Sadržaj

Uvod 1

1.1 Svet digitalne elektronike ... 3
Digitalni i analogni signali ... 3
Brojni sistemi i kodovi ... 5
Logička kola i logičke operacije ... 7
1.2 Osnovni pojmovi procesnog upravljanja ... 9
Proces ... 9
Upravljački sistem ... 10
Senzori ... 10
Aktuatori i izvršni uređaji ... 12
1.3 Softver – osnovni pojmovi ... 14
Inženjerski i aplikativni softver ... 14
Firmver ... 15
Verzija softvera, Update, Upgrade, SP, HSP ... 16
Download & Upload ... 16
1.4 Šta je PLC? ... 18
Programabilni relej, industrijski računar ili nešto sasvim drugo? ... 18
1.5 Način rada PLC-a – Sken ciklus ... 21
Procesori personalnih računara i procesori PLC-ova ... 21
1.6 Mesto i uloga PLC-a u savremenim upravljačkim sistemima ... 25
Upravljanje na lokalnom nivou ... 25
SCADA sistemi ... 25
1.7 DCS ... 29

Hardverska struktura PLC-a 31

2.1 Hardverska struktura PLC-a ... 33
2.2 Procesorski modul ... 35
Karakteristike CPU-a ... 35
Kompaktnost i modularnost ... 38
2.3 Memorija ... 40
Vrste memorija koje koristi PLC ... 40
Organizacija memorije ... 41
Tipovi podataka i adresiranje ... 43
2.4 Ulazno/izlazni (I/O) moduli ... 46
Digitalni I/O moduli ... 46
Analogni I/O moduli ... 48
2.5 Specijalni, funkcijski i tehnološki moduli ... 49

2.6 Komunikacioni interfejs....50
Komunikacioni moduli....50
Port za proširenje (ekspanzioni port)....51
Industrijske mreže....51
2.7 Distribuirane periferije....53
2.8 Redundantni sistemi....54
2.9 Konfigurisanje hardvera – konfiguratori....55

Instalacija i ožičenje PLC-a 57

3.1 Instalacija PLC-a....59
3.2 Ožičenje PLC-a....65
Odabir kablova uz poštovanje standarda....65
Povezivanje napajanja....65
Ožičenje ulaza i izlaza....66
Postavljanje komunikacionih kablova....69

Programiranje PLC-a 71

4.1 Programiranje PLC-a....73
4.2 Programski jezici za PLC-ove....77
Izbor programskog jezika....78
4.3 Lestvičasti dijagram....81
Konvencionalni i PLC lestvičasti dijagrami....81
Struktura i elementi lestvičastih dijagrama....82
Naredbe uslova i naredbe izvršenja....82
4.4 Repertoar naredbi....87
Bit naredbe....87
Ulazne bit naredbe....89
Normalno otvoreni (NO) kontakti....89
Normalno otvoren (NO) kontakt predstavljen u lestvičastom dijagramu....90
Normalno zatvoreni (NC) kontakti....91
Normalno zatvoren (NC) kontakt predstavljen u lestvičastom dijagramu....93
Izlazne bit naredbe....93
Lestvičasti dijagram s jednobitnim NO i NC naredbama....95
Logičke naredbe....96
Tajmer....98
Brojač....100
Naredbe poređenja....103
Matematičke naredbe....104
Naredbe premeštanja i pomeranja podataka....106
Naredbe za konverziju tipa podataka....107
Naredbe za realizaciju komunikacionih zadataka....108
Naredbe za realizaciju tehnoloških funkcija....109
4.5 Prekidi....110
4.6 Potprogrami....113

Primeri 115
5.1 Prvi koraci – novi projekat 117
5.2 Definisanje hardverske konfiguracije 118
5.3 Komunikacija PC–PLC, podešavanje IP adrese 120
5.4 Programi s jednobitnim naredbama 122
Uključivanje i isključivanje izlaznih uređaja 122
Samodržanje 123
Realizacija logičkih funkcija 126
5.5 Upotreba tajmera i brojača 129
Odmeravanje vremena tajmerom 129
Brojanje proizvoda na pokretnoj traci 130
5.6 Upravljanje motorom pomoću PLC-a 133
ON/OFF upravljanje 133
Upravljanje pomoću frekventnog regulatora 134
Upravljanje koračnim motorom i servomotorom 136
5.7 Regulacija temperature 142
Diskretna (ON/OFF) regulacija 142
PID regulacija 144
Širinsko impulsna regulacija (PWM) 147
5.8 Merenje težine 150
Merenje težine 150
PLC program 150
5.9 HMI 153
Konfigurisanje HMI uređaja u TIA portalu 153
Programski zadatak 154
5.10 Veb server aplikacija 158
Izrada korisničkih veb stranica 159
PLC – podešavanja i program 160

Dijagnostika i održavanje 163
6.1 Dijagnostika i održavanje 165
Dijagnostika 165
Preventivno i interventno održavanje 168
6.2 Daljinsko održavanje 170

Izrada projektne i tehničke dokumentacije 173
7.1 Uvod 175
7.2 Izrada projektne dokumentacije 176
7.3 Izrada ostale dokumentacije 178

Indeks 179

Uvod



1.1 Svet digitalne elektronike
1.2 Osnovni pojmovi procesnog upravljanja
1.3 Softver – osnovni pojmovi
1.4 Šta je PLC?
1.5 Način rada PLC-a – Sken ciklus
1.6 Mesto i uloga PLC-a u savremenim upravljačkim sistemima

1.1 Svet digitalne elektronike

Digitalni i analogni signali

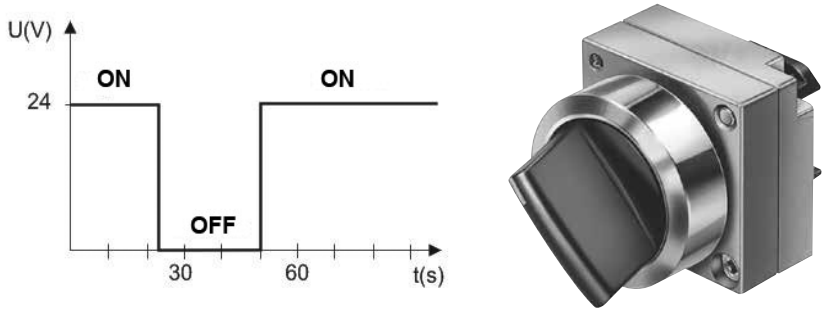
Bez sumnje se može reći da je današnji svet zapravo svet digitalne elektronike (slika 1.1). Konstrukcija personalnih računara je takva da CPU, memorija i sve druge komponente računara mogu da rade samo sa digitalnim signalima (signalima logičke nule i jedinice). Ali šta je sa automatikom i procesnim upravljanjem? Da li su digitalni signali i u toj oblasti preuzeli primat nad analognim?



Slika 1.1 *Živimo u digitalnom svetu*

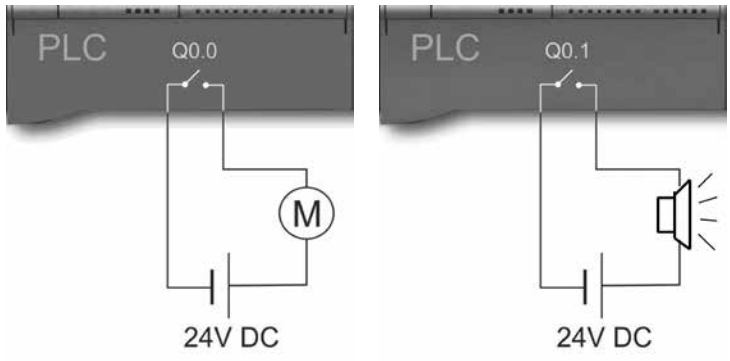
Ukoliko se posmatra proces obrade ili smeštanja i arhiviranja podataka koji dolaze sa procesa i pritom se ima na umu činjenica da su PLC-ovi u mnogim segmentima hardvera slični personalnim računarima, dolazi se do zaključka da se i u slučaju PLC-ova podaci mogu pamtiti i obrađivati samo u digitalnoj (binarnoj) formi. Međutim, ako se razmatra proces merenja ili praćenja nekih veličina na procesu koje se kontinualno menjaju, zaključuje se da se takve operacije ipak mogu obaviti samo pomoću analognih veličina i analognih senzora. Ali pre svega, neophodno je prvo razmotriti značenje pojmova digitalni i analogni signali, pa će se onda lakše razumeti i nekih drugi pojmovi iz oblasti digitalne elektronike.

Digitalni signali u automatici služe da ukažu na jedno od dva moguća stanja nekog ulaznog uređaja. Prekidač može biti pritisnut ili nepritisnut, senzor aktiviran ili neaktiviran – tj. i jedan i drugi se mogu naći u dva različita stanja. Ta dva stanja se opisuju binarnom informacijom, čija je vrednost logička nula ili logička jedinica (slika 1.2)



Slika 1.2 *Digitalni signal opisuje stanje prekidača i digitalnih senzora*

Digitalni signali takođe služe da se pomoću njih upravlja radom izlaznih uređaja (slika 1.3). Upisivanje logičke jedinice u neku izlaznu adresu koristi se kao signal za uključivanje odgovarajućeg izlaznog uređaja, a upisivanje logičke nule kao signal za njeno isključivanje.



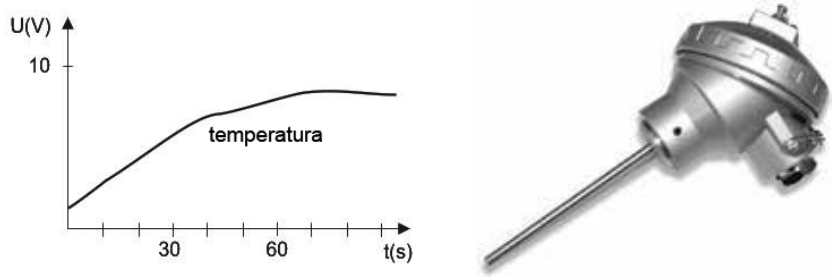
Slika 1.3 *Upravljanje izlaznim uređajima*

Analognim signalima se opisuju stanja određenih fizičkih veličina na procesu, koje se menjaju kontinualno u vremenu. Pritisak, temperatura i protok primeri su takvih fizičkih veličina (slika 1.4).



Slika 1.4 *Pritisak – fizička veličina koja se kontinualno menja*

Da bi PLC mogao koristiti te analogne signale (slika 1.5), pre njihovog upisivanja u memoriju PLC-a potrebno je obaviti A/D konverziju, kako bi oni bili predstavljeni u binarnoj formi – u vidu niza logičkih nula i jedinica. Analogni signali se mogu iskoristiti i za upravljanje određenim izlaznim uređajima.



Slika 1.5 *Analogni signal prati stanje analognog temperaturnog senzora*

Brojni sistemi i kodovi

Da bi se izrazile određene vrednosti koriste se brojni sistemi. U svakodnevnom životu koristimo decimalni brojni sistem kako bismo predstavili ili razumeli određene vrednosti. Decimalni brojni sistem ima 10 cifara (0, 1, 2,….9) i njegovu primenu uočavamo na svakom koraku. Cene proizvoda u prodavnicama, kursne liste, telefonski brojevi, vreme i datum – sve su to primeri u kojima se upotrebljava ovaj brojni sistem (slika 1.6).



Slika 1.6 *Telefonski brojevi*

Međutim, ne postoji samo decimalni brojni sistem. Iako je čoveku izuzetno pogodan za razumevanje i korišćenje, decimalni brojni sistem je teško i skoro nemoguće implementirati za memorisanje podataka u računarskim sistemima.

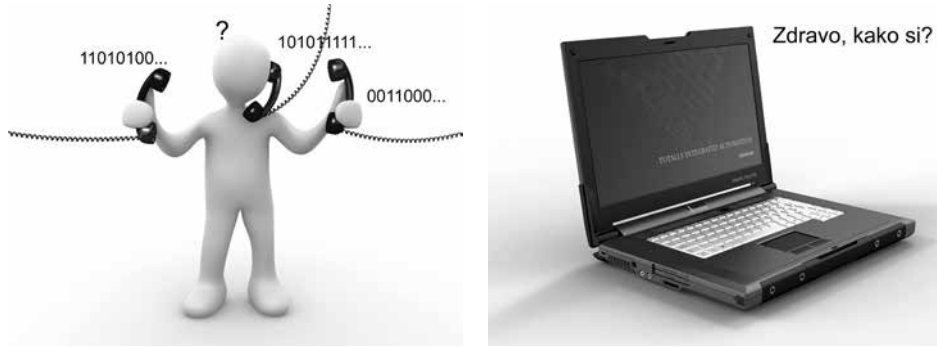
Postoje drugi brojni sistemi koji se mogu primeniti u oblasti računarskih sistema (slika 1.7). Sve memorije danas postojećih računarskih sistema

Decimalni	Binarni	Heksadecimalni
0	0000000	00
1	0000001	01
2	0000010	02
3	0000011	03
4	0000100	04
5	0000101	05
6	0000110	06
7	0000111	07
8	0001000	08
9	0001001	09
10	0001010	0A
11	0001011	0B
12	0001100	0C
13	0001101	0D
14	0001110	0E
15	0001111	0F
16	0001000	10

Slika 1.7 *Brojni sistemi*

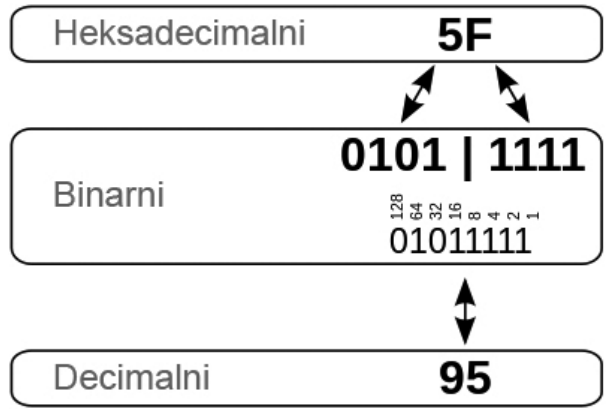
rade na principu pamćenja podataka u binarnoj notaciji, tj. u vidu memorisanja dva stanja: stanja logičke nule i stanja logičke jedinice, pa je samim tim i najlakši način reprezentovanja ovih signala binarni brojni sistem koji se sastoji od dve cifre: nule (0) i jedinice (1). Stanje logičke jedinice najčešće odgovara naponu od 5V DC (odnosno 24V DC u industriji), a stanje logičke nule naponu od 0V DC.

Svako bi trebalo da koristi jezik kojim će najefikasnije i najlakše predstaviti, skladištiti, obrađivati i razmenjivati podatke. Kad su u pitanju ljudi i njihova potreba da izraze vrednosti nekih veličina, tu je svakako nezamenjiv decimalni brojni sistem. Međutim, računari i računarski sistemi u tu svrhu koriste binarni brojni sistem i binarni jezik (slika 1.8).



Slika 1.8 *Računari koriste binarni jezik za komunikaciju*

Osim binarnog i decimalnog, najčešće se sreću još i oktalni i heksadecimalni brojni sistem, koji se najčešće primenjuju radi lakšeg i efikasnijeg predstavljanja binarnih podataka (slika 1.9).



Slika 1.9 *Pretvaranje iz jednog brojnog sistema u drugi*

Pored klasičnog pretvaranja decimalnih brojeva u binarne, vrlo često se primenjuje i kodiranje decimalnih brojeva. Kodiranje podrazumeva da se svaka cifra decimalnog broja predstavlja posebnim binarnim kodom. Najpoznatiji BCD (Binary Coded Decimal) kôd jeste težinski (8-4-2-1) kôd koji omogućava kodiranje svih deset cifara decimalnog brojnog sistema pomoću četiri binarne cifre.

Alfanumeričkii znakovi (brojevi, slova, interpunkcijski znakovi, specijalni znakovi itd.) kodiraju se drugim vrstama kodova koji iziskuju korišćenje većeg broja bitova. Neki od takvih kodova su ASCII i Unicode.

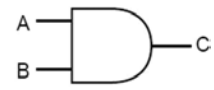
Logička kola i logičke operacije

Logička kola su elektronska kola koja imaju zadatak da realizuju logičke operacije. Logičkim operacijama su definisane funkcije nad jednom, dve ili više promenljivih. Promenljive su veličine jednog bita. Postoji samo jedna logička funkcija koja obavlja logičku operaciju nad jednom promenljivom i to je logička operacija negacije (NOT). Ostale logičke operacije (I, ILI, NI, NILI, ekskluzivno ILI) obavljaju se nad dve ili više promenljivih.

I (AND)

Na osnovu tabele se zaključuje da će izlaz logičke funkcije (C) biti u stanju logičke jedinice samo ako su oba ulaza (A i B) u stanju logičke jedinice (slika 1.10).

A	B	C
0	0	0
0	1	0
1	0	0
1	1	1

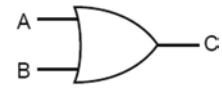


Slika 1.10 *Tabela i simbol za logičko kolo I*

ILI (OR)

Izlaz logičke funkcije (C) biće jednak logičkoj jedinici kad je na bar jednom od ulaza (A ili B) logička jedinica (slika 1.11).

A	B	C
0	0	0
0	1	1
1	0	1
1	1	1

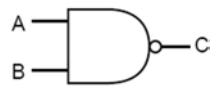


Slika 1.11 *Tabela i simbol za logičko kolo ILI*

NI (NAND)

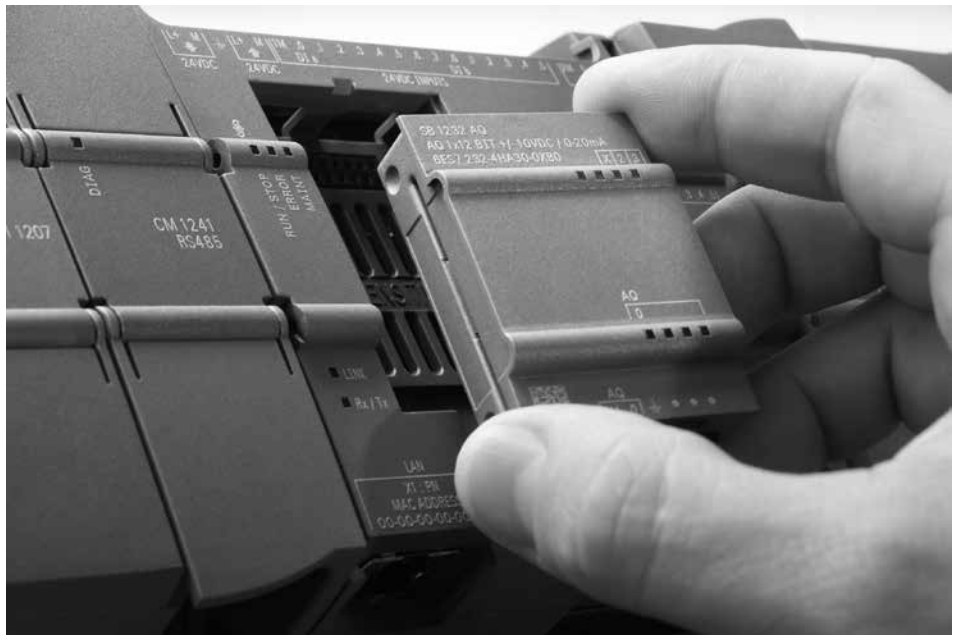
Izlaz NI logičke funkcije biće na logičkoj jedinici kad je bar jedan od ulaza (A ili B) na logičkoj nuli (slika 1.12).

A	B	C
0	0	1
0	1	1
1	0	1
1	1	0



Slika 1.12 *Tabela i simbol za logičko kolo NI*

Hardverska struktura PLC-a



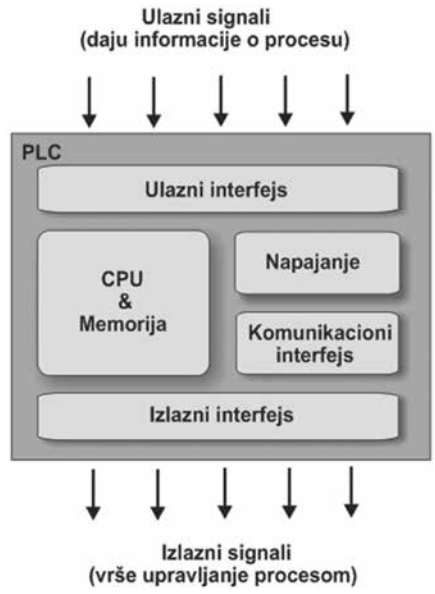
2.1 Hardverska struktura PLC-a
2.2 Procesorski modul
2.3 Memorija
2.4 Ulazno-izlazni (I/O) moduli
2.5 Specijalni, funkcijski i tehnološki moduli
2.6 Komunikacioni interfejs
2.7 Distribuirane periferije
2.8 Redundantni sistemi
2.9 Konfigurisanje hardvera – konfiguratori

2.1 Hardverska struktura PLC-a

Osnovna hardverska struktura PLC-a sadrži sledeće delove (slika 2.1.):

- procesorski modul (CPU)
- napajanje
- ulazni interfejs
- izlazni interfejs
- komunikacioni interfejs.

Procesorski modul je najvažniji deo PLC-a. On upravlja radom svih ostalih komponenti PLC-a i sinhronizuje njihov rad. U okviru procesorskog modula nalazi se i memorija u kojoj se čuvaju korisnički program i svi ostali podaci neophodni za rad PLC-a. Preko ulaznog interefejsa prihvataju se ulazni podaci i smeštaju se u odgovarajuće područje memorije. Ulazni interfejs je, dakle, spona između ulaznih uređaja i senzora koji daju informacije o stanju na procesu, i procesora PLC-a. Sličnu funkciju ima i izlazni interfejs, koji predstavlja sponu između procesora PLC-a i izlaznih uređaja. Pomoću izlaznog interfejsa, signali dobijeni na osnovu programa prenose se na izlazne linije PLC-a, a samim tim i na izlazne uređaje. Na taj način se realizuje upravljanje uređajima i mašinama na procesu. Komunikacioni interfejs služi za ostvarivanje komunikacije između PLC-a i nekog drugog uređaja (računara, drugog PLC-a, operaterskog panela i sl.).



Slika 2.1 *Hardverska struktura PLC-a*

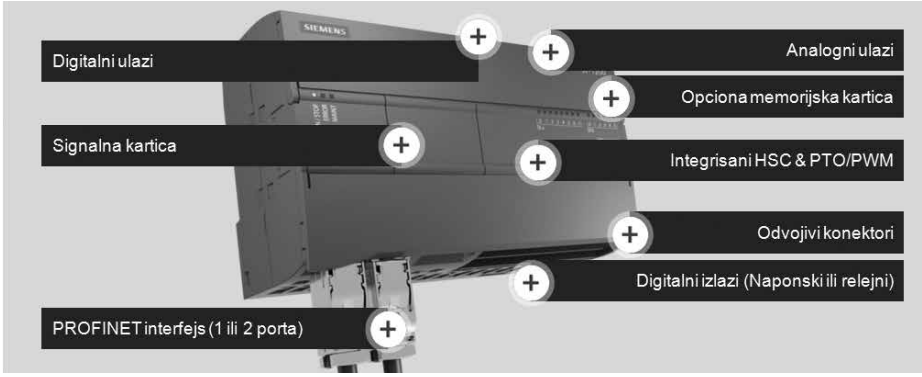
Kad bi se napravila analogija između ljudskog tela i PLC-a, moglo bi se reći da PLC sa svojim procesorom predstavlja mozak upravljačkog sistema jer upravlja svim radnjama koje upravljački sistem izvršava.

Po istoj analogiji, ulazni interfejs PLC-a sa senzorima i ulaznim uređajima bio bi ekvivalent čulima koje čovek poseduje i na osnovu kojih prima informacije iz svoje okoline, dok bi izlazni interfejs PLC-a zajedno s izlaznim uređajima predstavljao ekstremitete koji imaju zadatak da obavljaju određene radnje (slika 2.2).



Slika 2.2 *Analogija PLC-a s ljudskim telom*

Na slici 2.3 prikazana je hardverska struktura SIEMENS-ovog PLC-a Simatic S7-1200.



Slika 2.3 *Hardverska struktura S7-1200 PLC-a*

2.2 Procesorski modul

Karakteristike CPU-a

Procesor PLC-a ne razlikuje se mnogo od procesora standardnih računara jer i on ima aritmetičko-logičku jedinicu, upravljačku jedinicu i registre. Osnovna razlika između njega i procesora klasičnih personalnih računara leži u manjem broju instrukcija koje u svom repertoaru ima procesor PLC-a, a koje su prilagođene operacijama koje obavlja PLC. Pored toga, specifičnost procesora PLC-a je i njegov karakterističan ciklični rad, prilikom koga on neprestano ponavlja jedan isti niz operacija, tj. izvršava tzv. sken ciklus. Na slici 2.4 prikazan je procesorski modul CPU 1214C iz klase Simatic S7-1200 PLC-ova, kome su dodati komunikacioni i signalni (ekspanzioni) modul.



Slika 2.4 *Procesorski modul (u sredini) s komunikacionim modulom (levo) i ekspanzionim modulom (desno)*

U okviru procesorskog modula nalaze se dve najbitnije komponente za rad PLC-a:

- procesor (Central Processing Unit, CPU) i
- memorija.

Osnovne karakteristike procesorskog modula mogu se izraziti preko njegovih karakteristika. Za razliku od procesora klasičnih personalnih računara, čije su bitne karakteristike frekvencija radnog takta procesora, količina keš memorije, nanometarska tehnologija izrade i slično, za procesorski modul PLC-a važne su neke druge karakteristike koje nisu ve-

zane samo za CPU, već i za njegov ulazno/izlazni interfejs, memoriju, komunikacione mogućnosti, kao i mogućnosti programiranja različitim programskim jezicima (slika 2.5).

Table A- 62 General

Technical data	CPU 1215C AC/DC/Relay	CPU 1215C DC/DC/Relay	CPU 1215C DC/DC/DC
Order number	6ES7 215-1BG31-0XB0	6ES7 215-1HG31-0XB0	6ES7 215-1AG31-0XB0
Dimensions W x H x D (mm)	130 x 100 x 75	130 x 100 x 75	130 x 100 x 75
Shipping weight	550 grams	585 grams	520 grams
Power dissipation	14 W	12 W	12 W
Current available (SM and CM bus)	1600 mA max. (5 VDC)	1600 mA max. (5 VDC)	1600 mA max. (5 VDC)
Current available (24 VDC)	400 mA max. (sensor power)	400 mA max. (sensor power)	400 mA max. (sensor power)
Digital input current consumption (24VDC)	4 mA/input used	4 mA/input used	4 mA/input used

Table A- 63 CPU features

Technical data		Description
User memory[1]	Work	100 Kbytes
	Load	4 Mbytes, internal, expandable up to SD card size
	Retentive	10 Kbytes
On-board digital I/O		14 inputs/10 outputs
On-board analog I/O		2 inputs/2 outputs
Process image size		1024 bytes of inputs (I)/1024 bytes of outputs (Q)
Bit memory (M)		8192 bytes

Slika 2.5 *Neke od brojnih karakteristika CPU-a 1215C datih u uputstvu proizvođača*

U masi karakteristika kojima se opisuje procesorski modul PLC-a, sledeće se mogu izdvojiti kao najznačajnije:

- *set instrukcija* – obuhvata sve instrukcije kojima raspolaže određeni procesor (jednobitne instrukcije, logičke instrukcije, tajmeri, brojači, instrukcije poređenja, matematičke instrukcije, instrukcije pomeranja, instrukcije konverzije, instrukcije za komunikaciju itd.);
- *broj digitalnih i analognih ulazno/izlaznih linija* – ukazuje na maksimalan broj digitalnih i analognih signala koje može da primi ulazni interfejs PLC-a, kao i maksimalni broj digitalnih i analognih izlaznih linija;
- *veličina memorije* – opisuje veličinu memorije koja služi za smeštaj programa ili podataka;
- *mogućnosti komunikacije* – opisuje mogućnost procesora da podrži određene komunikacione protokole;
- *mogućnosti programiranja* – ukazuje na vrste programskih jezika koji su raspoloživi za dati procesor;

- *brzina skeniranja* – vreme potrebno za skeniranje određene veličine programa (tipično 1 KB) ili na vreme potrebno za izvršavanje jednog ulaznog skena (vreme potrebno za očitavanje ulaznih podataka i njihovo upisivanje u memoriju), kao i izlaznog skena (upisivanje podataka iz memorije na izlaz PLC-a).

Vreme skeniranja – CPU iz familije Simatic S7-1200

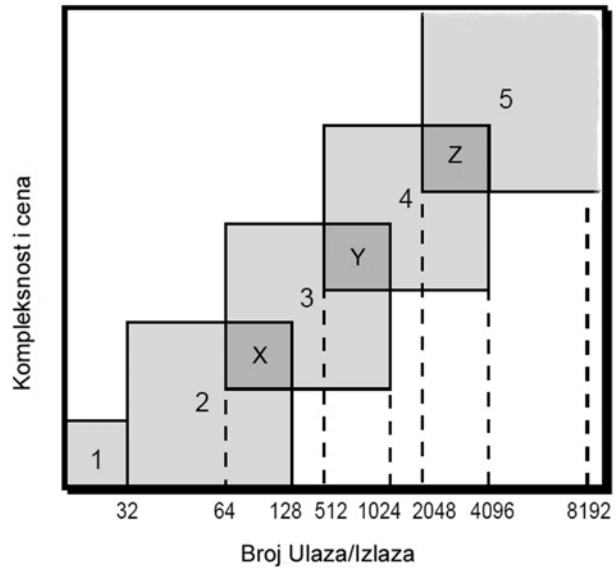
Vreme izvršenja binarnih instrukcija – 0.08 μs/operacija
Vreme izvršenja Word instrukcija – 1.7 μs/operacija
Vreme izvršenja instrukcija s realnim brojevima – 2.3 μs/operacija

Osim navedenih karakteristika, nekad je važno znati i koliki je broj prekida (engl. *interrupt*) moguće realizovati, opseg nivoa logičkih nula i jedinica, vreme izvršavanja pojedinih tipova naredbi i sl.

PLC-ovi se često klasifikuju u različite kategorije na osnovu karakteristika procesora. Na slici 2.6 prikazana je klasifikacija PLC-ova na osnovu broja ulazno/izlaznih linija. Ta klasifikacija je samo primer grube podele PLC-ova zavisno od broja I/O (Input/Output) linija. Različiti proizvođači imaju sopstvene standarde i kategorije u koje svrstavaju svoje PLC-ove, ali oni ipak ne odstupaju značajno od prikazane podele. Na osnovu ove klasifikacije, svi PLC-ovi se mogu podeliti u sledeće grupe:

- mikro PLC-ovi (do 32 I/O)
- mali PLC-ovi (32 do 128 I/O)
- srednji PLC-ovi (64 do 1024 I/O)
- veliki PLC-ovi (512 do 4096 I/O)
- veoma veliki PLC-ovi (više od 2048 I/O).

Primetićete da se kategorije preklapaju, pa se neki PLC-ovi mogu naći u dve kategorije. PLC koji, recimo, ima 80 I/O može se svrstati i u grupaciju malih i u grupaciju srednjih PLC-ova. U ovom slučaju, kriterijum za svrstavanje u određenu kategoriju nije broj ulazno/izlaznih linija datog PLC-a, već su to njegove dodatne funkcionalnosti – komunikacione mogućnosti, veličina memorije, dostupni programski jezici, napredne softverske funkcije i sl.



Slika 2.6 *Klasifikacija PLC-ova na osnovu broja ulazno/izlaznih linija*

Kompaktnost i modularnost

Izrazi kompaktan i modularan koriste se za opis kako celog PLC-a, tako i samog procesorskog modula.

Pod kompaktnim PLC-ovima podrazumevaju se oni čiji su svi osnovni hardverski delovi (CPU, ulazno/izlazni inerfejs, komunikacioni interfejs) strukturno i funkcionalno povezani u jedinstvenu celinu. Kompaktnim procesorskim (CPU) modulom smatra se onaj čiji je ulazno/izlazni interfejs ukomponovan u sâm procesorski modul. Za razliku od kompaktnog, modularni PLC nije izgrađen u vidu jedinstvene celine, već su svi njegovi delovi izrađeni u vidu posebnih modula. Takav tip PLC-a omogućava formiranje strukture PLC-a, onako kako to zahteva korisnik, odnosno sâm proces. Povezivanjem različitih modula formira se PLC čije karakteristike zavise od toga koji su moduli ukomponovani u zajedničku celinu. Ukoliko ima više ulaznih modula, PLC će moći da obrađuje više ulaznih signala. Ukoliko u PLC-u postoji analogni ulazni modul, PLC će moći da obrađuje i analogne signale. Vrlo često je modularni PLC izgrađen u vidu reka (engl. *rack*) koji predstavlja šasiju u koju se smeštaju moduli.

Rek se sastoji od više slotova, a u svaki slot smešta se jedan modul. Obično su prva dva slota u modulu predviđena za procesorski modul i modul napajanja, a raspored ostalih modula proizvoljan je i zavisi od toga kakvim se procesom upravlja.

Na slici 2.7 prikazana su dva SIEMENS procesorska modula CPU 1511 iz klase Simatic S7-1500, od kojih jedan ima osobinu kompaktnosti. Oznaka „C“ u imenu procesorskog modula ukazuje na njegovu kompaktnost.



Slika 2.7 *Obični PLC (CPU 1511-1 PN) i kompaktni PLC (CPU 1511C-1PN) sa integrisanim ulazno/izlaznim interfejsom*

Vrlo često jedan CPU ima obe karakteristike – i kompaktnost i modularnost. Kompaktnost podrazumeva postojanje ulaza i izlaza na samom procesorskom modulu, a modularnost mogućnost dodatnog povećanja broja ulaza i izlaza povezivanjem dodatnih ulazno/izlaznih modula. Na slici 2.8. prikazan je PLC Simatic S7-1200, koji je i kompaktan i modularan.



Slika 2.8 *Modularnost i kompaktnost u jednom PLC-u, SIEMENS Simatic S7-1200*

Maksimalan broj dodatnih modula koji se može povezati na CPU iz klase Simatic S7-1200:
– 8 SM (signalnih modula)
– 1 SB, CB ili BB (signalnih i drugih kartica – engl. *board*)
– 3CM (komunikacionih modula).

* Gore navedeno važi za CPU-ove 1214, 1215, 1217; ne važi za CPU 1211 koji nema mogućnost proširenja signalnim modulima i CPU 1212 koji može da se proširi samo sa dva signalna modula.

Programiranje PLC-a



4.1 Programiranje PLC-a
4.2 Programski jezici za PLC-ove
4.3 Lestvičasti dijagram
4.4 Repertoar naredbi
4.5 Prekidi
4.6 Potprogrami

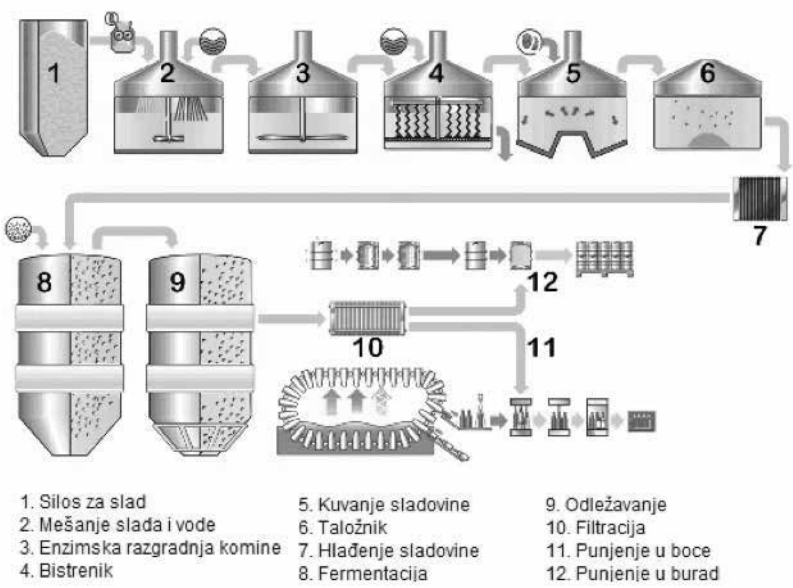
4.1 Programiranje PLC-a

Pod pojmom programiranje PLC-a podrazumevaju se svi postupci koji se obavljaju pri izradi softvera za PLC. Oni se mogu svrstati u dve faze:

- Izrada programa – pomoću posebnog inženjerskog softvera (alata) instaliranog na personalnom (ili industrijskom) računaru pravi se namenski aplikativni program koji PLC treba da izvršava.
- Prebacivanje programa u PLC (preuzimanje programa; engl. *download*) – program se iz računara prebacuje u PLC direktno preko kabla ili memorijske kartice.

Kako bi se pristupilo izradi programa prvo treba detaljno proučiti proces kojim PLC treba da upravlja, tj. tehnologiju samog procesa. Da bi se, recimo, automatizovao jedan proizvodni proces u pivari, tehnolog treba da dostavi detaljan opis procesa u tehnološkom smislu da bi programer stekao jasnu i kompletnu sliku o procesu – od toga kako se pripremaju sirovine i kako počinje sam proces, pa sve do glavnih i završnih faza, od procesa fermentacije i fiiltracije do punjenja flaša ili limenki (slika 4.1). Bez obzira na svoje znanje i veštine, programer je nemoćan bez poznavanja tehnologije proizvodnje ili pomoći tehnologa jer ne zna šta treba da uradi. Neki programeri su zahvaljujući dugogodišnjem iskustvu na istim ili sličnim programskim zadacima ovladali tehnologijom procesa koji automatizuju pa mogu izvesti programske zadatke i bez tehnologa. Oni su u izvesnom smislu i programeri i tehnolozi jer su se kroz višegodišnje iskustvo specijalizovali za određenu oblast.

U suštini, sve se svodi na to da treba sagledati i izdvojiti glavni programski zadatak ili više njih, i – ukoliko je to moguće – u okviru glavnog programskog zadatka izdvojiti i manje programske zadatke koji se mogu posmatrati kao posebne programske celine. Na taj način program će imati znatno jasniju strukturu. Da bi se napravio odgovarajući program, takođe je neophodno da programer tačno zna hardversku konfiguraciju PLC-a u smislu toga koliko je digitalnih i analognih ulaza i izlaza korišćeno, kao i to koji je senzor, prekidač, aktuator ili neki drugi uređaj priključen na koji ulaz ili izlaz PLC-a.

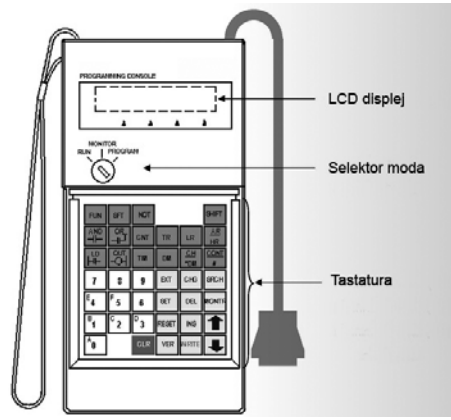


Slika 4.1 *Tehologija proizvodnje piva*

PLC se može programirati pomoću:

1. programskih konzola
2. personalnih računara
3. memorijskih kartica

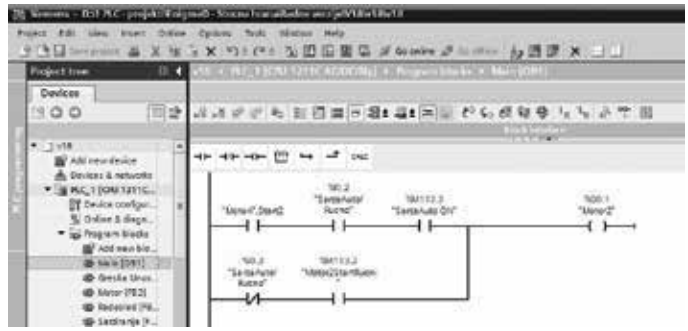
PLC se ranije mogao programirati i pomoću programskih konzola (slika 4.2) koje predstavljaju specijalnu vrstu terminala. One se ne mogu koristiti za izradu veoma složenih programa, ali su podesne za unos manjih izmena u programu ili monitoring nekih parametara. U suštini, većina proizvođača PLC-ova nikad nije ni imala programske konzole u ponudi, a danas su gotovo iščeznule s tržišta.



Slika 4.2 *Programska konzola (OMRON)*

Danas se program za PLC pravi na personalnim računarima a potom prebacuje u PLC direktnim povezivanjem s računarom preko kabla ili mreže.

Da bi se PLC programirao preko računara potreban je odgovarajući kabl za povezivanje PLC-a i personalnog računara, ali i odgovarajući softver (slika 4.3).



Slika 4.3 *TIA portal – Inženjerski softver*

Današnji laptop računari su idealno rešenje za programiranje i reprogramiranje PLC-a na licu mesta u samom fabričkom pogonu.

Proizvođači industrijske i procesne opreme vrlo često u svojoj ponudi imaju i specijalno prilagođene laptop računare za rad u industrijskom okruženju. Ti računari imaju zaštitu od elektromagnetnih smetnji, otporniji su na uslove rada u industrijskim halama, a imaju i instalirane licencirane softvere za programiranje PLC-ova, izradu SCADA aplikacija i sl. S obzirom na to da se koriste na terenu, odnosno u samom pogonu ili polju (engl. *field*), oni se često nazivaju i „field PG" (slika 4.4).



Slika 4.4 *Field PG*

PLC se može programirati i pomoću memorijskih kartica. Taj način je izuzetno pogodan kad programiranje treba da izvrši operater na procesu, odnosno lice koje nema nikakvog iskustva u programiranju. Programer koji je napravio program za PLC može taj program da snimi na memorijsku karticu, i da je zatim pošalje kurirskom službom ili na neki drugi način na udaljenu lokaciju na kojoj se nalazi PLC (slika 4.5).



Slika 4.5 *Memorijska kartica*

Da bi se PLC programirao pomoću memorijske kartice na kojoj je sačuvan program, treba samo

ubaciti memorijsku karticu u PLC, a u većini slučajeva i ispoštovati određenu jednostavnu proceduru koja se uglavnom sastoji u tome da se memorijska kartica ubaci u PLC, zatim da se PLC isključi s napajanja i ponovo uključi.

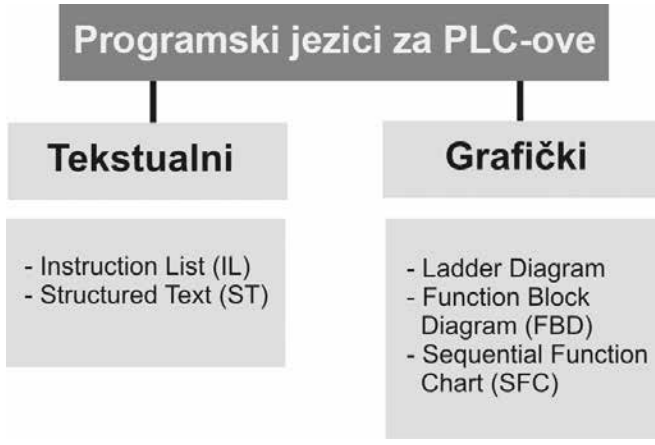
Kao preduslov za rad nekih PLC-ova neophodno je da u okviru CPU-a postoji memorijska kartica jer kod njih u okviru CPU-a ne postoji posebna memorija predviđena za smeštanje programa. Kod drugih tipova PLC-a, memorijska kartica služi samo za prebacivanje programa u PLC ili iz njega. Memorijska kartica PLC-a Simatic S7-1200 može se upotrebiti i u jednu i u drugu svrhu. Ukoliko se ona konfiguriše kao programska kartica (engl. *program card*), u njoj CPU čuva aplikativni program koji izvršava, a kad se konfiguriše kao prenosna kartica (engl. *transfer card*), služi samo za trenutno prebacivanje programa u PLC ili iz njega. Osim navedenih funkcija, ona može da se koristi i za ažuriranje/nadogradnju (engl. *update/upgrade*) firmvera, arhiviranje podataka itd.

4.2 Programski jezici za PLC-ove

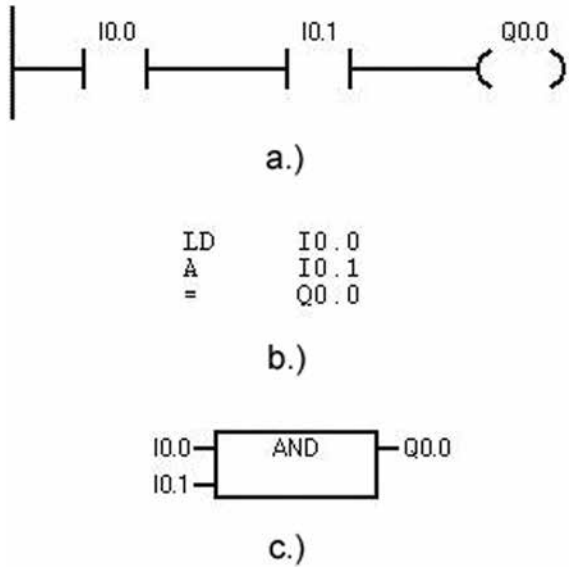
Stariji PLC-ovi su se uglavnom programirali pomoću lestvičastih dijagrama koji su po svojoj strukturi podsećali na šematske dijagrame kojima su se objašnjavale relejne mreže. Moderni PLC-ovi su doneli velike novine i poboljšanja, kako u svojoj hardverskoj strukturi, tako i u softverskim rešenjima. Današnji PLC-ovi se programiraju na više načina pa je zato bila potrebna i standardizacija programskih jezika. U skladu sa zahtevima koje su nametnuli tržište, korisnici, pa i sami proizvođači PLC-ova, usvojen je internacionalni standard IEC 61131-3 kojim je definisano i dozvoljeno sledećih pet programskih jezika za programabilne logičke kontrolere:

1. *Instruction List (IL)* – lista naredbi
2. *Structured Text (ST)* – strukturiran tekst
3. *Ladder Diagram* – lestvičasti dijagram
4. *Function Block Diagram (FBD)* – funkcijski blok dijagram
5. *Sequential Function Chart (SFC)* – sekvencijalni funkcijski grafikoni.

Prva dva jezika s ove liste pripadaju klasi tekstualnih jezika, pri čemu je IL programski jezik u klasi asemblerskih jezika, dok je ST programski jezik sličan Pascal programiranju. Ostala tri jezika pripadaju klasi grafičkih jezika, među kojima je jezik lestvičastih dijagrama (*Ladder Diagram*) i dalje ubedljivo najzastupljeniji programski jezik kada je u pitanju programiranje PLC-a (slika 4.6)



Slika 4.6 *Programski jezici za PLC-ove*



Slika 4.7 *Isti program napisan u vidu lestvičastog dijagrama (a), liste naredbi (b) i funkcijskih blok dijagrama (c)*

Izbor programskog jezika



Slika 4.8 *Programski jezici*

Izbor programskog jezika koji će programer koristiti za programiranje PLC-a zavisi od više faktora: od ličnih afiniteta programera, mogućnosti PLC-a, specifičnosti projektnog zahteva itd. U određenim okolnostima, pogodnije je koristiti jedan programski jezik, dok se u drugim situacijama kao bolje rešenje nameće neki drugi programski jezik (slika 4.8).

Lista naredbi (IL) programski je jezik blizak svakome ko ima iskustva s programiranjem mikroprocesora ili programiranjem u asemblerskom jeziku. Ovaj jezik se sadrži od koda s mnogo programskih linija, i u njemu svaka linija predstavlja jednu operaciju. Ukoliko programer koristi samo IEC definisane naredbe, program napisan na ovom jeziku lako se može prilagođavati različitom hardveru. IL je jezik najnižeg nivoa pa se kao takav i izvršava mnogo brže od grafički orijentisanih jezika, kao što je, recimo, jezik lestvičastih dijagrama. Pored toga, prednost ovog jezika je i to što program napisan na njemu zahteva mnogo manje prostora u memoriji. Međutim, bez obzira na prednosti koje obezbeđuje programerima, njegov veliki nedostatak je taj što nije omiljen među krajnjim korisnicima, inženjerima održavanja i servisnim

inženjerima jer nema preglednu grafičku strukturu koja omogućava laku preglednost i brzo delovanje u slučaju kvarova i eventualnih izmena.

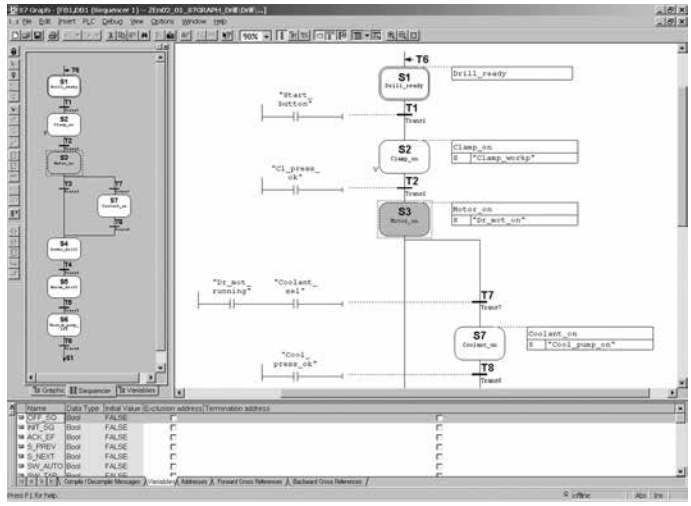
ST je sličan višim programskim jezicima kao što su PASCAL ili C. Trigonometrijska izračunavanja i analize podataka mogu se obavljati značajno lakše u ovom jeziku nego u jeziku lestvičastih dijagrama ili jeziku IL. Njegova prednost je u tome što danas većina tek diplomiranih inženjera ima više iskustva u računarskim programskim jezicima, nego u razumevanju relejnih i električnih mreža, pa samim tim mogu biti i produktivniji u programiranju na jeziku ST nego na jeziku lestvičastih dijagrama. Mana mu je to što nije blizak većini inženjera koji imaju dugogodišnje iskustvo u radu sa PLC-ovima, to jest, struktura programa napisanog u ST-u nije poznata većini inženjera koji rade na održavanju ili servisiranju PLC-ova.

Činjenica je da su lestvičasti dijagrami najzastupljeniji jezici za programiranje PLC-ova. Njihova ogromna prednost ogleda se u tome što su slični konvencionalnim električnim relejnim dijagramima, pa uz malo truda mogu biti razumljivi i ljudima koji nemaju programerska iskustva i sposobnosti, a poseduju znanja iz oblasti elektrotehnike. Njihova mana je to što se u slučaju značajnog povećanja veličine programa sve teže čitaju i razumeju, pa je neophodno dobro i promišljeno dokumentovati i komentarisati programske linije i delove programa.

Posle lestvičastog dijagrama, drugi po zastupljenosti je zasigurno jezik funkcijskih blok dijagrama (FBD). Pomoću njega se formira struktura programa koju je lako pratiti. Sa izlaza jednog bloka ide se na ulaz drugog bloka, tako da se praćenjem različitih putanja u okviru programa može lako ustanoviti funkcija određenih delova programa. Ovaj jezik predstavlja dobar izbor za početnike koji imaju samo osnovna znanja iz digitalne elektronike, a nikakvog iskustva u radu s relejnom logikom.

SFC ima jednostavan koncept. Ceo program je podeljen i grupisan u akcione blokove (slika 4.9). Akcioni blok u okviru koga se nalazi programski kôd, napisan na jeziku po izboru programera, aktivan je sve dok sledeći akcioni blok ne počne da se izvršava. Za prelazak na sledeći akcioni blok, neophodno je naravno da se ispune odgovarajući uslovi. S druge strane, ovaj stil programiranja nije pogodan za bilo koju aplikaciju jer korišćenje ovog jezika može u nekim slučajevima uneti nepotrebnu kompleksnost u strukturu programa. Dodatni nedostatak ovakvog programiranja jeste taj što ono iziskuje mnogo vremena za pripremu i planiranje pre nego što se uopšte počne s programiranjem jer, u suprotnom, SFC može biti izuzetno težak za praćenje. Program napisan na SFC-u izvršava se sporije od programa pisanih na drugim jezicima, a nemoguća je i njegova konverzija u druge jezike. IL, funkcijski blokovi i lestvičasti dijagrami lako se mogu prevoditi

jedan na drugi, dok sa SFC-om to nije slučaj. Sve u svemu, ovaj programski jezik može biti dobro rešenje u slučajevima kad je hardver PLC-a moćan, tj. kad ima brzinu i memoriju dovoljnu da isprati zahteve SFC-a, i naravno kad su lični afiniteti i sposobnosti programera ili krajnjeg korisnika više okrenute ovom programskom jeziku. Još jedna prednost ovog jezika je i to što je pogodan za krajnjeg korisnika i servisera koji brzo može da ustanovi u okviru kog se programskog akcionog bloka javio zastoj ili kvar.

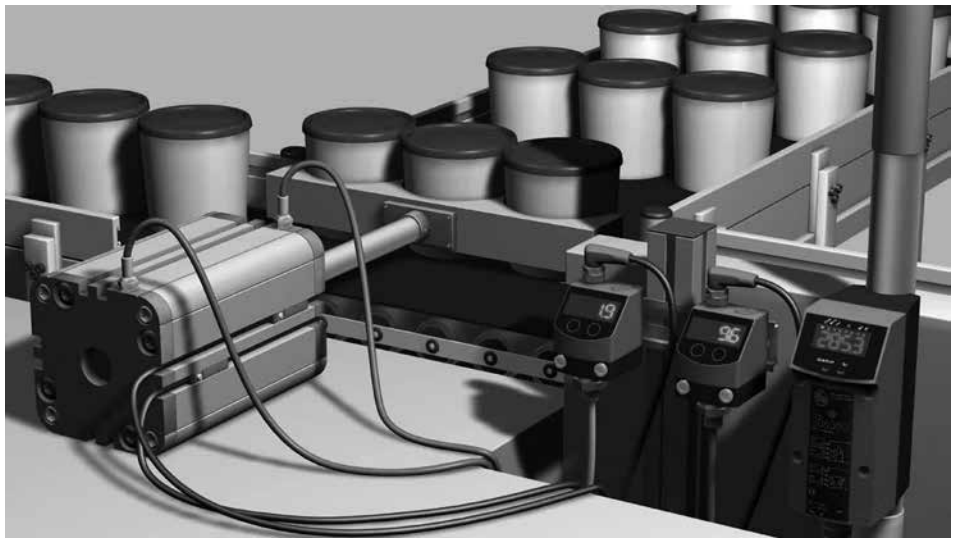


Slik 4.9 *Siemens-ov S7-Graph jezik (u klasi jezika SFC)*

SIEMENS-ov programski jezik STEP 7, koji predstavlja inženjerski softver za programiranje PLC-ova, potpuno je kompatibilan sa standardom IEC 61131-3, tako da on programeru nudi mogućnost korišćenja svih programskih jezika definisanih ovim standardom. Osim programa Ladder, FBD i IL (Statement List), Step 7 omogućava i korišćenje jezika S7-SCL (koji je zapravo jezik ST) i S7-Graph (koji je po IEC 61131-3 standardu jezik SFC).

Kriterijum za izbor programskog jezika	**Programski jezik**
Jednostavna upotreba za krajnje korisnike	SFC
Univerzalno prihvaćen jezik	Ladder
Brzina izvršavanja u PLC-u	IL & ST
Aplikacije sa pretežno digitalnim I/O	Ladder & FBD
Lako naknadno menjanje programa	Ladder
Lako korišćenje kompleksnih matematičkih operacija	ST

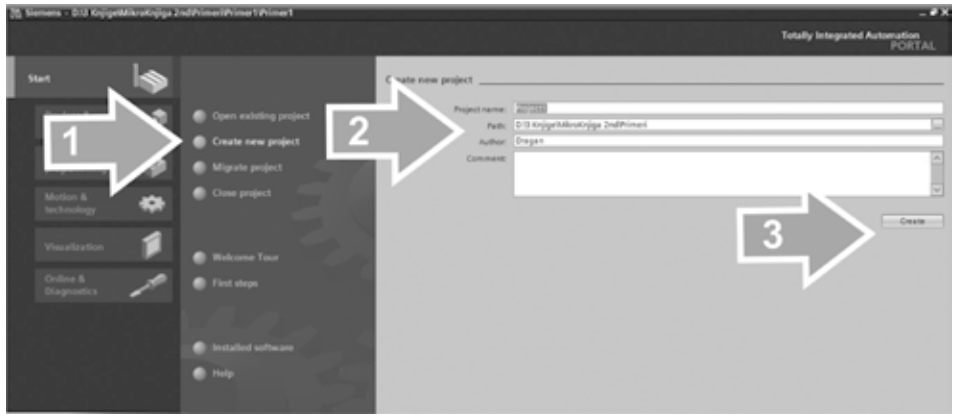
Primeri



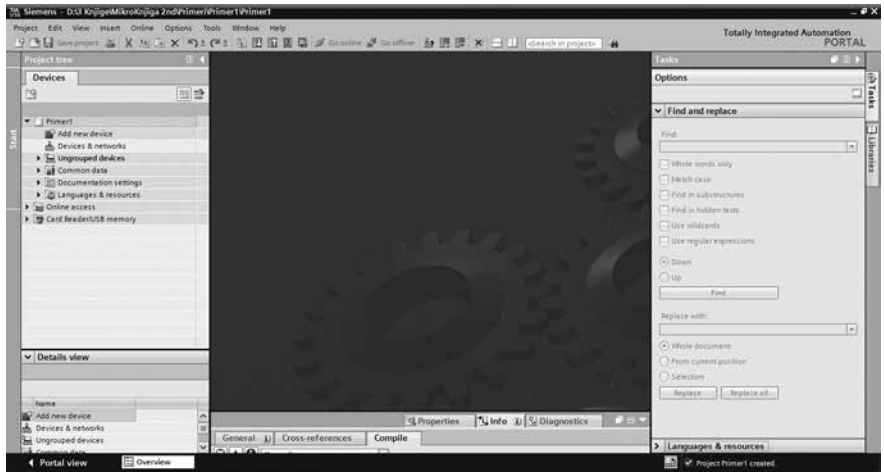
5.1 Prvi koraci – Novi projekat
5.2 Definisanje hardverske konfiguracije
5.3 Komunikacija PC–PLC, podešavanje IP adrese
5.4 Programi s jednobitnim naredbama
5.5 Upotreba tajmerskih i brojačkih naredbi
5.6 Upravljanje motorom pomoću PLC-a
5.7 Regulacija temperature
5.8 Merenje težine
5.9 HMI
5.10 Veb server aplikacija

5.1 Prvi koraci – novi projekat

Na slici 5.1 prikazan je postupak otvaranja novog projekta u TIA portalu. TIA portal ima dva početna ekrana: Portal View i Project View. Ovde je prikazan ekran Portal View, koji zapravo samo nudi prečice ka ekranu Project View. Na početku izrade projekta, prvo treba odabrati opciju „Create new project“ (1), nakon čega se biraju ime projekta i lokacija na kojoj će on biti smešten (2). Nov projekat se pravi pritiskom na dugme Create (3). Nakon ovih koraka izgled ekrana biće kao na slici 5.2.



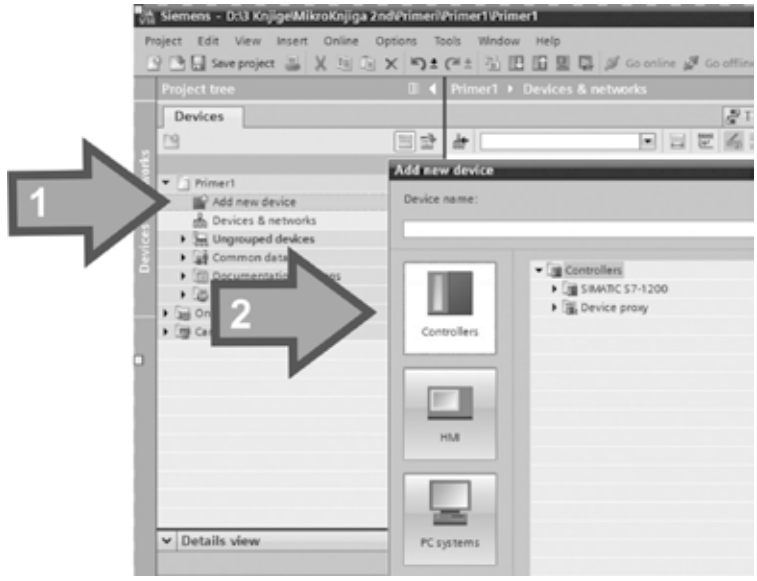
Slika 5.1 *Izrada projekta u TIA portalu*



Slika 5.2 *Izgled ekrana po završetku opisanih koraka*

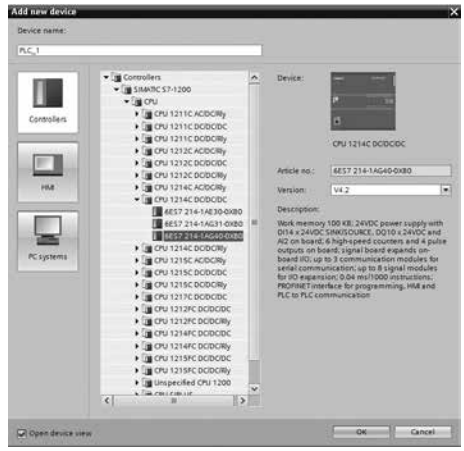
5.2 Definisanje hardverske konfiguracije

Pre nego što se pristupi izradi programa za PLC, treba definisati hardversku konfiguraciju u inženjerskom softveru. U slučaju nekih proizvođača, odnosno tipova PLC-ova, nije neophodno definisati hardversku konfiguraciju u okviru PLC programa – projekta. Takva praksa je predviđena jedino za male i mikro PLC-ove. Za PLC-ove iz klase Simatic S7-1200 neophodno je definisati kompletan hardver, kako onaj koji se odnosi na CPU i dodatne I/O module, tako i onaj koji se odnosi na komunikacione module. Slika 5.3 prikazuje postupak definisanja PLC hardvera u TIA portalu. Izborom opcije „Add new device" (1) otvara se novi prozor u kome programer bira uređaj koji želi da doda. Pošto mi hoćemo da u projekat dodamo PLC, biramo opciju „Controllers" (2).

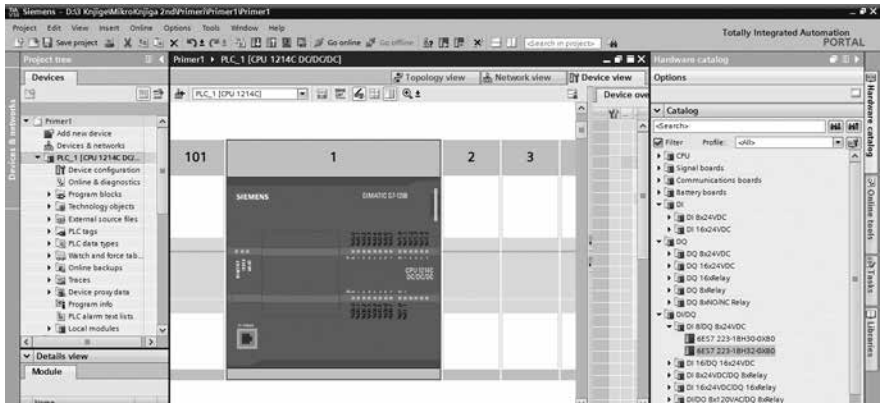


Slika 5.3 *Dodavanje novog uređaja u projekat*

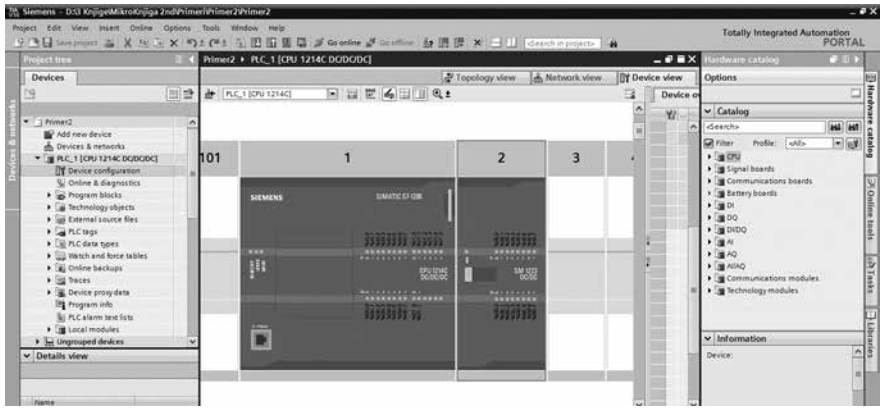
Nakon izbora PLC uređaja iz menija (slika 5.4), u okviru projekta se pojavljuje odgovarajući CPU (slika 5.5), a u vizuelnom prikazu hardverske konfiguracije vide se i mesta za dodatne I/O module. Jednostavnim prevlačenjem modula s desne strane (iz kataloga) na odgovarajuće mesto u hardverskoj konfiguraciji formira se PLC konfiguracija, koja se zajedno s gotovim PLC programom prebacuje u PLC (slika 5.6).



Slika 5.4 *Dodavanje novog PLC-a (CPU-a) u projekat*



Slika 5.5 *PLC nakon dodavanja u TIA portalu*



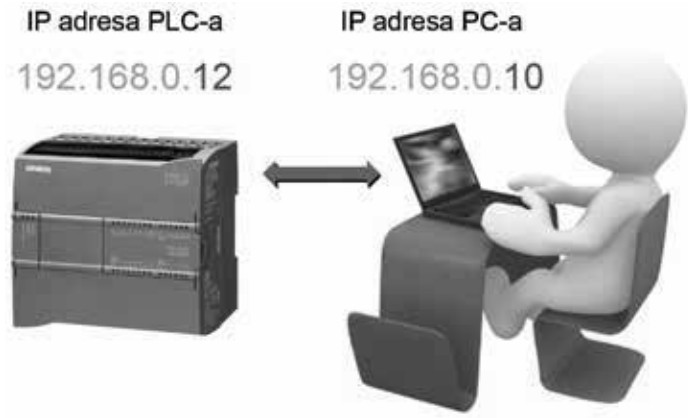
Slika 5.6 *PLC nakon dodavanja dodatnog modula u TIA portalu*

5.3 Komunikacija PC–PLC, podešavanje IP adrese

Veza između računara na kome se pravi program i PLC-a uspostavlja se pomoću odgovarajućeg kabla. Stariji tipovi PLC-a uglavnom su u tu svrhu koristili neku vrstu serijske komunikacije zasnovanu na protokolu RS232 ili RS485, dok novi PLC-ovi danas prisutni na tržištu uglavnom komuniciraju na bazi TCP/IP protokola, to jest putem eterneta.

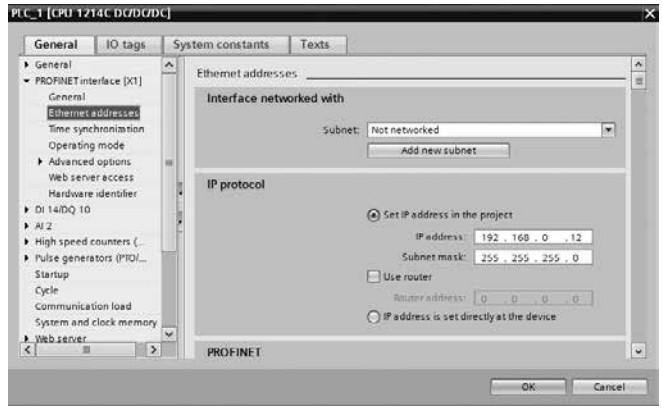
Budući da se navedena komunikacija zasniva na korišćenju IP adresa, programer mora posedovati osnovno znanje o toj komunikaciji. Suština uspostavljanja komunikacije jeste da IP adrese PLC-a i PC-a moraju pripadati istoj mreži.

IP adresa ima dva dela; prvi deo, s leve strane (192.168.0) predstavlja adresu mreže, dok je drugi deo, s desne strane (u našem slučaju 12 ili 10) adresa hosta, tj. adresa konkretnog uređaja koji se nalazi u toj mreži. Da bi uređaji mogli međusobno da komuniciraju, njihove adrese treba da pripadaju istoj mreži, tj. mrežni (levi) deo adrese treba da im bude potpuno isti, dok adresa hosta mora biti jedinstvena (slika 5.7).



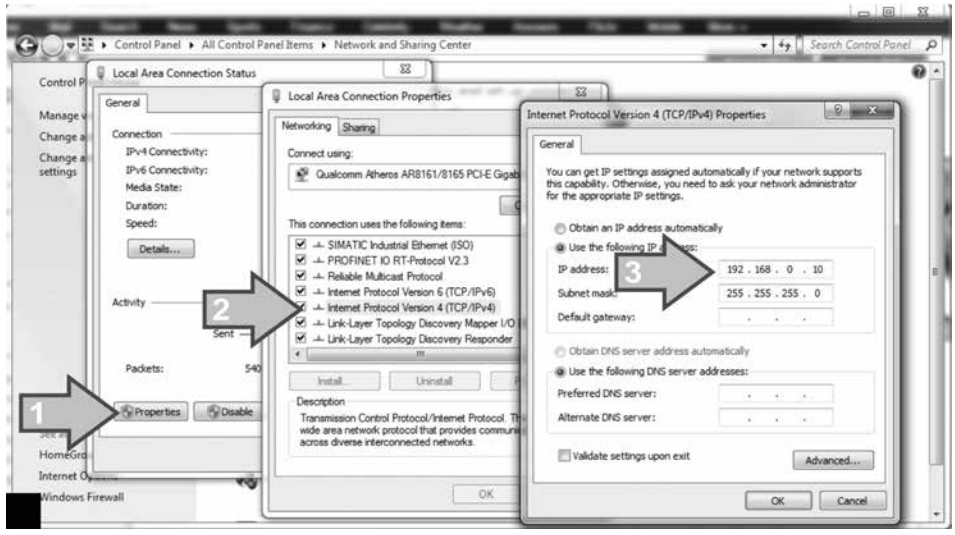
Slika 5.7 *IP adrese PLC-a i PC-a*

IP adresa PLC-a u TIA portalu podešava se u okviru opcije Properties, koja se aktivira tako što se PLC pritisne levim tasterom miša kako bi se izabrao; zatim se pritiskom na desni taster miša otvori padajući meni u kome se nalazi opcija Properties. U okviru opcija za podešavanje PLC-a, nalazi se i mogućnost podešavanja Eternet (IP) adrese i to pod opcijom PROFINET interface (slika 5.8).



Slika 5.8 *Podešavanje IP adrese PLC-a u TIA portalu*

Procedura podešavanja IP adrese u računaru zavisi od operativnog sistema. Na slici 5.9 prikazano je podešavanje u operativnim sistemima Windows. Na PC računaru, IP adresa se podešava preko Control Panela. U okviru Control Panela nalazi se stavka „Network and Sharing Center", unutar koje je dostupna i opcija Change Adapter Settings. Pritiskom na nju pristupa se ikonici Local Area Connection. Dvostukim pritiskom na tu ikonicu otvara se prozor Local Area Connection Status (1), u okviru koga se pritisne dugme Properties. Kad se otvori prozor Local Area Connection Properties, treba potvrditi opciju Internet Protocol Version 4 (2), čime se otvara mogućnost upisa odgovarajuće IP adrese (3).



Slika 5.9 *Postupak podešavanja IP adrese na računaru*