

# Upravljanje pozicijom uz pomoć apsolutnog davača pozicije rotirajućeg tipa

---

Ivković, Luka

Undergraduate thesis / Završni rad

2019

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

*Permanent link / Trajna poveznica:* <https://um.nsk.hr/um:nbn:hr:200:505284>

*Rights / Prava:* [In copyright](#) / [Zaštićeno autorskim pravom.](#)

*Download date / Datum preuzimanja:* **2024-04-18**

*Repository / Repozitorij:*

[Faculty of Electrical Engineering, Computer Science and Information Technology Osijek](#)



**SVEUČILIŠTE JOSIPA JURJA STROSSMAYERA U OSIJEKU**

**FAKULTET ELEKTROTEHNIKE, RAČUNARSTVA I INFORMACIJSKIH TEHNOLOGIJA**

**OSIJEK**

**Stručni studij**

**UPRAVLJANJE POZICIJOM UZ POMOĆ  
APSOLUTNOG DAVAČA POZICIJE ROTIRAJUĆEG  
TIPA**

**Završni rad**

**Luka Ivković**

**Osijek, 2019.**

## Sadržaj

|                                                            |    |
|------------------------------------------------------------|----|
| 1. UVOD .....                                              | 1  |
| 2. UPRAVLJANJE POZICIJOM.....                              | 2  |
| 2.1. Aktuatori i senzori .....                             | 2  |
| 2.2. Programabilni logički kontroler (PLC).....            | 2  |
| 3. DAVAČI POZICIJE .....                                   | 6  |
| 3.1 Linearni davači pozicije.....                          | 6  |
| 3.2 Davači pozicije rotirajućeg tipa .....                 | 7  |
| 3.2.1 Inkrementalni davači pozicije rotirajućeg tipa ..... | 7  |
| 3.2.2 Apsolutni davači pozicije rotirajućeg tipa .....     | 8  |
| 4. SPAJANJE DAVAČA POZICIJE .....                          | 11 |
| 4.1 Spajanje „Open-collector“ izlaza.....                  | 12 |
| 5. PROGRAMSKO RJEŠENJE .....                               | 14 |
| 5.1 Step 7 .....                                           | 15 |
| 5.2 Pretvorba Gray koda u binarni zapis .....              | 16 |
| 5.3 Kalibracija .....                                      | 17 |
| 5.4 Pokretanje i zaustavljanje teretnih kolica .....       | 19 |
| 5.5 Pozicioniranje .....                                   | 27 |
| 6. ZAKLJUČAK .....                                         | 31 |
| LITERATURA.....                                            | 32 |
| SAŽETAK.....                                               | 33 |
| ABSTRACT .....                                             | 34 |
| ŽIVOTOPIS .....                                            | 35 |
| PRILOG .....                                               | 36 |

## 1. UVOD

U današnje vrijeme, temeljna disciplina i glavni faktor u tehnološkom razvoju i industrijalizaciji je automatsko upravljanje. Načela automatskog upravljanja primjenjuju se u mnogim tehničkim i netehničkim dinamičkim sustavima te bez njih ne bi obavljali željenu zadaću na odgovarajući način. Upravo zbog široke primjene načela i svakidašnje potrebe za automatskim upravljanjem može se reći da smo naveliko zakoračili u doba automatizacije.

Kao što je već navedeno, automatsko upravljanje primjenjuje se u različitim dinamičkim sustavima. Dinamički sustav predstavlja funkcijsku cjelinu za obradu i prijenos energije, materije, informacije i kapitala, gdje se ulazne veličine sustava promatraju uzrokom, a izlazne veličine sustava njegovom vremenskom posljedicom [1].

Sustav upravljanja pozicijom je jedan primjer koji pripada u dinamički sustav. U ovom radu opisat će se sustav upravljanja pozicijom koji je univerzalan i moguće ga je primijeniti na mnoge tehničke sustave.

## **2. UPRAVLJANJE POZICIJOM**

Upravljanje pozicijom je proces koji ima veliku važnost u automatizaciji industrije. Često automatizirani industrijski pogon ima potrebu za upravljanjem pozicije, počevši od onih jednostavnijih poput pokretne trake pa sve do onih najzahtjevnijih kao što su robotske ruke.

Upravljanje pozicijom je opisano i primijenjeno na sustavu koji ima široku primjenu u industriji, sustav koji može biti prikazan kao industrijska teretna kolica (vagon), pokretna traka, sustavi za zavarivanje ili rezanje, preslagivanje robe, te bilo koji drugi sustav koji ima mogućnost pozicioniranja. U svim navedenim sustavima neizostavna je primjena upravljanja pozicijom. Upravljanje pozicije obuhvaća nekoliko važnih čimbenika koje je potrebno uzeti u obzir prilikom realizacije tog sustava. Prije svega potrebno je znati s kakvim sustavom se upravlja, ima li taj sustav stalnu masu ili je masa promjenjiva kao u slučaju s teretnim kolicima. Prilikom pokretanja sustava koji se upravlja važno je uzeti u obzir ubrzanje i usporavanje zbog inercije sustava koja može dovesti do pogreške prilikom pozicioniranja.

Osnovni dijelovi sustava pomoću kojeg se upravlja pozicijom su aktuator, regulator i mjerni član.

### **2.1. Aktuatori i senzori**

U sustavima gdje se primjenjuje upravljanje pozicijom glavni član takvog sustava je aktuator. Aktuator je izvršni član u sustavima upravljanja koji ima zadatak pretvoriti upravljački signal u određenu radnju kao što je mehanički pomak ili uključivanje/isključivanje nekog elementa.

Senzor je uređaj koji detektira događaj ili promjenu u sustavu te mjerenu veličinu pretvara u odgovarajući signal. Senzori za detektiranje pozicije objekta mogu raditi na različitim principima koristeći različite fizikalne pojave za mjerenje pozicije kao što su elektromagnetizam, jačina svjetla, zvuka itd. Njihova je upotreba specifična za različita okruženja i primjene [2].

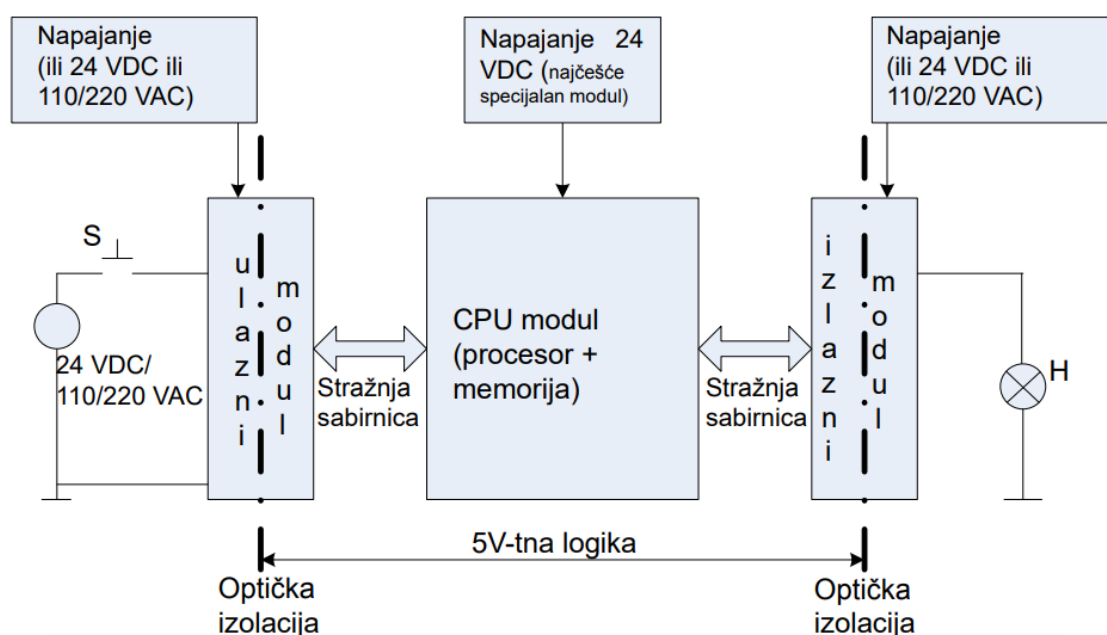
### **2.2. Programabilni logički kontroler (PLC)**

Svakim automatiziranim sustavom upravlja regulator koji šalje upravljačke signale i tako uzrokuje promjenu u sustavu te njegovu regulaciju. Logika koja se nekoć upotrebljavala za izvedbu takvih sustava za upravljanje bila je relejna logika. Relejna logika se izvodi tako da se žicama spajaju komponente i releji pomoću kojih se izvršava zadana logika. Najveći nedostatak relejne logike je veličina upravljačkog djela, mehanička i električna potrošnost komponenata, potrošnja energija te zahtjevnost promjene u programskoj logici. Nastojeći riješiti takve probleme javila se potreba za

uvođenjem računala u upravljačke sustave kao što je programabilni logički kontroler (PLC) kod kojega se pogreške u logici rješavaju u programskom kodu [2].

Arhitektura programabilnog logičkog kontrolera (Slika 2.1) je standardnog oblika koji se pronalazi i kod digitalnog računala. Glavni dijelovi PLC-a su:

- Procesor (CPU)
- Memorija
- Ulazno izlazne jedinice (I/O)
- Sabirnica.



Slika 2.1 Arhitektura PLC-a

Glavni dio programabilnog logičkog kontrolera je CPU (engl. *Central Processing Unit*) koji je vođen programskim naredbama pomoću kojih izvodi osnovne radnje nad podatcima. Uz CPU je smještena memorija koja najčešće dolazi u dvije izvedbe, RAM (engl. *Random Access Memory*) i EPROM (engl. *Erasable Programmable Read-Only Memory*). RAM memorija je primarni oblik računalne memorije u koju je moguće upisivati podatke te čitati podatke koji se nalaze unutar memorije. EPROM je vrsta memorije čija je glavna osobina trajna pohrana podataka, čak i u slučaju gubitka napajanja. Podaci koji su trajno spremljeni mogu se samo čitati sve dok se ne izbrišu i upišu novi.

Sustav PLC-a se sastoji od više modula koji su smješteni na sabirnicu (engl. *Rack*) i međusobno su povezani. Prvi modul je namijenjen za napajanje, koji pretvara mrežni napon u istosmjerni izvor napajanja potreban za rad ostalih modula. Na mjestu drugog modula se nalazi CPU, dok su ostali moduli memorijski, komunikacijski moduli za digitalne ulaze i izlaze, moduli za analogne ulaze i izlaze, itd. Ovaj oblik konstrukcije sustava omogućuje lako proširenje dodavanjem potrebnog modula. U složenijim sustavima moguće je spojiti vanjske module s dodatnim ulazno /izlaznim signalima koji su međusobno povezani u mrežu. Ulazno/izlazne jedinice PLC-a mogu biti digitalne i analogne, one ovise o tipu kartice, odnosno modula koji je smješten na sabirnici.

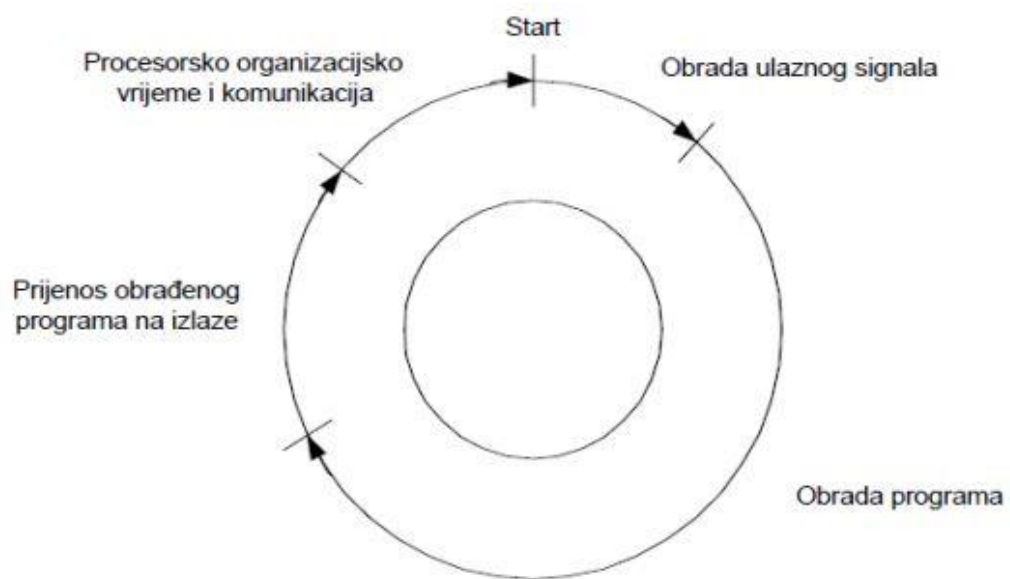
Izlazni analogni moduli mogu biti strujni i naponski. Strujni moduli su češće u uporabi kako bi se umanjio efekt pada napona na priključnim vodovima uslijed veće udaljenosti izvršnog člana. Kako za analogne izlaze tako i za analogne ulaze postoje različiti standardi. Najčešći iznos strujnog analognog ulaza je u rasponu od 4 do 20 mA. Iznos od 4 mA ima zaštitnu ulogu koja omogućava detekciju prekida žice. Naponski analogni moduli, čiji raspon vrijednosti je od 0 do 5 V, od 0 do 10 V ili  $\pm 10$  V također se koriste kako kao izlazni signali tako kao i upravljački signali. Ulazni analogni signal potrebno je diskretizirati kako bi ga PLC mogao obraditi. Diskretiziranje se postiže pomoću A/D i D/A pretvornika tj. ulazna kartica ima ulogu pretvorbe te pretvaranja mjerenje veličine u digitalni zapis.

Digitalni moduli se najčešće upotrebljavaju za spajanje diskretnih senzora kao što su tipkala, prekidača, induktivnih pretvarača itd..

Logičke jedinice i nule se dobivaju pomoću različitog iznosa napona, a iznos napona ovisi o tehnologiji koju modul koristi. Tehnologije koje se upotrebljavaju su HTL (engl. *High Threshold Logic, Push-Pull*) i TTL (engl. *Transistor–transistor logic*). Iznos napona kod HTL tehnologije za logičku jedinicu je od 13 V do 24 V, dok je kod TTL tehnologije od 2.4 do 5 V.

Princip rada PLC-a se temelji na ciklički kontinuiranom prikupljanju i obradi podataka u cirkularnoj petlji. PLC mijenja stanje na izlazu s obzirom na promjenu stanja na ulazu. Način na koji će mijenjati izlaz zadano je u korisničkom programu.

Na početku ciklusa rada PLC sprema u memoriju procesora podatke koje prikupi na ulazima, obrađuje ih te na osnovi korisničkog programa, obrađuje signale nakon čega postavlja izlazne signale. Ciklus rada PLC-a je vidljiv na slici 2.2.



*Slika 2.2 Ciklus rada PLC-a*

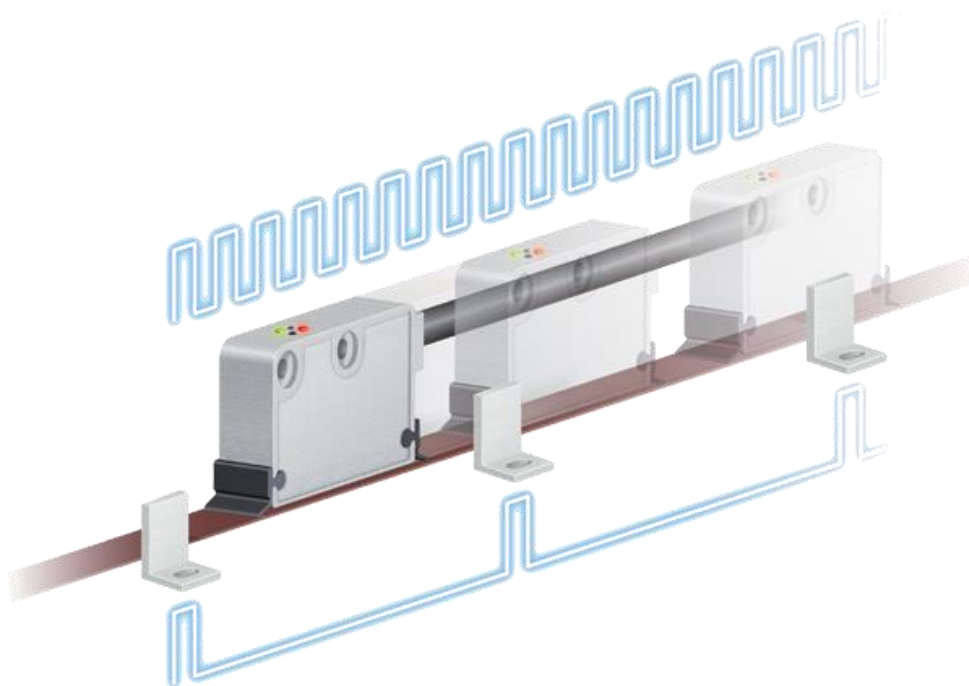


### 3. DAVAČI POZICIJE

Davači pozicije su mjerni pretvarači koji mehaničko gibanje (poziciju) pretvaraju u digitalni oblik. Glavna podjela davača pozicije je na apsolutne i inkrementalne te po izvedbi se dijele na linearne i rotacijske.

#### 3.1 Linearni davači pozicije

Linearni davači pozicije su elektromehanički uređaji koji pretvaraju pravocrtno gibanje u analogni ili digitalni signal. Informacija o položaju i brzini dobiva se tako što senzor očitava ljestvicu te se očitana vrijednost dekodira i prikazuje pomoću digitalnog čitača. Brzina pomicanja senzora je proporcionalna frekvenciji izlaznog signala. Kodiranje položaja može se postići upotrebom raznih tehnologija. Tehnologije koje se primjenjuju su magnetska, optička, induktivna, kapacitivna i tehnologija bazirana na vrtložnim strujama. Izgled linearnog davača pozicije je prikazan na slici 3.1.



*Slika 3.1 Linearni davač pozicije [9]*

Linearni davači pozicije imaju visoku pouzdanost u zahtjevnim uvjetima rada iz razloga što prašina, ulje i krhotine ne utječu na njegov rad. Najčešće se ugrađuju u strojeve za obradu drveta, lijevanje plastike, pisalice, razne hidrauličke strojeve, vage te utovarivače.

### 3.2 Davači pozicije rotirajućeg tipa

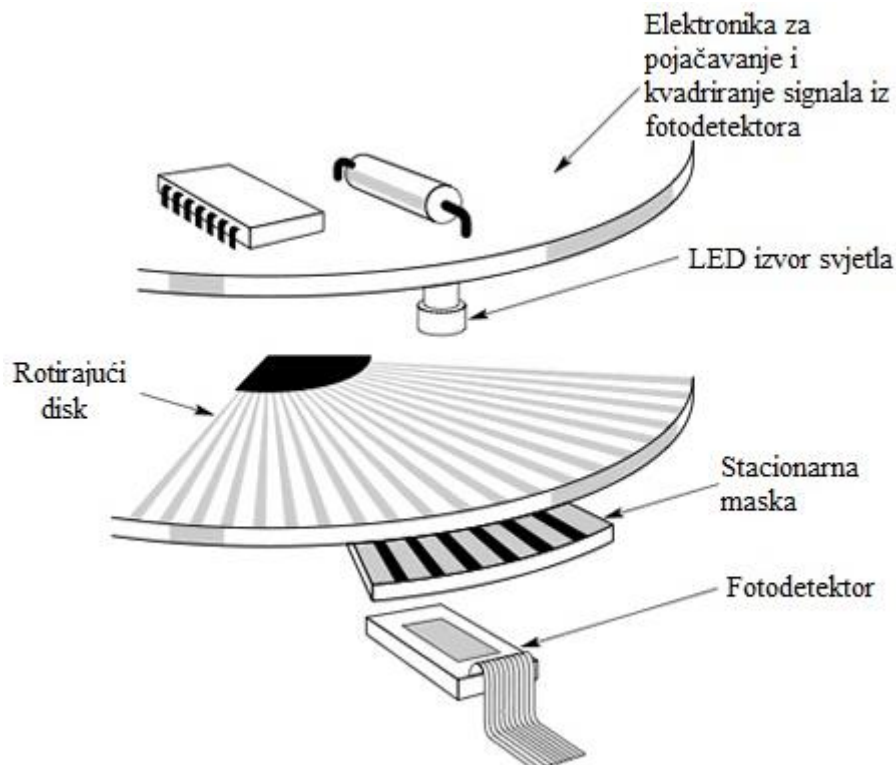
Davač pozicije rotirajućeg tipa pretvara kružno gibanje u analogni ili digitalni signal. Najčešće se postavlja na osovinu gibajućeg uređaja, te slično kao kod linearnih, mjeri kutni pomak na osovini.

#### 3.2.1 Inkrementalni davači pozicije rotirajućeg tipa

Inkrementalni davači pozicije postavljaju se u sustavima sa zatvorenom petljom gdje generiraju podatke potrebne za određivanje smjera vrtnje kao i brzinu vrtnje. Inkrementalni davač pozicije pretvara kutno gibanje u digitalni signal koji se lako povezuje s programabilnim logičkim kontrolerom ili računalom.

Osnovni dijelovi inkrementalnog davača pozicije rotirajućeg tipa su prikazani na slici 3.2 i to su:

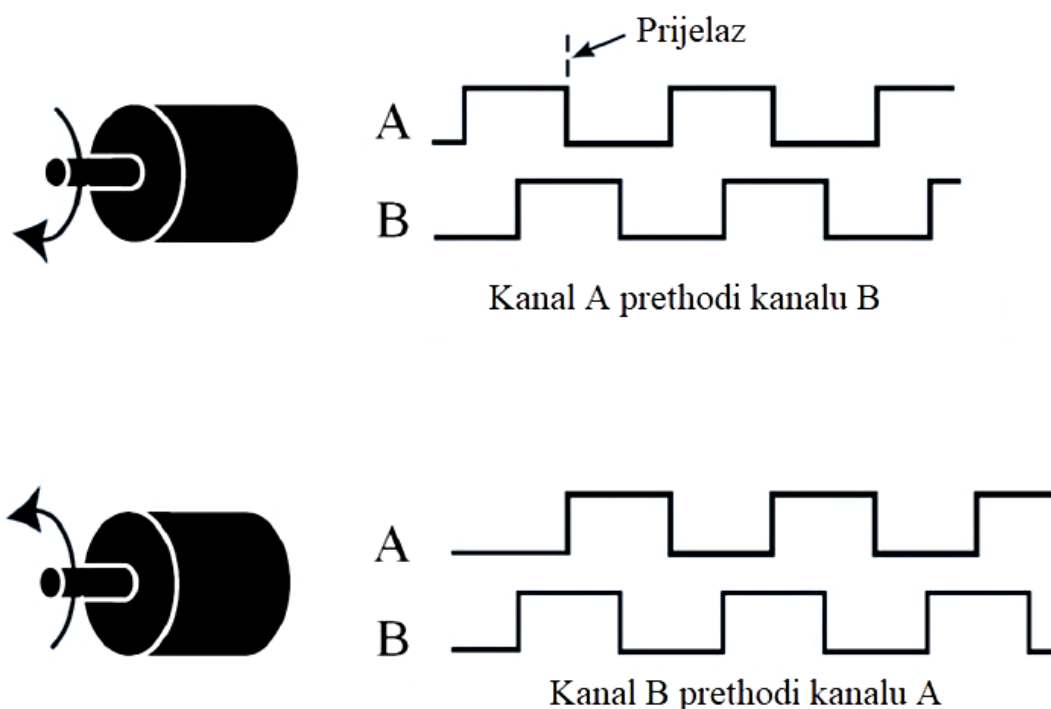
1. LED izvor svjetla
2. Rotirajući disk
3. Stacionarna maska
4. Fotodetektor
5. Elektronika za pojačavanje i kvadriranje signala iz fotodetektora.



*Slika 3.2 Dijelovi inkrementalnog davača pozicije rotirajućeg tipa [7]*

Vrtnjom diska ispred stacionarne maske zatvara se protok svjetla iz LED izvora svjetla. Fotodetektor prima svjetlosnu energiju i proizvodi impulse na izlazu koji se pretvaraju u potrebni digitalni signal.

Jedan kanal inkrementalnog davača pozicije je dovoljan za određivanje brzine vrtnje, drugi kanal je potreban za određivanje smjera vrtnje. Kod nekih izvedbi postoji i treći kanal (svaki puni krug daje jedan puls) koji je moguće koristiti za potrebe pozicioniranja.



*Slika 3.3 Smjer vrtnje inkrementalnog davača pozicije rotirajućeg tipa sa dva kanala[8]*

Smjer vrtnje inkrementalnog davača pozicije određuje se pomoću dva kanala izlaznog signala koji su pomaknuti  $90^\circ$  jedan u odnosu na drugi. Na kojem kanalu se prvom pojavi rastući brid, u tome smjeru se davač pozicije okreće, što je vidljivo na slici 3.3.

Dodavanjem komplementiranih kanala smanjuju se utjecaj vanjskih smetnji. Ukoliko PLC ostane bez napajanja on gubi vrijednost pozicije te kao takav senzor se ne može koristiti za očitavanje apsolutne pozicije.

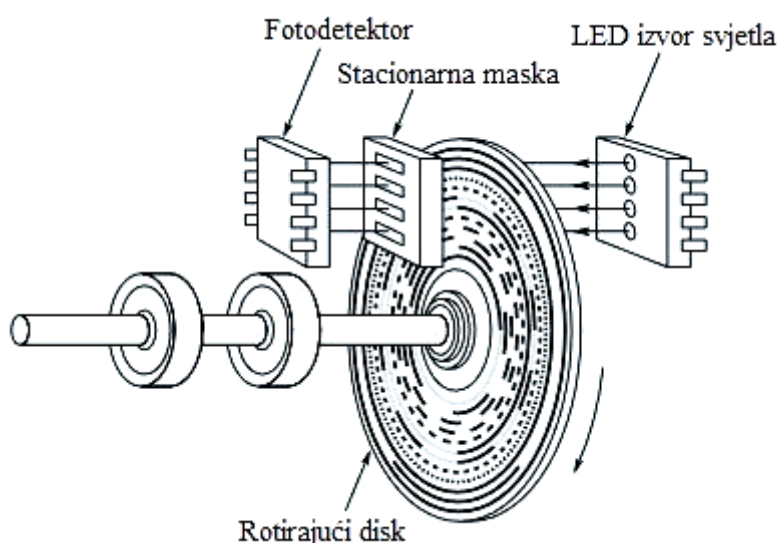
### **3.2.2 Apsolutni davači pozicije rotirajućeg tipa**

Apsolutni davači pozicije su po principu rada slični inkrementalnim davačima pozicije. Rotirajući disk prekida protok svjetlosti do fotodetektora i na taj način formira izlazni signal. Svaka pozicija kod apsolutnih davača pozicije, za razliku od pozicije kod inkrementalnih davača je jedinstvena.

Pozicija se ne određuje brojanjem impulsa već čitanjem određenih kodova koji su jedinstveni za svaki položaj. Velika prednost apsolutnih davača je ta što nakon prekida napajanja pozicija ostaje ista, odnosno kôd koji apsolutni davač daje na izlazu se ne gubi ukoliko ostane bez napajanja.

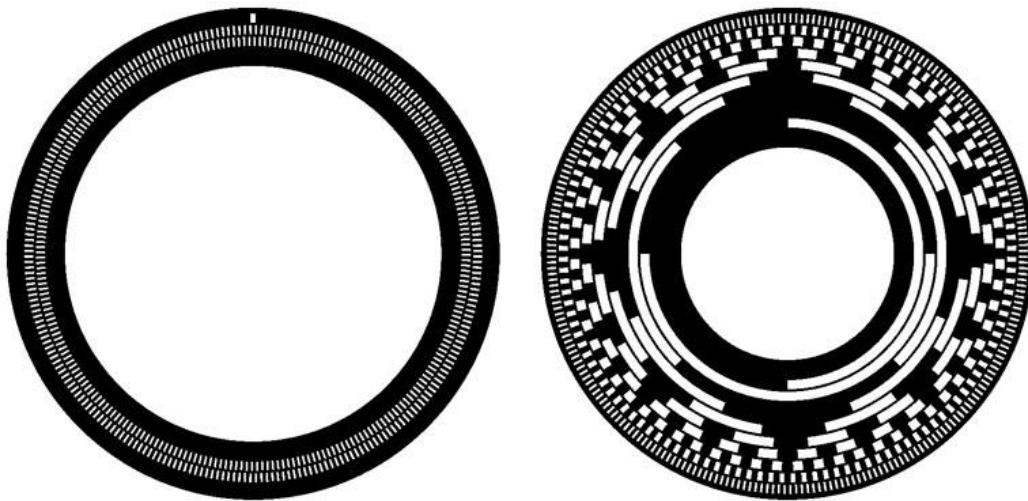
Glavni dijelovi apsolutnog davača pozicije rotirajućeg tipa su:

1. LED izvor svjetlosti
2. Rotirajući disk
3. Stacionarna maska
4. Fotodetektor.



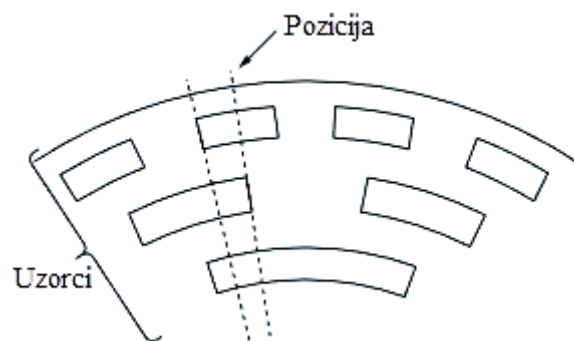
*Slika 3.4 Dijelovi apsolutnog davača pozicije rotirajućeg tipa [9]*

Za razliku od jednostavnog jednotračnog (engl. „*Single track*“) diska kod inkrementalnog davača, apsolutni davač sadrži rotirajući disk od nekoliko koncentričnih tragova. Razlika između diskova prikazana je na slici 3.5. Tragovi koji se nalaze na disku su načinjeni od prozirnih i neprozirnih uzoraka koji daju jedinstvenu vrijednost za svaki položaj. Razlika između diskova inkrementalnog i apsolutnog davača pozicije vidljiva je na slici 3.5. Fotodetektor prima svjetlost koja prođe kroz rotirajući disk i kroz stacionarnu masku. Signal koji šalje fotodetektor je kodiranog oblika, najčešće u formatu oblika „Gray“. Rezolucija, odnosno razlučivost apsolutnog davača pozicije je vrijednost koju davač pozicije daje jednim okretom, te je ujedno to i preciznost koju davač pozicije ima.



*Slika 3.5 Disk inkrementalnog i apsolutnog davača pozicije*

Uzorci na disku apsolutnog davača pozicije su raspodijeljeni u redove, odnosno prstenove gdje svaki prsten predstavlja jedan bit. Prozirni dio uzorka koji je prikazan na slici 3.6 propušta svjetlost i na taj način postavlja bit u logičku jedinicu, dok neprozirni dio ne propušta svjetlost i on postavlja bit u logičku nulu. Skup svih bitova daje izlaznu „riječ“ (engl. *Word*). Apsolutni davač upotrebljen za realizaciju ovog završnog rada je razlučivosti 256, odnosno izlazna riječ je veličine 8 bitova, što bi značilo da ima 8 prstenova na disku. Da bi omogućilo čitanje diska, za svaki prsten se postavlja fotodetektor koji emitira logičku nulu ili logičku jedinicu.



*Slika 3.6 Čitanje pozicije [9]*

Ukoliko se pojavi potreba za proširenjem opsega čitanja senzora, moguće je upotrebljavati „Multi-turn“ davač pozicije. „Multi-turn“ apsolutni davač pozicije se sastoji od više diskova, najčešće dva, što omogućuje brojanje pozicije u rasponu od više krugova. Kada jedan disk napravi puni krug, tada drugi disk napravi pomak za jedan korak razlučivosti. Rad „Multi-turn“ davača pozicije najjednostavnije usporediti sa satom, gdje kazaljke zamjenjuju diskove.

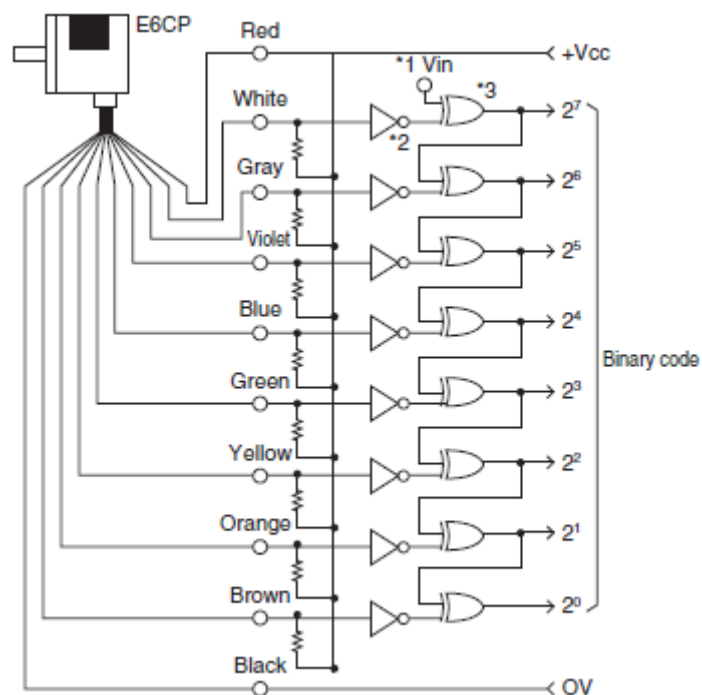
#### 4. SPAJANJE DAVAČA POZICIJE

Da bi se postigao ispravan rad apsolutnog davača pozicije rotirajućeg tipa, potrebno je obratiti pozornost na nekoliko stavki koje mogu znatno utjecati na njegov rad. Upotrebljeni davač pozicije je marke Omron koji je vidljiv na slici 4.1, te njegova rezolucija iznosi 256, odnosno 8 bita. Davač pozicije ima 8 izlaza koji su fizički vidljivi kao 8 vodiča različitih boja. Svaki vodič, odnosno izlaz, predstavlja jedan bit, stoga je potrebno obratiti pažnju na redoslijed spajanja signala na karticu digitalnih ulaza programabilnog logičkog kontrolera.



*Slika 4.1 Omron apsolutni davač pozicije rotirajućeg tipa [10]*

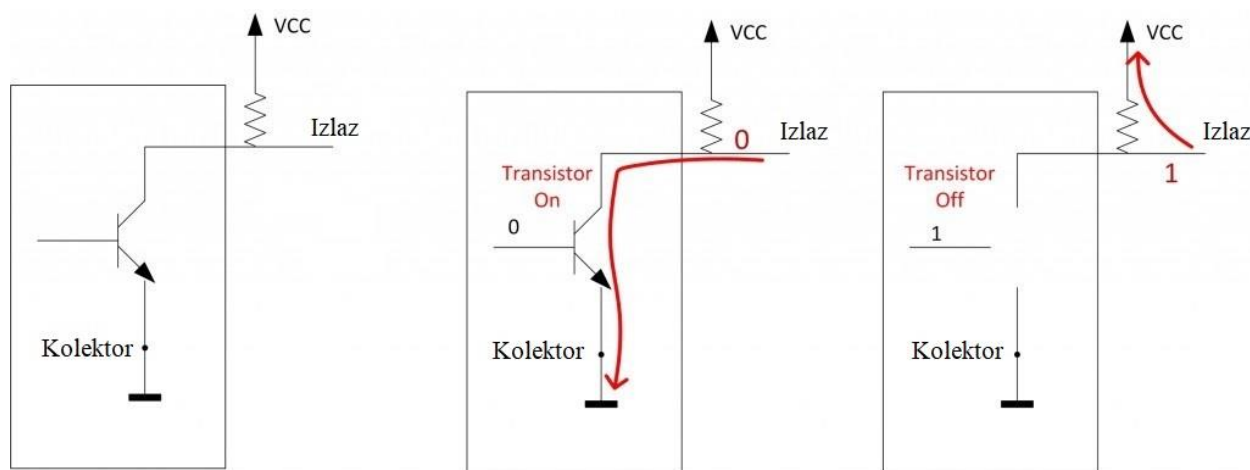
Također je potrebno obratiti pozornost na vrijednosti napona i struje prilikom spajanja davača pozicije na PLC zato što je izlaz upotrebljenog davača pozicije tzv. „Open-collector“. Takav izlazni krug nije u potpunosti sukladan s digitalnim ulazima na postojećoj ulaznoj kartici te ga je potrebno prilagoditi. Potrebno je od pasivnog signala stvoriti aktivni signal (24V) kako bi ga ulazna kartica uspjela očitati. Stoga je upotrebljen sklop koji će povećati napon i struju do vrijednosti koju PLC može čitati. Vrijednost koja se dobije nakon čitanja svih 8 izlaza je u obliku Gray-ovog koda, koji je potrebno pretvoriti u binarni kako bi se vrijednost mogla čitati i u obliku prirodnog broja. Pretvorba Gray-ovog koda u ovom radu se obavila programski i opisana je u poglavlju 5. Pretvorbu Gray koda moguće je učiniti i sklopovski upotrebom određenih logičkih sklopova gdje se svakom od izlaza apsolutnog davača pozicije dodaju sklopovi NE i XILI na način prikazan na slici 4.2. Na slici je također prikazan način spajanja davača pozicije s „Open-Collector“ izlazima.



Slika 4.2. Spajanje davača pozicije s pretvorbom Gray koda

#### 4.1 Spajanje „Open-collector“ izlaza

„Open-collector“ izlaz se ponaša kao sklopka koja je isključena ili je priključena na masu. Kolektor tranzistora je otvoren i doveden na pin integriranog sklopa, dok je emiter uzemljen. Izlazni oblik ovakvog sklopa čini otvoreni krug ili spajanje s masom. Elektronička shema „Open-collector“ izlaza je prikazana na slici 4.3.



Slika 4.3. Shema "Open-collector" izlaza

Budući da je kartica koja se upotrebljavala za digitalne ulaze na koje se spajao davač pozicije pasivna, odnosno potreban joj je signal napona od 24 V, provedena je prilagodba. Korišten je sistem „Pull up“ otpornika prikazano kao na slici 4.3. Kada tranzistor provede između kolektora i

emitera, iznos napona spojenog na ulaznu karticu pada blizu 0 V. Kad tranzistor ne vodi onda se napon prebacuje na izlaz kartice, umanjen za pad napona na otporniku. Iznos tog napona ovisi o otporu kartice te iznosu upotrebljenog “pull up” otpornika. Prilikom proračuna otpora treba pripaziti na maksimalne dozvoljene vrijednosti napona i struje koje se smiju dovesti na izlaz integriranog kruga, kako ne bi došlo do trajnog oštećenja integriranog kruga.



## 5. PROGRAMSKO RJEŠENJE

Zadatak završnog rada bio je realizirati sustav upravljanja pozicijom pomoću apsolutnog davača pozicije rotirajućeg tipa u programskom paketu Step7 te ga testirati na već postojećoj maketi. Maketa je predstavljala pokretna industrijska teretna kolica koja se gibaju između dvije pozicije. Za određivanje pozicije upotrebljen je apsolutni davač pomoću čijeg se izlaznog signala određuje položaj zaustavljanja kolica. Izlazni signal davača pozicije je bilo potrebno pretvoriti u binarni zapis, što je opisano u potpoglavlju 5.2. Dobivenu poziciju davača je moguće preskalirati na željeno područje rada. Istovremeno se može obaviti i rekalkibracija pozicije preko induktivnih davača postavljenih na transportnu liniju. Korekcija pozicije (rekalkibracija) je postupak koji se primjenjuje u sustavima gdje može doći do odstupanja mjerenja od stvarne pozicije uslijed proklizavanja prijenosa između davača i uređaja koji se mjeri.

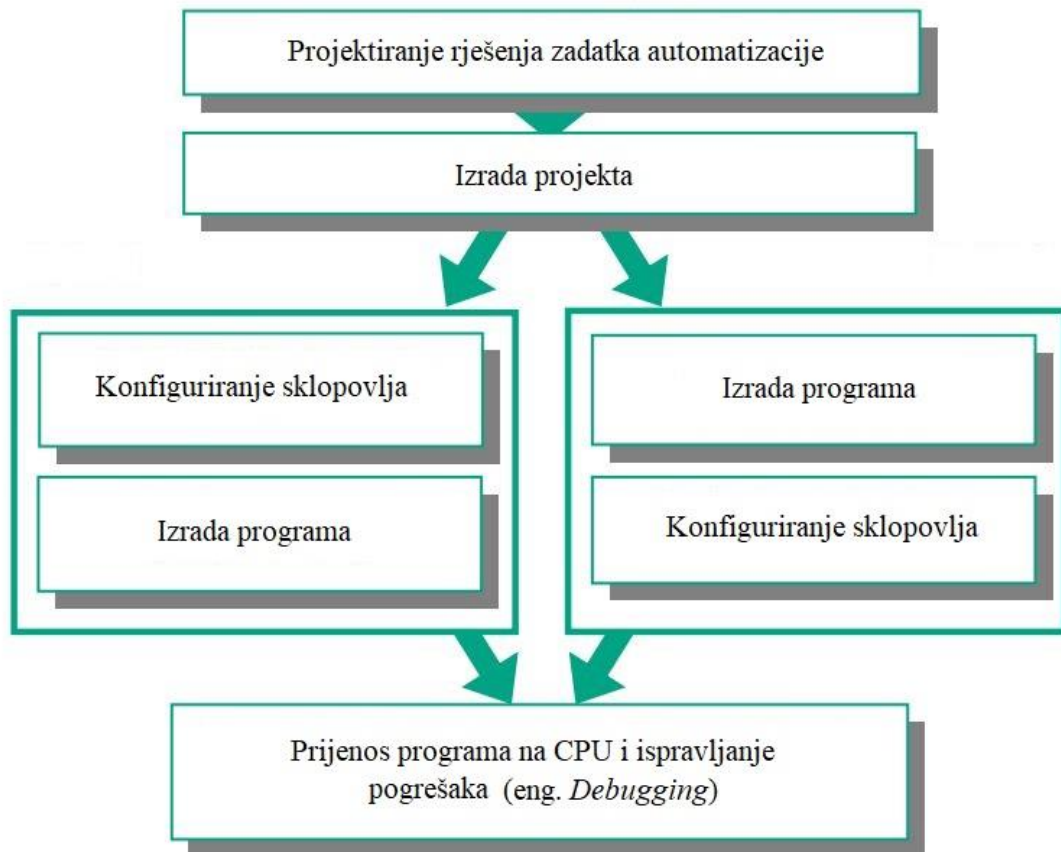
Pozicije su nazvane „work“ i „home“ te su zadane programski gdje je omogućeno upravljati njima putem kontrolne ploče. Kontrolna ploča je u ovom radu zamijenjena virtualnom, odnosno VAT tablicom gdje je moguće mijenjati stanje ulaznih bitova koji predstavljaju tipkala na kontrolnoj ploči. Prilikom realizacije sustava bilo je potrebno obratiti pozornost na koji će se način kolica početi kretati te zaustaviti na određenu poziciju.

Pomak kolica do željene pozicije moguće je u dva režima rada. Prvi režim rada je ručni (engl. „*Manual*“) koji omogućava da se kolica gibaju samo u slučaju dok je tipkalo na kontrolnoj ploči pritisnuto, te se puštanjem tipkala teretna kolica zaustavljaju. Drugi režim rada je automatski (engl. „*Automatic*“) u kojem se pritiskom tipkala kolica gibaju do odredišta bez da tipkalo mora biti pritisnuto. Odabir režima rada moguć je pomoću tipkala za odabir režima rada.

Naredbom za automatsko pozicioniranje, teretna kolica se prema toj poziciji počinju gibati maksimalnom postavljanom brzinom. U trenutku kada se kolica približe postavljenoj poziciji, tada kreće gibanje sa sporijom brzinom postavljenom od strane korisnika. Sporom brzinom se kolica gibaju do određene granice nakon koje nastupa zaustavljanje na prethodno određenu poziciju. Početna brzina kretanja i brzina približavanja su određene u programu, kao i trenutak kada nastupa sporija brzina i granica nakon koje se kolica počinju zaustavljati. Prilikom gibanja kolica onemogućena je promjena pozicije sve dok kolica ne stignu na prethodno zadanu poziciju i ne zaustave se. Zaustavljanje kolica u bilo kojem trenutku moguće je pritiskom na tipku „*Emergency Stop*“.

## 5.1 Step 7

Step7 je programski paket kojim se omogućuje programiranje PLC-ova. Program je osmišljen od strane tvrtke Siemens i pripada porodici Simatic S7 koji se temelji na IEC 61131-3 standardu. Sa Step 7 programom se može konfigurirati sklopovlje (engl. *Hardware*) PLC-a, postaviti adrese i parametre modula te konfigurirati mrežne postavke. Podaci su organizirani u obliku projekta čiji redoslijed stvaranja je prikazan dijagramom na slici 5.1.



Slika 5.1 Redoslijed stvaranja Step 7 projekta

Step 7 omogućuje programiranje u nekoliko vrsta programskih jezika, a osnovni su:

- LAD (engl. *Ladder Logic*)
- FBD (engl. *Function Block Diagram*)
- STL (engl. *Statement List*).

STL programski jezik se temelji na naredbama kojima se opisuju operacije koje PLC izvršava. Svaki red programskog koda predstavlja jednu naredbu.

Za razliku od STL jezika, FBD je grafički oblik programiranja gdje su naredbe prikazane funkcijskim blokovima čijim povezivanjem i konfiguriranjem se opisuju operacije koje PLC izvršava.

LAD programski jezik je najpopularniji jezik programiranja u programskom paketu Step 7 te je najviše u upotrebi. Upravo ovaj programski jezik će se upotrebljavati za realizaciju programa upravljanja pozicijom u ovom radu. Ladder Logic predstavlja prilagodbu relejnih upravljačkih krugova pri čemu se elementi povezuju kao kod električnih shema. Bitna stavka kod LAD programskog jezika je ta da se program izvršava u dva smjera, od gore prema dolje te s lijeva na desno.

Neki od dodatnih programskih jezika su SCL (engl. *Structured Control Language*) visoki programski jezik (sličan „Pascal“ programskom jeziku), S7-GRAPH, S7-HiGraph i CFC koji je sličan FBD programiranju.

## 5.2 Pretvorba Gray koda u binarni zapis

Kako bi se mogao razumljivo čitati izlazni signal bilo ga je potrebno pretvoriti u binarni zapis koji se nakon toga pretvarao u cjelobrojni zapis odnosno „Integer (INT)“ tip podatka. Iz tog razloga pretvorba Gray koda je prvi korak u programskom rješenju.

Gray-ov kôd je još također poznat pod izrazima „*reflected binary code*“ (RBC) i „*reflected binary*“ (RB). Ovakav kôd predstavlja redoslijed binarnog brojevnog sustava na način da se dvije uzastopne vrijednosti razlikuju samo u jednoj binarnoj znamenki (jednom bitu). Gray kôd je dizajniran da spriječi lažne izlaze iz elektromehaničkih prekidača. Široku primjenu ima za olakšavanje ispravaka greški u digitalnoj komunikaciji. Gray kôd se upotrebljava u linearnim i rotirajućim davačima položaja. Glavna svrha mu je da se izbjegne pogrešno čitanje bitova prilikom mijenjanja više bitova odjednom. Upotrebom Gray koda na izlazu davača pozicije smanjuje se pogreška ukoliko dođe do kvara na jednom od izlaza davača pozicije, odnosno ukoliko se jedan bit izgubi.

Jedan od načina pretvorbe Gray koda pomoću logičkih sklopova je taj da se na svaki izlaz apsolutnog davača pozicije spoji ekskluzivni ILI (XILI) sklop. Prije spajanja ekskluzivnog ILI sklopa potrebno je negirati svaki od izlaza. Način pretvorbe Gray koda putem LAD programskog jezika je prikazan u prilogu 1.

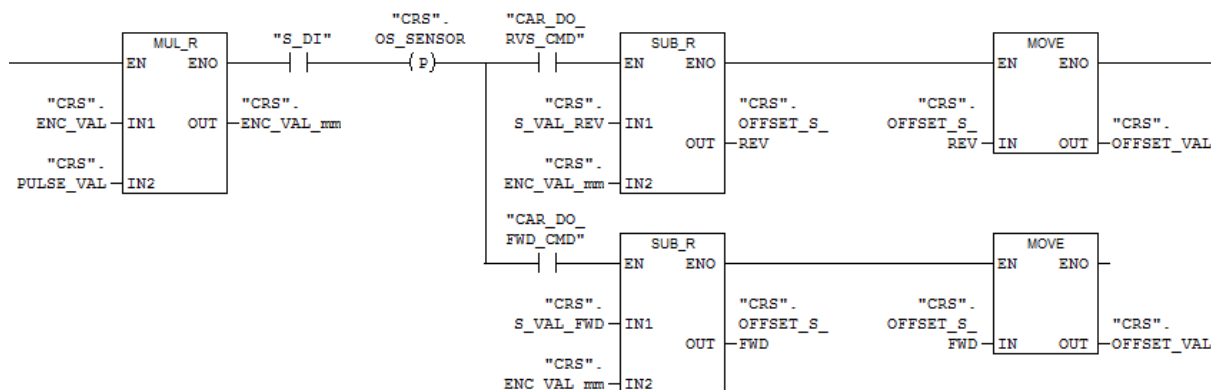
Ekskluzivni ILI se postiže određenom kombinacijom NO (engl. *Normally Open*) i NC (engl. *Normally Closed*) kontakata koji su „bool“ tipa podataka. Kontaktima se pridružuju vrijednosti dobivene negacijom izlaznih signala davača pozicije. Negacija se postigla dodavanjem NE logičkog sklopa koji je u LAD programskom jeziku realiziran na način prikazan u prilogu 2.

Grupiranjem pretvorenih bitova dobiva se cjelobrojni tip podataka (INT). Nakon prikazane pretvorbe Gray koda, signal koji davač pozicije daje na izlazu prikazan je kao INT tip podataka koji je bilo potrebno pretvoriti u REAL tip podatka kako bi se omogućila njegova lakša upotreba dalje u programu. Pretvorba INT u REAL je izvršena upotrebom sustavnog funkcijskog bloka namijenjenom za tu pretvorbu. Sustavni funkcijski blokovi su osmišljeni kako bi olakšali i ubrzali programiranje a samim time i čitanje programskog koda u bilo kojem od programskih jezika. Takvi blokovi se nalaze u bibliotekama koje program sadrži te ih je moguće uvesti iz drugih projekata.

### 5.3 Kalibracija

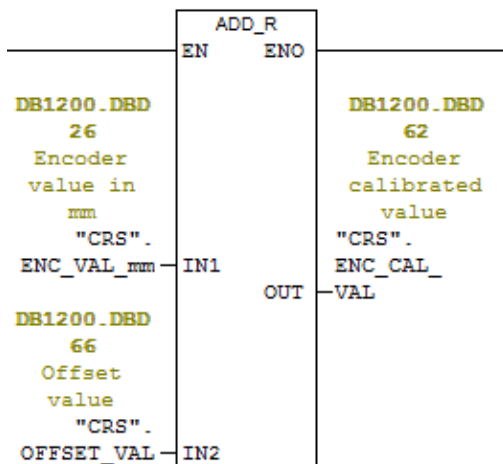
Kalibracija se provodi kako bi se impulsi koje daje davač pozicije na izlazu pretvorili u stvarnu vrijednost. Vrijednost u programu nazvana „CRS.ENC\_VAL“ predstavlja vrijednost davača pozicije na izlazu i ona može biti od 0 do 256, kolika je i rezolucija upotrebljenog apsolutnog davača pozicije. Ta vrijednost se naziva inženjerska jedinica iz razloga što je univerzalna te nema mjernu jedinicu. U industrijskim postrojenjima se najčešće prije postavljanja davača pozicije dodaju razni prijenosnici kako bi se povećao raspon vrijednosti jednog kruga davača pozicije rotirajućeg tipa. U ovom slučaju se nije dodavao dodatni prijenos, već se kalibracija realizirala na način da se izmjerio promjer koloture na kojoj je bio pričvršćen davač pozicije te na koju se smještala remenica koja je pomicala maketu teretnih kolica. Nakon izmjerene promjera bilo je moguće izračunati opseg koloture koji je bio potreban kako bi se dobila vrijednost jednog impulsa. Vrijednost jednog impulsa davača pozicije se dobila tako da se opseg podijelio s brojem impulsa koju davač pozicije ima, odnosno s rezolucijom davača pozicije koja je u ovom slučaju 256. Kalibracija je realizirana programski i prikazana je na slikama 5.2 i 5.3. Prvi korak kalibracije je da se vrijednost koju davač pozicije daje na izlazu, pomnoži s vrijednošću jednog impulsa koja je u milimetrima, nakon čega se dobije stvarna vrijednost „CRS.ENC\_VAL\_mm“ u milimetrima. Kako bi se poboljšala preciznost makete, dodan je senzor koji ispravlja vrijednost davača pozicije ukoliko dođe do odstupanja koje je moguće zbog klizanja remenice što uzrokuje okretanje koloture bez da se kolica pomiču. Kada kolica aktiviraju senzor, u programu se uspoređuje vrijednost između davača pozicije i vrijednosti koja je zadana na senzoru. Vrijednost na senzoru je stalna i ona predstavlja vrijednost koju davač pozicije treba imati u toj poziciji na kojoj se nalazi senzor.

Usporedba se obavlja prilikom gibanja naprijed i natrag pri čemu se vrijednost na senzoru razlikuje u ta dva slučaja. Usporedba se odvija tako da se od vrijednosti na senzoru oduzme vrijednost koju davač pozicije prikazuje, te se tako dobiva razlika koja je nazvana „offset“ i u programu je označena kao „CRS.OFFSET\_VAL“.



Slika 5.2 Dobivanje „offset“ vrijednosti

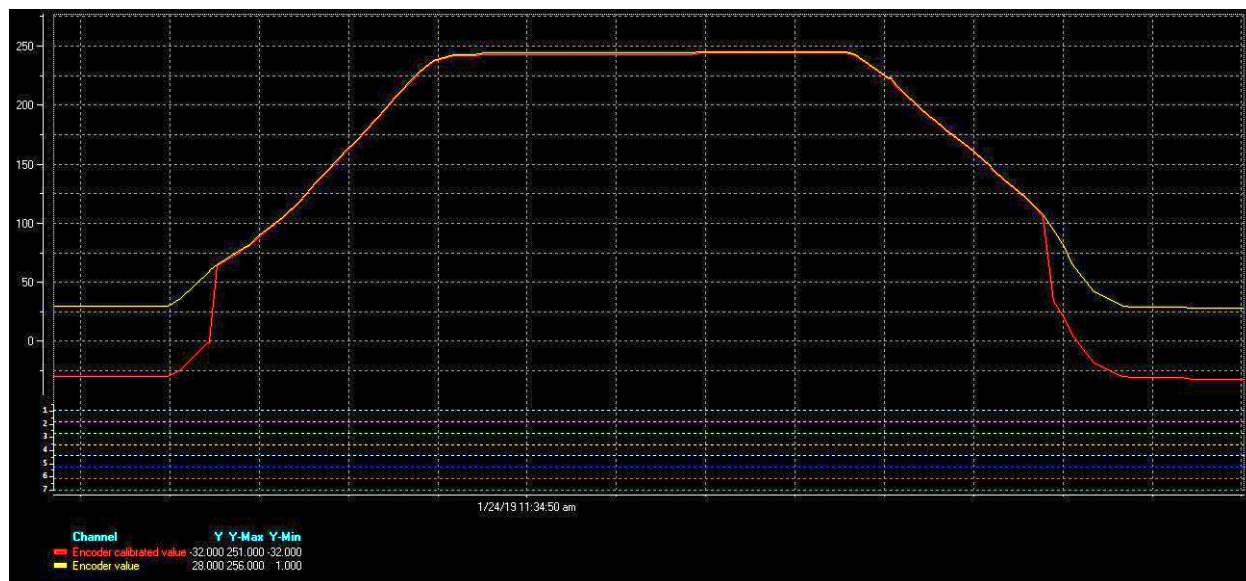
Kako bi se dobila kalibrirana vrijednost, potrebno je „offset“ vrijednost dodati vrijednosti koju prikazuje davač pozicije i time ju ispraviti. Ta radnja se izvršava blokom za zbrajanje i ona je prikazana na slici 5.3.



Slika 5.3 Dobivanje kalibrirane vrijednosti davača pozicije

Snimanje odziva kalibrirane i nekalibrirane vrijednosti je prikazano na slici 5.4 gdje se jasno vidi trenutak u kojem kolica aktiviraju senzor koji ispravi vrijednost davača pozicije. Na grafu je prikazan odziv kretanja kolica do „work“ pozicije u kojoj su mirovala nekoliko trenutaka, nakon čega su vraćena u „home“ poziciju. Žuti odziv predstavlja ne kalibriranu vrijednost, dok crvena predstavlja kalibriranu vrijednost. Tokom snimanja prikazanog grafa, kolica su dva puta aktivirala

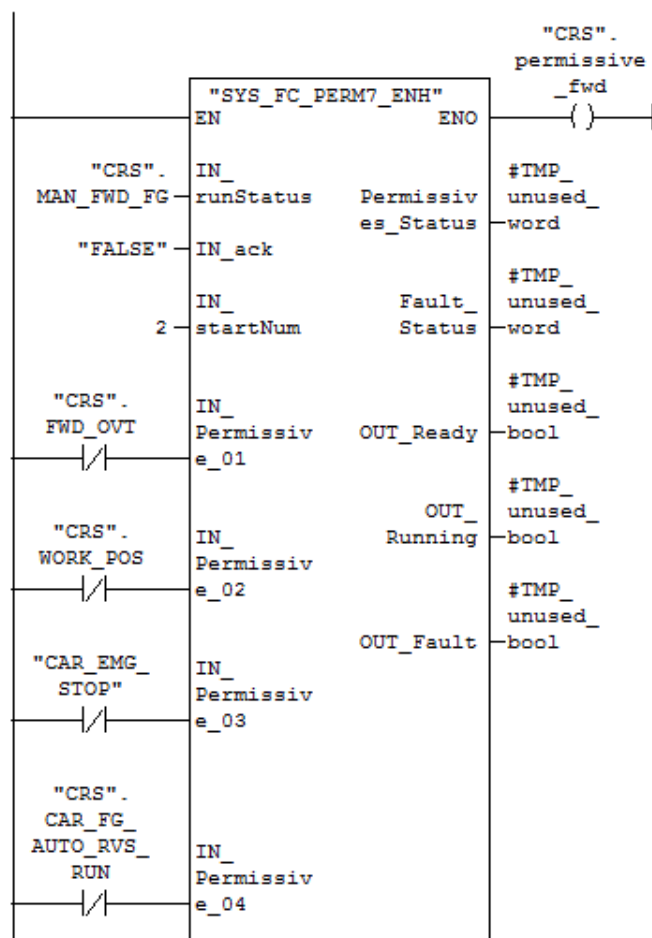
senzor, što je izazvalo skokovitu promjenu na crvenom odzivu, te je to dokaz ispravljanja vrijednosti koju davač pozicije prikazuje.



*Slika 5.4 Razlika između kalibrirane i ne kalibrirane vrijednosti*

## 5.4 Pokretanje i zaustavljanje teretnih kolica

Prije samog pokretanja teretnih kolica, kao i svakog uređaja koji se pozicionira u industriji, potrebno je ispuniti određene uvjete kako bi se pokretanje izvršilo. Uvjeti koje je potrebno ispuniti postavljaju se na ulaz funkcijskog bloka, te kada su svi uvjeti ispunjeni na izlazu se postavlja bit koji se naziva „permissive“ i koji je glavni uvjet za pozicioniranje. Funkcijski blok upotrebljen za dobivanje „permissive“ bita je sustavni i on je prikazan na slici 5.5 skupa s uvjetima postavljenim na ulazu.



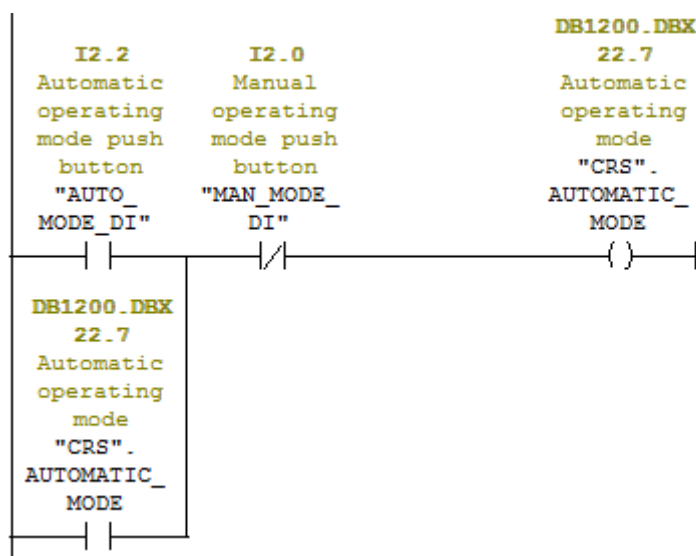
Slika 5.5 Blok „permissive-a“ za gibanje naprijed

„Permissive“ bit će prijeći u logičku jedinicu u trenutku kada su svi uvjeti na ulazu zadovoljeni. Uvjeti koji se moraju zadovoljiti za dobivanje „permissive“ bita su prikazani pomoću „Normally Closed“ kontakata te su smješteni na ulaz funkcijskog bloka „SYS\_FC\_PERM7\_ENH“. Signal „CRS.FWD\_OVT“ je jedan od uvjeta koji mora biti zadovoljen, odnosno dokle god bit „CRS.FWD\_OVT“ ne postane logička jedinica, uvjet će biti zadovoljen. „CRS.FWD\_OVT“ postaje logička jedinica u trenutku kada teretna kolica izađu iz područja pozicioniranja na način da premaše graničnu vrijednost pozicije u kojoj se trebaju zaustaviti. Premašivanje vrijednosti je moguće zbog ne prilagođene brzine zaustavljanja kolica, što će detaljnije biti opisano u potpoglavlju 5.5. „CRS.WORK\_POS“ predstavlja bit koji postaje logička jedinica kada kolica budu na „work“ poziciji, što znači da se „permissive“ bit može ostvariti pod uvjetom da kolica već nisu na željenoj poziciji. „CAR.EMG\_STOP“ je signal koji postaje aktivan u trenutku pritiskanja „emergency“ tipkala koji zaustavlja čitav sustav. Kako se teretna kolica gibaju u dva smjera, naprijed i nazad, bilo je potrebno postaviti „permissive“ bitove za oba smjera kretanja. Jedan od uvjeta za dobivanje tih bitova je da se kolica ne gibaju, odnosno ukoliko se kolica gibaju u jednom

smjeru, nije moguće poslati zahtjev za promjenu smjera gibanja dok kolica ne dostignu prethodno odabranu poziciju, te za to služi bit „CRS.CAR\_FG\_AUTO\_RVS\_RUN“. „CRS.permissive\_fwd“ je bit pomoću kojeg se izvršava zahtjev za gibanje naprijed do kojeg dolazi prilikom slanja kolica u „work“ poziciju, dok se prilikom slanja kolica u „home“ poziciju šalje zahtjev za gibanje unatrag pri čemu se „permissive“ bit naziva „CRS.permissive\_rev“.

Kako bi se prešlo u automatski način rada, potrebno je pritisnuti tipkalo koje je u programu označeno s „AUTO\_MODE\_DI“. Iz automatskog načina rada se izlazi pritiskom na tipkalo za ulazak u ručni način rada koje je nazvano „MAN\_MODE\_DI“.

Način na koji je to ostvareno prikazan je na slici 5.6, dok je na slici 5.7 prikazan koji uvjet mora biti zadovoljen da bi se prešlo u ručni režim rada.



Slika 5.6 Odabir automatskog režima rada

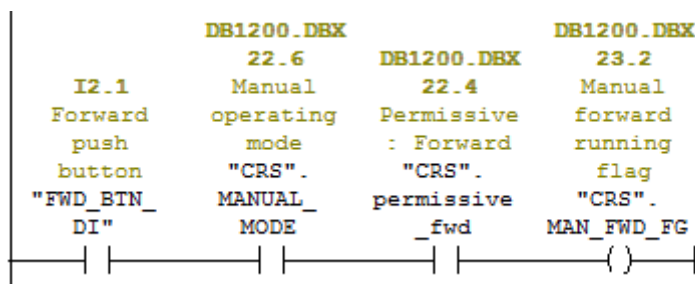
Bit koji potvrđuje da je aktivan automatski način rada je „CRS“\_AUTOMATIC\_MODE, te čim je taj bit jednak logičkoj nuli, započinje ručni način rada koji je označen sa „CRS“\_MANUAL\_MODE.



Slika 5.7 Ulazak u ručni način rada

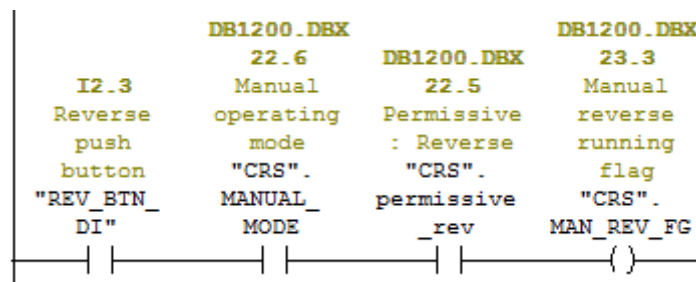


Nakon što je odabran ručni način rada i zadovoljeni svi uvjeti za gibanje, pritiskom tipkala za gibanje naprijed ili unatrag postaviti će se logička jedinica u bit koji je nazvan „zastavica“. Bitovne zastavice postoje za oba smjera kretanja i u oba režima rada, što znači da ih ukupno ima 4.



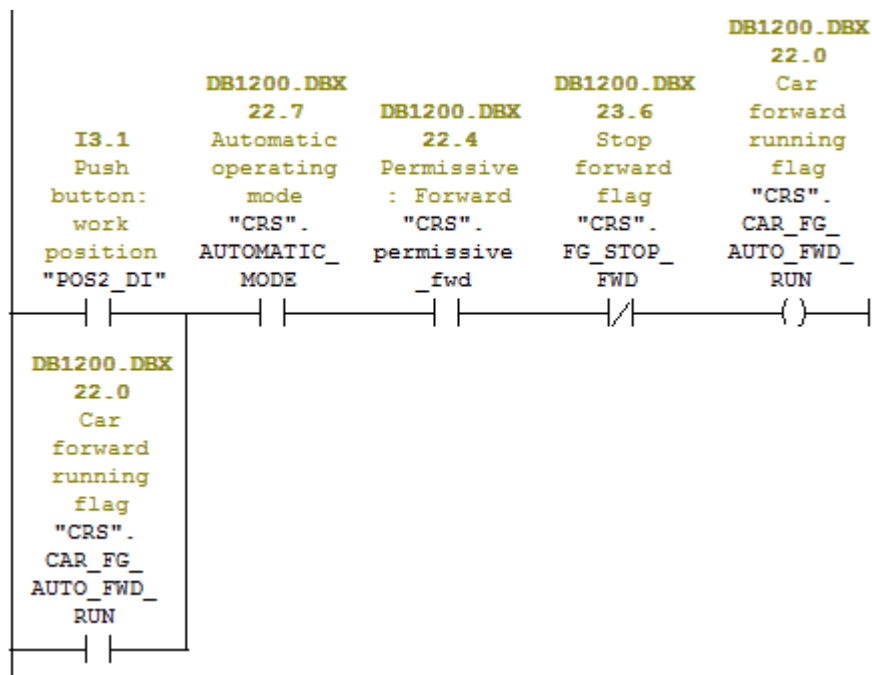
Slika 5.8 Bitovna zastavica za kretanje naprijed u ručnom načinu rada

Bitovne zastavice dolaze s oznakom „FG“ te je bitovna zastavica za gibanje naprijed u ručnom režimu rada nazvan „CRS.MAN\_FWD\_FG“. Kako bi taj bit prešao u stanje logičke jedinice potrebno je da bude pritisnuto tipkalo za kretanje naprijed čime se ulazni bit „FWD\_BTN\_DI“ postavlja u logičku jedinicu. Uz pritisnuto tipkalo mora biti aktiviran ručni režim rada te ispunjeni svi uvjeti za ostvarivanje „CRS.permissive\_fwd“ bita.



Slika 5.9 Bitovna zastavica za kretanje unatrag u ručnom načinu rada

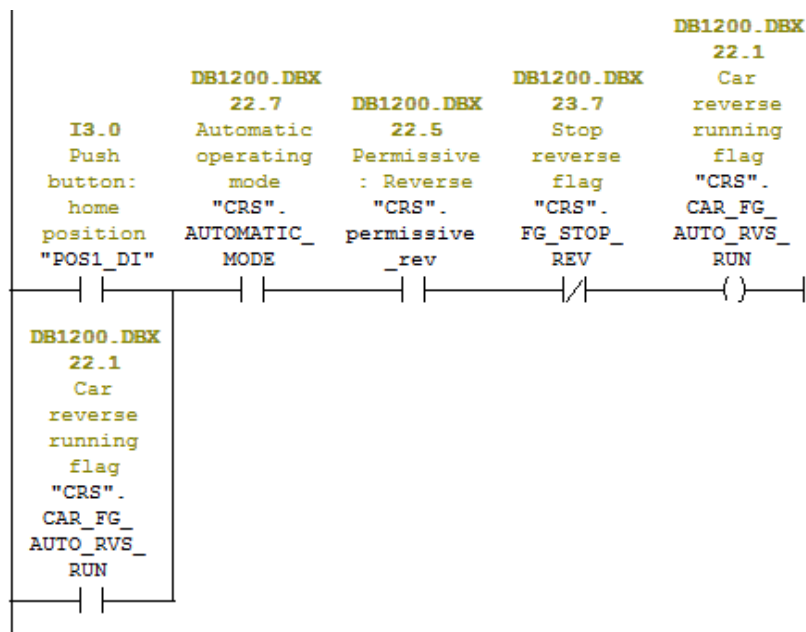
Bitovna zastavica za gibanje unatrag u ručnom režimu rada nazvan je „CRS.MAN\_FWD\_FG“. Kako bi taj bit prešao u stanje logičke jedinice, potrebno je da bude pritisnuto tipkalo za kretanje unatrag čime se ulazni bit „REV\_BTN\_DI“ postavlja u logičku jedinicu. Uz pritisnuto tipkalo mora biti aktiviran ručni režim rada te ispunjeni svi uvjeti za ostvarivanje „CRS.permissive\_rev“ bita.



Slika 5.10 Bitovna zastavica za kretanje naprijed u automatskom načinu rada

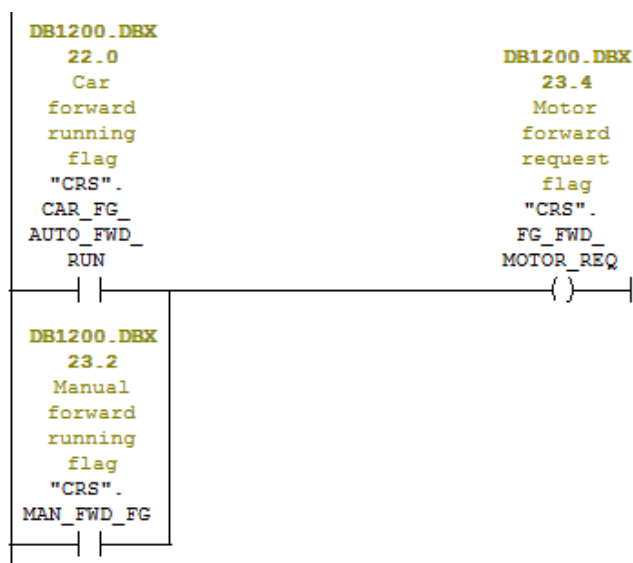
Za dobivanje bitovne zastavice u automatskom režimu rada potrebno je imati potvrdu da je automatski režim rada uključen, potvrdu o tome daje bit „CRS.AUTOMATIC\_MODE“. Osim potvrde o režimu rada, potrebno je ispuniti uvjete za dobivanje „CRS.permissive\_fwd“ ili „CRS.permissive\_rev“ bitova. U logici prikazanoj na slikama 5.9 i 5.10 primijenjen je način tzv. samodržanja koji omogućuje da bitovna zastavica zadrži vrijednost nakon puštanja tipkala, odnosno nakon što bit koji predstavlja stanje tipkala poprime vrijednost logičke nule.

Bitovi koji predstavljaju stanje tipkala su „POS2\_DI“ za gibanje naprijed i „POS1\_DI“ za gibanje unatrag. Samodržanje stanja bitovne zastavice za gibanjem naprijed se prekida bitom „CRS.FG\_STOP\_FWD“ dok se kod bitovne zastavice za gibanje unatrag samodržanje prekida bitom „CRS.FG\_STOP\_REV“ na način da bitovi poprime vrijednost logičke jedinice. Uvjeti za prekidanje stanja bitovnih zastavica su prikazani na slikama 5.16 i 5.17.



Slika 5.11 Bitovna zastavica za kretanje unatrag u automatskom načinu rada

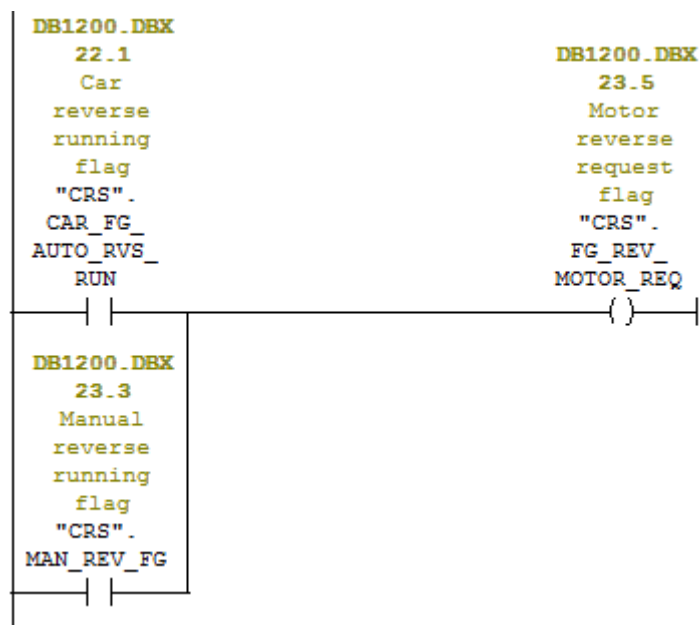
Nakon što je ostvarena jedna od bitovnih zastavica, šalje se zahtjev za gibanjem naprijed ili nazad. Zahtjev za gibanjem naprijed se postiže aktivacijom „CRS.CAR\_FG\_AUTO\_FWD\_RUN“ ili „CRS.MAN\_FWD\_FG“ bita što je prikazano na slici 5.12.



Slika 5.12 Zahtjev za gibanjem naprijed

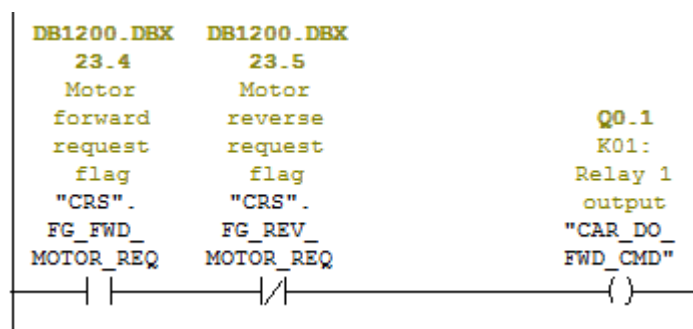
Zahtjev za gibanjem naprijed nosi ime „CRS.FG\_FWD\_MOTOR\_REQ“ i on se prekida čim oba bita poprima vrijednost logičke nule.

Zahtjev za gibanjem unatrag se postiže aktivacijom „CRS.CAR\_FG\_AUTO\_RVS\_RUN“ ili „CRS.MAN\_REV\_FG“ bita što je prikazano na slici 5.13. Zahtjev za gibanjem unatrag nosi ime „CRS.FG\_REV\_MOTOR\_REQ“ i on se također prekida kad su oba bita vrijednost logičke nule.

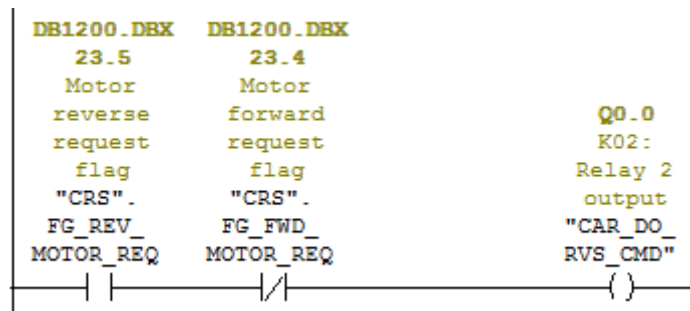


Slika 5.13 Zahtjev za gibanjem unatrag

Pomoću zahtjeva za gibanjem se aktiviraju releji te, ovisno o tome koji je relej upaljen, motor se giba naprijed ili nazad. Releji su spojeni na izlaze PLC-a i nazvani su „CAR\_DO\_FWD\_CMD“ i „CAR\_DO\_RVS\_CMD“. Da bi se releji uklopili, potrebno je izvršiti zadnji uvjet za pokretanje kolica, a to je da bude aktivan samo jedan zahtjev za gibanjem što je vidljivo na slikama 5.14 i 5.15.



Slika 5.14Uvjet za uklapanje releja gibanja naprijed



Slika 5.15 Uvjet za uklapanje releja gibanja unatrag

Isklapanje releja se postiže u trenutku ostvarivanja željene pozicije ili pritiskom tipkala „CAR\_STOP\_DI“. Kada kolica dođu u „work“ poziciju, bit „CRS.FG\_STOP\_FWD“ prelazi u logičku jedinicu, čime onemogućava gore spomenuto samodržanje za kretanje naprijed.



Slika 5.16 Prekid bitovne zastavice za kretanje naprijed

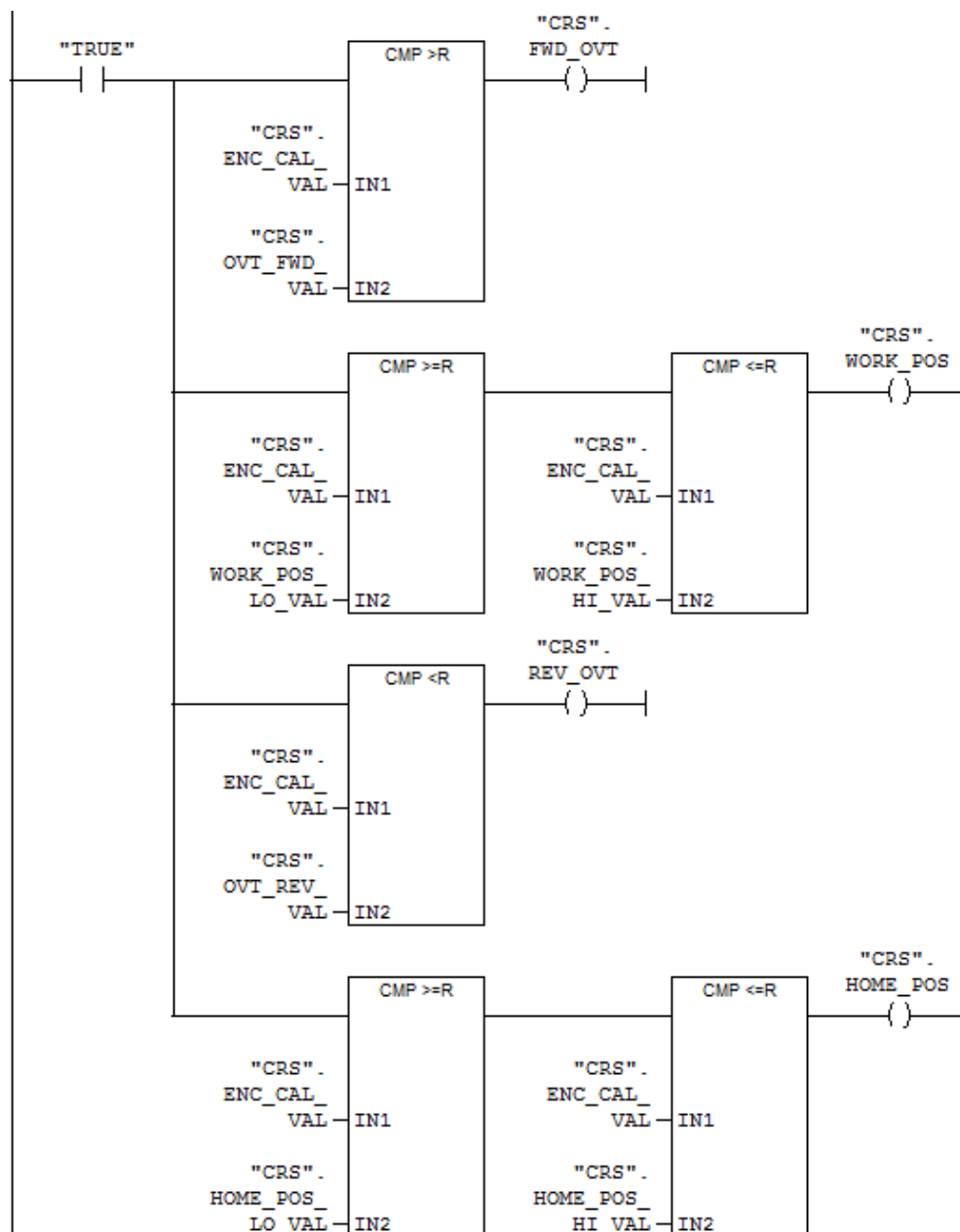
U trenutku kada kolica dođu u „home“ poziciju, bit „CRS.FG\_STOP\_REV“ prelazi u logičku jedinicu, čime onemogućava samodržanje za kretanje unatrag.



Slika 5.17 Prekid bitovne zastavice za kretanje unatrag

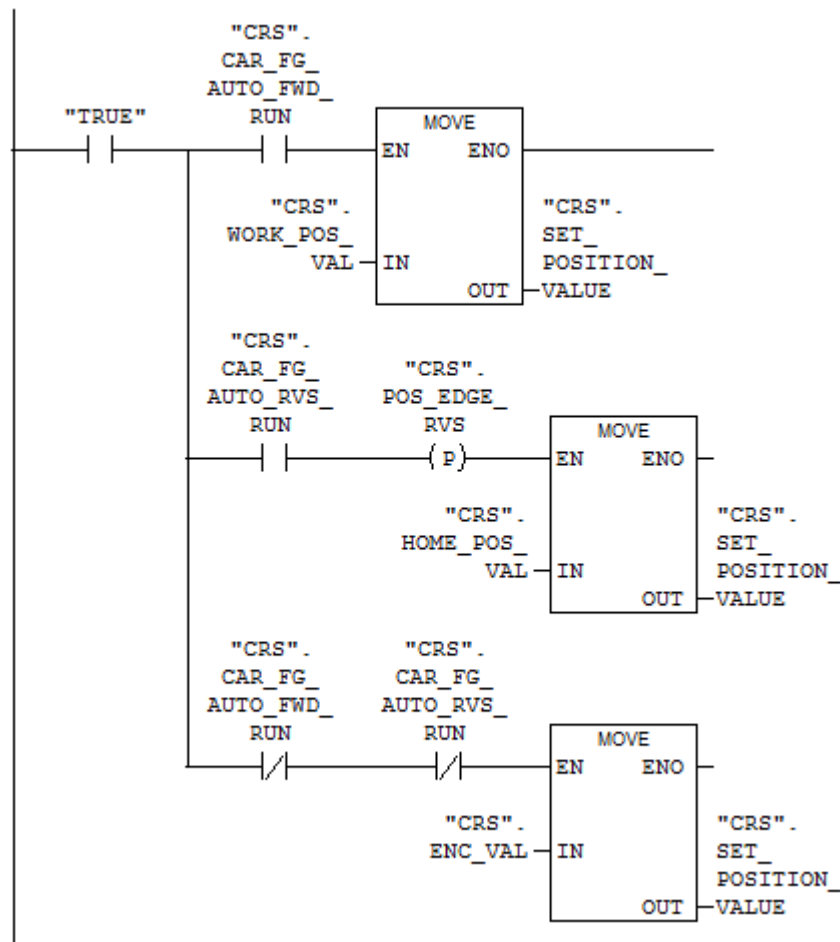
## 5.5 Pozicioniranje

Pozicioniranje u ovom programskom rješenju obuhvaća određivanje pozicije na koje se kolica trebaju zaustaviti, kao i način na koji će se zaustaviti. Pozicije na koje se kolica zaustavljaju su već spomenute ranije u tekstu, one su „work“ i „home“ pozicije, te njihova vrijednost određuje se na način prikazan na slici 5.18. Uspoređivanjem trenutne vrijednosti s graničnim vrijednostima pozicije dobiva se bit koji potvrđuje kada kolica dođu u određenu poziciju. Bit koji za potvrdu „work“ pozicije daje logičku jedinicu je „CRS.WORK\_POS“, dok je bit za potvrdu „home“ pozicije „CRS.HOME\_POS“.



Slika 5.18 Određivanje pozicije teretnih kolica

Ukoliko kolica pređu graničnu vrijednost pozicije, aktiviraju se bitovi koji onemogućuju ponovno pokretanje dok se kolica ne vrate u područje pozicije. Bit koji kazuje na to da su kolica premašila vrijednost „work“ je „CRS.FWD\_OVT“, dok za „home“ poziciju služi „CRS.REV\_OVT“. U slučaju kada je aktiviran automatski način rada, potrebno je upisati vrijednost pozicije na koju se kolica trebaju zaustaviti. Nakon zahtjeva za gibanje naprijed ili nazad u automatskom načinu rada, postavlja se vrijednost pozicije koja je stalna i ona je zadana u programu, a način na koji se to odvija prikazan je na slici 5.19.

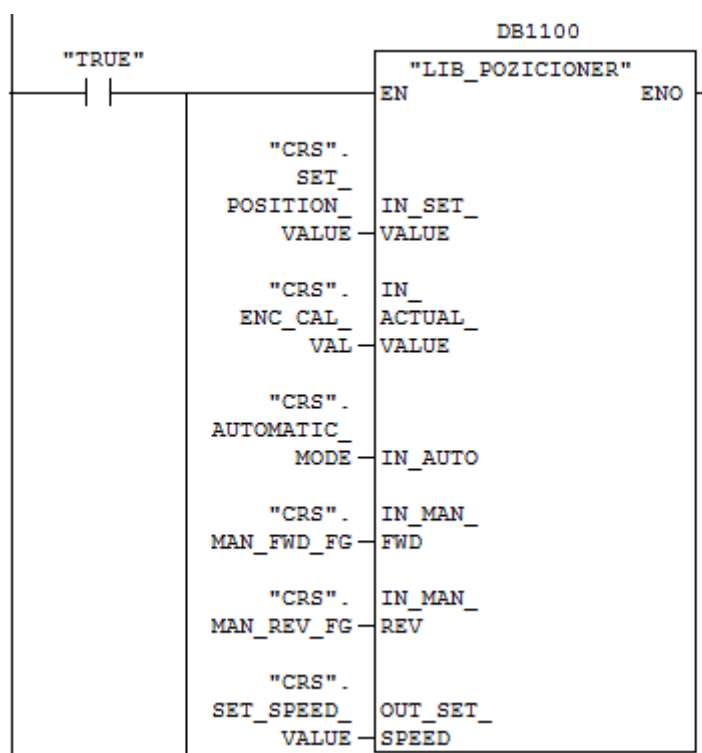


Slika 5.19 Postavljanje vrijednosti pozicije

„CRS.SET\_POSITION\_VALUE“ predstavlja varijablu u koju se upisuje vrijednost pozicije na koju se kolica trebaju zaustaviti. Vrijednost koja će se upisati ovisi o tome koja je bitovna zastavica aktivirana. U slučaju da je aktivirana bitovna zastavica za kretanje naprijed, u „CRS.SET\_POSITION\_VALUE“ se upisuje vrijednost varijable „CRS\_WORK\_POS\_VAL“, dok u slučaju kada je aktivirana bitovna zastavica za kretanje unatrag upisuje se vrijednost „CRS\_HOME\_POS\_VAL“. Ukoliko nije aktivirana niti jedna bitovna zastavica, upisuje se trenutna vrijednost davača pozicije, što znači da će u trenutku ako se pritisne „STOP“ tipkalo, kolica ostati na mjestu gdje jesu, neće se nastaviti gibati. Nakon postavljanja vrijednosti u

varijablu, ona se dovodi na ulaz u blok koji se naziva „LIB\_POZICIONER“ i on je prikazan na slici 5.20.

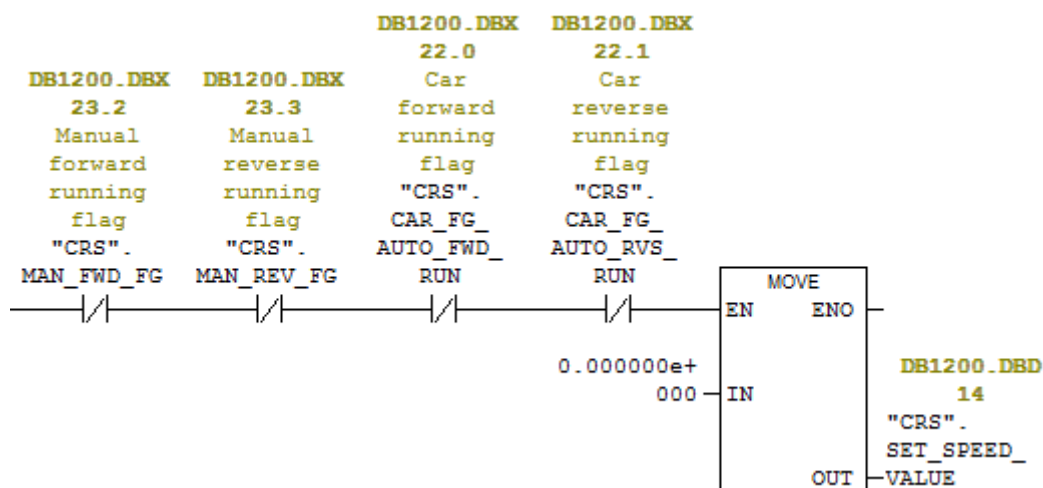
Zaustavljanje kolica je vrlo bitna stavka u stvarnim sustavima iz razloga što teretna kolica mogu biti namijenjena za transport velikih masa, i nepravilno rukovanje njima može prouzročiti krivo pozicioniranje. Zaustavljanje se postiže s dva stupnja promjene brzine, brzinom približavanja i brzinom zaustavljanja. Kada će kolica početi usporavati na određenu brzinu, te koliku će brzinu poprimiti, podešeno je unutar pozicionera. Unutar bloka smještena je logika pomoću koje se smanjuje ili povećava brzina kretanja teretnih kolica.



Slika 5.20 Pozicioner

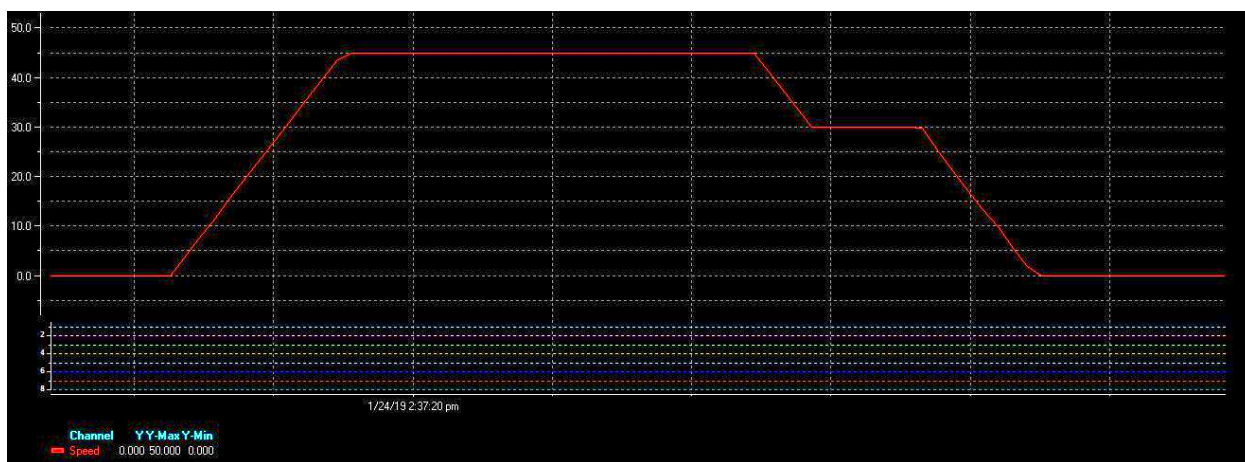
Ulazne vrijednosti su vrijednost pozicije koju je potrebno postići i trenutna pozicija teretnih kolica. Na ulaz se također dovode bitovi koji govore o tome u kojem se načinu rada nalazi sustav. Da bi se logika unutar pozicionera počela izvršavati, potrebno je da jedna od četiri bitovne zastavice bude ostvarena. Ukoliko niti jedna bitovna zastavica ne poprими logičku jedinicu, tada se brzini pridružuje vrijednost 0 kako je prikazano na slici 5.21.





Slika 5.21 Postavljanje brzine u nulu

Brzina prije nego što dođe na izlaz pozicionera, koji je nazvan „CRS.SET\_SPEED\_VALUE“, prolazi kroz blok koji se naziva rampa. Rampa, kada dobije vrijednost brzine na ulazu koju je potrebno postići, povećava ili smanjuje vrijednost trenutne brzine do vrijednosti koja se nalazi na ulazu. Jedan od bitnih parametara za postizanje željene brzine je akceleracija koja određuje koliko brzo će se odvijati promjena brzine. Koeficijenti ubrzavanja i usporavanja se dovode na ulaz rampe te se oni mogu mijenjati po potrebi. Snimljeni odziv brzine koji izlazi iz pozicionera je prikazan na slici 5.22, gdje je na y osi brzina, a na x osi vrijeme. Vrijednost brzine koju daje pozicioner dovodi se na ulaz „drivea“ motora koji pokreće kolica.



Slika 5.22 Odziv brzine

Na prikazanom odzivu je vidljivo kako brzina postepeno raste do željene brzine kojom se kolica gibaju do vremena kada počinje prvo usporenje nakon kojega nastupa brzina približavanja. Brzina približavanja traje neko vrijeme nakon kojeg nastupa drugo usporavanje koje dovodi do kontroliranog zaustavljanja kolica.

## 6. ZAKLJUČAK

U ovom završnom radu realiziran je i opisan sustav upravljanja pozicijom koji je neizostavan sustav u današnjoj industriji gdje je široko primijenjena automatizacija. Sustav upravljanja opisan u ovom radu se zasnivao na upravljanju pomoću apsolutnog davača pozicije rotirajućeg tipa koji je bio glavna stavka opisane teme. Kako bi se realizirao bilo kakav oblik automatskog upravljanja, potrebno je posjedovati teorijsku podlogu o sustavu kojim se želi upravljati. Upravo iz tog razloga se u radu opisao cijeli sustav kao i sve komponente potrebne za pravilan rad sustava upravljanja.

Upravljanje sustavom je ostvareno pomoću programabilnog logičkog kontrolera (PLC-a) na koji su se spajale sve komponente opisane u radu. Na PLC se smješta programsko rješenje koje predstavlja skup naredbi čijim izvršavanjem ostvaruje automatsko upravljanje zadanim sustavom.

Realizacija sustava je odrađena pomoću programskog alata Step7 gdje se pisalo programsko rješenje za zadani sustav upravljanja. Programsko rješenje je ključna stavka kod ovakvog načina upravljanja pozicijom. Tijekom izrade programskog rješenja moralo se pripaziti na organizacijsko rješenje upravljačkog programa, upotrebljene komponente prilikom projektiranja te i na parametre kako bi rješenje bilo zadovoljavajuće. Programsko rješenje opisano u radu je testirano na maketi pokretnih industrijskih kolica, čime se potvrdila ispravnost rješenja.

## LITERATURA

- [1] Prof.dr.sc Nedeljko Perić, AUTOMATSKO UPRAVLJANJE, Zagreb, 2004.
- [2] Benjamin Žagar, UPRAVLJANJE POZICIJOM U OTVORENOJ PETLJI POMOĆU DIGITALNIH DAVAČA POZICIJE TE DIGITALNIH I ANALOGNIH AKTUATORA, Diplomski rad, Osijek, 2016.
- [3] <file:///H:/Drives%20manuals/Manuals/Encoders/absolutencoder.html>
- [4] <https://www.theautomationstore.com/encoder-basics/>
- [5] [https://en.wikipedia.org/wiki/Open\\_collector](https://en.wikipedia.org/wiki/Open_collector)
- [6] Working with STEP 7, SIMATIC, SIEMENS
- [7] <https://www.designworldonline.com/precision-machines-count-on-robust-optical-encoders/>
- [8] [https://www.dynapar.com/technology/encoder\\_basics/quadrature\\_encoder/](https://www.dynapar.com/technology/encoder_basics/quadrature_encoder/)
- [9] <https://www.anaheimautomation.com/manuals/forms/encoder-guide.php>
- [10] [https://www.nexinstrument.com/OMRON-E6CP-AG5C-256-5M-Encoder\\_p\\_107438.html](https://www.nexinstrument.com/OMRON-E6CP-AG5C-256-5M-Encoder_p_107438.html)

## SAŽETAK

Ovaj završni rad opisuje proces upravljanja pozicijom koji je vrlo čest primjer automatskog upravljanja u industriji. Proces je opisan na primjeru industrijskih teretnih kolica. Kolica su se pozicionirala na dvije pozicije pomoću apsolutnog davača pozicije rotirajućeg tipa čiji način rada je opisan u radu. U radu su prikazane i opisane osnovne komponente sustava za upravljanje pozicijom koji se široko primjenjuju u industriji. Programsko rješenje je izrađeno u programskom paketu Step7 koji služi za programiranje programabilnih logičkih kontrolera te je uvid u ispravnost rješenja dan testiranjem na već postojećoj maketi teretnih kolica izrađenoj od strane tvrtke Danieli Systec.

**Ključne riječi:** automatsko upravljanje, davači pozicije, Step7, PLC

## **ABSTRACT**

### **Position control using absolute rotary encoder**

This thesis describes position control which is very common in automation in industry. The control process is described using an example of an industrial freight car. The car is positioned in two positions by means of an absolute rotary encoder. The principle of operation is described in the work. This paper presents and describes the basic components of the position control system that are widely applied in industry. The program solution was developed in the program package Step7 that was used to program the programmable logic controllers. The accuracy of the solution and testing was done on an existing industrial freight car model manufactured by Danieli Systec company.

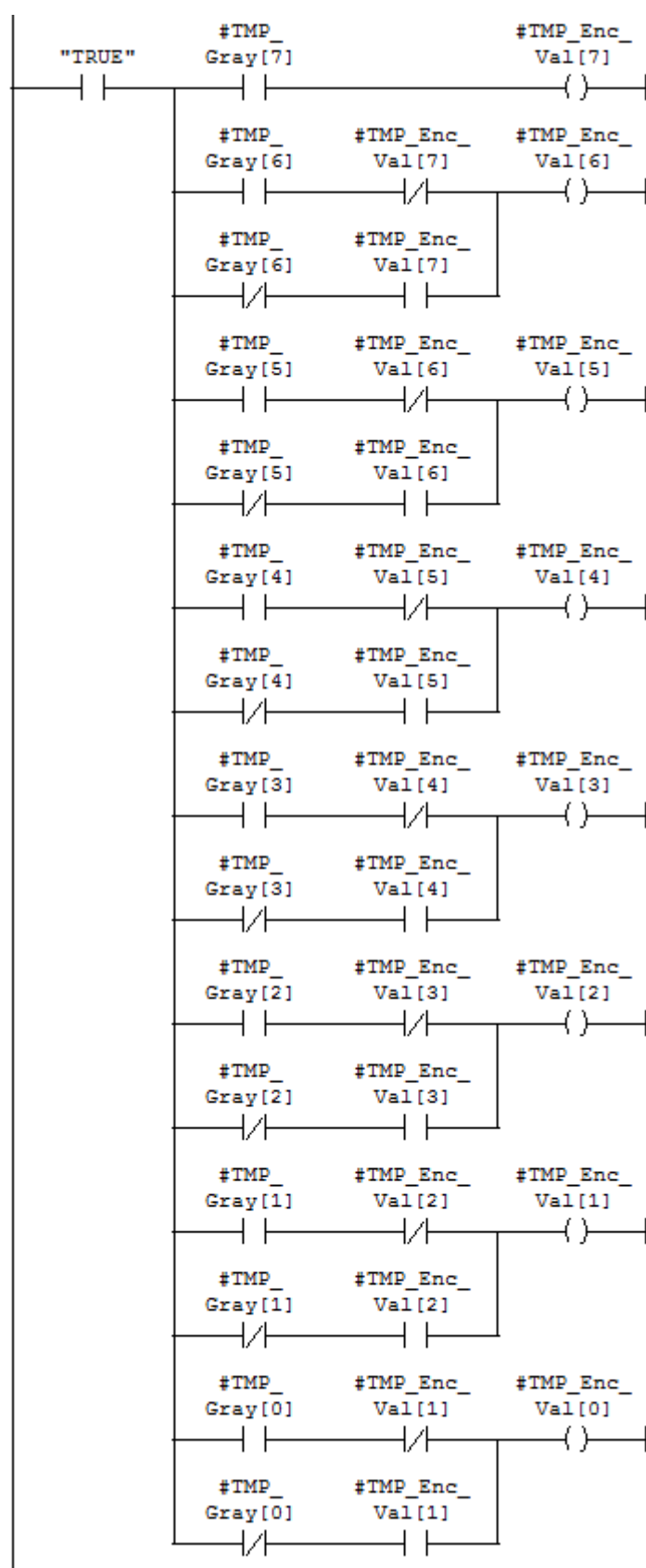
**Keywords:** Automation, Encoders, Step7, PLC

## **ŽIVOTOPIS**

Luka Ivković rođen je u Somboru 07. travnja 1997. godine. Osnovnu školu završava 2011. godine u mjestu Šećerana nakon čega se upisuje u Prvu srednju školu Beli Manastir. Srednju školu završava 2015. godine s odličnim uspjehom čime stječe zvanje Elektrotehničar. 2015. godine upisuje Fakultet elektrotehnike, računarstva i informacijskih tehnologija u Osijeku, smjer automatika. Stručnu praksu je odradio u tvrtki Danieli-Systec.

## PRILOG

Prilog 1: *Ekskluzivni ILI sklop na svakom od izlaza davača pozicije*



Prilog 2: NE logički sklop na svakom od izlaza apsolutnog davača pozicije

