



KATEDRA ZA PRIMENJENU MEHANIKU I AUTOMATSKO
UPRAVLJANJE

SEMINARSKI RAD

na temu

LOGO! -SIEMENS
Programiranje i primena

Predmet : **Projektovanje Sistema**
Automatskog upravljanja

Predmetni nastavnik:
Dr Milan Matijević, docent

Student:
Nenad Đorčević, 81/97

Kragujevac , mart 2004. godine

S A D R Ž A J

Uvod.....	3
1. Programabilni logički kontroler (PLC).....	3
1.1. Opšti odredbe.....	3
1.2. PLC u sistemima automatskog upravljanja.....	4
1.3. Sistematski prilaz projektovanju sistema upravljanja koristeći PLC.....	5
2. Logički modul LOGO!.....	6
2.1. Šta je LOGO!?.....	6
2.2. Rukovanje LOGO! kontrolerom.....	7
2.2.1. Osobine i povezivanje kontrolera.....	9
2.2.2. Instaliranje LOGO! modula.....	9
2.2.3. Programski i parametarski mod.....	9
2.3 LOGO!Soft Interfejs.....	10
2.3.1. Instaliranje LOGO!Soft – a.....	10
2.3.2. Rad u LOGO!Soft – u.....	11
2.4. Uspostavljanje veze LOGO! – PC.....	15
3. Programiranje LOGO! modula.....	16
3.1. Funkcije sadržane u programskoj memoriji LOGO! modula.....	17
3.1.1. Konektori.....	18
3.1.2. Osnovne funkcije.....	16
3.1.3. Složene funkcije.....	20
3.2. Programiranje LOGO! modula pomoću LOGO!Soft – a	25
3.2.1. Osnovne komande i preporuke u radu sa softverom.....	25
3.2.2. Unošenje novog programa.....	27
3.2.3. Promena postojećeg programa.....	30
3.2.3.1. Brisanje blokova u shemi.....	31
3.2.4. Brisanje programa iz modula.....	32
3.3. Parametarski mod.....	32
3.3.1. Setovanje sata – tajmera.....	33
3.3.2. Setovanje parametara.....	34
3.4. Komunikacija između LOGO! i PC -ja.....	35
4. Urađeni primeri za vežbu.....	35
5. Simulacija rada MOGO! modula.....	45
6. Nove generacije LOGO! modula. Tendencije razvoja.....	47
7. Zaključak.....	48
Literatura.....	49

Uvod

Osnovni cilj ovog seminarskog rada je opis **programabilnog logičkog kontrolera LOGO!** firme **SIEMENS**, kako bi se budući čitalac upoznao sa osnovnim principima rada, načinu programiranja i mogućnostima koje pruža ovaj kontroler.

Polazeći od pretpostavke da korisnik ovog rada poseduje osnovna znanja iz oblasti projektovanja sistema automatskog upravljanja, u prvom delu napravljen je samo kratak pregled osnovnih osobina programabilnih logičkih kontrolera.

U drugom delu, već se prelazi na programabilni logički kontroler LOGO!, gde se govori o osnovnim tehničkim podacima, rukovanju ovim modulom, povezivanju sa PC računarom, mogućnostima primene načinu instaliranja softvera i dr.

U trećem poglavlju dat je detaljan opis funkcija koje su integrisane u samom modulu – u, kao i njihova fizička interpretacija. Poseban deo ovog poglavlja govori o načinu programiranja ovog logičkog modula.

Četvrti deo sadrži deset urađenih primera, gde je dat opis sistema kao i jedno od mogućih rešenja.

Na kraju je opisan i postupak simulacije rada programa, kao i savremeni trendovi razvoja ovog modula. U zaključku je dat opšti utisak kao i prednosti i mane ovog modula.

1. Programabilni logički kontroleri (PLC)

1.1. Opšte odredbe

Jedna od najbitnijih svrha PLC – a je zamena relea (elektromagnetskih prekidača) kao hardverskih komponenti.

Prema IEC (International Electrotechnical Commission) PLC se definiše kao:

"... digitalni elektronski sistem, projektovan da se koristi u industrijskom okruženju koji upotrebljava programabilnu memoriju za interno memorisanje, uskladištenje korisnički orijentisanih instrukcija za implementiranje specifičnih funkcija takvih kao što su: logičke, sekvencijalne, brojačke, tajming i aritmetičke do upravljačkih, kroz digitalne ili analogne ulaze za upravljanje raznih tipova mašina i

procesa. PLC i njemu pridruženi periferni uređaji projektovani su tako da budu lako ugradivi u industrijski sistem upravljanja i da omogućavaju laku upotrebu svojih funkcija."

PLC se sastoji od centralne procesorske jedinice (CPU), memorije, koja sadrži program po kojem se generiše upravljačko dejstvo, i ulaznih i izlaznih modula koji su direktno povezani sa uređajima.

CPU (u užem smislu) je mikroprocesor koji koordiniše aktivnosti PLC sistema. CPU izvršava program, obrađuje ulazno/izlazne signale i komunicira sa spoljnim uređajima, dok je u memoriju smešten program, kao i odgovarajuće vrednosti tajmera, brojača...

Za PLC se često kaže da predstavlja srce upravljačkog sistema, jer PLC pomoću programa koji je učitao u njegovu memoriju, stalno prati stanje sistema kroz povratni signal i na osnovu logike programa određuje kako će se izlazne veličine menjati. U ovom radu će biti obrađena prekidačka (ON/OFF) logika. Karakteristično za prekidačku logiku je to da se za ulaze i izlaze koriste digitalni signali.

Treba dodati da se PLC može proširiti modulima koji generišu upravljački signal drugom logikom – kao što su fuzzy, PID zakon upravljanja i dr; mogu se sprežiti sa drugim kontrolerima; nadgraditi tako da može da daje i prima kontinualne signale i dr.

1.2. PLC u sistemima automatskog upravljanja

Sistem upravljanja uopšte, predstavlja zbir elektronskih uređaja i opreme koji se koriste za obezbeđivanje stabilnosti, tačnosti i glatkog prelaza procesa ili proizvođačke aktivnosti. Zahvaljujući savremenim tehnološkim dostignućima, moguće je ostvariti upravljanje veoma komplikovanim procesima, odnosno sistemima pomoću PLC – a i računara.

PLC je u sistemu spregnut sa odgovarajućim ulaznim i izlaznim uređajima preko svojih ulaznih, odnosno izlaznih modula. Ulazni signali, u opštem slučaju mogu biti signali sa uređaja poput senzora, davača..., ali to mogu biti i ručno uneti input – i preko tastatura, tastera, i drugih vrsta prekidača preko kojih operater uspostavlja vezu sa PLC – om. Izlazni signali su najčešće usmereni na izvršne uređaje kao što su motori, solenoidi, releji, alarmi itd. Kroz aktiviranje motora ili solenoida PLC može da kontroliše najrazličitije procese: od najprostijih kao što je uzimanje i spuštanje nekog objekta do sistema

za servo pozicioniranje. Sijalice, displej i alarmi su takođe opcije koje postoje kao izlaz iz PLC – a, ali ne utiču na sam sistem i služe za obaveštavanje operatera.

1.3. Sistematski prilaz projektovanju sistema upravljanja koristeći PLC

Koncept upravljanja sistemom je vrlo lak zadatak. On uključuje sistematski prilaz koristeći sledeću proceduru:

1. Definisanje i analiziranje objekta upravljanja: *Prvo*, treba odrediti kojim se objektom želi upravljati. Prevažodna svrha programabilnog kontrolera je da upravlja nekim objektom. Ovaj objekat može biti neka mašina ili proces i naziva se upravljani sistem. Promena parametara upravljanog sistema se stalno prati ulaznim uređajima (senzorima...) koji daju određeno stanje i šalju signal PLC-u. Kao odziv, kontroler šalje signal izlaznim uređajima (aktuatorima...) koji upravljaju sistemom kako je programom predviđeno. Znači, potrebno je odrediti sekvencu operacije crtajući odgovarajući algoritam.

2. Dodela ulaza i izlaza: *Drugo*, treba odrediti sve ulazne i izlazne uređaje koji se vezuju na PLC. Ulazni uređaji mogu biti razni prekidači, senzori, itd. Po identifikovanju ulaznih/izlaznih uređaja, svakom uređaju treba dodeliti adresu. Sam postupak adresiranja zavisi od vrste PLC – a koji se koristi.

3. Pisanje programa: Program se može uneti pomoću softvera za pisanje ladder dijagrama ili preko konzole na PLC – u . Ali i ovo je korak koji je specifičan za određenu vrstu PLC – a. Ipak je najčešći slučaj pisanje ladder dijagrama.

4. Pokretanje sistema: Pre pokretanja sistema, proveriti da li su svi ulazno/izlazni uređaji dobro povezani za PLC-om. Posle pokretanja, možda je potrebno još fino podešavanje upravljačkog sistema. Sistem je potrebno podešavati sve dok se ne utvrdi da je rukovanje njime apsolutno bezbedno.

Za detaljnije informacije u vezi ove teme, konsultovati Seminarski rad Vladimira Đorđevića, "Programabilni logički kontroleri", na osnovu kojeg je prvo poglavlje i napisano.

2. Logički modul LOGO!

Do sada smo napravili samo uvod na temu programabilnih logičkih kontrolera, a sada se prelazi na konkretan modul **LOGO!** – proizvod firme **SIEMENS**.

2.1. Šta je LOGO!?

LOGO! je univerzalni logički modul, koji se koristi u inženjerskim električnim aplikacijama. Nastao je sredinom devedesetih godina prošlog veka, tačnije 1996. godine, i stalno je razvijan i usavršavan, tako da danas predstavlja jedan od najboljih modula ove vrste u svetu. Jedan od kontrolera novije generacije, sa dodatnim modulima prikazan je na **slici 1**.



Slika 1 – LOGO! model 230RC sa dodatnim modulima

Koristi se pre svega da zameni razne vrste prekidača, počev od relea, preko tajmera i dr. Mogućnosti ovog kontrolera su proširene i zbog toga što ima integrisane praktične funkcije u svojoj internoj memoriji.

Odlikuje se izuzetno **dobrim performansama**, koje su bitne za ovu vrstu kontrolera:

- ima mogućnost proširivanja – dodavanjem novih modula,
- veliki broj INPUT i OUTPUT priključaka, koji su međusobno izolovani (broj ovih priključaka zavisi od generacije modula, a može se uvećati pomenutim proširivanjem),
- lako se instalira,
- lako se programira i reprogramira – izuzetno fleksibilan u pogledu programiranja,
- žičenje je svedeno na minimum,
- otporan na vibracije,

- radi u različitim ambijentalnim uslovima u smislu temperature,
- zadovoljava sve standarde,
- veoma je malih dimenzija – standardan (4 modula) 72x90x55mm i dr...

Nešto više o ovim karakteristikama biće reči i kasnije.

Zahvaljujući brojnim i raznovrsnim pozitivnim osobinama, **oblasti primene** LOGO! kontrolera su gotovo sve vrste električnih aplikacija:

- u zgradama i instalacijama (kontrola svetla, vrata, kapija, garaža, ventilacionih sistema...),
- u prekidačkim ormarima,
- u malim mašinama (za kontrolu pumpi, malih motora i presa, kompresora...),
- u kontrolnim sistemima staklenih bašta...
- za preprocesiranje signala kod kontrolnih sistema....

Kao potvrda svih ovih navedenih karakteristika, može se navesti podatak proizvođača da je u upotrebi više od 100.000 jedinica u oblasti trgovine i industrije.

2.2. Rukovanje LOGO! kontrolerom

Rukovanje ovim kontrolerom je izuzetno jednostavno. Na **slici 2** prikazan je LOGO! na kome su strelicama prikazane određene pozicije.



gde su:

- 1-INPUT konektori obeleženi kao I1 do I8
- 2-OUTPUT konektori obeleženi kao Q1 do Q4
- 3-Tasteri kursora
- 4- L1 i N1 strujni priključci za
- 5- OK i ESC tasteri
- 6- Displej

Slika 2 – Ulazi, izlazi i tasteri

Strujni priključci su označeni kao L1 i N1 i vezuju se za napajanje. **Input konektori** su obeleženi oznakama I1 do I8 (u ovom slučaju) i to sa leva na desno. Za njih se vezuju odgovarajući izvori ulaznog signala. I~ In.

Output konektori su obeleženi oznakama Q1 do Q4 (u ovom slučaju). Za njih se priključuju odgovarajući izvršni organi. Oznaka Q je upotrebljena umesto O~ Out da bi se razlikovala od 0 (nula).

Tasteri kursora služe za kretanje po meniju koji se pojavljuje na displeju.

Tasteri OK i ESC služe za potvrdu i izlaz iz selekcije, respektivno.

Na **displeju** se pojavljuju informacije o modu u kome se nalazi kontroler, sat, i stanja ulaza i izlaza – **Slika 3**.



Slika 3 – displej

Na **slici 3** je dat primer gde možemo videti da je u prvoj liniji displeja prikazana oznaka I:1 2 3 4 5 6 što odgovara ulazima a brojne oznake koje su markirane crno znače da su ti ulazi "uključeni"- što odgovara stanju logičke 1 (u primeru sa slike, 1 i 3 su uključeni). Analogno su obeleženi i izlazi, u poslednjoj liniji displeja, kao

Q: 1 2 3 4, gde vidimo da postoji samo izlaz 2. Inače izlazi su normalno otvoreni.

Na **slici 3** se vidi i da je u drugoj liniji displeja prikazano **vreme** (We 2:08) kao i funkcija **RUN** (u donjem desnom uglu), koja znači da kontroler ustanovi ulaze i, po programu, odgovarajuće izlaze.

Treba reći da se iznad tastera kursora nalazi i konektor za vezu **LOGO! – PC** (zeleni pravougaonik na **slici 3**), koji kada ova veza nije ostvarena treba da bude prekriven zaštitnim poklopcem.

2.2.1. Osobine i povezivanje konektora

LOGO! ima konektore za napajanje, input i output.

LOGO! 230R i LOGO! 230 RC su podešeni za napone od 115V, 120V, 230V i 240V kao i za opseg frekvencija od 47 do 63 Hz.

Bitno **ograničenje za input konektore** da signali moraju da budu iste faze kao i glavno napajanje kontrolera, jer bi se u suprotnom pojavio visoki napon koji bi uništio kontroler.

Izlazi su releji koji su izolovani jedan od drugog, zbog čega možemo da kontrolišemo različite faze na izlazu – na primer na svakom izlazu možemo imati signal sa drugom fazom.

Ograničenja za izlazne releje su takva da ih možemo izložiti strujom od 8A, u slučaju omskog opterećenja, odnosno 2A u slučaju kapacitivnog opterećenja.

2.2.2. Instaliranje LOGO! modula

Samo instaliranje, odnosno postavljanje LOGO! kontrolera u sistem kojim upravljamo je krajnje jednostavno, obzirom na to da na sebi ima standardnu šinu širine 35mm. U tom slučaju, treba samo voditi računa o tome da su nam komande za upravljanje ovim kontrolerom pristupačne, kao i da možemo da pratimo stanje na displeju.

Važna praktična napomena je da prilikom rukovanja kontrolerom ne smemo dodirivati ulaz za PC jer iako LOGO! nije priključen, moguće je da se napon zadrži na tom priključku, pa je zato kada god je to moguće, ovaj priključak zatvoren tj. prekriven zaštitom.

2.2.3. Programski i parametarski mod

Ukoliko želimo da unesemo logičko kolo u kontroler, potrebno je da kontroler prebacimo u **Programming mode**. U ovom modu možemo da kopiramo, odnosno brišemo deo kola, setujemo vreme, ostvarimo komunikaciju LOGO! – PC i dr. O ovim aktivnostima će detaljnije u delu 3. Programiranje LOGO! modula.

Programming mode se **uključuje** istovremenim pritiskom tastera LEFT, RIGHT i OK – **Slika 4**. Ovakav izbor je napravljen da bi onemogućio uključivanje ovog moda greškom. Autor je ovaj način uključivanja nazvao **kombinacija tri prsta**.



Slika 4 – Uključivanje Programme moda

Ukoliko smo već uneli program, instalirali kontroler, i ukoliko želimo da promenimo parametre programa – na primer vreme setovanja tajmera ili kašnjenje kod relea, neophodno je kontroler prebaciti u **Parametrization mode**. Da bi uključili ovaj mod, potrebno je istovremeno pritisnuti tastere OK i ESC – **Slika 5**.



Slika 5 – Uključivanje Parametrization moda

Ova kombinacija nazvana je **kombinacijom 2 prsta**. U ovom modu, LOGO! nastavlja da izvršava logičko kolo odnosno kontroler je još uvek uključen.

Kada se kontroler nalazi u parametarskom modu, na displeju se mogu pratiti vrednosti promenljivih – na primer možemo videti koliko je tekuće vreme kašnjenja T_a . Ovo je naročito korisno prilikom testiranja logičkog kola.

2.3. LOGO!Soft Interfejs

2.3.1. Instaliranje LOGO!Soft – a.

Prvi korak u radu sa LOGO! kontrolerom, predstavlja instaliranje softvera **LOGO!Soft** koji služi za programiranje

kontrolera i komunikaciju LOGO! – PC. Inače program za instaliranje ovog softvera dobija se uz sam kontroler na flopi disku odnosno CD – romu. Ovaj softver ima određene zahteve u pogledu konfiguracije računara, ali uzimajući u obzir konfiguracije računara koji su danas u upotrebi, ovi zahtevi postaju trivijalni.

Instaliranje obuhvata nekoliko standardnih koraka:

- izaberemo drajv na kome se nalazi instalacija (u našem slučaju to će biti CD Rom, u odgovarajućem direktorijumu Logo),
- dupli - klik na **SETUP.EXE**, posle čega se otvara prozor u kome potvrdimo lokaciju direktorijuma u koji se smesta ovaj program,
- pratimo instrukcije koje slede na monitoru.

Instaliranjem ovog softvera, u posebnom direktorijumu (definisanom prilikom instaliranja), pored standardnog sadržaja, instaliraju se i neki gotovi primeri. Deinstalacija podrazumeva brisanje svih fajlova nastalih prilikom instalacije.

Važno je reći da je sa razvojem novih verzija LOGO! modula, razvijan i ovaj softver, tako da se LOGO!Soft paketi mogu bitno razlikovati – u najnovijim verzijama ovaj paket ima mogućnost i pisanja ladder¹ dijagrama. U ovom radu koristi se LOGO!Soft v1.1, mada će biti dati i primeri neki novijih verzija.

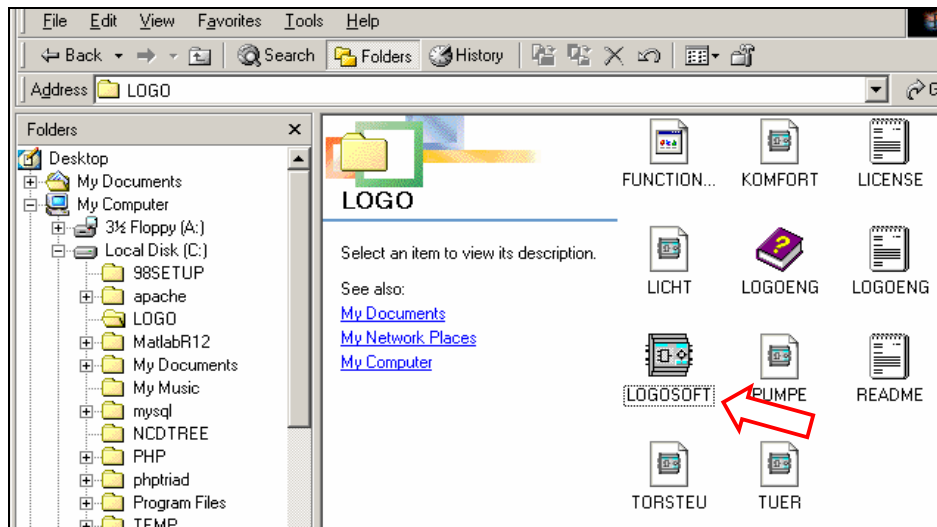
2.3.2. Rad u LOGO!Soft – u

Pošto smo instalirali program, na određenoj lokaciji iz direktorijuma **LOGO**, startujemo ga duplim klikom na ikonicu – **slika 6**, posle čega se otvara prozor kao na **slici 7**. Kao što se na **slici 6** može videti, u direktorijumu LOGO se nalaze help – LOGOENG, kao i primeri KOMFORT, LIGHT, PUMPE, TORSTEU, TUER...

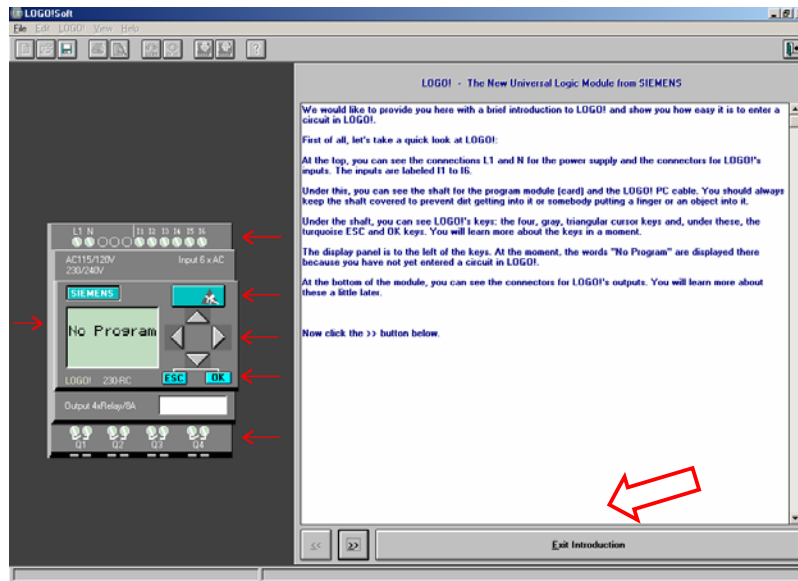
Na **slici 7** vidimo da se u desnom delu prozora, otvara "Brief Introduction"², u kome su ukratko date neke instrukcije u vezi sa radom LOGO! kontrolera. Ukoliko prvi put pokrećete program, dobro je proći kroz ovaj deo, u suprotnom ga treba ugasiti izborom opcije označene na slici **Exit Introduction**. Pošto ste to uradili u prozoru ostaje samo prikaz LOGO kontrolera.

¹ ladder na engleskom znači lestvica

² Brief Introduction na engleskom znači kratko predstavljanje

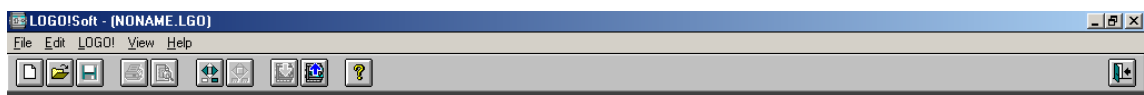


Slika 6 – Startovanje programa LOGO!Soft



Slika 7 – Prozor programa LOGO!Soft

Prozor sadrži standardnu liniju padajući menija u kojoj se nalaze opcije: **File**, **Edit**, **LOGO!**, **View** i **Help** – Slika 8.



Slika 8 – Meni LOGO!Soft - a

Padajući meni **File** sadrži standardne opcije: New, Open, Save,

Save As, Printer Setup, Print i Exit

Edit sadrži sledeće opsije:

- **Program** – služi za prelazak u programski mod
- **Parameterize** – služi za zadavanje odgovarajućih parametara.

LOGO! sadrži sledeće opcije:

- **Download Program...** – služi za prenos programa sa PC – a na kontroler
- **Upload Program...** – služi za prenos programa sa kontrolera na PC
- **Connection Setup** – definiše serijski port veze PC – LOGO!

View sadrži opcije:

- **Program Comments...** – služi za unošenje komentara u vezi sa ulazima, odnosno izlazima,
- **Diagram Overview...** – služi za shematski prikaz logičkog kola,
- **Button Bar...** – standardna opcija koja se odnosi na prikaz dugmića – može biti čekirana ili ne,
- **Status Line...** – služi za prikaz tekuće veličine programa – takođe može biti čekirana ili ne
- **Introduction at program start up...** – odnosi se na već pomenuti "intro".

Help je standardna opcija u padajućem nmeniju i sadrži sledeće opcije:

- **Contens** – sadržaj help funkcije (može se pozvati i funkcijom F1),
- **Window Interface** – otvara u standardnom prozoru funkciju Help
- **Working with LOGO!** – deo Help –a koji se odnosi na rad sa kontrolerom,
- **LOGO! Functions** – prikaz funkcija integrisanih u programskoj memoriji,
- **Technical Data** – prikaz tehničkih podataka,
- **Order From** - sadrži informacije o porudžbini,
- **About LOGO!Soft**– prikazuje opšte odredbe o LOGO!Soft-u

Odmah ispod ove linije padajućih menija, nalazi se "Button bar" odnosno linija koju čine ikonice, preko kojih se direktno pozivaju opcije koje su najviše u upotrebi. Ukoliko želimo da vidimo koju funkciju predstavlja ta ikona , možemo da pozicioniramo kursor tj.

miš na istu i u žutom pravougaoniku, pojaviće se informacija. Za detaljnije informacije, potrebno je startovati Help (preko funkcijskog dugmeta F1, na primer), i u prozoru izabrati opciju

Working With LOGO!Soft>Overview of the LOGO!Soft Window

gde možemo klikom na svaku od opcija u novom teks – prozoru dobiti informacije o toj poziciji.

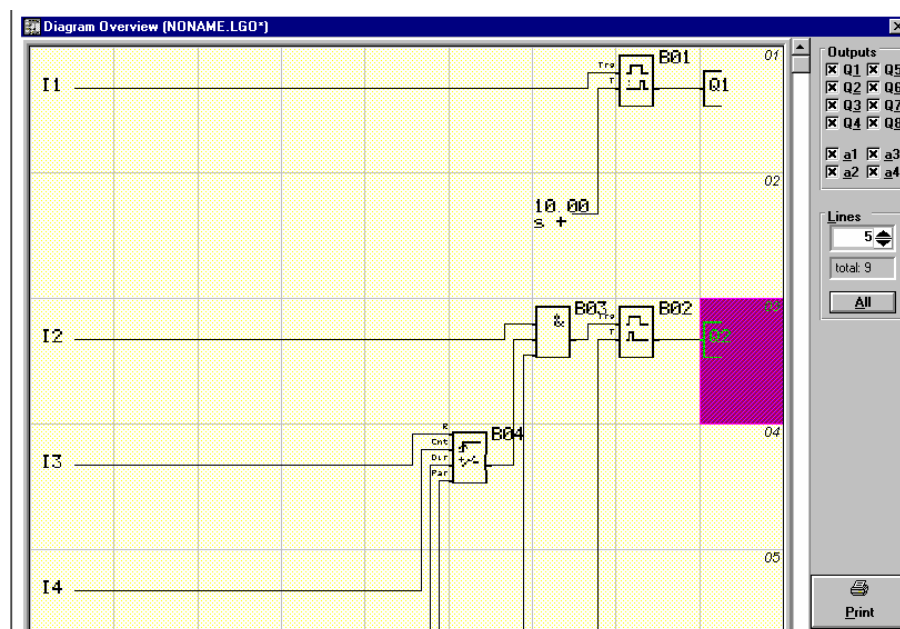
Čitav meni može se pozvati i desnim klikom na polju prozora LOGO!Soft – a gde se u padajućem meniju pojavljuju sve opcije.

"Status line", kao što je već rečeno, prikazuje veličinu tekućeg programa – **Slika 9**. Postoji opcija da klikom na deo ove linije dobijemo više informacija o istom.

Block: 26/30 Para: 20/27 RAM: 21/24 REM: 3/7 Timer: 7/10

Slika 9 – Status Line u donjem delu LOGO!Soft prozora

Jedan od glavnih razloga upotrebe LOGO!Soft –a je mogućnost vizuelizacije logičkog kola učitanoog programa – **Slika 10**.

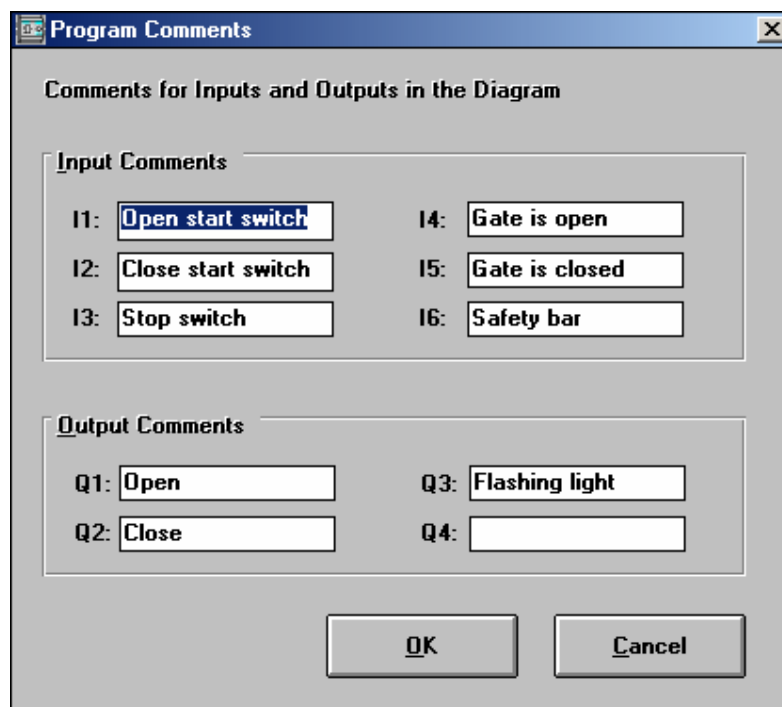


Slika 10 – Diagramski prikaz logičkog kola pomoću LOGO! Soft – a

Ovaj prikaz se dobija izborom opcije **Diagram Overview** iz padajućeg menija **View**, ili izborom odgovarajuće ikonice. Ružičasto polje na **slici 10** pokazuje tekuću poziciju, odnosno onu koju trenutno

možemo videti na displeju.

Ukoliko želimo da unesemo komentare za input i output, izborom opcije **Comments** iz padajućeg menija **View**, posle čega se otvara prozor kao na **slici 11** (dat je prozor koji odgovara primeru Gate).



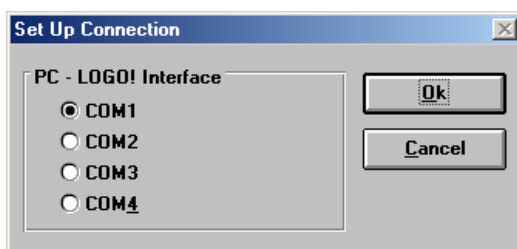
Slika 11 – Prozor za unos komentara

2.4. Uspostavljanje veze LOGO! – PC

Da bi povezali LOGO! kontroler sa računarom, potrebno je da imamo kabl za ovu konekciju, koji se od dobavljača dobija uz sam kontroler.

Prvo je potrebno kablom povezati PC sa LOGO! kontrolerom, zatim uključiti kontroler, čime se kontroler odmah startuje u PC/LOGO modu.

Kada smo uključili kontroler, izaberemo opciju **Connection Setup** iz padajućeg menija **LOGO!**, čime se otvara prozor – **Slika 12**. U novootvorenom prozoru, biramo opciju, odnosno serijski port preko koga je uspostavljena veza.



Slika 12 – Prozor Set Up Connection LOGO! / PC

Kada smo definisali vezu, možemo da preko opisanih opcija **Download Program...** i **Upload Program...** iz padajućeg menija **LOGO!** ostvariti transfer sa PC na LOGO! odnosno obrnuto. Ne startovati ove opcije, ukoliko niste prethodno konektovali LOGO na PC, jer će softver prijaviti grešku.

Na kraju, kako će programiranje LOGO! kontrolera za potrebe ovog rada biti realizovano kroz LOGO!Soft, u odeljku o samom programiranju, možemo se bolje upoznati sa mogućnostima i primenom ovog programa.

3. Programiranje LOGO! modula

3.1. Funkcije sadržane u programskoj memoriji LOGO! modula

Kao što je već ranije više puta naglašeno, jedna od glavnih osobina ovog kontrolera je da u sebi ima sadržane neke osnovne logičke funkcije. Poznavanje ovih funkcija je od suštinskog značaja za programiranje kontrolera, bilo da govorimo o programiranju pomoću LOGO!Soft – a ili o direktnom unosu programa u logo.

Broj integrisanih funkcija zavisi od generacije modela koji programiramo, ali to i ne treba da predstavlja neki problem, obzirom da se mnoge složene logičke funkcije mogu izraziti kombinacijom više osnovnih funkcija. Ipak, veći broj integrisanih funkcija bitno olakšava sam postupak programiranja.

Kako ovde govorimo o prvoj verziji LOGO! modula, ukratko će biti predstavljene funkcije koje su integrisane u njemu. Ovaj

modulator sadrži:

- **6 osnovnih logičkih funkcija i**
- **8 specijalnih logičkih funkcija.**

Pre nego predstavimo ove funkcije, prvo ćemo definisati konektore.

Oznake u zagradi odgovaraju oznakama koje figurišu u LOGO! modulu.

3.1.1. Konektori (Co)

Konektori se mogu povezati sa ulazima u blokove. LOGO! modul prepoznaje pet vrsta konektora:

x

Ako želimo provodan input, koristimo signal **x** koji u logičkoj tabeli ima vrednost **x=1** za **AND**, **NOT**, **NAND**. U ostalim slučajevima vrednost je **x=0**.

I1, I2, I3, I4, I5, I6, I7, I8, I9, I10, I11, I12, Ia1, Ia2, Ia3, Ia4

Ako želimo spojiti jedan od ulaza sa blokom, potrebno je samo uneti oznaku tog inputa (u ovom slučaju na raspolaganju je 12 inputa).

Inputi sa oznakom većom od I6 na raspolaganju su samo u verzijama LOGO!...L...; inputi Ia1 do Ia4 postoje samo kod LOGO!...LB11 verzija.

Q1, Q2, Q3, Q4, Q5, Q6, Q7, Q8, Qa1, Qa2, Qa3, Qa4

Analogan je slučaj i sa autput – ima u smislu povezivanja u programskoj shemi. Da bi neki aut vezali za blok, potrebno je samo navesti oznaku tog auta.

Autputi veći od Q4 postoje kod LOGO!...L... verzija; autputi Qa1 do Qa4 odgovaraju verzijama LOGO!...LB11.

hi

Ako želimo da stanje inputa bloka bude uvek '1', input je hi – skraćenica od high³.

lo


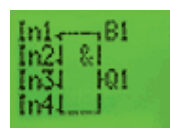
³ high na engleskom visok

Ako želimo da stanje inputa bloka bude uvek '0', input je **lo** skraćenica za low⁴.

3.1.2. Osnovne funkcije (GF)

Logička funkcija I (AND)

Serijska veza normalno otvorenih kontakta je u dijagramu kola prikazana kao na **slici 13 - a**. U LOGO! – u je ovo **AND** blok – **slika 13 – b**. **Slika 13 – c** daje logičku tabelu .

		<table><tr><th>I1</th><th>I2</th><th>I3</th><th>Q</th></tr><tr><td>0</td><td>0</td><td>0</td><td>0</td></tr><tr><td>0</td><td>0</td><td>1</td><td>0</td></tr><tr><td>0</td><td>1</td><td>0</td><td>0</td></tr><tr><td>0</td><td>1</td><td>1</td><td>0</td></tr><tr><td>1</td><td>0</td><td>0</td><td>0</td></tr><tr><td>1</td><td>0</td><td>1</td><td>0</td></tr><tr><td>1</td><td>1</td><td>0</td><td>0</td></tr><tr><td>1</td><td>1</td><td>1</td><td>1</td></tr></table>	I1	I2	I3	Q	0	0	0	0	0	0	1	0	0	1	0	0	0	1	1	0	1	0	0	0	1	0	1	0	1	1	0	0	1	1	1	1
I1	I2	I3	Q																																			
0	0	0	0																																			
0	0	1	0																																			
0	1	0	0																																			
0	1	1	0																																			
1	0	0	0																																			
1	0	1	0																																			
1	1	0	0																																			
1	1	1	1																																			
a	B	C																																				

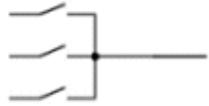
Slika 13 – Logička funkcija I

Blok, odnosno funkcija se zove I (AND) zato što aut Q postoji (Q=1) samo u slučaju kada sva tri ulaza I1, I2 i I3 postoje (=1).

U ovom slučaju se x konektor koristi za ulaz koji nije u upotebi, pa je za ovu funkciju x=1.

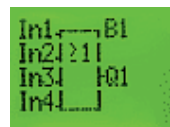
Logička funkcija ILI (OR)

Paralelna veza normalno otvorenih kontakta je u dijagramu kola prikazana kao na **slici 14 - a**. U LOGO! – u je ovo **OR** blok – **slika 14 – b**. **Slika 14 – c** daje logičku tabelu tabelu.



A ladder logic diagram showing three parallel normally open contacts connected to a single output line.

a



A screenshot of the OR block in the LOGO! software. The block has four input slots labeled In1, In2, In3, and In4. It has two output slots labeled Q1 and Q2. The first output slot is labeled B1.

b

I1	I2	I3	Q
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	1

c

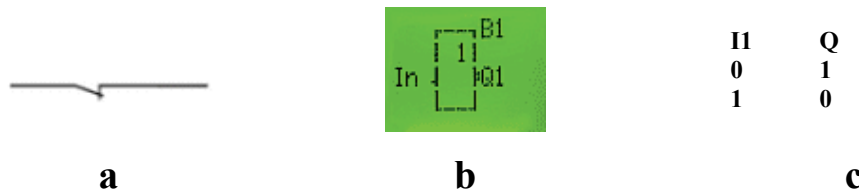
Slika 14 – Logička funkcija ILI

Logička funkcija se zove ILI zato što je za postojanje outa, Q, dovoljno da postoji samo jedan input. Kod ovog bloka x=0.

⁴ low na engleskom nizak

Logička funkcija NE (NOT) – Invertor

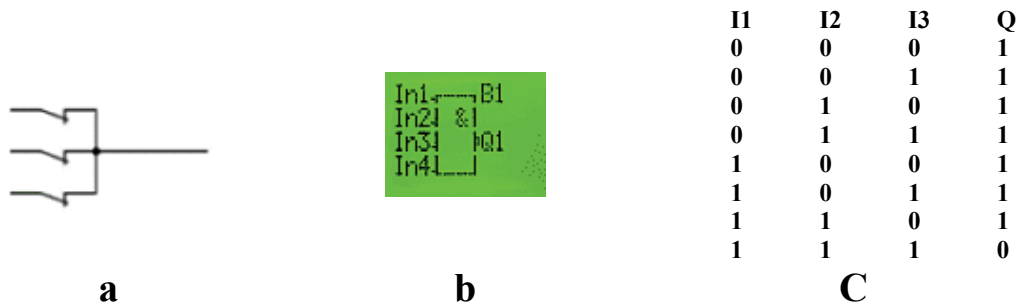
Logička funkcija koja **invertuje** ulaz sa odgovarajućim prikazima data na **Slici 15**. Naziva se **NE (NOT)**.



Slika 15 – Logička funkcija NE - Invertor

Logička funkcija NI (NAND)

Logička funkcija NI ima vrednost $Q=0$ samo kada su svi inputi setovani – **Slika 16**. Predstavlja paralelnu vezu normalno zatvorenih ulaza. Kada je bilo koji od ulaza nije setovan ($I=0$), output postoji ($Q=1$) – tabela na **slici 16 -c**.

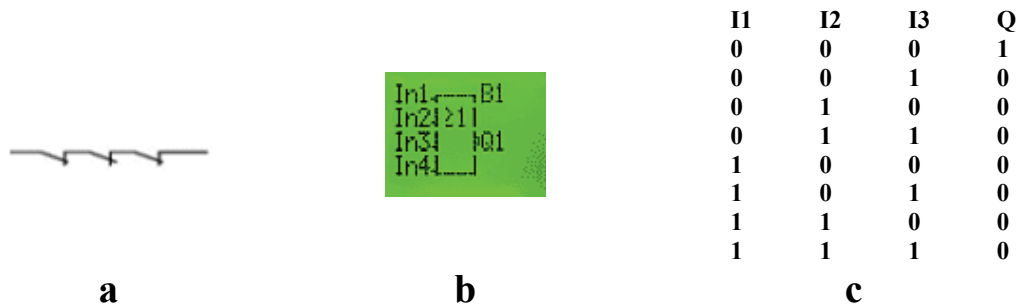


Slika 16 – Logička funkcija NI

Logička funkcija NILI (NOR)

Predstavlja serijsku vezu normalno zatvorenih ulaza. Odgovarajući prikaz ove funkcije dat je na **slici 17**.

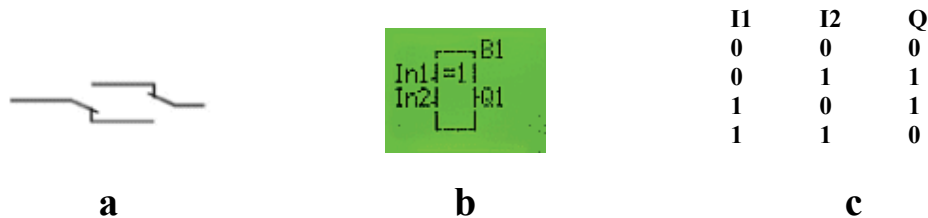
Autput iz ovog bloka je setovan ($Q=1$) samo kada su svi inputi u stanju koje odgovara logičkoj nuli, što se može videti u tabeli na **slici 17 – c**. Kada je bilo koji od ulaza setovan, output je nula.



Slika 17 – Logička funkcija NILI

Logička funkcija "isključivo ILI" (OR ELSE=XOR)

Ova logička funkcija ima vrednosti na izlazu Q=1, samo kada su inputi različiti – tabela na **slici 18**.



Slika 18 – Logička funkcija NE - Invertor

Napomenimo samo da novije generacije LOGO! modula imaju implementirane još neke funkcije koje spadaju u grupu osnovnih, a njihov prikaz biće dat u delu 6. Nove generacije LOGO! modula.

3.1.3. Specijalne funkcije (SF)

On – Delay

Ova funkcija predstavlja **kašnjenje** signala koji izlazi iz bloka u odnosu na ulazni signal definisano parametrom T. Prikaz ovog bloka u LOGO! – u, odnosno vremenska zavisnost ulaznog i izlaznog signala dati su na **slici 19** (a i b respektivno).



Slika 19 – On Delay

Kada se input **Trg** (skraćenica od trigger) setuje, tajmer počinje da meri vreme T_a – tekuće vreme u LOGO! modulu. Ukoliko Trg signal traje dovoljno dugo, kada tekuće vreme T_a dostigne setovani parametar **T**, pojavljuje se signal na izlazu. Ako se signal na ulazu resetuje tj. pređe iz stanja 1 u 0, onda će se resetovati i vreme **T**, tako da signala na izlazu neće ni biti.

Off – Delay

Ovom funkcijom se postiže produženje signala na izlazu iz bloka u odnosu na signal koji se nalazi na ulazu, za određeni parametar **T**. Prikaz ovog bloka u LOGO! – u, odnosno vremenska zavisnost ulaznog i izlaznog signala dati su na **slici 20** (a i b respektivno).



Slika 20 – Off Delay

Kada ulaz, **Trg**, uzme vrednost 1 izlaz **Q** se odmah uključuje tj. uzme vrednost 1. Ako se vreme **Trg** promeni od 1 do 0, LOGO! – ovo tekuće vreme T_a počinje da teče i izlaz je i dalje setovan. Tek kada tekuće vreme T_a dostigne vrednost parametra **T**, dolazi do resetiranja izlaza **Q**; $Q=0$.

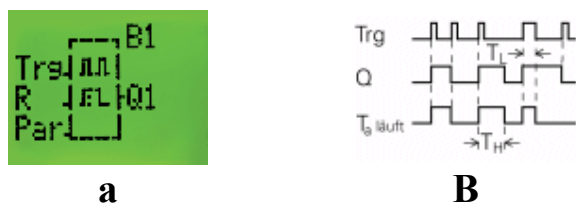
Ako se **Trg** ponovo setuje i resetuje, proces se ponavlja.

Međutim, ako se setuje ulazni signal **R** na stanje 1, onda će doći do resetovanja izlaza, pre isteka vremena **T**. Ovo je posledica toga što **svaki ulaz označen sa R (R~reset) ima "prednost" nad ostalim ulazima**.

Pulsni rele (Pulse Reley)

Ova vrsta relea se negde označava i kao konvencionalni prekidač. Prikaz ovog bloka u LOGO! – u, odnosno vremenska zavisnost ulaznog i izlaznog signala dati su na **slici 21**(a i b respektivno).

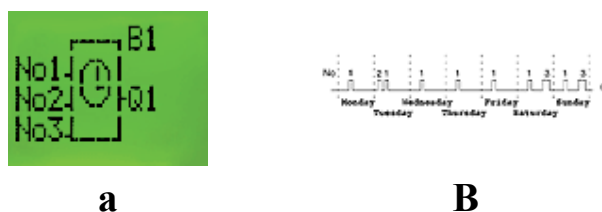
Svaki put kada ulazni signal promeni stanje sa 0 na 1 stanje izlaza se promeni (što može biti On ili Off).



Slika 21 – Pulse Relay

Tajmer – Clock (Time switch)

Tajmer je funkcija koja je dostupna kod verzija modula koje imaju sat, odnosno oznaku C u nazivu (Primer LOGO! 230RC). Prikaz ovog bloka na **slici 22**.



Slika 22– Tajmer

Parametri su No1, No2, No3. Svaki od parametara služi za setovanje vremena uključivanja i isključivanja tajmera u pridruženu memoriju sata (ima ih tri).

Parametri se zadaju tako što se prvo zadaje dan, zatim vreme uključivanja i na kraju vreme isključivanja.

Dan se može zadati kao

Mo (~Monday) – jedan dan

Mo...Sa – od ponedeljka do subote.

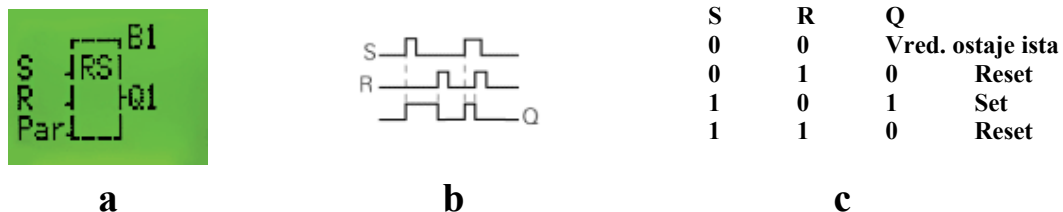
Vreme se standardno zadaje, a oznaka --:-- znači interval od 00:00 do 23:59 sata. Ako u polju za vreme uključivanja stavimo navedeni simbol, znači da nema uključivanja. Analogno i za slučaj zadavanja vremena isključivanja.

Dobra karakteristika tajmera je da on meri vreme i kada nestane napajanje zahvaljujući rezervnoj snazi.

Ukoliko dođe do preklapanja vremena definisanih preko pomenuta tri tajmera, pa se instrukcije preklapaju, postoji hijerarhija među parametrima tako da No3 ima prioritet u odnosu na No2, a No2 u odnosu na No1.

R – S rele

Ovaj rele zadovoljava čestu potebu da se jednom setovano stanje izlaza zadržava. Prikaz ovog bloka dat je na **slici 23**.



Slika 23 – R –S Relej

Ovaj rele predstavlja binarni flip – flop kod koga izlaz zavisi od stanja signala na ulazu, kao i od prethodnog stanja izlaza. Samo se u slučaju postojanja ulaznog signala R (reset), izlaz sigurno resetuje, zato što **ulazni signal R ima prioritet!** - tabela sa **slike 23**.

Generator impulsa (Clock Pulse Generator)

Radi se o bloku koji generiše impulse jednakog trajanja, koje je definisano parametrom T. Blok i vremenski dijagram ulaza i izlaza dati su na **slici 24**.



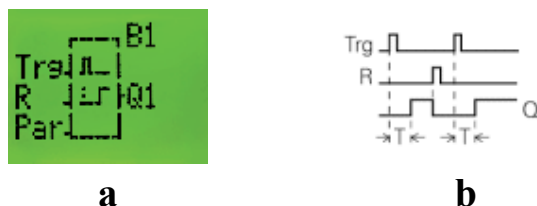
Slika 24 – Generator impulsa

Parametar T služi za definisanje trajanja signala odnosno pauze na izlazu. Pomoću ulaznog signala En (enable) praktično uključujemo generator. Generator setuje izlaz na 1 za vreme T, a zatim ga resetuje Q=0 za isti vremenski interval i tako dalje, sve dok se stanje ulaznog signala ne promeni sa 1 na 0, kada se i ovaj generator, odnosno njegov izlaz resetuje.

Bitna napomena je da se vremenski parametar zadaje tako da ne može biti manji od 0.10 sekundi – u slučaju manjeg intervala, vreme se neće setovati – primer T=0.05s ili T=0.00s.

Retentive On – Delay

Još jedan blok koji definiše **kašnjenje**, a čiji je prikaz i vremenski dijagram dat na **slici 25**.



Slika 25 – Retntive On - Delay

Ako se ulazni signal Trg promeni od 0 na 1, počinje da teče vreme T_a . Kada tekuće vreme T_a dostigne vreme T tj. $T_a = T$ izlaz Q je setovan na vrednost 1. Sledeće setovanje ulaza Trg neće uticati na izlaz Q, sve dok se input R ponovo ne promeni na stanje 1, kada dolazi do resetovanja izlaza. Tek sada bi mogao naredni Trg da setuje izlaz kako je prethodno opisano.

Brojač (Up And Down Counter)

Simbol brojača kao i dijagramski prikaz principa rada dat je na **slici 26**.



Slika 26 – Brojač

Na slici se vidi da imamo :

input **R**, koji služi da internu vrednost brojača resetuje (i ovde ima prioritet u odnosu na druge ulaze).

Cnt input je ulaz čije promene stanja sa 0 na 1 broji brojač – ne broji promene sa 1 na 0.

Dir parametar koji definiše smer brojanja. Za Dir = 0, brojač broji u napred, dok za Dir=1, brojanje u nazad

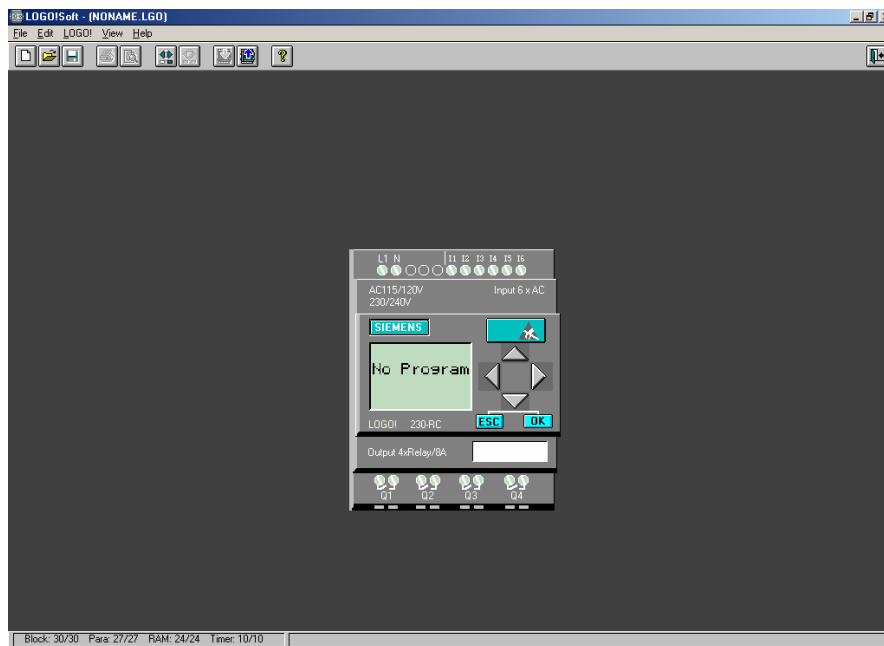
Par parametar koji definiše granicu do koje će brojač brojati, posle čega će pustiti izlaz Q, odnosno setovati ga. Može imati bilo koju vrednost od 0 do 9999 što predstavlja ograničenja brojača.

3.2. Programiranje LOGO! modula pomoću LOGO!Soft – a

Kako smo se do sada već upoznali sa osnovnim karakteristikama LOGO! modula, prelazimo konkretno pisanje programa, odnosno na programiranje ovog modula pomoću Logo!Soft v1.1. paketa. Način instaliranja ovog programskog paketa, osnovne funkcije, kao i meni opsioni su u odeljku 2.3. LOGO!Soft interfejs.

3.2.1. Osnovne komande i preporuke u radu sa softverom

Startovanjem programa LOGO!Soft otvara se prozor kao na **slici 27**.

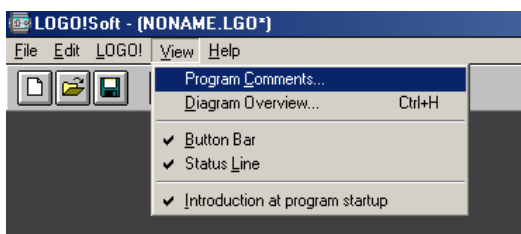


Slika 27 – Glavni prozor LOGO!Soft v1.1

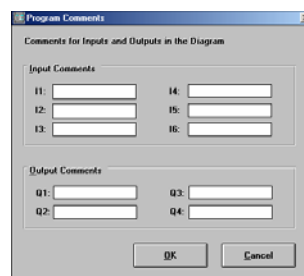
Kao što se na **slici 27** može videti, u ovom prozoru je prikazan jedan LOGO! modul. Zato se sam postupak programiranja pomoću ovog softvera može primeniti i u radu sa samim modulom, a neke specifične aktivnosti koje treba preduzeti, kada je reč o modulu, biće posebno naznačene.

Na **slici 27** se takođe vidi da se prilikom startovanja ovog programa otvara potpuno novi fajl. Pretpostavljajući da je upravljanje

određenim sistemom projektovano u skladu sa redosledom datim u odeljku 1.3. – Sistematski prilaz projektovanju sistema upravljanja, projektant ima već definisane ulaze i izlaze. U skladu sa tim, preporuka je da se u prvom koraku pisanja programa imenuju ulazi, odnosno izlazi. Ovo se postiže izborom opcije **Program Comments...** padajućeg menija **View** – **Slika 28**, posle čega se otvara prozor **slika 29**, gde unosimo odgovarajuće nazive (vidi primer na **slici 11**).

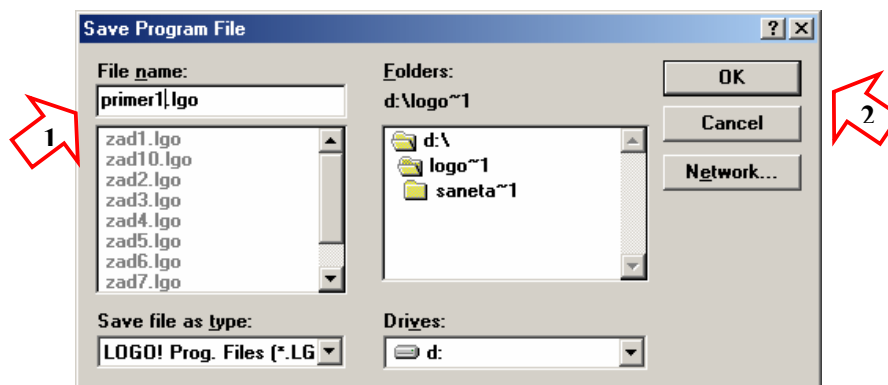


Slika 28 – Putanja za Program Comments..



Slika 29 – Prozor za imenovanje ulaza i izlaza

Pošto smo definisali ulaze i izlaze, možemo sačuvati fajl pod odgovarajućim imenom. Izaberemo opciju **Save As...** padajućeg menija **File**. U novootvorenom prozoru, definišemo mesto gde ćemo sačuvati fajl i unesemo ime fajla u odgovarajućem polju – **Slika 30**.



Slika 30 – Zapisivanje fajla

Ovo smo mogli da uradimo i u nekom narednom koraku, ali je povoljno sačuvati fajl što pre. U toku rada sa LOGO! Soft – om, povremeno treba birati opciju **Save** padajućeg menija **File**, kako bi sve promene bile sačuvane.

Sada već možemo da pređemo na unošenje sheme logičkog kola.

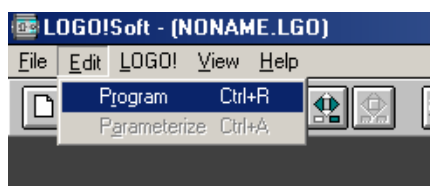
3.2.2. Unošenje novog programa

Kada startujemo program LOGO! Soft, možemo videti da na displeju modula prikazanog u prozoru, nalazi poruka **No Program**– **Slika 31**.



Slika 31 – Početno stanje modula

Da bi uneli novi program, odnosno da bi pomoću softvera prešli u **Programme mode**, iz padajućeg menija **Edit**, izaberimo opciju **Program** – **slika 32**, posle čega se na displeju otvara glavni meni – **slika 33**.



Slika 32 – Putanja za Program mode



Slika 33 – Glavni meni

Ako želimo da unesemo novi program direktno preko modula, potrebno je kombinacijom 3 prsta preći u **Programme mode**.

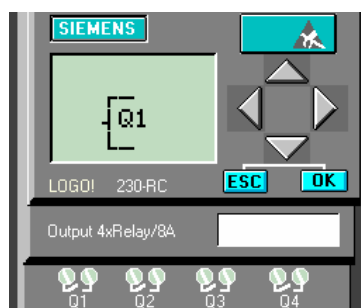
Iz glavnog menija prikazanog na displeju, izaberemo opciju **Program...** Da bi ovo učinili, koristimo kursor tastere za kretanje pozicionog znaka >, koji stoji levo od naziva opcije, i pritiskom na **OK** taster u prozoru, ili jednostavnim pritiskom **↵ Enter na tastaturi**, ulazimo u označeni podmeni. Sada se na displeju pojavljuje

sadržaj podmenija Program... – **Slika 34**. Treba naglasiti da unošenje programa uvek počinje od izlaza Q1.

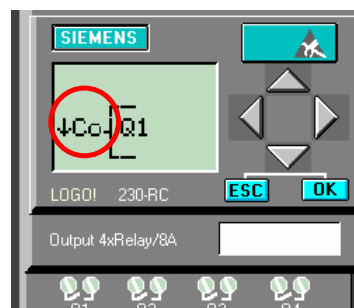


Slika 34 – Sadržaj podmenija Program...

Pomoću kursora se obeleži opcija **Edit Prg** i klikne **OK** (ili ↵), tako da se sada na displeju pojavljuje oznaka za prvi izlaz, Q1 – **slika 35**. Pritiskom **OK**, otvara se meni prikazan na **slici 36**, gde treba odabrati jednu od opcija koju želimo da povežemo sa ovim ulazom.



Slika 35 – Blok izlaza Q1



Slika 36– Povezivanje izlaza Q1

Strelica koja stoji sa leve strane markiranog dela, ↓, pokazuje smer kretanja u listi opcija. Dakle, pomoću kursora Δ i ∇ i OK, biramo jednu od ponuđenih opcija:

Co – lista konektora – sadrži **x, I1...I6, Q1...Q4, hi, lo** (vidi 3.1.1.)

GF – lista **osnovnih funkcija** (vidi 3.1.2.),

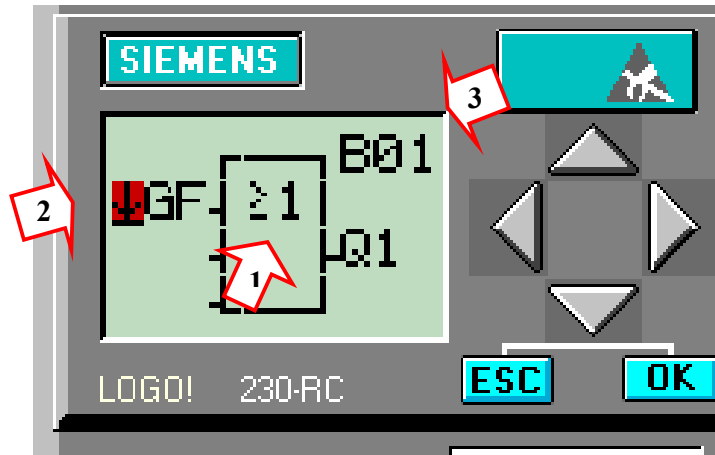
SF – lista **složenih funkcija** (vidi 3.1.3.),

BN – lista već **postojećih blokova**.

Izaberemo jednu od opcija – biramo konektor, odnosno funkciju koja odgovara datom bloku, koji će biti ubačen u program. Na primer u ↓GF, možemo izabrati opciju koja odgovara ILI funkciji, odnosno ILI bloku– **slika 37 pozicija 1**. Kada se pojavi željeni blok, pritisnemo **OK**, posle čega se kursor nalazi na prvom ulazu u izabrani blok. Pritiskom na **OK**, otvara se već poznati meni (**sl 37 – poz. 2**). U ovom meniju opet možemo da izaberemo neku funkciju za ulaz, odnosno neki od konektora. Kada smo pronašli željenu opciju, pritisnemo **OK**,

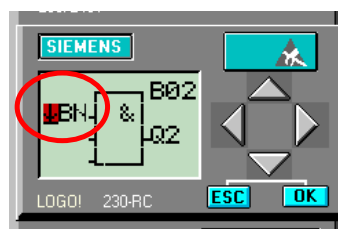
posle čega se kursor premešta na drugi ulaz u bloku itd.

Na **slici 37** se može videti da je blok definisan u programu numerisan kao B01 (**sl 37 pozicija 3**). Svaki sledeći blok će dobijati



Slika 37 – Definisanje ulaza u blokove

ove "adrese" i to po redu unošenja u program. Ovu pogodnost možemo iskoristiti u slučaju kada želimo izlaz iz nekog već postojećeg bloka, da dovedemo na ulaz posmatranog bloka. Izborom opcije ↓BN otvara se meni kao na **slici 38**, gde biraмо blok čiji izlaz želimo da dovedemo na ulaz posmatranog bloka. Tada se praktično u shemi kopira deo kola od ulaza do navedenog bloka i u dijagramu je ovaj deo prikazan crveno – **Slika 39**.

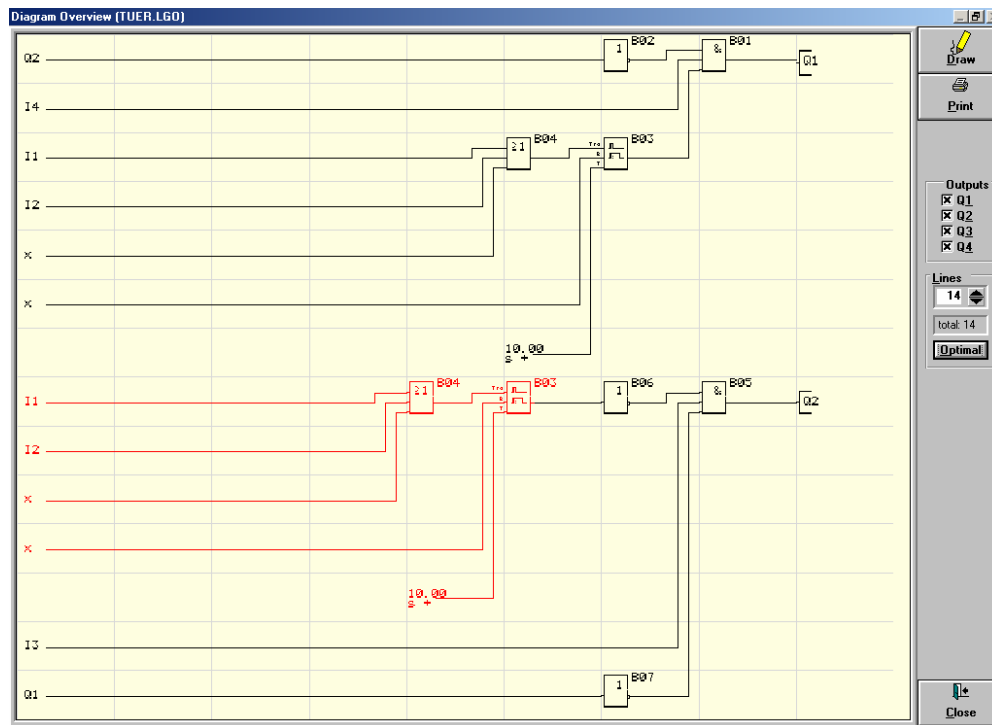


Slika 38 – Opcija ↓BN

Sledeći gore opisanu proceduru, unosimo program prema odgovarajućoj logičkoj shemi.

Tokom unošenja programa, LOGO! se ponaša tako da nas vodi kroz postupak. Drugim rečima, kada god specificiramo neki unos (napr. ulaz u neki blok), potvrđivanjem tog unosa, prelazi na poziciju na kojoj očekuje novi unos itd... Ovo je naročito korisno kod

definisanja složenijih logičkih kola.



Slika 39 – Kopirani deo dijagrama iz primera TUEP

Kada završimo unos neke logičke sheme od odgovarajućih ulaza do datog izlaza (npr Q1), LOGO! se pozicionira na definisani izlaz (Q1), pa prikaz odgovara **slici 35**. Pomeranjem kursora na dole, ▾, na displeju se prikazuje izlaz Q2, za koji pravimo odgovarajuću logičku vezu, sledeći prethodno opisan postupak.

Kada smo uneli željeno logičko kolo, pritiskom na dugme **ESC**, LOGO! se vraća korak unazad, odnosno u podmeni opcije Program.

Ukoliko želimo da pogledamo shemu unetog programa, izaberemo opciju **Diagram Overview...** padajućeg menija **View**. U ovom prikazu je ispod svakog ulaza, odnosno izlaza, ispisan uneti "Program Comments".

3.2.3. Promena postojećeg programa

Ukoliko želimo da promenimo već uneti program, potrebno je ući u programski mod. Iz menija **Program** preba izabrati opciju **Edit Prg** – isto kao i kod unošenja novog programa (**Slika 34**).

Na displeju modula, pojavljuje se prvi izlaz (uobičajna pozicija

kursora, kada izaberemo pomenutu opciju). Koristeći kursorske dugmiće, dolazimo do tačke u kojoj želimo da napravimo promenu. Pritiskom **OK** aktivira se data pozicija, što se identifikuje crvenim kvadratićem, gde bismo željnu opciju – unosimo novi blok ili definišemo vrstu konektora. Ako na datom mestu unosimo novi blok, blok odnosno input koji se prethodno nalazio na tom mestu (stara shema) postaje prvi input novog bloka.

Ako selektujemo blok u želji da ga zamenimo blokom sa drugom funkcijom, postoji ograničenje u pogledu broja ulaza ova dva bloka. Naime, dva bloka koja menjamo jedan drugim moraju imati isti broj inputa, odnosno ulaznih parametara. Na primer, ne možemo menjati funkciju I i RS blok.

Na kraju treba uzeti u obzir i memorijska ograničenja koja se odnose na veličinu programa. Ona su različita za različite generacije i verzije modula. U našem slučaju, modul LOGO! 230RC verzija 1.1 može imati sedam blokova od jednog ulaza do izlaza. Ako program sadrži dati broj blokova u kolu, onda ne možemo da unesemo novi blok prilikom editovanja tog programa.

Kada završimo sa promenama, iz ove opcije izlazimo, kao i u prethodnom slučaju, klikom na **ESC**.

3.2.3.1. Brisanje i zamena blokova

Ukoliko želimo da obrišemo blok iz neke sheme, potrebno je postaviti kursor na ulaz bloka za koji je vezan blok koji brišemo. Ako je blok B02 povezan za B01, postavimo kursor na ulaz B01 tj na slovo B u B02. Klikom **OK**, otvara se poznata lista opcija, gde bismo \downarrow **Co**, a zatim **x**. Tako smo blok B02 obrisali zajedno sa svim elementima koji za njega bili vezani.

Kada koristimo seriju blokova na više pozicija u programu, i ako obrišemo jedan od blokova iz ove serije postoje dva karakteristična slučaja. Prvi, ako brišemo prvi blok iz ove serije, gledano od izlaza serije, ne postoji nikakav uticaj na ostali deo kola gde se koristi serija. Drugi, kada obrišemo neki od blokova unutar kopirane serije, onda će se to odraziti na svaki deo kola u kome je kopirana serija blokova.

Zamena bloka nesrodnim blokom može se obaviti tako što prvo obrišemo postojeći blok, a onda unesemo novi blok na tu poziciju.

3.2.4. Brisanje čitavog programa iz modula

Brisanje čitavog programa iz modula postiže se u programskom modu izborom druge opcije **Clear Prg** – Slika 40.



Slika 40 – Brisanje programa iz modula

Kursor pozicioniramo na **Clear Prg**, kliknemo **OK**, posle čega se otvara dijalog u kome treba da potvrdimo akciju, izborom opcije **YES** – Slika 41.



Slika 41 – Potvrđivanje opcije Clear

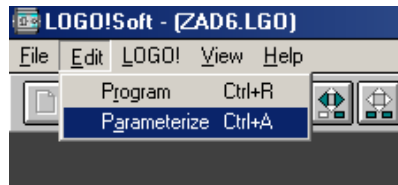
Iza ovoga, modul je vraćen u programski mod. Sada možemo ovde uneti neki novi program.

3.3 Parametarski Mod

Parametarski mod se uključuje već ranije pomenutom kombinacijom dva prsta – istovremeno se pritisnu **ESC** i **OK**, kada govorimo o modulu, odnosno izborom opcije **Parametrize** padajućeg menija **Edit** u slučaju rada sa LOGO!Soft – om (Slika 42). Da bi ušli u ovaj mod, potrebno je da startujemo program izborom opcije **Start** – Slika 43.

U ovom modu možemo da menjamo parametre procesa, a to su vremenski parametri No, parametri kašnjenja T i parametri brojača

Par.



Slika 42 – Putanja za parametar mod



Slika 43 – Startovanje programa

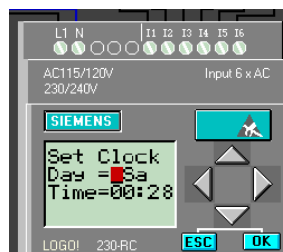
U parametarskom modu imamo dve opcije – Set Clock i Set Param kao što se može videti na slici 44.



Slika 44 – Parametar mod

3.3.1. Setovanje sata – tajmera

Ovde praktično govorimo o setovanju vremena uključivanja tajmera. Da bi zadali ovo vreme, potrebno je kursor postaviti na jedan od **No** parametara. Klikom na **OK**, LOGO! otvara prozor za set prikazan na slici 45.



Slika 45 – Setovanje tajmera

Vidi se da je kursor postavljen na mesto koje odgovara definisanju dana u nedelji. Pomoću kursora ∇ i Δ izaberemo dan u nedelji. Ovde je moguće definisati i više dana u nedelji u smislu od...do. Kursori za

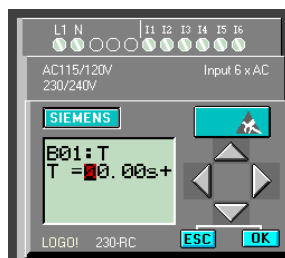
prelazak sa linije koja odgovara danu na liniju koja odgovara vremenu uključivanja T su < i >. I ovde se kursori ∇ i Δ koriste za promenu vrednosti na datoj poziciji. Ovde se može selektovati i opcija --:-- , koja označava da nema uključivanja. Kada unesemo željene vrednosti, klopom na OK, potvrđujemo operaciju, čime se LOGO! vraća u parametarski mod (Slika 44).

3.3.2. Setovanje parametara

Zadavanje vrednosti parametara se postiže izborom opcije **Set Param** (Slika 44). Ovako možemo menjati, samo one parametre koji imaju oznaku + tj one koji su "vidljivi", dok parametre označene znakom – ne možemo menjati.

Kako je način zadavanja parametara **T** odnosno **Par** isti, ovde je opisan samo postupak zadavanja parametra kašnjenja T.

Postoje dve mogućnosti – jedna se odnosi na zadavanje ove vrednosti u programskom modu, u samom procesu unošenja programa u modul, a drugi je promena istih u parametarskom modu. U oba slučaja otavra se prozor za definisanje parametara koji je dat na **slici 46**.



Slika 46 – Definisanje parametra T

Kao što se na **slici 46** može videti, prva linija se odnosi na redni broj bloka u dijagramu i vrstu parametra (ovde je B01:T), dok se u drugoj liniji bira vrednost datog parametra koristeći tastere ∇ i Δ kada se kursor nalazi na određenoj poziciji (crveni kvadratić). Naravno, klik OK, potvrđuje unos, dok ESC ga poništava. Ozanka + na kraju pokazuje dali postoji mogućnost promne datog parametra ili ne. Na potpuno analogan način vrši se unos i drugih parametara.

3.4. Komunikacija između LOGO! modula i PC

Na kraju ovog poglavlja obrađujemo i komunikaciju LOGO! – PC. Već je u odeljku 2.4. Uspostavljanje veze LOGO! – PC opsiana hardverska veza, tako da ćemo ovde opisati mogućnosti transvera programa između ovih elemenata.

Razmena podataka između LOGO! i računara vrši se kada je LOGO! uključen i kada je izaberemo opcija PC/Card – **Slika 47a**, posle čega definišemo vrstu komunikacije modula i smer **Slika 47b**.



Slika 47 – Definisanje veze modula sa PC

Kopiranje programa sa PC na LOGO! vrši se automatski, kada je veza uspostavljena i LOGO! uključen. Za vreme trajanja transvera, pojavljuje se indikator koji pokazuje postojanje akcije. Ovde treba samo izabrati pravu opciju u meniju koji je prikazan na **slici 47b**.

4. Urađeni primeri za vežbu

Ovde su urađeni primeri koji su dati u radu Vladimira Đorđevića "Programabilni logički kontroleri", kako bi uporedili projektovanje sistema za Omron PLC i LOGO! modul.

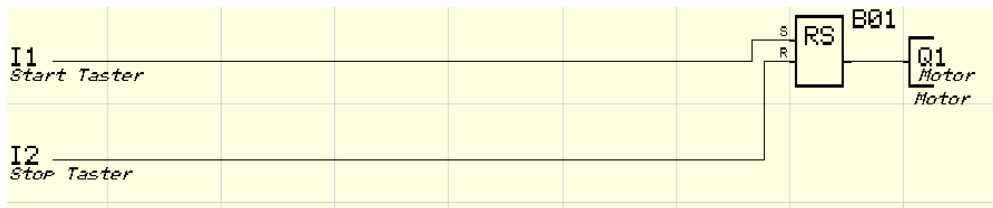
Jedina razlika je to što je 10. zadatak iz pomenutog rada zamenjen drugim primerom, zato što je broj ulaznih veličina prevazilazi mogućnosti ovog modula – obzirom da radimo u prvoj verziji softvera, ovaj problem može biti rešen korišćenjem bilo novije verzije modula (sa većim brojem ulaza), ili proširivanjem onog koji nam je na raspolaganju. Zato je umesto 10. zadatka dat jedan primer sa siemens – ovg sajta.

Ovde je dat samo dijagramski prikaz rešenja, a sam postupak projektovanja sproveden je u skladu sa opsivanim postupcima u ovom radu.

Primer 1

Postavka: Napraviti program za PLC kojim se na pritisak tastera startuje motor, a pritiskom na taster stop, zaustavlja.

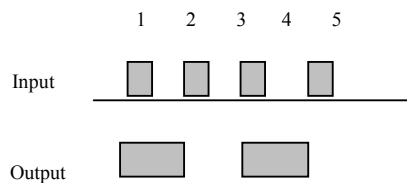
Rešenje: Kada je pritisnut **start** taster (I1) a **stop** taster (I2) nije pritisnut, **motor** (Q1) radi jer je RS rele setovan. Kada se taster **start** pusti, **motor** i dalje radi zato što RS rele ostaje u prethodnom stanju a tek kada je pritisnut taster **stop** (I2) **motor** prestaje da radi.



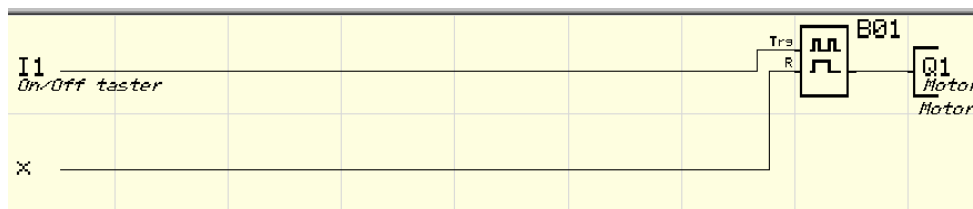
Slika 48 P_1 – Rešenje primera 1

Primer 2

Postavka: Napisati program kojim će se motor pokretati pritiskom na taster, i na isti taster zaustavljati.



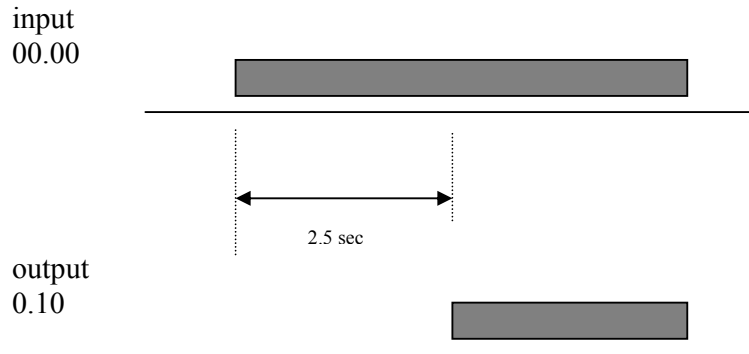
Rešenje: Rešenje ovog primera takođe je krajnje jednostavno, obzirom da se svodi na korišćenje jednog bloka - Pulsni rele



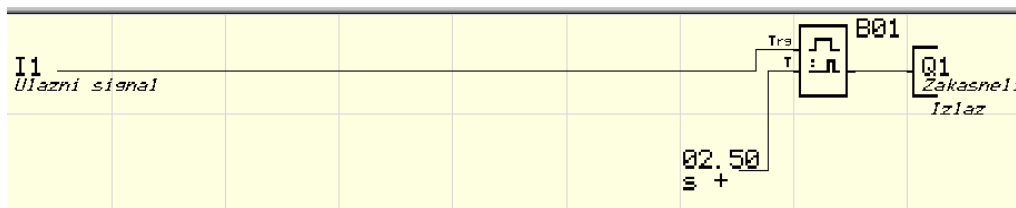
Slika 48 P_2 – Rešenje primera 2

Primer 3

Postavka: Napisati program kojim će se ulazni signal zakasniti 2.5 sekunde.



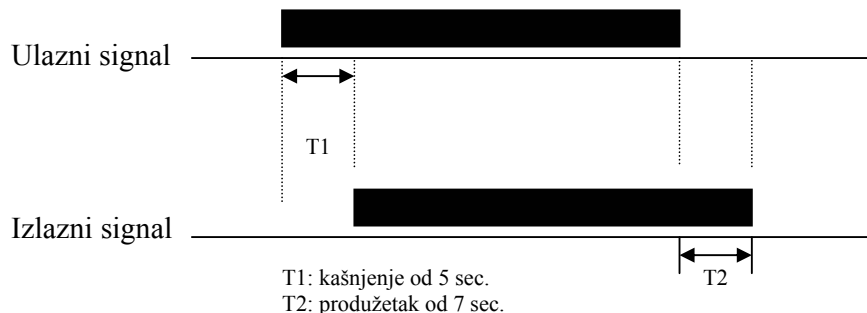
Rešenje: U ovom zadatku je korišćena funkcija kašnjenja **On - Delay**. Njena funkcija je da "**zakasni**" signal onoliko vremena koliko je u njoj definisano. Ovde je to 2.5 s a ovaj parametar se može menjati što pokazuje znak +



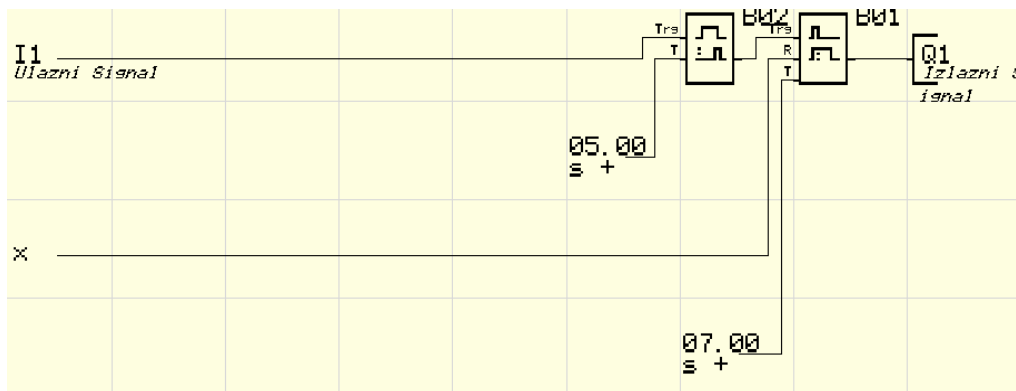
Slika 48 P_3 – Rešenje primera 3

Primer 4

Postavka: Napisati program kojim izlazni signal nastaje 5 sekundi posle početka ulaznog signala i koji će delovati još 7 sekundi po njegovom završetku.



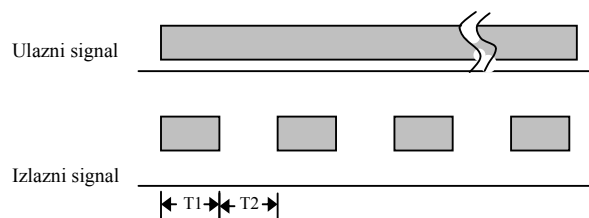
Rešenje: To se postiže pomoću dva tajmera i to jedan ON – Delay B02, za prednje kašnjenje od 5 sekundi i Off – Delay B01 za zadržavanje signala za još 7 sekundi. Izlaz iz On – Delay bloka javlja se tek posle 5 sekundi i setuje drugi tajmer. Kada se ugasi ulazni signal resetuje se prvi tajmer, tako da na ulazu u drugi nemamo signal. Ali signal iz drugog tajmera će postojati sve dok ne istekne zadato vreme T – ovde je to 7 sekundi.



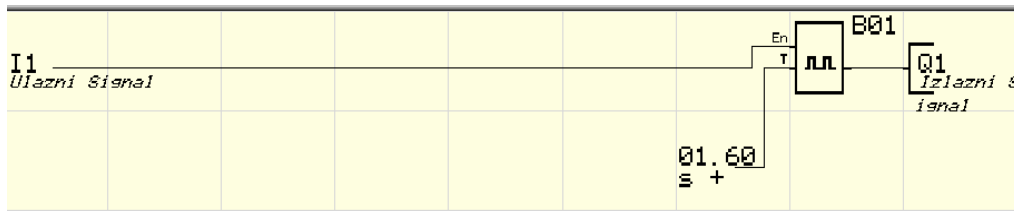
Slika 48 P_4 – Rešenje primera 4

Primer 5

Postavka: napisati program koji ponavlja izlazne signale u određenim intervalima kada se na ulaz dovodi signal.



Rešenje: I ovde se rešavanje ovog problema svodi na upotrebu jednog bloka – generatora impulsa čiji je način rada već ranije opisan. Dok postoji ulazni signal, na izlazu se generišu impulsi sa trajanjem intervala signal/pauza definisanim parametrom T. U ovom primeru je uzeto da je T=1.6 sekundi a može se menjati.

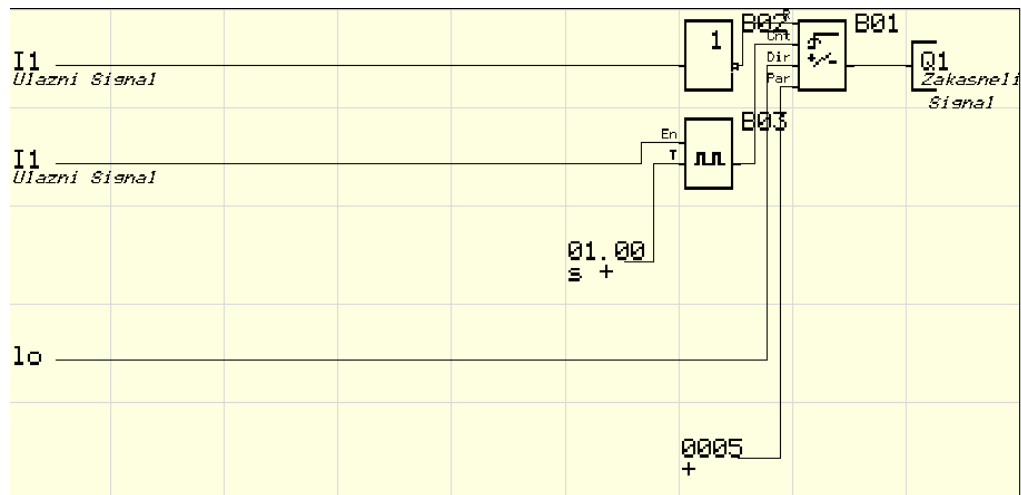


Slika 48 P_5 – Rešenje primera 5

Primer 6

Postavka: napisati program koji omogućava duže kašnjenje signala od 99.99 sekundi (maksimuma koji može da se definiše).

Rešenje: Kolo u ovom zadatku daje jedno od rešenja. Za njegovo postizanje iskorišćena je kombinacija brojača (*counter* – **CNT**) i generatora impulsa. Funkcija **CNT (blok B01)** broji onoliko ulaznih signala koliko je u njoj definisano (u našem primeru 5). Pored ulaza koji se prebrojavaju f-ja CNT mora da ima i signal koji je resetuje. U našem slučaju je to **zatvoreni ulaz**.



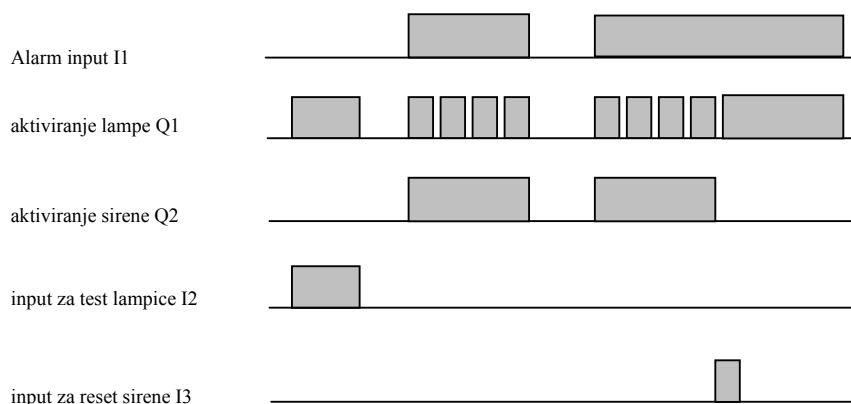
Slika 48 P_6 – Rešenje primera 6

Ideja koja je ovde realizovana je da ulazni signal dolazi na generator impulsa koji ima definisan parametar $T=0.5$ sekundi, koji će na izlazu posle svake sekunde davati impuls koji prebrojava brojač. Kada brojač dostigne vrednost definisanu za Par (u našem slučaju $Par=5$), brojač daje signal na izlazu, uz uslov da signal na ulazu postoji – brojač nije resetovan. Znači, u odnosu na ulazni, izlazni

signal pojavio se sa kašnjenjem od 1X5 sekunde. Naravno kolo sa ovolikim kašnjenjem smo mogli da izvedemo i sa običnim On – Delay blokom, a 1 sekunde i 5 su vrednosti izabrane zbog simulacije rada modula (da ne bi dugo čekali). Umesto njih su mogle da stoje 99.99 sekundi i 9999 brojanja ciklusa. Tada bi ukupno kašnjenje bilo 99.99X9999 sekundi.

Primer 7

Postavka: Napisati program koji na način prikazan na vremenskom dijagramu pali i gasi alarmno svetlo i sirenu u slučaju potrebe.



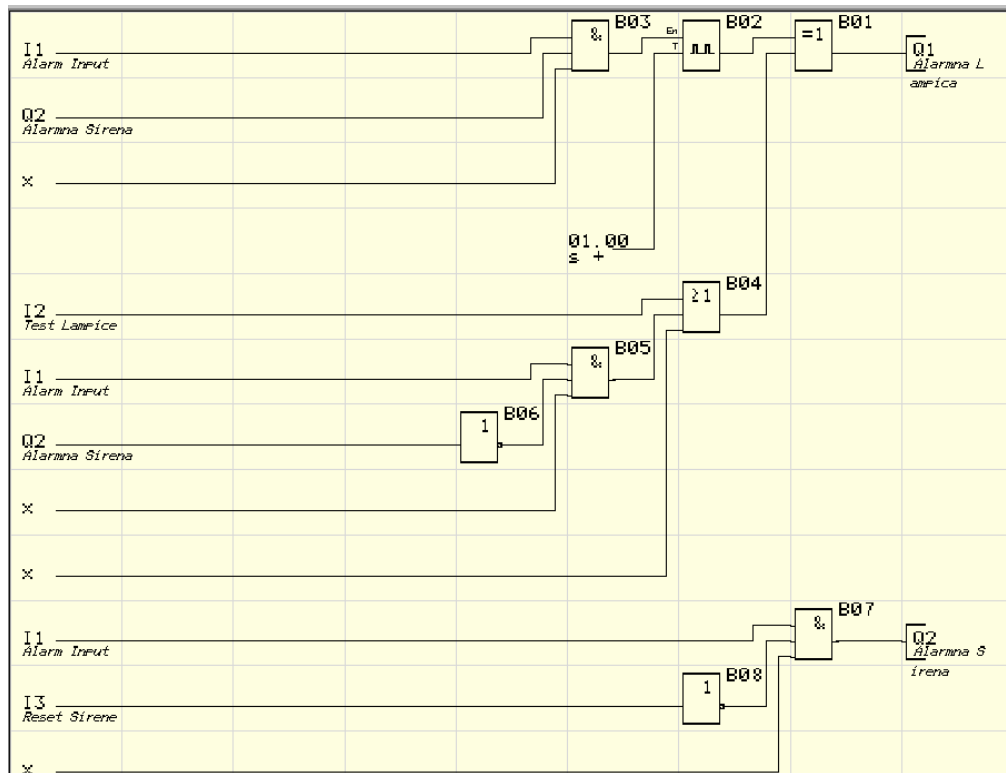
Rešenje: Alarmna sijalica se pali kada je pritisnut taster za testiranje sijalice (I2).

Ako postoji alarm (I1) i generator impulsa B02 daje pulsirajući signal (trajanje/pauza 1 s) i taster za potvrdu alarma nije pritisnut – alarmna sijalica (Q1) zasvetli.

Sijalica će takođe svetleti ako alarm postoji u periodu između pritiska tastera za potvrdu alarma i prekida alarma.

Alarmna sirena (Q2) će biti upaljena ako postoji alarm (I1) i nije pritisnut taster za potvrdu alarma (I3).

U slučaju kada se pritisne taster za potvrdu alarma, alarmna sirena se gasi (za to vreme sijalica i dalje radi), operater na mašini otklanja razlog zbog kog je alarm upaljen, alarm se gasi, i na kraju se gasi i sijalica.



Slika 48 P_7 – Rešenje primera 7

Primer 8

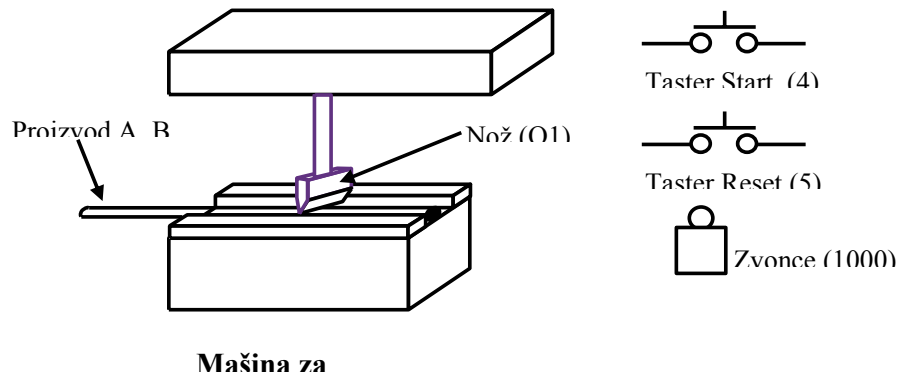
Merenje veka trajanja noža za sečenje

Nož se koristi za isecanje tri vrste proizvoda: A, B i C. Zbog istupljenosti, menja se posle 1000 odsecanja komada A, 500 komada B ili 100 komada C. Delovi mogu nasumično dolaziti na red za isecanje. Kada se nož istupi, aktivira se zujalica.

Za izvođenje ovakvog zadatka koriste se tri senzora, svaki za po jednu vrstu komada, i jedan senzor koji je indikator završene operacije sečenja. Tasterom se startuje celokupan proces. Ulazi i izlazi dati su u tabeli

Input Comments	
I1: Start taster	I4: Senzor Komada C
I2: Senzor Komada A	I5: Kraj Operacije
I3: Senzor Komada B	I6: Reset Brojaca

Output Comments	
Q1: Secenje	Q3:
Q2: Zujalica	Q4:

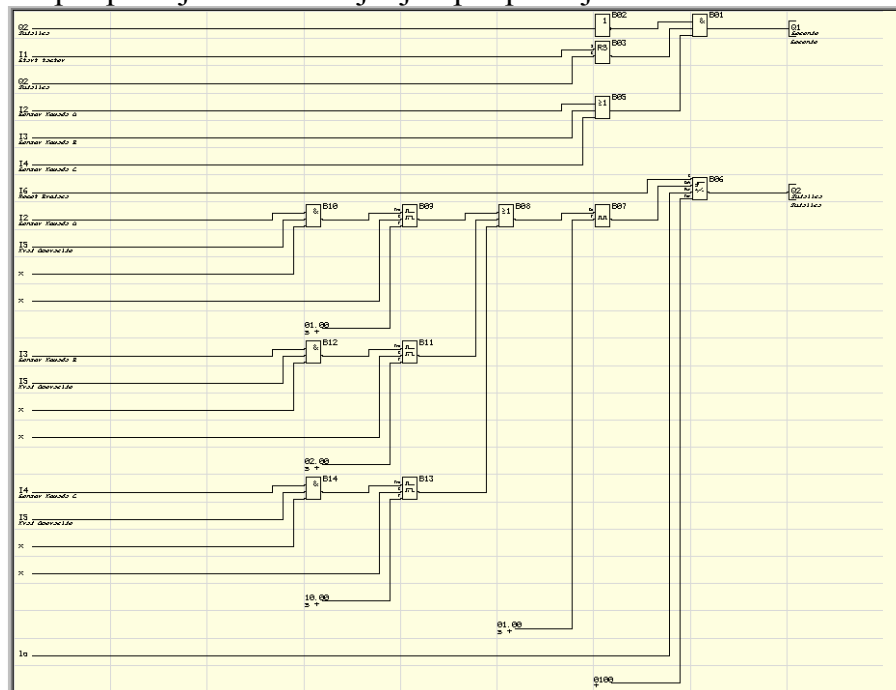


Po startovanju mašine neophodno je pritisnuti taster reset da bi se resetovao brojač B06.

Kada je pritisnut taster start (I1) setuje se rele B03, sve dok se operacija sečenja ne završi (senzor 0.00).

Ako signali sa B02 i B03 postoji i jedan od senzora za tip proizvoda je zatvoren (I2, I3 ili I4) uključuje se izlaz sečenje Q1.

Sada, u drugom delu dijagrama, proverava se da li je isečeni deo A, B ili C. Sistem je napravljen uz pretpostavku da signali sa senzora komada nisu duži od 0.5 sekundi. Ako je A onda se od vrednosti na izlazu iz bloka B09 kraći od 1.5, pa će generator B07 dati jedan impuls koji će izbrojati brojač B06, ako je B onda je izlaz iz B11 kraći od 2.5, pa brojač broji 2 impulsa, odnosno ako je C broji 10 impulsa jer je dužina signala na izlazu iz B13 manja od 10.5 sekundi. Kada se izbroji 1000 impulsa u brojaču, ovaj daje izlaz koji uključuje zujalicu, operator mašine bi zamenio nož za sečenje i pritisnuo taster reset. Tada vrednost u brojaču opet postaje 1000 i brojanje opet počinje.

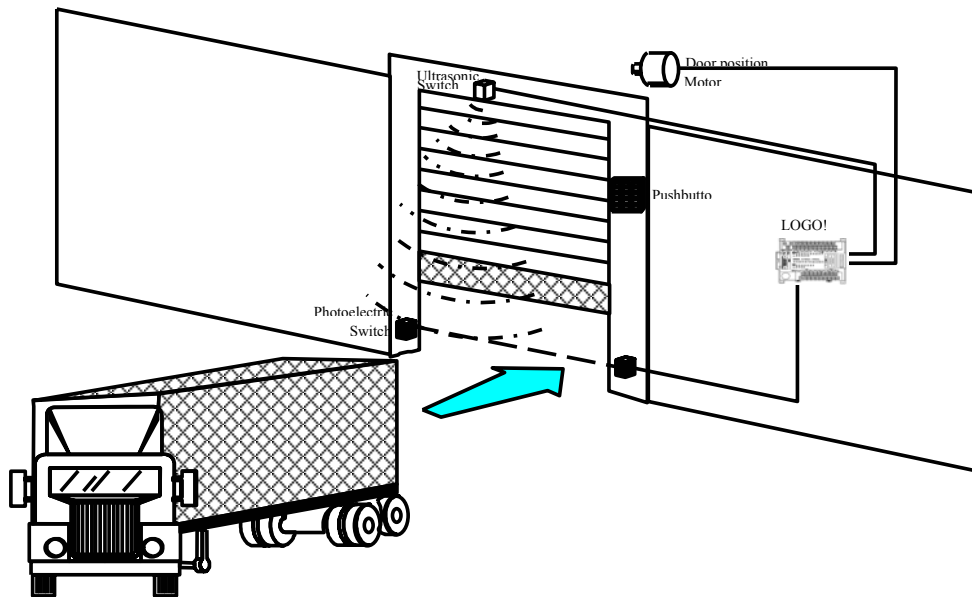


Slika 48 P_8 – Rešenje primera 8

Primer 9

Postavka: Napisati program kojim se upravlja otvaranje i zatvaranje vrata skladišta. Problem je opisan na slici P.9.

Kao detektor vozila koje prilazi, koristi se ultrasonični prekidač. Odvojeni fotosenzori detektuju prolaz vozila preko prekida svetlosnog zraka. Kao odziv ovih signala, upravljačka kola generišu signale koji pogone motor koji spušta i podiže vrata.

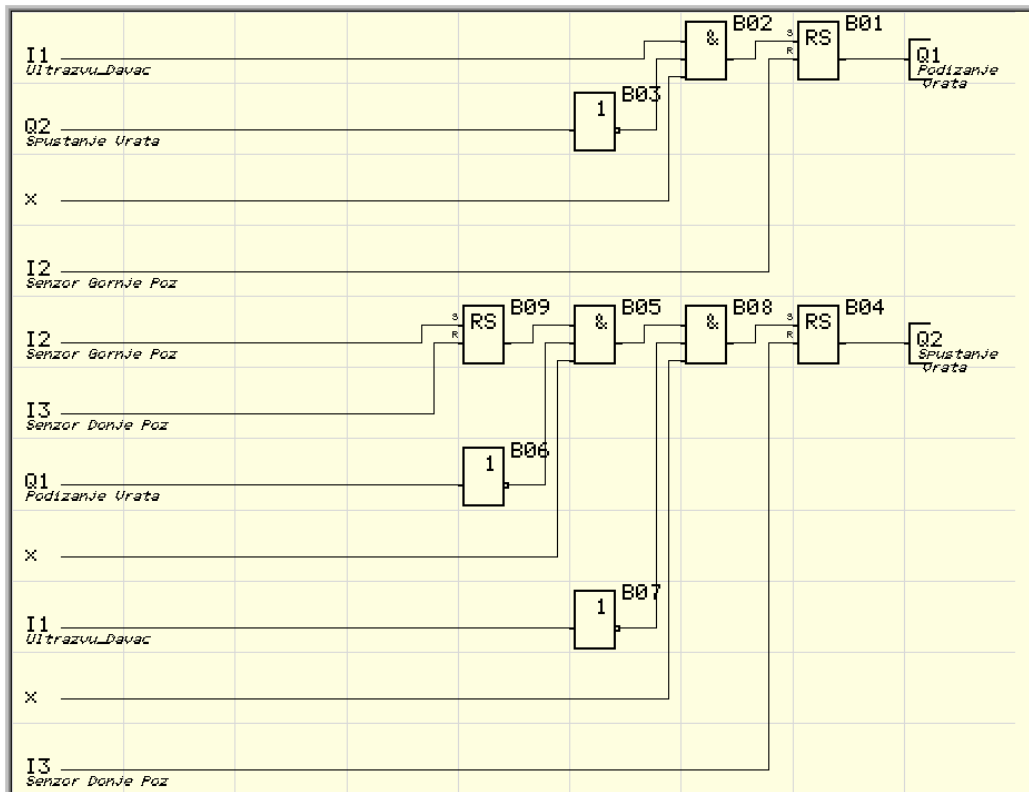


Podizanje vrata

Kada je ultrazvučni davač (I1) setovan, setuje se B01 i vrata se podižu, sve dok vrata ne dostignu gornji položaj (senzor za gornji položaj I2 resetuje B01). Pošto postoji signal sa I1 i vrata su stigla u gornju poziciju, vrata ostaju ovtorena.

Spuštanje vrata

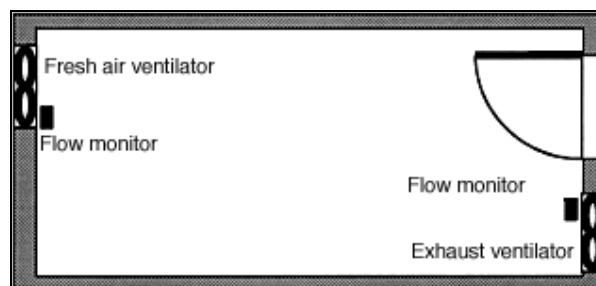
Signal sa senzora za gornji položaj vrata, setuje rele B09, pa ako se vrata ne podižu ($Q1=0$), odnosno signal sa I1=0, setuje se B04 - vrata se spuštaju. Ovaj rele B04 resetuje signal sa senzora donje pozicije I3, čime se postupak završava.



Slika 48 P_9 – Rešenje primera 9

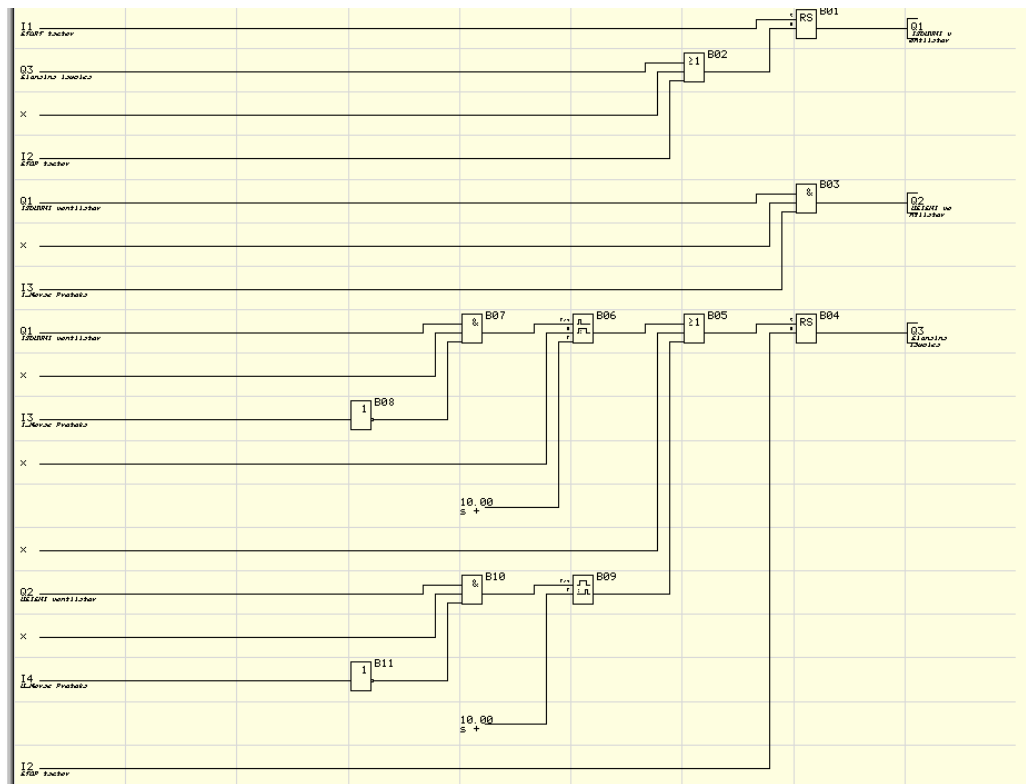
Primer 10 (Siemens – ov primer ventilatora)

Postavka: Prostorija ima dva ventilatora – izduvni i usisni. Oba ventilatora su kontrolisana od strane kontrolnog uređaja koji meri protok. Usisni ventilator ne sme biti uključen ako merač protoka ne registruje regularan rad izduvnog ventilatora. Ugrađena je još i kontrolna lampica koja ukazuje kvar na ventilatorima.



Rešenje: Ako je uključen start taster (I1), biće setovan izduvni ventilator. Njega će resetovati ili signalna lampa (Q1) ili Stop Taster (I2). Kada radi Q1 i signal sa izduvnog merača protoka (I3) uključice

ventilator Q2 – usisni.

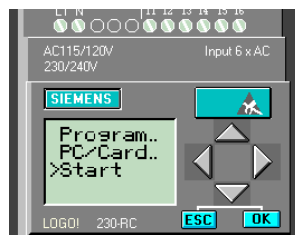


Slika 48 P_10 – Rešenje primera 10

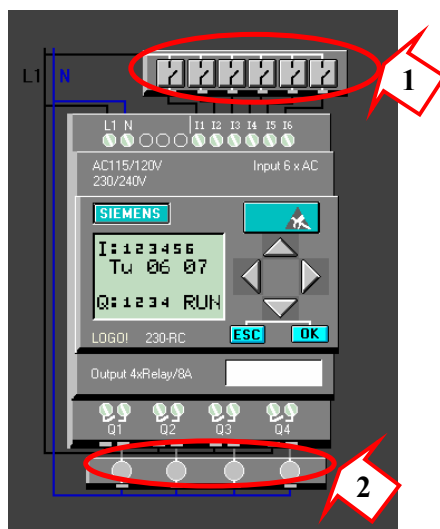
Kada za vreme rada ventilatora nemamo signale sa odgovarajućih davača, uključice se signalna lampica, koja se resetuje pritiskom na taster stop.

5. Simulacija rada LOGO! modula

Simulacija rada programa koji smo napisali u LOGO! Soft – u, je krajnje jednostavna. Potrebno je samo da se izabere opcija Start iz glavnog menija modula – **Slika 49**, posle čega se otvara prozor dat na slici 50.

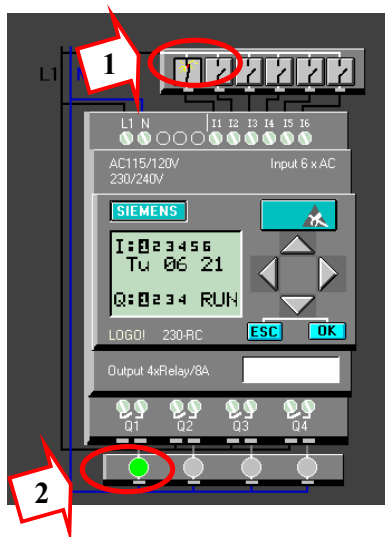


Slika 49 – Startovanje programa



Slika 50 – Prikaz startovanog modula

Kao što se na **slici 50** može videti, prikazani su ulazi (**pozicija 1**), odnosno izlazi (**pozicija 2**), kao i povezani vodovi za napajanje. Ulazi su prikazani kao normalno otvoreni. Klikom na ikonice koje odgovaraju datim ulazima, zatvaraju se veze, oni postaju zatvoreni – **Slika 51 pozicija 1** ulaz I1 je zatvoren. Kada se pojavi signal na izlazu, krug koji odgovara datom izlazu postaje zelen – **Slika 51 poz 2** signal na izlazu Q1.



Slika 51 – Ulazi i izlazi u toku simulacije

Stanje ulaza i izlaza može se pratiti na displeju modula (crni kvadratići označavaju ulaze i izlaze koji imaju signale).

Iz ove opcije se izlazi izborom opcije **Program** padajućeg menija **Edit**.

6. Nove generacije LOGO! modula. Tendencije razvoja

Za manje od deset godina, koliko su u upotrebi, LOGO! moduli su dosta razvijani i stalno poboljšavani u pogledu performansi i oblasti primene.

Do sada su razvijene četiri generacije ovog kontrolera, koje su nastajale tako da su upotpunjavale dijapazon mogućnosti, zadržavajući osnovne pozitivne karakteristike.

Prvo, svaka generacija je imala veći broj integrisanig funkcija, tako da sada najnovije generacije imaju integrisanih preko 30 funkcija – proširena je ponuda specijalnih funkcija u pogledu novih vrsta brojača tajmera i dr. Primer nekih od njih dat je na **slici 52**.



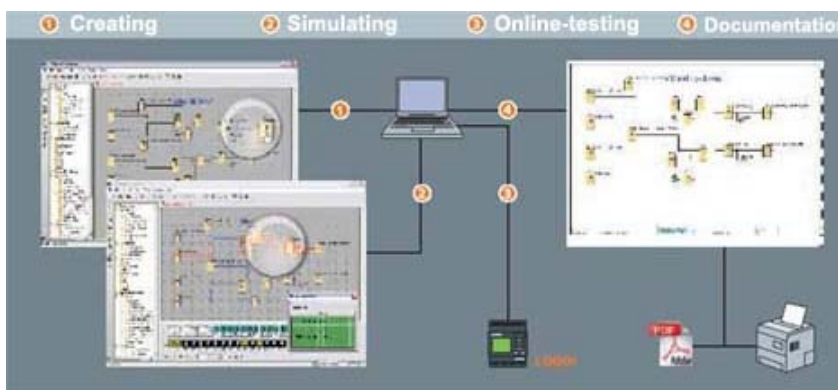
Slika 52 – Nove funkcije u LOGO modulu!

Na **slici 52** možemo konstatovati sve veću zasptupljenost i posvećenost tretmanu analognih signala pomoću LOGO! modula. Sadašnja dostignuća su takva da se ovaj modul može vezati i za analogne izvršne organe.

Kada govorimo o nadgradnji, tj. prošitrivanju osnovne jedinice modula, treba istaći da se velika pažnja poklanja i dodatnim modulima. Sada se dodatni moduli dele na analogne, digitalne, i komunikacijske.

Dosta je usavršavan i prateći softver za programiranje i simulaciju, tako da sada imamo jednu složenu strukturu, koja se može prikazati kao na **slici 53**. Najveća prednost novog softvera je krajnje jednostavna upotreba. Softver podržava i programiranje ranijih generacija modula.

LOGO! modul se danas može koristiti i sistemu sa drugim vrstama kontrolera, tako da je i u tom smislu izuzetno fleksibilan.



Slika 53 – Najnoviji LOGO!Soft

7. Zaključak

Programabilni logički kontroleri predstavljaju nezaobilazni deo inženjerskih aplikacija iz domena upravljanja. Prednosti u odnosu na klasične električne komponente su ušteda energije, vremena, realizacija različitih funkcija i dr.

Logički modul LOGO! kompanije SIEMENS pripada svetskom vrhu kada je reč o programabilnim logičkim kontrolerima. Ovakvoj poziciji najviše su doprinele izuzetne karakteristike, kao i intenzivan rad na usavršavanju i proširenju oblasti primene.

Kao najbitnije osobine ovog kontrolera istakao bih – male dimenzije, postojanost u raznim ambijentalnim uslovima, jednostavno rukovanje i programiranje, mogućnost direktnog programiranja i reprogramiranja, lako instaliranje, otpornost na mehaničke vibracije, dr. Ove karakteristike su omogućile primenu u jednostavnijim sistemima upravljanja, za koji su inače projektovani PLC kontroleri.

O eventualnim manama LOGO! modula, ne bih mogao mnogo da pričam, obzirom da je upotrebljena literatura publikacija proizvođača, ali je ipak potvrda kvaliteta i broj jedinica koje su do sada instalirane (preko 100.000). Ne treba izostaviti i činjenicu da je ovde predstavljena prva verzija modula, koja je očigledno na znatno nižem nivou u odnosu na novije generacije.

Na kraju, sadržaj ovog rada ukazuje samo na neke osnovne tehnike i princip rada LOGO! modula, pa ako se čitalac ovog rada opredeli da primeni baš ovaj modul u nekoj konkretnoj instalaciji, onda je primarni cilj autora ostvaren.

L I T E R A T U R A

- [1] **Matijević, Milan**, *konsultacije*
- [2] **Draganović, Ljubiša**, *Projektovanje sistema automatskog upravljanja*, Lola insatitut Beograd, 2000.
- [3] **LOGO!Soft Help System**
- [4] **Đorđević, Vladimir**, *Programabilni logički kontroleri*, MF Kragujevac, Kragujevac, 2001.
- [5] **Petrović, Đorđe**, *Projektovanje digitalno upravljano laboratorijskog modela procesa za sušenje vazduha*, MF Kragujevac, Kragujevac, 2001.
- [6] **www.ad.siemens.de/logo**
- [7] **www.siemens.de/logo**