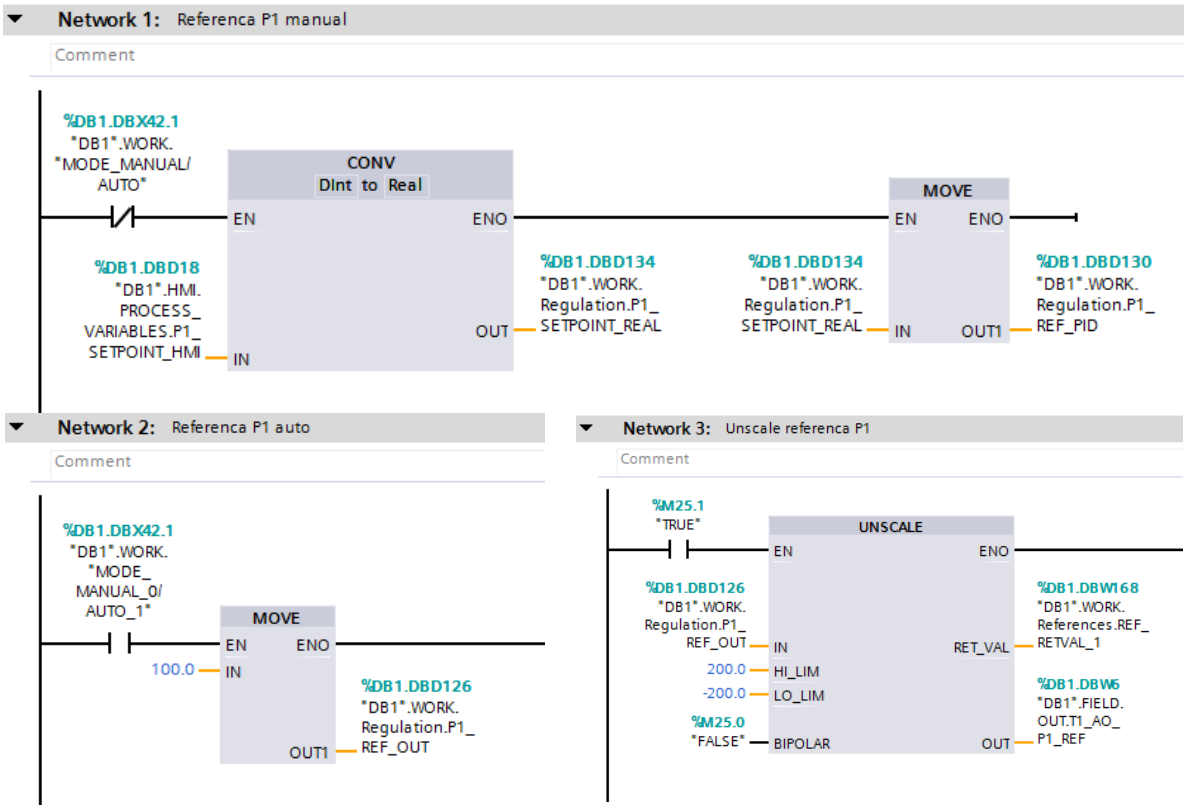
Kako bi se otklonio utjecaj atmosferskog tlaka (korišteni senzor radi na principu mjerenja hidrostatskog tlaka) u varijablu TI_LVL_MIN_VALUE spremljen je iznos koji daje senzor u trenutku kada donji senzor krajnje razine signalizira da je spremnik prazan. U varijablu T1_LVL_MAX_VALUE spremljena je vrijednost u trenutku kada gornji krajnji senzor razine signalizira da je spremnik pun kako bi se kasnije ta vrijednost koristila za skaliranje razine u postotke. Pomoću matematičkih operacija izrađena je logika pretvorbe mjerenog signala u željeni oblik. Izuzet je samo korišteni dio mjernog raspona senzora, te se, trenutna vrijednost razine u omjeru sa vrijednosti punog spremnika i množenjem sa 100 dobiva raspon razine u granicama od 0 do 100 % popunjenosti spremnika. Ta je vrijednost kasnije korištena za prikaz na HMI sučelju, te kao povratna informacija o stvarnom stanju tekućine u spremniku kod načina rada za podešavanje razine u glavnom spremniku. Navedeni postupak pretvorbe signala prikazan je slikom 4.7. koja prikazuje logičku interpretaciju navedenog postupka.




*Slika 4.7. Programsko rješenje za skaliranje vrijednosti razine u spremniku*

### 4.1.3. Zadavanje referentnih vrijednosti za upravljanje protokom crpki

Kako bi se omogućilo željeno vladanje crpki glavnog kruga potrebno im je zadavati referentnu vrijednost protoka koja predstavlja željenu radnu točku. Kako je prikazano slikom 4.8. referentne vrijednosti za upravljanje protokom crpki glavnog kruga u ručnom načinu rada (varijabla „MODE_MANUAL/AUTO“ u stanju logičke nule) zadaju se od strane korisnika preko HMI sučelja u rasponu od 0 do 100% i spremaju u varijable unutar DB1. P1_SETPOINT_HMI varijabla je u koju je spremljena referentna vrijednost crpke P1 (jedne od crpki glavnog kruga). Pomoću funkcije „Slider“ vrijednost se zadaje u DInt formatu pa ju je potrebno pretvoriti u Real tip podatka funkcijom „CONV“. Nakon toga funkcijom „MOVE“ vrijednost se prebacuje u varijablu P1_REF_PID koja se koristi kao referentna vrijednosti regulatora protoka opisanog u idućem potpoglavlju. U drugoj mreži (network 2) zadana je fiksna referentna vrijednost od 100% jer je u automatskom načinu rada izvedeno podešavanje razine glavnog spremnika, te je potrebna veća brzina djelovanja radi brže reakcije sustava na podešenu vrijednost razine. Funkcija „UNSCALE“ ima suprotnu ulogu od funkcije „SCALE“ spomenute u poglavlju 4.1.2., točnije, ona se koristi kako bi raspon zadan točkama „HI_LIM“ i „LO_LIM“ (u ovom slučaju između -200 i 200) pretvorila u inženjerske jedinice pogodne za slanje na izlaznu analognu karticu u kojoj se izvršava analogno/digitalna pretvorba, te se signal pretvara u analogni procesni signal za slanje informacija postavnim članovima procesa. Granice -200 i 200 odabrane su kako bi se vrijednost u obliku postotaka ograničila na raspon izlaznog naponskog signala od 0 do 5 V (prikazano u poglavlju 3.2.4., slika 3.5.) što je napon koji odgovara logičkoj razini napona mikroupravljača Arduino Nano.

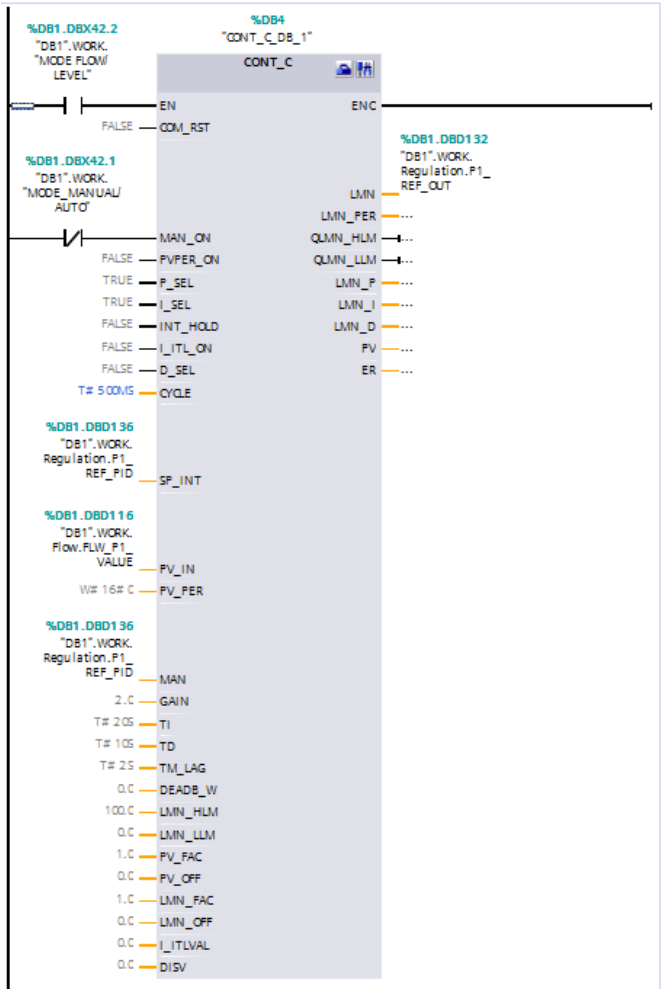



*Slika 4.8. Programsko rješenje za zadavanje referentnih vrijednosti za upravljanje protokom crpki*

### 4.1.4. PID regulator

Kao što je spomenuto u potpoglavlju 2.3. ulogu regulatoru u regulacijskom krugu ima softverska funkcija unutar razvojnog okruženja. Za tu namjenu odabrana je funkcija (instrukcija) „CONT_C“ prikazana slikom 4.9. čija je uloga biti softverski PID regulator za tehničke sustave s kontinuiranim ulaznim i izlaznim signalima. Ima relativno velik broj ulaznih i izlaznih parametara, a u odnosu na unaprijed dodijeljene parametre potrebno je bilo dodati referentne vrijednosti za protok crpki (parametar „SP_INT“), vrijednost mjernog signala protoka (parametar „PV_IN“), regulirani izlazni signal iz regulatora (parametar „LMN“), povezati parametar „MAN_ON“ sa varijablom „MODE_MANUAL/AUTO“ kako bi se iz grafičkog sučelja mogao uključivati i isključivati utjecaj regulatora kako bi se odzivi sa regulatorom i bez njega mogli usporediti. Za regulaciju protoka svake od tri crpke glavnog kruga korištena je po jedna funkcija regulatora, te su postavljene u OB32 što je funkcija cikličkog prekida.

Ciklički prekid jest prekid koji se pokreće u periodičnim razdobljima i inicira izvođenje organizacijskog bloka za obradu cikličkog prekida. Ciklički prekid može se koristiti za izvođenje programa u vremenskom intervalu koji je neovisan o vremenu izvođenja cikličkog programa. Kod CPU 300 za cikličke prekide koriste se organizacijski blokovi od OB32 do OB 35. Kod CPU 400 koriste se organizacijski blokovi od OB 30 do OB 38. Koji su od tih devet organizacijskih blokova zapravo dostupni, ovisi o korištenom CPU-u. [9]




*Slika 4.9. Funkcija CONT_C korištena kao regulator protoka*

### 4.1.5. Mjerenje protoka

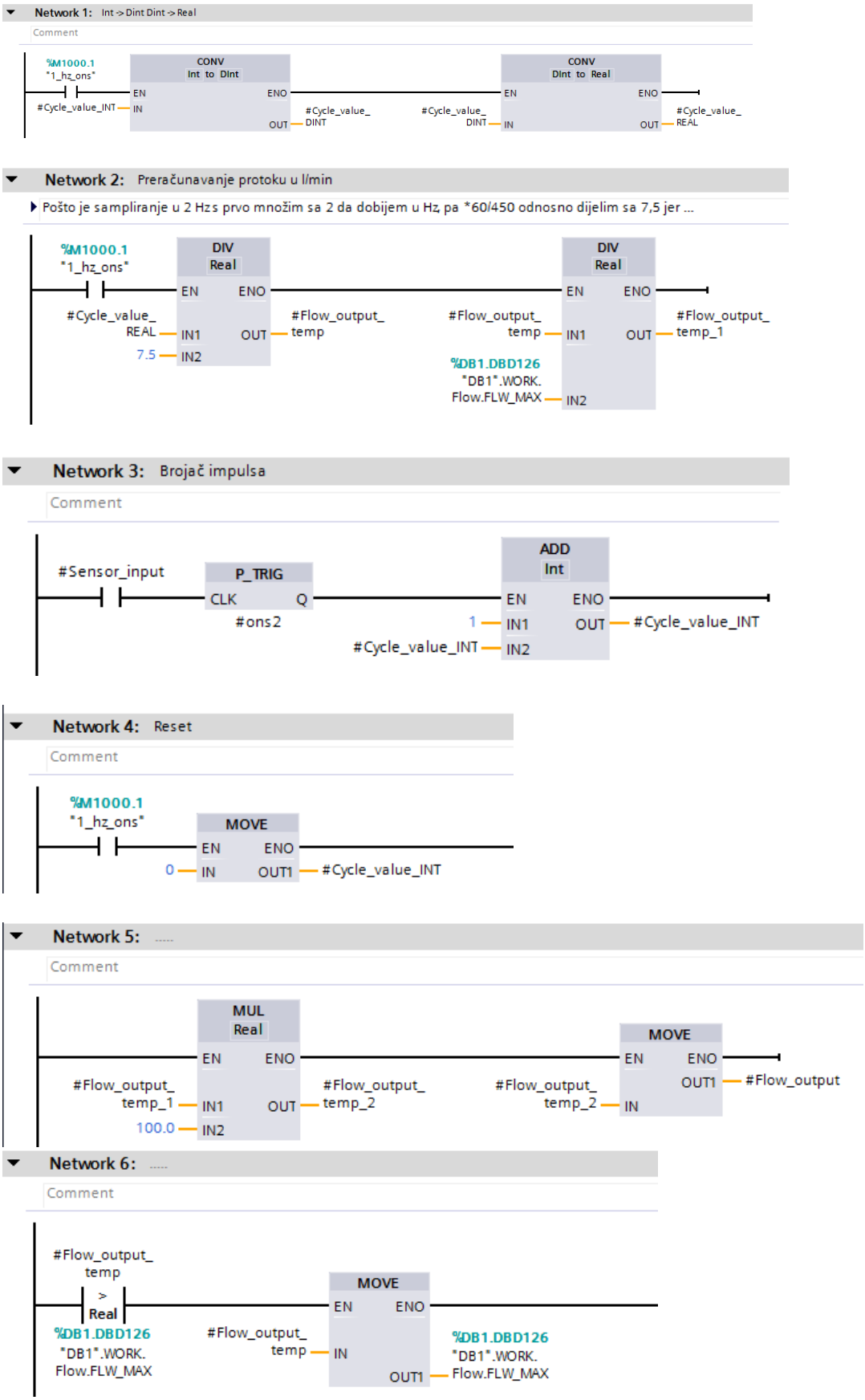
Slikom 4.10. prikazano je kako je softverski izvedeno mjerenje protoka fluida koristeći mjerač protoka sa Hall-ovim senzorom opisan u potpoglavlju 2.1.2. koji kao rezultat mjerenja daje niz impulsa koji odgovara omjeru 450 impulsa po litri. Mreža 8 (network 8) prikazuje kako izgleda poziv funkcijskog bloka FB1 „Mjerenje protoka“, gdje varijabla T1_DI_FLW1 predstavlja vezu s digitalnim ulaznim modulom PLC-a na koji je spojen senzor. Na ONS spojeno je memorijsko mjesto koje služi funkciji za detekciju pozitivnog brida, a rezultat mjerenja kao real tip podatka spremljen je u varijablu FLW_P1_VALUE. Mjerenje je izvedeno tako što su preko detekcije pozitivnog brida brojani impulsi brojačem CTU i spremljeni u lokalnu varijablu Cycle_value_INT koja sprema vrijednost jednog ciklusa uzorkovanja koje se izvršava bitom 3 taktnog memorijskog bajta koji je, u konfiguraciji CPU uključen na adresi 200. U tablici 4.1. navedene su dostupne frekvencije i odgovarajući bitovi u taktnom memorijskom bajtu pomoću kojih ih je moguće koristiti.

Mjerenje protoka odnosi se na logičko rješenje brojanja niza impulsa koje, kao rezultat okretanja kola s lopaticama daje mjerač protoka zasnovan na Hall-ovom senzoru. Slikom 4.7. prikazan je način na koji se izvodi to preračunavanje. U network-u 3 detektiraju se pozitivni bridovi impulsa sa senzora, te se funkcijom ADD za svaki pozitivni brid pribroji 1 i spremi u lokalnu varijablu Cycle_value_in. U prvom network-u ta se vijednost preračunava najprije u Dint, a zatim u Real format kako bi bila pogodna za dijeljenje.U drugom network-u frekvencijom 1 Hz poništava se ciklička vrijednost s brojača, te se na taj način omogućava uzorkovanje (engl. sampling) signala frekvencijom 1 Hz. U šestom network-u sprema se najveća vrijednost protoka kako bi se u network-u 5 u odnosu na nju mogao odrediti protok u postotcima

*Tablica 4.1. Vrijednosti frekvencije pulsiranja bitova taktnog memorijskog bajta*

| R. br. | bit | Frekvencija pulsiranja |
|---|---|---|
| 1 | .0 | 10 Hz |
| 2 | .1 | 5 Hz |
| 3 | .2 | 2,5 Hz |
| 4 | .3 | 2 Hz |
| 5 | .4 | 1,25 Hz |
| 6 | .5 | 1 Hz |
| 7 | .6 | 0,625 Hz |
| 8 | .7 | 0,5 Hz |

Vrijednost ciklusa osvježava se dakle frekvencijom 1 Hz, tako da se resetira vrijednost brojača, te izvedu matematičke operacije i dijeljenja sa 7.5 (množenje sa 60 i dijeljenje sa 450) da se dobije vrijednost protoka u litrama u minuti. Taj se iznos nakon toga skalira na vrijednost od 0 do 100 % kako bi se u takvom obliku postavio kao parametar regulatora.

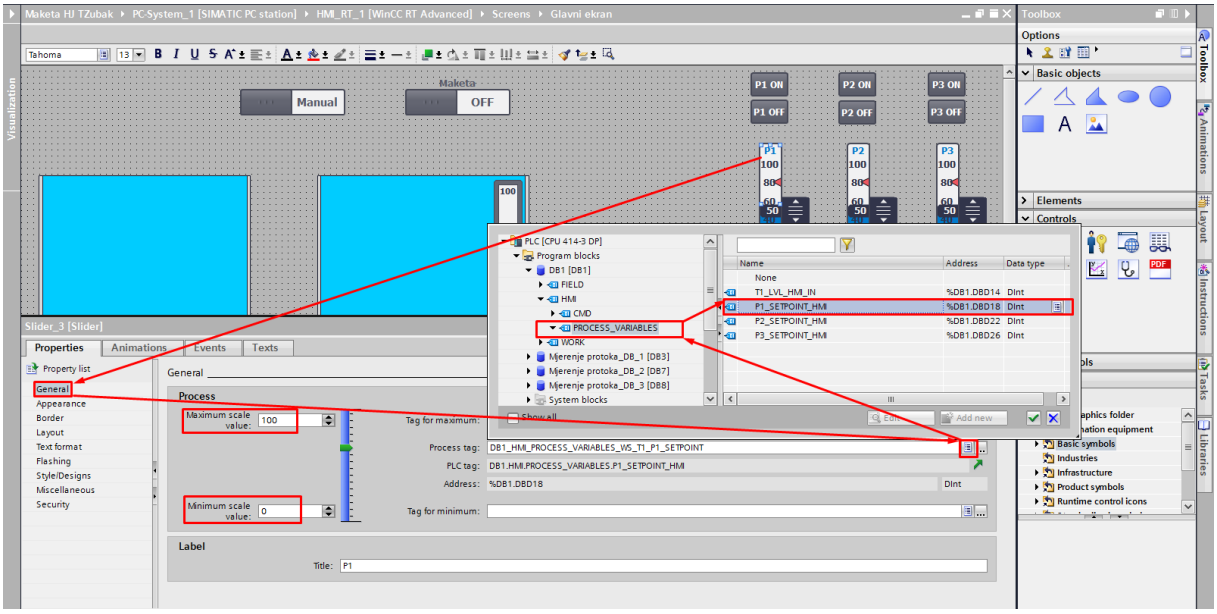



*Slika 4.10. Mjerenje protoka fluida mjeračem protoka sa Hall-ovim senzorom*

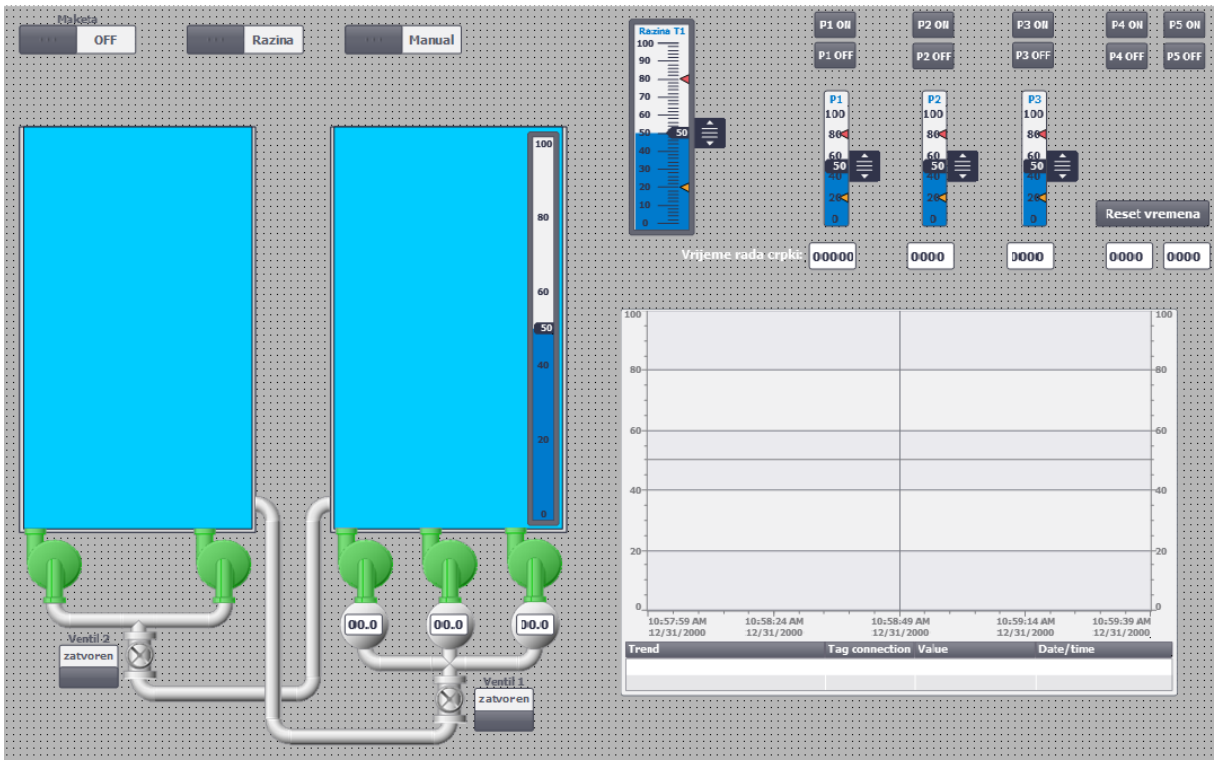
### 4.1.6. Grafičko sučelje WinCC Advanced

**WinCC unutar TIA portala** dostupan je u verzijama Basic, Confort, Advanced i Professional. Inženjerski alati uvelike su određeni uređajem koji je potrebno konfigurirati, pri čemu viša verzija softvera za inženjering uključuje mogućnosti niže verzije softvera. Nadogradnja na višu verziju dostupna je s pomoću *PowerPacka*, osim za WinCC Basic. Integracija unutar TIA portala omogućuje isto radno okruženje za konfiguriranje HMI jedinica i programiranje PLC jedinica, što omogućava zajedničke simbole, zajedničko upravljanje podacima i dijagnostiku sustava kao sastavni dio. Sučelje za inženjering ima isti izgled za HMI i PLC jedinice i na taj se način povećava jednostavnost uporabe. Nema više razlika između HMI i PLC projekata. HMI i PLC jedinice (također i PC jedinice) nalaze se u jednom projektu i mogu se umrežiti preko komunikacijske veze koja se jednom konfigurira.[9]

Pri izradi ovog rada korištena je verzija softvera WinCC RT Advanced V13 integrirana unutar TIA Portala, a u nastavku je prikazan način na koji se dodaju grafički elemnti sučelja, na koji se način grafički elementi povezuju sa varijablama iz projekta, te konačno grafičko sučelje makete hidrauličke jedinice. U projektnom stablu pod SIMATIC PC station nalazi se mapa „Screens“. Klikom na „Add new screens“ dodaje se novi ekran grafičkog sučelja. Projekt može imati više ekrana koji se međusobno povezuju i na željeni način uključuju i isključuju. Slika 4.11. prikazuje izgled sučelja za uređivanje prozora gdje se, u središnjem okviru nalaze raspoređeni elementi sučelja, desno se nalaze objekti vizualizacije podijeljeni u skupine. Objektima (elementima) vizualizacije dodjeljuju se grafička svojstva poput dimenzija, pozicije, transparentnosti itd. Uz elemente vežu se varijable PLC-a, a način dodjeljivanja varijable elementu označen je na slici. Slikom 4.12. prikazan je izgled konačnog grafičkog sučelja.



Slika 4.11. Izgled uređivača grafičkog sučelja, te dodjeljivanje varijable grafičkom elementu




*Slika 4.12. Izgled konačnog grafičkog sučelja*

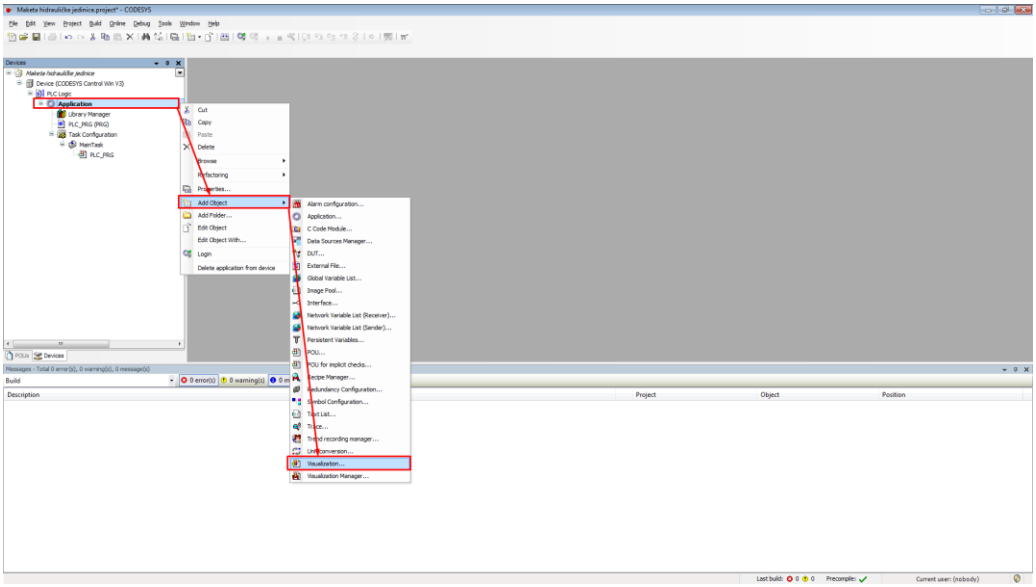
## 4.2. CODESYS

CODESYS (engl. *Controller Development System*) razvojno je okruženje za razvoj programskih rješenja (programiranja PLC-ova, i stvaranja vizualizacije) za sustave u kojima se primjenjuju programabilni logički kontroleri. Proizvedeno je od strane njemačke kompanije 3S-Smart Software Solutions, a prva verzija programa puštena je na tržište 1994. godine. Okruženje je temeljeno na IEC 61131-3 standardu spominjanom u potpoglavlju 3.2., te podržava programiranje u šest standardnih programskih jezika: ljestvičasti dijagram (LD), strukturirani tekst (ST), funkcijski blok dijagram (FBD), instrukcijsku listu (IL), sekvencijalni funkcijski dijagram (SFC) i neprekidni funkcijski dijagram (CFC). Programsko okruženje moguće je besplatno preuzeti sa stranice službene stranice CODESYS (https://www.codesys.com) te se može koristiti nakon besplatne registracije.

U programskom alatu CODESYS verzije V3.5 SP10 izrađena je simulacija rada makete hidrauličke jedinice po uzoru na programsko rješenje u programskom okruženju Siemens TIA portal. Zbog sličnosti u logici programskog rješenja nisu prikazana logička rješenja.

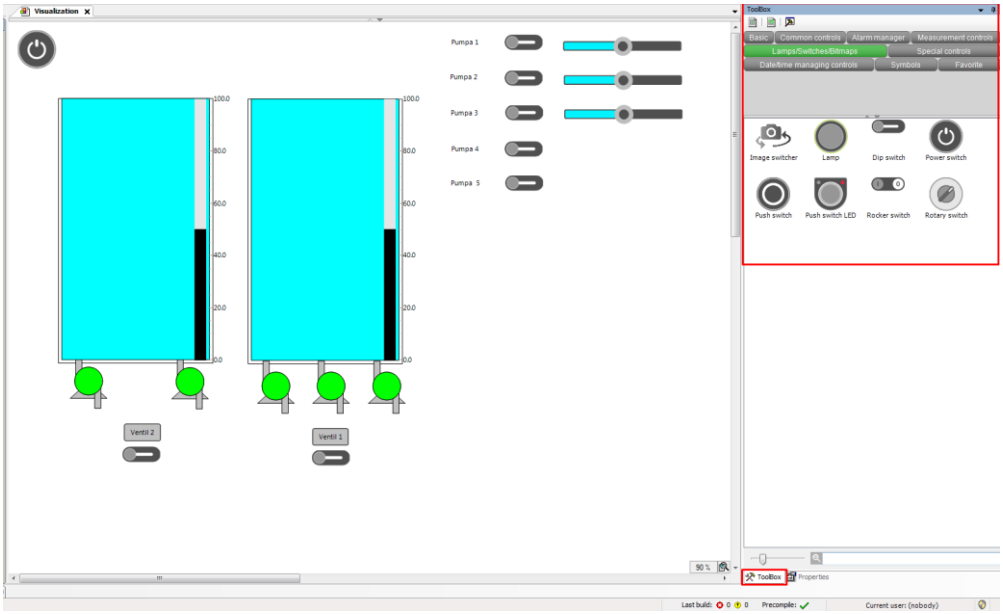
### 4.2.1. Izrada sučelja vizualizacije

Nakon dodavanja novog projekta, izbora predloška „Standard project“ i dodavanja imena projekta i lokacije, željenog uređaja (CODESYS Control Win V3) i programskog jezika (Ladder Logic Diagram (LD)) otvara se sučelje programa prikazano slikom 4.13. Klikom na „Aplication“, „Add Object“, „Visualization“, te dodavanja biblioteke simbola VisuSymbols (System) započinje izrada cjelokupnog projekta razvijanja logike upravljanja simulacijom makete hidrauličke jedinice i grafičkog sučelja za upravljanje. Nakon dodavanja blok potprograma vizualizacije nalazi se pod grupom „Visualization Manager“.



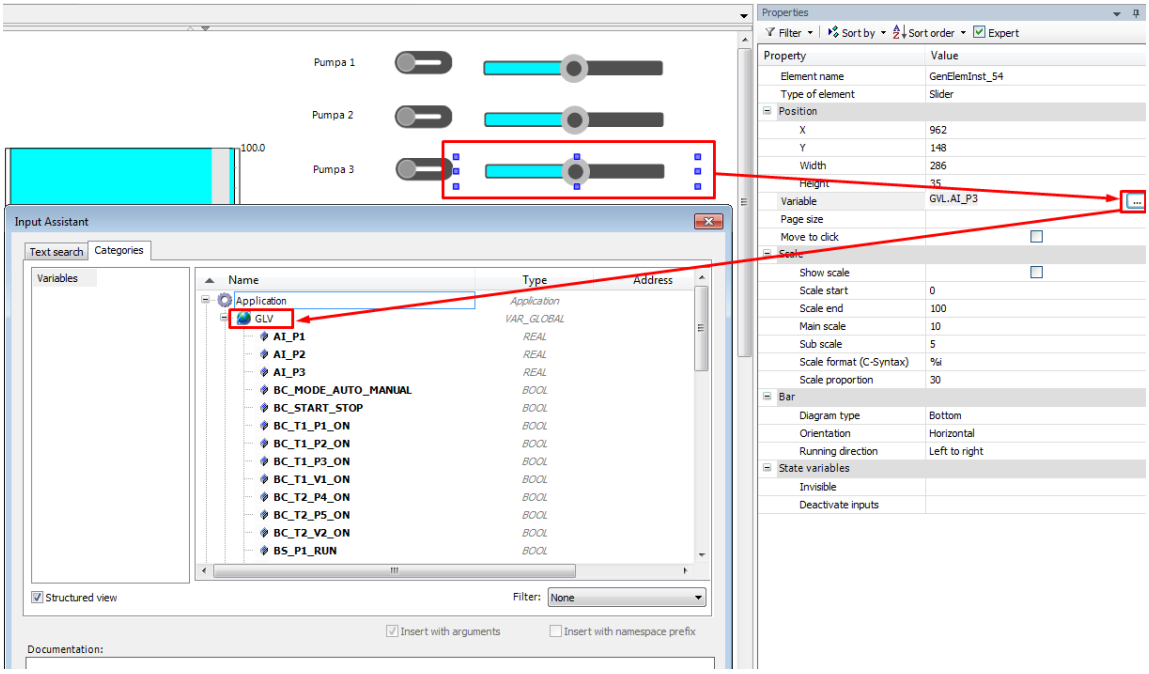
*Slika 4.13. Dodavanje objekta za konfiguriranje vizualizacije*

Sučelje za izradu vizualizacije prikazano je slikom 4.14. te prikazuje središnji prozor gdje se raspoređuju grafički elementi vizualizacije, te prozor „Toolbox“ gdje su ponuđeni grafički elementi u grupama. Nakon raspoređivanja elemenata slijedi parametriranje pojedinog elementa.

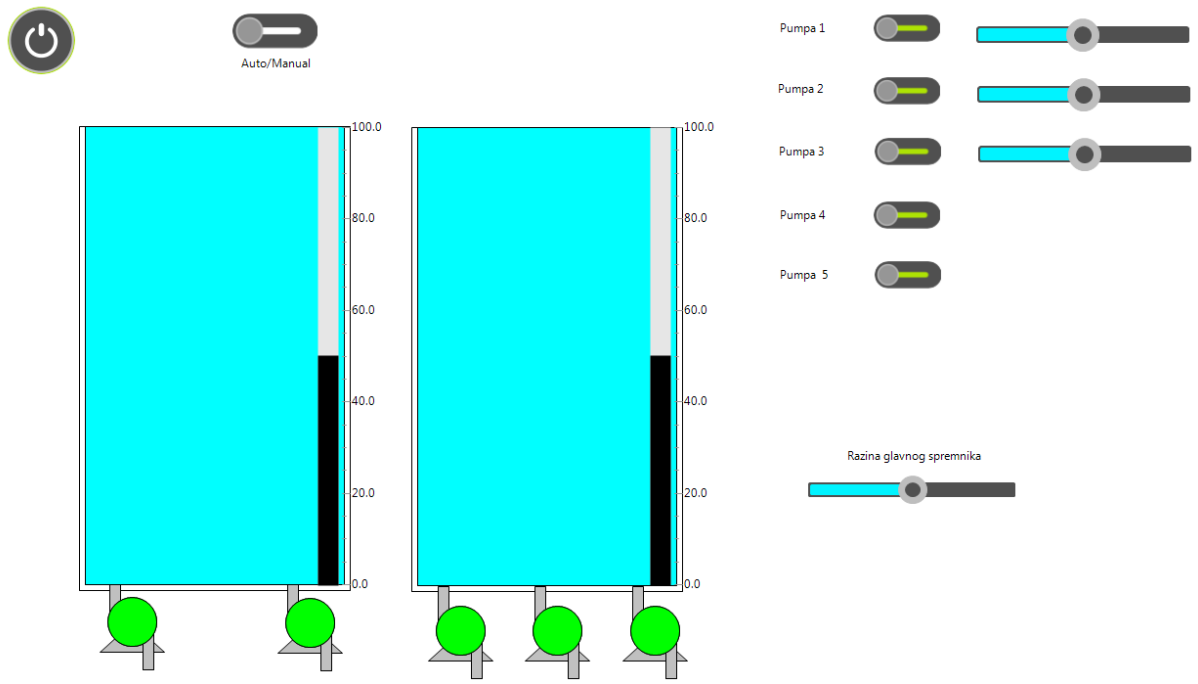



*Slika 4.14. Izgled prozora za uređivanje grafičkog sučelja*

Nakon dodavanja pojedinog elementa potrebno je odrediti nejgova svojstva. Dvostrukim klikom na element otvara se prozor „Properties“ unutar kojega je moguće odrediti stvojstva elementa poput dimenzija, položaja, boje, vezati ga uz procesnu ulaznu ili izlaznu varijablu, podešavati vidljivost elementa varijablom itd. Slikom 4.15. prikazano je dodljeljivanje svojstava slideru za upravljanje protokom pumpe P3. Klikom na označenu ikonu otvara se „Input Assistant“ unutar kojega se nalazi definirana globalna lista varijabli iz koje se pridružuje željena varijabla, a slikom 4.16. konačni izgled grafičkog sučelja u CODESYS-u.



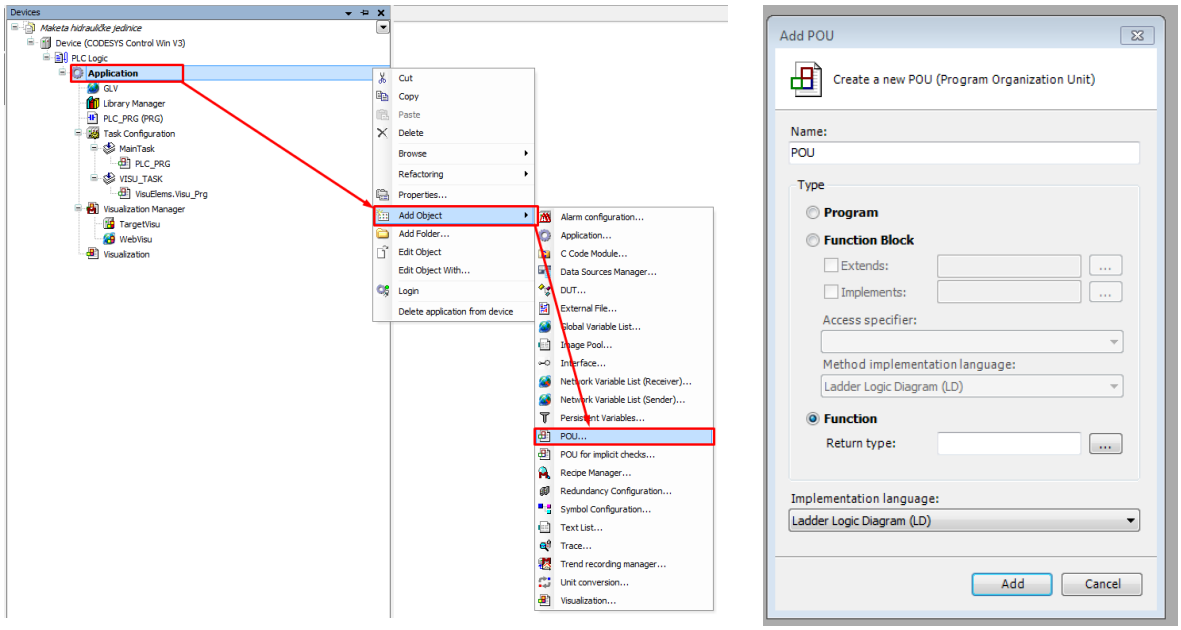

*Slika 4.15. Dodjeljivanje svojstava grafičkom elementu*



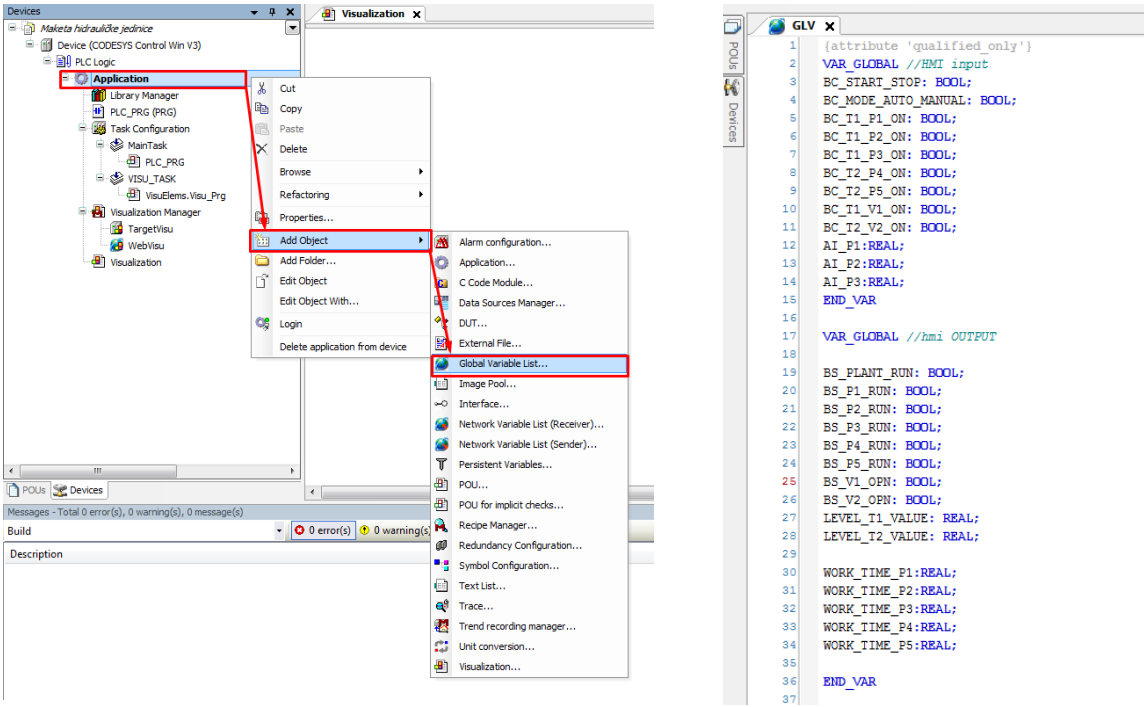

*Slika 4.16. Konačni izgled grafičkog sučelja*

### 4.2.2. Programske organizacijske jedinice i globalne varijable

Dodavanje programske organizacijske jedinice prikazano je slikom 4.17. Moguće je izabrati između tri POU definirane IEC standardnom, a uz odabir jedinice potrebno je definirati programski jezik kojim se želi programirati, te tip podatka koji jedinica vraća. Na slici 4.18. prikazano je kako se dodaje nova globalna lista varijabli.




*Slika 4.17. Dodavanje programske organizacijske jedinice*




*Slika 4.18. Dodavanje globalne liste varijabli (lijevo) i način grupiranja globalnih varijabli u listi (desno)*

# 5. Zaključak

U ovom radu opisan je proces izrade makete hidrauličke jedinice koja predstavlja primjer tehničkog procesa. Potrebno je bilo, nakon izrade makete omogućiti upravljanje maketom pomoću programabilnog logičkog kontrolera (PLC), te vizualizaciju procesa pomoću grafičkog (HMI) sučelja. Hardverska konfiguracija sastoji se od PLC-a Siemens S7 400, te distribuiranog ulazno/izlaznog uređaja ET200S. Kako bi se omogućilo upravljanje iznosom protoka crpki potrebno je bilo načiniti elektronički sklop temeljen na Arduino Nano mikroupravljaču kako bi se omogućila pretvorba analognih signala u pulsno širinski moduliran naponski signal (PWM) za upravljanje srednjom vrijednosti napona. Realiziran je regulacijski krug za regulaciju protoka pojedine crpke koji se sastoji od softverskog regulatora, crpki sa elektroničkim sklopom za upravljanje kao izvršnim članom i mjeračem protoka kao mjernog člana za davanje informacije o trenutnom iznosu protoka koji se povratnom vezom šalje na „ulaz“ regulacijskog kruga. Opisani su programski alati korišteni za razvoj programskog rješenja, te dijelovi programskog rješenja i rješenja izrade grafičkog sučelja.

# Literatura

[1] N. Perić, I. Petrović, M. Vašak, Procesna automatizacija, FER, Zagreb 2013.
[2] Z. Valter, Procesna mjerenja, ETFOS, Osijek, 2008., ISBN: 978-953-6032-59-4
[3] J. Weidauer, Električna pogonska tehnika, Graphis d.o.o., Zagreb, 2013., ISBN: 978-953-279-032-0
[4] Electronicshub, Valni oblici pulsno-širinski moduliranog signala
url: http://www.electronicshub.org/wp-content/uploads/2016/11/Arduino-PWM-2.png
[5] El-component, tranzistor IRF530N
url: http://www.el-component.com/images/mosfet/irf530n-pinout.jpg
[6] Makezine, Raspored pinova mikroupravljača Arduino Uno
https://makezine.com/wp-content/uploads/2013/02/arduino_uno_pinout_web.png
[7] Perić, Automatsko upravljanje – predavanja; Sveučilište u Zagrebu, FER, Zagreb, 2004.
[8] G. Malčić, Programirljivi logički kontroleri, Tehničko veleučilište u Zagrebu, Elektrotehnički odjel, Zagreb, 2007.
url: http://nastava.tvz.hr/gmalcic/PLC_skripta_TVZ.pdf
[9] H. Berger, SIMATIC automatizacijski sustavi, SIEMENS, Graphis d.o.o., Zagreb, 2013., ISBN: 978-953-279-031-3

# Sažetak

U ovom diplomskom radu izrađena je maketa hidrauličke jedinice koja se sastoji od dva spremnika tekućina opremljenih crpkama, ventilima, senzorima razine i protoka. Ta maketa predstavlja primjer tehničkog procesa nad kojom provodi proces automatizacije. Maketa je upravljana PLC-a Siemens SIMATIC S7 400, a upravljanje i nadzor vrše se pomoću HMI sučelja. Programsko rješenje logike upravljanja i vizualizacije razvijeno je u programskim alatima Siemens STEP 7 i WinCC, te CODESYS. Opisane su komponente od kojih se maketa sastoji kao i princip rada. Opisan je postupak mjerenja protoka i razine tekućine u spremniku pomoću PLC-a, sklop temeljen na Arduino Uno mikroupravljaču koji služi za upravljanje protokom pumpe, te regulacijski krug za regulaciju protoka uz ručno dodan poremećaj. Osim navedenog opisani su i programabilni logični kontroleri i pripadajući IEC standard te komunikacijske mreže korištene u industriji.

**Ključne riječi:**

Programabilni logički kontroler (PLC), Siemens STEP 7, Siemens WinCC, TIA Portal, CODESYS, IEC 61131-3, PROFINET, PROFIBUS, HMI sučelje, PID regulator, PWM modulacija, Arduino

# Abstract

A hydraulic unit model consisting of two fluid reservoirs equipped with pumps, valves, level and flow sensors is created. This model represents an example of an automatically controlled technical process.. The model is controlled by a Siemens SIMATIC S7 400 PLC, and the control and monitoring are performed using a HMI interface. The program solution for control and visualization was developed in Siemens STEP 7, WinCC and CODESYS. The components of which the model is built are described and their operating principle is explained. The flow and level measurement procedure in the tank is described. The flow and level measurement is implemented using PLC, an Arduino Uno-based microcontroller assembly for pump flow management, and a flow control circuit with a manual disruption option. Furtheremore, the programmable logic controllers, the corresponding IEC standard and communication networks used in the industry are described.



**Keywords:**

Programmable logic controller (PLC), Siemens STEP 7, Siemens WinCC, TIA Portal, CODESYS, IEC 61131-3, PROFINET, PROFIBUS, HMI, PID controller, PWM modulation, Arduino

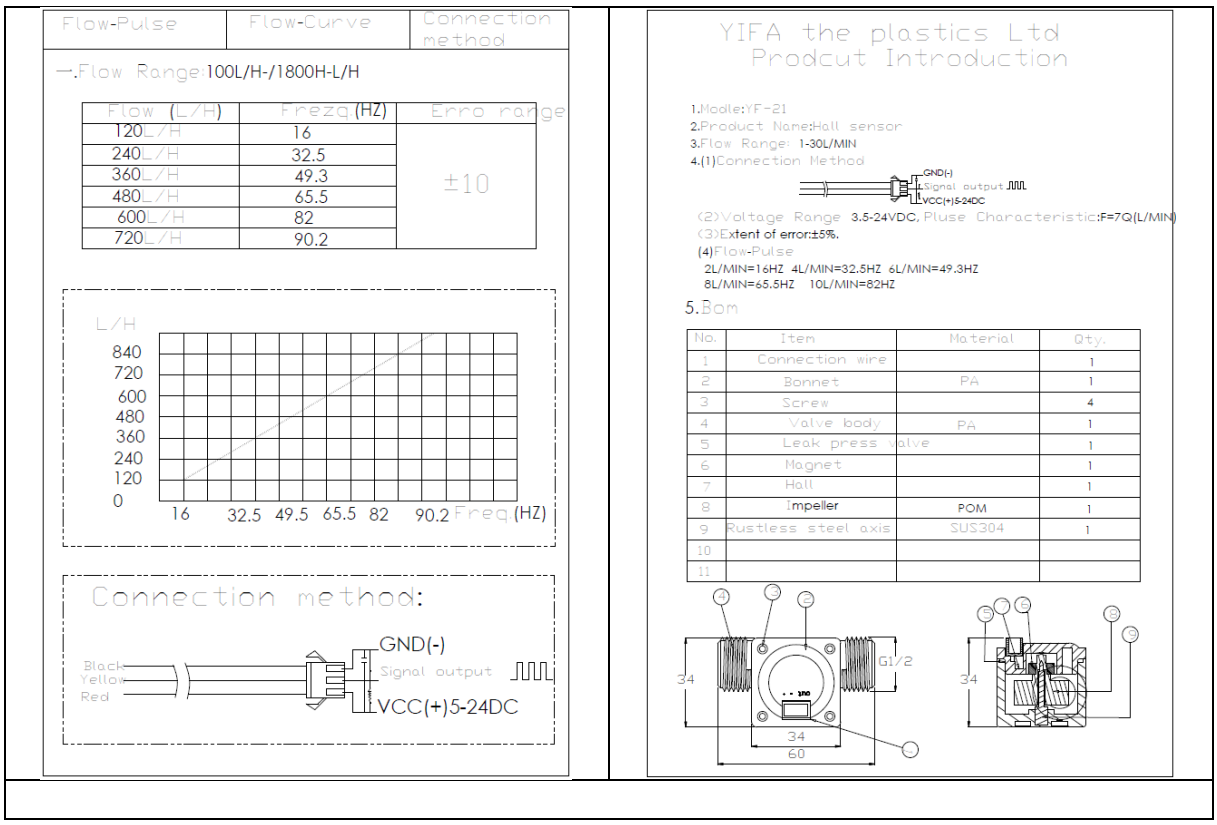
# Životopis

Tomislav Zubak rođen 1.6.1993. iz Daruvara student je druge godine diplomskog studija elektrotehnike na Fakultetu elektrotehnike, računarstva i informacijskih tehnologija u Osijeku smjer industrijska elektroenergetika. Završio je srednju Tehničku školu u Daruvaru, te godine 2012. upisao preddiplomski studij elektrotehnike.
Informatički je pismen, te je tokom studiranja stekao vještine rada u alatima poput Siemens SIMATIC STEP7 i WinCC, AutoCAD, MATLAB, Eplan.
Engleski jezik poznaje u govoru i pismu, te posjeduje vozačku dozvolu B kategorije. Za sebe smatra da je perspektivna i motivirana osoba željna novih znanja iz područja tehničke struke. U slobodno vrijeme svira trombon, te je završio srednju glazbenu školu Brune Bjelinskog u Daruvaru.

Tomislav Zubak

# Prilozi

Prilog 1. Tehničke specifikacije mjerača protoka



Flow-Pulse | Flow-Curve | Connection method

一.Flow Range:100L/H-/1800H-L/H

| Flow (L/H) | Freza (HZ) | Erro range |
|---|---|---|
| 120L/H | 16 | ±10 |
| 240L/H | 32.5 | |
| 360L/H | 49.3 | |
| 480L/H | 65.5 | |
| 600L/H | 82 | |
| 720L/H | 90.2 | |



Connection method:



YIFA the plastics Ltd
Prodcut Introduction

1.Modle:YF-21
2.Product Name:Hall sensor
3.Flow Range: 1-30L/MIN
4.(1)Connection Method



(2)Voltage Range 3.5-24VDC, Pluse Characteristic:F=7Q(L/MIN)
(3)Extent of error:±5%.
(4)Flow-Pulse
2L/MIN=16HZ 4L/MIN=32.5HZ 6L/MIN=49.3HZ
8L/MIN=65.5HZ 10L/MIN=82HZ

5.Bom

| No. | Item | Material | Qty. |
|---|---|---|---|
| 1 | Connection wire | | 1 |
| 2 | Bonnet | PA | 1 |
| 3 | Screw | | 4 |
| 4 | Valve body | PA | 1 |
| 5 | Leak press valve | | 1 |
| 6 | Magnet | | 1 |
| 7 | Hall | | 1 |
| 8 | Impeller | POM | 1 |
| 9 | Rustless steel axis | SUS304 | 1 |
| 10 | | | |
| 11 | | | |

Prilog 2. Kod za mikroupravljač Arduino Nano

```
//Motor 1
int PWM_1 = 11;
int AI_1 = A0;

//Motor 2
int PWM_2 = 3;
int AI_2 = A1;

//Motor 3
int PWM_3 = 10;
int AI_3 = A2;

int ain1=0;
int ain2=0;
int ain3=0;

float omjer1=0;
float omjer2=0;
float omjer3=0;

int rounded_1=0;
int rounded_2=0;
int rounded_3=0;
```

```
void setup()
{

  pinMode(AI_1, INPUT);
  pinMode(STBY1_IN, INPUT);
  pinMode(STBY1_OUT, OUTPUT);
  pinMode(PWM_1, OUTPUT);

  pinMode(AI_2, INPUT);
  pinMode(STBY2_IN, INPUT);
  pinMode(STBY2_OUT, OUTPUT);
  pinMode(PWM_2, OUTPUT);

  pinMode(AI_3, INPUT);
  pinMode(STBY3_IN, INPUT);
  pinMode(STBY3_OUT, OUTPUT);
  pinMode(PWM_3, OUTPUT);

}
```

```
void loop()
{
ain1 = analogRead(AI_1);
ain2 = analogRead(AI_2);
ain3 = analogRead(AI_3);


omjer1=ain1*(255.0/1023.0);
rounded_1= round(omjer1);
analogWrite(PWM_1, rounded_1);

omjer2=ain2*(255.0/1023.0);
rounded_2= round(omjer2);
analogWrite(PWM_2, rounded_2);

omjer3=ain3*(255.0/1023.0);
rounded_3= round(omjer3);
analogWrite(PWM_3, rounded_3);
}
```