**TEHNIČKA ŠKOLA ŽUPANJA**

**Županja, Veliki kraj 42**

Završni rad

**NASLOV ZAVRŠNOG RADA**

**Mentor: Učenik:**

Ime i prezime mentora, mag.ing.el. (prof.) Ime i prezime učenika, 4a

**Županja, lipanj 2017**

**SADRŽAJ**

[1. UVOD 3](#_Toc328486303)

[2. PRINCIP RADA PLC-a 4](#_Toc328486304)

[2.1. Ulazni dio ( digitalni , analogni ulazi ) 4](#_Toc328486305)

[2.2. Izlazni dio ( digitalni, analogni izlaz ) 5](#_Toc328486306)

[2.3. Centralna procesorska jedinica 6](#_Toc328486307)

[2.4. Modul napajanja 6](#_Toc328486308)

[2.5. Način rada PLC uređaja 6](#_Toc328486309)

[2.6. Komunikacijsko sučelje 7](#_Toc328486310)

[2.7. Memorijski blok za program i podatke 8](#_Toc328486311)

[2.8. Moduli za proširenje 8](#_Toc328486312)

[3. PROGRAMIRANJE 10](#_Toc328486313)

[3.1. STL ( Statement liste ) 10](#_Toc328486314)

[3.2. Kontaktni dijagrami 13](#_Toc328486315)

[4. ZAKLJUČAK 18](#_Toc328486316)

[5. LITERATURA 19](#_Toc328486317)

#

# UVOD

 U ovom seminarskom radu govorit ćemo o strojevima upravljanim PC-om, konkretnije o upravljanju strojem korištenjem programskih logičkih kontrolora ili skraćeno PLC-a.  PC  je osobno računalo koje se kao što već znamo sastoji od ulazno/izlaznih uređaja kao što su (tastatura,miš,monitor…),procesora (CPU), memorije. Isto tako i programski logički kontrolori imaju  neke svoje  ulazno/izlazne  uređaje, memoriju, procesor samo u odnosu na PC (računalo) su manjih dimenzija pa se mogu nazvati i maleni PC-ovi.Krajem 60-tih godina industrijski proizvodni pogoni su uglavnom bili upravljani sustavima zasnovanim na relejnim krugovima (relejna logika).
 Svaki put kada bi se promijenio proizvodni program morale su se napraviti prilagodbe upravljačkih sklopova što bi iziskivalo veliko vrijeme praznog hoda proizvodnje, a s time i velike troškove. Iz toga se vidi da su relejni upravljački sustavi bili vrlo nefleksibilni, te da bi se promijenila funkcija upravljanja jednog takvog relejnog sklopa nije svaki put bilo dovoljno promijeniti njegovo ožičenje, nego je ponekad trebalo krenuti sa sastavljanjem novog sklopa. Otprilike u isto vrijeme je i razvoj mikroprocesora došao do određenog  nivoa  te se pojavila ideja o izradi elektroničko-kompjuterskog upravljačkog sustava koji bi se jednostavno sa promjenom proizvodnog programa dao reprogramirati. Tada su napravljeni prvi programski logički kontrolori skraćeno PLC i vrlo brzo pokazali izuzetne prednosti u odnosu na relejni upravljački sustav –  pouzdaniji je od relejnog sustava jer nema mehaničkih pokretnih dijelova; fleksibilniji jer ga pri promjeni proizvodnje treba samo reprogramirati, a ne mijenjati ožičenje; smanjeni je opseg ožičenja i greške u ožičenju; dimenzije su višestruko manje jer su vremenski releji,  brojači i ostale relejne upravljačke komponente riješene softverski. Osim toga PLC kao industrijsko računalo otporan je na razne nepovoljne utjecaje iz proizvodnje kao što su prašina, vlaga, visoka temperatura, vibracije, elektromagnetski utjecaji jer je samim svojim ustrojem napravljen tako da se postavi u neposrednoj blizini procesa kojim upravlja. Kao takvi sa visokim stupnjem fleksibilnosti PLC-ovi su vrlo brzo bili široko prihvaćeni.Prvobitna primjena PLC-ova je bila u automobilskoj industriji (u tvornicama automobila).Imali su prioritet da smanje vrijeme zastoja prilikom promjene proizvodnog procesa.

# PRINCIP RADA PLC-a

 Da bi se objasnio način rada PLC uređaja potreban je kratak pregled njegovih osnovnih cjelina. Svi PLC uređaji od mikro PLC-a do najvećih PLC sustava od preko 1000 U/I signala imaju, u principu, istu hardversku strukturu, odnosno iste osnovne cjeline:

* ulazni dio (digitalni, analogni ulaz
* izlazni dio (digitalni, analogni izlazi)
* CPU, tj. centralnu procesorsku jedinicu
* memorijski blok za program i podatke
* mrežni dio za napajanje te komunikacijsko sučelje
* moduli za proširenje

**Slika 2.1.** *Osnovne cjeline PLC uređaja*

* 1. Ulazni dio ( digitalni , analogni ulazi )

 Ulazni dio PLC-a su priključuje stezaljkama na vijke koje se spajaju na dolazni signali iz procesa preko kojeg se upravlja. Informacije koje PLC prima na svojim ulaznim stezaljkama mogu biti digitalne (diskretne) i analogne. Digitalna ulazna informacija može biti npr. signal s krajnje sklopke, senzora, tipkala i sl. dok analogna ulazna informacija može biti npr. naponski signal 0-10 V s mjernog pretvornika tlaka,temperature i sl. Za digitalnu informaciju visoko stanje iznosi 14-30 V, a nisko stanje 0-5 V. Analogna informacija može biti u raznim oblicima – strujni 0-20 mA, strujni 4-20 mA,naponski 0-10 V, naponski –10 - +10 V uz određenu rezoluciju (8 ili 16 bitni A/D pretvornik). Prilagodba signala s uobičajenog ulaznog napona od 120-230 V uključuje opto izolaciju signala, što je vrlo važno kako bi se galvanski odvojili strujni krugovi, čime se sprječava protok struje uslijed potencijalnih razlika  strujnih krugova, te filtriranje signala kako bi se smanjile visokofrekventne smetnje, odnosno smetnje uslijed statičkih pražnjenja.

* 1. Izlazni dio ( digitalni, analogni izlaz )

Izlazni dio PLC-a su priključne vijčane stezaljke na koje se spajaju izvršni uređaji iz procesa kojima PLC šalje digitalne i analogne signale te na taj način upravlja procesom. Na digitalne izlaze iz PLC-a su najčešće spojeni magnetni svici, releji, sklopnici, motorne sklopke,signalne lampe, pneumatski razvodnici i sl., dok na analogni izlaze mogu biti spojeni npr. strujni signal za prikaz neke veličine na  pokaznom instrumentu, referenca brzine za frekvencijski pretvarač, PID regulirana veličina itd. Izlazne stezaljke također su opto izolirane od procesorske jedinice radi galvanske izolacije električnih krugova. Digitalni izlazi najčešće su izvedeni kao relejni, tranzistorski ili pomoću trijaka, a svaki od njih ima svoje prednosti ima ne:• relejni izlazi mogu se koristiti za sklapanje istosmjernih i izmjeničnih tereta, za struje do nekoliko ampera. Releji dobro podnose naponske udare i obzirom na zračni razmak između njihovih kontakata ne postoji mogućnost pojave pulsirajućih struja. Releji su međutim relativno spori prilikom sklapanja te imaju vijek trajanja (mjeren maksimalnim brojem sklapanja) manji od trijaka i tranzistora. Tranzistorski izlazi služe za sklapanje istosmjernih tereta, nemaju pokretnih dijelova koji se troše i bešumni su. Vrijeme reakcije im je brzo, ali mogu sklapati uglavnom struje do 0.5 A. Izlazi sa trijcima služe za sklapanje izmjeničnih tereta, a karakteristike su im slične kao tranzistorima.

* 1. Centralna procesorska jedinica

 Centralna procesorska jedinica s memorijom glavna je jedinica PLC uređaja. Najkraće rečeno procesorska jedinica čita stanja svih ulaza PLC uređaja (analognih i digitalnih), logički ih obrađuje u skladu s programom izrađenim od strane korisnika, te upravlja izlazima prema rezultatima dobivenim nakon logičke obrade.

* 1. Modul napajanja

 Kao i na svakom računalu modul napajanja je najrobusniji i najteži njegov dio. Neosjetljiv je na smetnje koje dolaze iz električne mreže kao i na kraće ispade mrežnog napona (trajanja 10-15 ms). Standardni ulazi napajanja PLC uređaja su: 120/230 V i 24 V.

* 1. Način rada PLC uređaja

 Pošto PLC prema promjeni stanja na njegovim ulazima mora kontinuirano korigirati stanja izlaza kako je to određeno logikom u korisničkom programu on tu internu obradu podataka vrti ciklički u beskonačnoj petlji. U osnovi ciklus obrade podataka podijeljen je na nekoliko dijelova:

**1. Obrada ulaznog stanja –** očitanje stanja ulaza te prijenos podataka ulaznog stanja u ulazni memorijski registar procesorske jedinice;

**2. Obrada programa** – programska obrada ulaznih stanja prema logički korisničkog programa te slanje rezultata u izlazni memorijski registar procesorske jedinice;

**3. Prijenos obrađenog programa na izlaze** – prijenos obrađenih podataka iz izlaznog memorijskog registra na fizičke izlaze PLC-a;

1. **Procesorsko organizacijsko vrijeme i komunikacija** – odvijaju se operacije potrebne za funkcioniranje operativnog sustava PLC uređaja te komunikacija sa vanjskim jedinicama. Vrijeme jednog ciklusa za oko 500 programskih naredbi se kreće oko 1,5 ms.



Slika 2.4.1. Ciklus rada PLC-a

2.6. Komunikacijsko sučelje

 Komunikacijsko sučelje ima višestruku namjenu. Prva i osnovna je komunikacija sa nadređenim PC računalom na kojem se piše upravljački program, šalje u PLC te dijagnosticira stanje rada ( slika 2.6. ).

****

**Slika 2.6.** *Povezivanje računala s PLC-om*

Ostale mogućnosti su komunikacija sa ostalim PLC uređajima i raznim senzorima preko njihove interne mreže (npr Device Net), komunikacija sa raznim vrstama operatorskih panela te komunikacija modemskom vezom. Gotovo svi PLC uređaji imaju ugrađen serijski port za komunikaciju ( RS-232 – električki standardi), a komunikacija se vrši preko protokola koji ovisi o proizvođaču uređaja (najčešće full duplex serijska veza).

2.7. Memorijski blok za program i podatke

 PLC korisnik prilikom programiranja koristi dva segmenta memorije procesorske jedinice – programske datoteke i datoteke podataka. Programske datoteke koriste korisnički definirane programe, potprograme i datoteku za dojavu i obradu grešaka. Datoteke podataka služe za memoriranje programski ovisnih podataka kao što su U/I status, postavljene i trenutne vrijednosti brojača i vremenskih članova te ostale memorijske konstante i varijable. Podaci programske datoteke i datoteke podataka pohranjuju se u dvije vrste memorije; RAM (eng. random access memory -memorija s izravnim pristupom) i EEPROM (eng. electricaly erasable programable read only memory – električki obrisiva programska memorija namijenjena isključivo za čitanje). RAM memorija u PLC uređajima obično je podržana baterijom kako se po nestanku napona napajanja ne bi izgubili podaci (koji se ipak mogu izgubiti ako se istroši baterija), dok EEPROM memorija trajno sprema podatke bez obzira na napon napajanja. Korisnički programi izvode se iz RAM memorije, a dobra je praksa da se pohrane i u EEPROM memoriji te da se učitavaju u RAM svaki put kada se uključuje PLC, ili u slučaju gubitka podataka iz RAM memorije (iz bilo kojeg razloga). Sistemski program i memorija za upravljanje radom PLC uređaja nisu vidljivi i dostupni korisniku, ali su od ključne važnosti za njegov učinkovit rad.

2.8. Moduli za proširenje

 U osnovi PLC uređaj je od jednog dijela te na sebi ima ograničeni broj ulaznih i izlaznih stezaljki. Kada je za proces potrebno više ulaza ili izlaza nego ih na sebi ima osnovni uređaj koriste se moduli za proširenje (slot). Modul za proširenje je poseban uređaj koji se spaja na PLC i koji na sebi ima dodatne ulazne i/ili izlazne stezaljke. Na taj način se PLC uređaj uvijek može proširiti bez da se nabavlja novi. Najčešće se moduli za proširenje prodaju kao moduli za digitalne ulaze i/ili izlaze te moduli za analogne ulaze i/ili izlaze.

Preporuča se da osnovni uređaj i moduli za proširenje koriste isti izvor napajanja. U pogonu moduli mogu biti udaljeni od osnovnog uređaja te se veza ostvaruje komunikacijskim kabelom. Broj modula koji se mogu spojiti osnovni uređaj ovisi o proizvođaču.



Slika 2.8.1. Modul za proširenje

# PROGRAMIRANJE

 Jednako kao i ostala industrijska računala i PLC izvodi program i prema njemu upravlja procesom, odnosno kontrolira ulaze i upravlja izlazima. Pisanje programa najčešće e izvodi preko nadređenog PC računala na kojem je instaliran softver za korišteni PLC-a. U editoru se napiše programski kod u nekom od programskih jezika te se zatim provjeri njegova sintaksa (sve u PC računalu). Ako program nema sintaksnih grašaka softver ga šalje u RAM memoriju PLC-a te je on spreman za rad.Komunikacija između PC računala i PLC-a je najčešće serijska(RS-232) te može biti aktivna iza vrijeme izvođenja programa na PLC-u. Na taj način na ekranu PC računala uvijek možemo pratiti stanja ulaza i izlaza te zadavati naredbe direktno preko tipkovnice i miša.PLC se također može programirati i preko ručnih programatora koji posjeduju mali LCD displej uz skromnu tipkovnicu. Takvi se uređaji direktno spajaju na PLC te se mogu koristiti za kraće programe ili za manje izmjene programa kada se to mora obaviti u pogonu.

 Za neke jednostavnije procese postoje PLC uređaji koji posjeduju na sebi i displej i par funkcijskih tipki pa se mogu programirati na licu mjesta.Kako bi uspješno proveli programiranje PLC-a koji će potom upravljati procesom moramo na neki način program ispitati. Ispitivanje programa možemo vršiti samo tako da na ulaze u PLC dovedemo stanje veličina iz realnih uvjeta u procesu. Za to se koriste tzv. simulatori stanja PLC-a. Simulator stanja nije ništa drugo nego niz prekidača i kontrolnih lampica koji se zasebno spoje na ulaze i izlaze PLC-a. Kada želimo simulirati rad nekog senzora bitnog za proces npr. protusmrzavajući senzor (preklopi kada je temperatura < 0°C) na mjestu njegovog ulaza u PLC preklopimo prekidač koji će umjesto njega simulirati da je temperatura ispod0°C. Jednako tako će kontrolna lampica svijetliti kada je aktiviran izlaz na koji je ona spojena. Na taj način dobivamo simulaciju rada PLC-a vjernu stvarnim uvjetima u procesu kako bi smo mogli ispitati program.

* 1. STL ( Statement liste )

STL (statement liste) je programski jezik koji omogućava programerima da služeći se jednostavnim naredbama na nivou asemblera opisuju operacije koje treba izvršiti PLC da bi upravljao procesom (kao i kod programiranja u asembleru svaki redak programskog kod-a predstavlja jednu naredbu za procesor). Modularna građa ovog jezika omogućuje nam da rješavamo kompleksne probleme na najučinkovitiji način. STL programski jezik ima određenu hijerarhiju pa nam ga je prema njoj najlakše opisati:

1. **PROGRAM**
2. **KORAK (STEP)**
3. **NAREDBA**
4. **UVJETNI DIO**
5. **IZVRŠNI DIO**

Svaki STL program se sastoji od niza koraka koji se izvršavaju slijedom od prvog prema zadnjem. Riječ korak se može tumačiti na više načina, ali većina STL programa korak tumači kao zasebnu naredbu. Korak ustvari predstavlja logički blok unutar kojega je napisan neki programski kod i kada se izvršavaju naredbe iz tog logičkog bloka kažemo da se izvršava određeni korak. U STL programskom jeziku se može programirati i bez koraka odnosno da cijeli program bude jedan korak, ali je to prilično nezgodno jer se teže kontrolira tok informacije,a i ne može se u potpunosti iskoristiti mogućnost programskog jezika. Svaki korak ima svoju oznaku (label) po kojoj ga se unutar programa može pozvati.

Naredbe su najosnovniji dio programa i zadaju se tako da imaju uvijetni i izvršni dio pa se u informatičkom jeziku takav raspored naziva rečenica (sentence).

Uvijetni (conditional) dio je lista jednog ili više uvijeta čije se stanje provjerava u toku programa (stanje ulaza i izlaza,brojača, timera, itd). Uvjetni dio uvijek počinje riječju IF iza koje slijede uvijeti koji na kraju daju rezultat – visoko ili nisko. Ako su svi uvijeti ispunjeni uvijetni dio šalje logičko stanje visoko u izvršni (executive) dio koji potom izvršava zadane zadatke (npr. upravlja izlaznim stezaljkama PLC-a). Izvršni dio uvijek počinje riječju THEN. Na slici je prikazan STL programski kod koji se može nalaziti unutar jednog koraka.



Na ovaj način možemo napraviti cijeli upravljački program odnosno da su sve rečenice dio jednog koraka, no da bi programiranju pristupili imalo ozbiljnije moramo se koristiti koracima. Programiranje sa koracima ima jedno pravilo sa kojim možemo određivati tok izvođenja programa. Pravilo govori da ako uvijetni dio zadnje rečenice unutar koraka nije ispunjen odnosno ako se izvršni dio zadnje rečenice ne izvršava program ne ide na sljedeći korak nego se vraća na prvu rečenicu unutar istog koraka. To bi značilo iz primjera sa slike da ako bi I:0/2 bio aktivan i O:0/4 bio neaktivan, OR funkcija bi davala logičko stanje nisko i izvršni dio ne bi radio, a program ne bi prešao na sljedeći korak nego bi se vratio na prvu rečenicu istoga odnosno provjeravao stanje I:0/1 itd. Pravilo ćemo najlakše prikazati na blokovskom dijagramu.



Slika 3.1.1. Blokovski dijagram rada PLC-a

Dakle program će se vrtiti u petlji unutar jednog koraka sve dok se uvijet iz zadnje rečenice ne zadovolji.

* 1. Kontaktni dijagrami

 Kontaktni dijagrami (ladder diagrami) nastali su na bazi strujnih upravljačkih shema kojima se prikazuje protok struje u strujnom krugu i koje služe električarima kao podloga za ožičenje istog.

Strujna shema se sastoji od dva horizontalna mrežna vodiča, a struja kroz krug teče odozgo prema dolje. Svaki strujni krug u strujnoj shemi prikazan je kao zaseban strujni put, a svaki strujni put sadrži minimalno jedan upravljani uređaj (npr. motor,relej,žarulja ili slično). Iz strujnog puta se vidi da je rad upravljanog uređaja određen uvjetima(npr. tipkala,pomoćni kontakti i slično) za njegovo uključenje. Kontaktni dijagram PLC programskog jezika vrlo je sličan strujnom putu iz strujne sheme.



Slika 3.2.1. Kontakta shema čvrsto ožičenog start-stop upravljačkog kruga

Kako se na slici vidi strujna shema se sastoji od dva horizontalna mrežna vodiča, a struja kroz krug teče odozgo prema dolje. Svaki strujni krug u strujnoj shemi prikazan je kao zaseban strujni put, a svaki strujni put sadrži minimalno jedan upravljani uređaj (npr. motor,relej,žarulja ili slično). Iz strujnog puta se vidi da je rad upravljanog uređaja određen uvjetima(npr. tipkala,pomoćni kontakti i slično) za njegovo uključenje. Kontaktni dijagram PLC programskog jezika vrlo je sličan strujnom putu iz strujne sheme.



Slika 3.2.2. Kontaktni dijagram PLC programskog jezika

Kod strujne sheme simboli predstavljaju stvarne uređaje (kontakte) i njihovo ožičenje, dok kod kontaktnih dijagrama koji koriste slične simbole oni predstavljaju naredbe u programu. Kontaktni dijagram je dio upravljačkog softvera PLC-a za razliku od strujne sheme koja predstavlja stvarni tok struje u strujnom krugu. Još jedna razlika između kontaktnog dijagrama i strujne sheme je u tome što strujna shema prikazuje stanje kontakata (otvoreno ili zatvoreno) dok se u kontaktnom dijagramu ispituje da li je neka naredba istinita '1' ili neistinita '0' što ne mora imati veze sa stanjem kontakata priključenih na ulazne stezaljke PLC uređaja.

Svaki programski logički put u kontaktnom dijagramu mora imati najmanje jednu izlaznu naredbu, a obično sadrži jedan ili više uvijeta koji moraju biti zadovoljeni da bi se izvršila izlazna naredba. Uvjeti (uvjetne naredbe) su najčešće signali koji dolaze sa uređaja priključenih na ulaz PLC-a u kombinaciji sa statusom izlaza, pomoćnih memorijskih varijabli, vremenskih i brojačkih članova. Na desnoj strani svakog logičkog puta nalazi se izlazna naredba koja se aktivira/deaktivira s obzirom na stanje uvjeta. Izlazne naredbe su npr. 'uključi izlaz' naredba koja uključuje npr. izlazni relej PLC uređaja, interne naredbe PLC-a kao manipulacija bitovima, vremenskim i brojačkim članovima te matematičke naredbe. Na slici dan je primjer veze ozmeđu fizičkih kontakata PLC-a i kontaktnog dijagrama.



Slika 3.2.3. Veza između fizičkih kontakata na PLC-u i kontaktnog djagrama

Program će stalno kontrolirati stanje fizičkog ulaza I/1 i prema tome upravljati izlazom O/1. Ovo je najbanalniji primjer kontaktnog dijagrama ali i kod programiranja se u principu koriste samo dvije osnovne kombinacije i njihovi izvodi.

Osnovne kombinacije naredbi čine 'I' i 'ILI' logičke operacije. Kontaktni dijagram serijski povezanih naredbi odnosno 'I' logičke operacije dan je na slici. Primjer ovakve operacije je pokretanje lifta. Da bi se lift pokrenuo moraju biti zadovoljena dva osnovna uvjeta – vrata od lifta su zatvorena i lift nije preopterećen težinom. Ovdje je izlaz O/1 (output 1 – start motora lifta) u logičkom stanju'1' kada su uvijeti I/1 ulaz(input 1 - senzor zatvorenih vrata) i I/2 ulaz (input 2 – senzor preopterećenja lifta) u logičkom stanju '1'. Kada bilo koji od ta dva uvijeta nije ispunjen gubi se logički kontinuitet i lift se neće pokrenuti.



Slika 3.2.4. Serijski povezane naredbe (logički I)

Kontaktni dijagram paralelno povezanih naredbi odnosno 'ILI' logičke operacije dan je na slici. Primjer ovakve operacije su automatska vrata na robnim kućama. Vrata će se otvoriti kada bilo koji od dva uvijeta bude ispunjen - ili senzor sa vanjske ili unutarnje strane vrata detektira osobu. Ovdje je izlaz O/1 (output 1 – start motora automatskih vrata) u logičkom stanju'1' kada je uvijet I/1 ulaz (input 1 – senzor sa vanjske strane vrata) ili I/2 ulaz (input 2 – senzor sa unutarnje strane vrata) u logičkom stanju '1'. Kada niti jedan od ta dva uvijeta nije ispunjen gubi se logički kontinuitet i vrata neće biti otvorena.



Slika 3.3.5. Logički ILI funkcija

Najčešće se osnovne logičke operacije kombiniraju pa iz njih možemo izvesti kompletnu boleovu algebru. Takav pristup daje nam bezbroj mogućnosti kod samog programiranja. Primjer ovakve kombinacije dan je na slici gdje je prikazana XOR logička funkcija kombinirana sa I logičkom funkcijom.



Slika 3.2.6. XOR logička funkcija

Programi se sastoje od desetak i više logičkih krugova pa treba naglasiti da se program napisan u kontaktnom dijagramu izvodi odozgo prema dolje tj. od prvog logičkog kruga prema zadnjem.

Često se u praksi za pokretanje i zaustavljanje raznih uređaja (motori, ventili itd) koriste tipkala. Da bi PLC nakon pritiska na tipkalo za startanje to zapamtio i prenio na odgovarajući izlaz potrebno je napisati sljedeći logički krug.



Dakle u trenutku kada se stisne tipkalo START, a nije stisnuto tipkalo STOP prema uvijetnom dijelu logičkog kruga uključiti će se motorska sklopka i MOTOR će proraditi. Već u sljedećem prolazu ciklusa rada PLC-a (ms) tipkalo START ne mora biti pritisnuto jer će MOTOR sam sebe držati uključenim (samoodržanje). Prekid rada motora možemo ostvariti pritiskom na tipkalo STOP. Dobro je primjetiti da se izlazno stanje motora (MOTOR) koristi u logičkoj operaciji odlučivanja.

# ZAKLJUČAK

 Smatram da je ovo jedno od najboljih rješenja za automatsko upravljanje složenih automatskih proizvodnih sustava, te da je to je budućnost automatike i industrijske proizvodnje.

 PLC ( programski logički kontrolori ) su pojednostavili način proizvodnje tijekom dosadašnjih godina. Smatram da PLC ima još svojih svojstava koja će se istaknuti u budućnosti te će još više olakšati proizvodnju. Moja jedina zamjerka za PLC je ta što je programiranje pomalo komplicirano te što postoji više načina programiranja a ne samo jedan, ali znam i da to sve ima svoje „ zato „ te da za takve stvari postoje visoko obrazovani ljudi te razni stručnjaci iz tog područja. Ipak svođenjem programiranja PLC-a na statement dijagrame pojednostavljuje se proces programiranja, tj. nije potrebno učiti nove programske jezike pošto su statement liste za različite PLC-ove (proizvođžače) gotovo identične. U bliskoj budućnosti PLC uređaji koristit će se sve više i u kućanstvima za upravljanje uređaja u svrhu smanjenja potrošnje električne energije, tj. uključivanja uređaja po jeftinim tarifama, upravljanje el. vrata, nadzornim sustavima itd.

# LITERATURA

[1] Goran Malčić; Programirljivi logički kontroleri

[2] Velibor Ravlić; Automatika

[3] Ratimir Žanević; Mjerni pretvornici u procesnoj industriji