DETEKTIRANJE VISINE PREDMETA POMOĆU IR SENZORA

Lamešić, Petar

Undergraduate thesis / Završni rad

2021

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **Karlovac University of Applied Sciences / Veleučilište u Karlovcu**

Permanent link / Trajna poveznica: https://urn.nsk.hr/urn:nbn:hr:128:280650

Rights / Prava: In copyright / Zaštićeno autorskim pravom.

Download date / Datum preuzimanja: **2024-04-26**

VELEUČILIŠTE U KARLOVCU
Karlovac University of Applied Sciences

Repository / Repozitorij:

Repository of Karlovac University of Applied Sciences - Institutional Repository

zir.nsk.hr

dabar
DIGITALNI AKADEMSKI ARHIVI I REPOZITORIJI

DETEKTIRANJE VISINE PREDMETA POMOĆU IR SENZORA

Lamešić, Petar

Undergraduate thesis / Završni rad

2021

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **Karlovac University of Applied Sciences / Veleučilište u Karlovcu**

Permanent link / Trajna poveznica: https://urn.nsk.hr/urn:nbn:hr:128:280650

Rights / Prava: In copyright / Zaštićeno autorskim pravom.

Download date / Datum preuzimanja: **2023-02-14**

VELEUČILIŠTE U KARLOVCU
Karlovac University of Applied Sciences

Repository / Repozitorij:

Repository of Karlovac University of Applied Sciences - Institutional Repository

zir.nsk.hr

dabar
DIGITALNI AKADEMSKI ARHIVI I REPOZITORIJI

VELEUČILIŠTE U KARLOVCU

STROJARSKI ODJEL

STRUČNI STUDIJ MEHATRONIKE

PETAR LAMEŠIĆ

DETEKTIRANJE VISINE PREDMETA POMOĆU IR SENZORA

ZAVRŠNI RAD

KARLOVAC, 2021.

KARLOVAC UNIVERSITY OF APPLIED SCIENCES
MECHANICAL ENGINEERING DEPARTMENT
PROFESSIONAL UNDERGRADUATE STUDY OF MECHATRONICS

PETAR LAMEŠIĆ

DETECTING OBJECT HEIGHT USING AN IR SENSOR

FINAL PAPER

KARLOVAC, 2021.

VELEUČILIŠTE U KARLOVCU

STROJARSKI ODJEL

STRUČNI STUDIJ MEHATRONIKE

PETAR LAMEŠIĆ

DETEKTIRANJE VISINE PREDMETA POMOĆU IR SENZORA

ZAVRŠNI RAD

Mentor: Dr.sc. Adam Stančić, viši predavač

KARLOVAC, 2021.

PREDGOVOR

Izjavljujem da sam završni rad samostalno izradio na temelju stečenih znanja tijekom studija i korištenjem navedene literature

Zahvaljujem se mentoru dr.sc Adamu Stančiću na stručnoj pomoći i savjetima pri izradi ovog rada.

Potpis:

Karlovac, 02.09.2021

SAŽETAK

Tema završnog rada je detektiranje visine predmeta pomoću IR senzora. U teoretskom dijelu opisan je princip rada i logika sustava po kojemu se detektiraju predmeti, određuje njihova visina te se razvrstavaju u određene spremnike. Opisani su senzori, Arduino mikrokontroler te LCD zaslon na kojemu se vrši prikaz trenutnog stanja sustava. U praktičnom dijelu završnog rada izrađen je sustav od 3 IR Break Beam senzora koji detektira predmet. U trenutku kada neki od senzora detektira predmet, prikazuje se poruka na zaslonu LCD-a koji ga je senzor detektirao.

Ključne riječi: Arduino, LCD, PLC, senzor

SUMMARY

The topic of the final paper is detecting the height of objects using IR sensors. The theoretical part describes the principle of operation and logic of the system according to which objects are detected, how their height is determined and how they are classified into certain containers. The sensors, the Arduino microcontroller and the LCD display on which the current state of the system is displayed are described. In the practical part of the final work, a system of 3 IR Break Beam sensors was made to detect the object. The moment one of the sensors detects an object, a message is displayed on the LCD screen that shows which sensor has detected it.

Keywords: Arduino, LCD, PLC, sensor

SADRŽAJ

1. UVOD 1

2. IZRADA LOGIČKOG MODELA 2

3. ODREĐIVANJE VISINE I RAZVRSTAVANJE PAKETA 8

3.1 Tipovi korištenih funkcija 17

3.2 Brojanje predmeta u spremniku 22

4. PRAKTIČNI DIO ZAVRŠNOG RADA 26

4.1 Povezivanje mikrokontrolera s računalom 26

4.2 Software za rad s Arduinom 28

4.3 Izrada programa za detektiranje predmeta pomoću IR Break Beam senzora 33

4.3.1 LCD zaslon 16x2 33

4.3.2 Testna pločica 36

4.3.3 Arduino Uno R3 pločica 38

4.3.4 IR Break Beam senzori 41

5. PROGRAM ZA DETEKCIJU VISINE PREDMETA 44

5.1 KOD PROGRAMA 47

5.2 Mogućnost proširenja projekta 53

6. ZAKLJUČAK 54

LITERATURA 55

POPIS SLIKA

Slika 1. IR Break Beam senzor [1] 2
Slika 2. Siemens LOGO! PLC [2] 3
Slika 3. Sučelje programa LOGO!Soft 4
Slika 4. Prikaz raspoloživih funkcija 5
Slika 5. Prikaz funkcija unutar radnog prostora 5
Slika 6. Pokretanje simulacije 6
Slika 7. Status simulacije 6
Slika 8. Primjer simulacije programa 7
Slika 9. 3D model sustava 9
Slika 10. Detekcija visine pomoću infracrvene zavjese 10
Slika 11. Prikaz sheme PLC programa 11
Slika 12. Poruka o nepravilnom obliku paketa 13
Slika 13. Prikaz aktivacije cilindra 2, slučaj 1 14
Slika 14. Prikaz aktivacije cilindra 1, slučaj 2 15
Slika 15. Poruka o neodgovarajućoj visini paketa 16
Slika 16. Logičke funkcije [4] 17
Slika 17. Logičke funkcije I, ILI, NE 18
Slika 18. Prikaz funkcije brojača 19
Slika 19. Prikaz funkcije flag 20
Slika 20. Prikaz funkcije Off-Delay 20
Slika 21. Prikaz funkcije Message Texts 21
Slika 22. Brojač paketa u spremniku 22
Slika 23. Trenutno stanje brojača 23
Slika 24. Prikaz aktivacije mehanizma za promjenu spremnika 24
Slika 25. Izgled finalne verzije programa 25
Slika 26. USB B kabel [9] 27
Slika 27. Provjera uspješnog povezivanja s mikrokontrolerom 27
Slika 28. Odabir pločice mikrokontrolera 28
Slika 29. Testiranje veze između računala i mikrokontrolera 29
Slika 30. Odabir Blink testa u aplikaciji 30
Slika 31. Generirani Blink kod 30
Slika 32. Prikaz opcije za verifikaciju programa 31
Slika 33. Slanje programa na mikrokontroler 31
Slika 34. Prikaz upaljene lampice na mikrokontroleru 31
Slika 35. Prikaz ugašene lampice na mikrokontroleru 32
Slika 36. LCD zaslon 16x2 34
Slika 37. Shema spajanja LDC zaslona i Arduino mikrokontrolera [15] 35
Slika 38. Prikaz spojenog LCD-a na testnu pločicu 36
Slika 39. Unutrašnjost testne pločice [16] 37
Slika 40. Metalna traka [16] 38
Slika 41. Dijelovi Arduino mikrokontrolera 40
Slika 42. IR Break Beam senzor [19] 41
Slika 43. Shema spajanja IR Break Beam senzora 42
Slika 44. Prikaz pull-up otpornika u sustavu [25] 43

Slika 45. Shema spajanja dijelova 44
Slika 46. Izgled spojenog sustava 45
Slika 47. Primjeri rada sustav 45

1. UVOD

Kod razvrstavanja robe u skladištu, pogonu ili na bilo kojem mjestu gdje se velika količina robe transportira, premješta i slaže bitno je što brže po određenim uvjetima rasporediti točno gdje koja vrsta paketa, materijala ili proizvoda treba biti raspoređena. Prije obrade paketa potrebno je znati njegovu veličinu zbog ograničenog prostora.

U završnom radu bit će razrađen primjer razvrstavanja paketa prema visini. U obzir će se uzeti 3 visine paketa. Paketi koji bi po visini odstupali od 3 uvjeta bili bi odloženi u poseban spremnik koji bi se po drugim uvjetima kasnije obrađivali. Za registraciju visine paketa koristit će se infracrveni Break Beam senzori.

U trenutku kada paket prođe pokraj senzora, ovisno koliko je paket visok pojavit će se signal na izlazima senzora koji su ga detektirali. Izlazi iz tih senzora će odlaziti na ulaze PLC-a (engl. Programmable Logic Controller - hrv. Programabilni Logički Kontroler). Izlaz iz PLC-a će biti određen prema kombinaciji signala na ulazima u PLC. Signali koji se pojavljuju na izlazima PLC-a mogu služiti za aktivaciju cilindara koji bi izbacivali pakete u spremnike. Detaljni opis kombinacija ulaza i kakvi sve mogu biti izlazi iz PLC-a će biti u nastavku rada.

Svaki od spremnika će imati brojač paketa koji ulaze u isti te nakon određenog broja paketa s brojača će biti poslan signal koji će aktivirati promjenu spremnika tako da spremnik koji je napunjen može ići na obradu (transport, slaganje, obrada paketa koji se u spremniku nalaze).

2. IZRADA LOGIČKOG MODELA

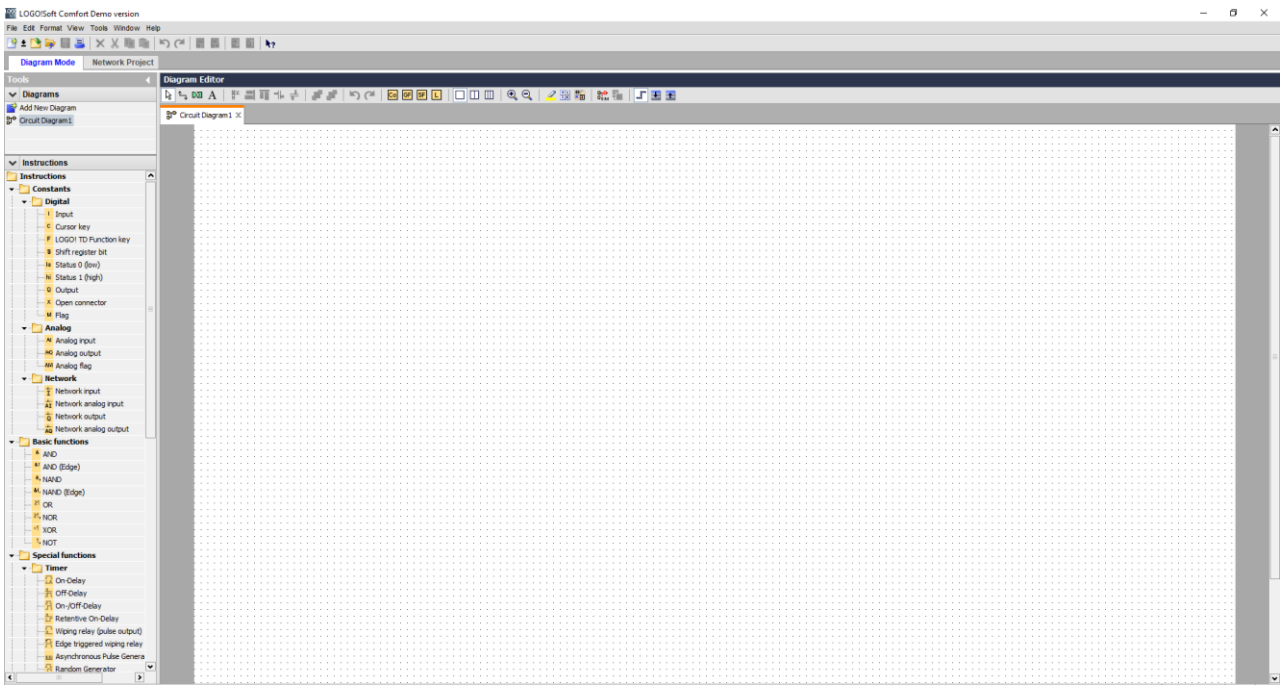
Za određivanje o kolikoj visini paketa se radi bit će korišteni IR Break Beam senzori. U trenutku kada se paket nađe između senzora on će prekinuti snop svjetlosti koji ide od predajnika prema prijemniku te će se pojaviti signal na izlazu iz prijemnika. [1] Signal s prijemnika će ići na jedan od ulaza u PLC (I1, I2, I3, I4) koji će sudjelovati u određivanju na koji izlaz iz PLC-a će biti poslan signal za aktivaciju pojedinog cilindra (Izlazi iz PLC-a su označeni s Q1, Q2, Q3, Q4).

Slika 1. IR Break Beam senzor [1]

Na slici iznad prikazani su predajnik i prijemnik infracrvenog senzora koji će činiti infracrvenu „zavjesu“ za registriranje predmeta. Bijela žica od prijemnika je žica koja služi za davanje signala ako se snop svjetlosti prekine. [1] Signal s bijele žice će ići na ulaz I1 PLC-a. Siemens LOGO! PLC ima 8 raspoloživih ulaza i 4 izlaza. Ulazi su označeni s I1, I2, I3, I4, I5, I6, I7 i I8, a izlazi s Q1, Q2, Q3 i Q4.

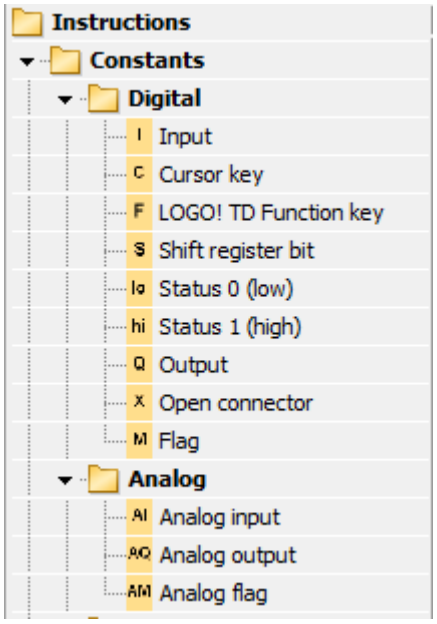
Slika 2. Siemens LOGO! PLC [2]

Siemens LOGO! PLC prikazan na slici 2. se programira u programu LOGO!Soft Comfort. Tu se određuje broj ulaza i izlaza, koje sve elemente će sadržavati i po kojoj logici će sustav sortirati pakete. Može se koristiti logika funkcijskih blok dijagrama ili ljestvičarska logika. U radu će se koristiti logika funkcijskih blok dijagrama. [3]

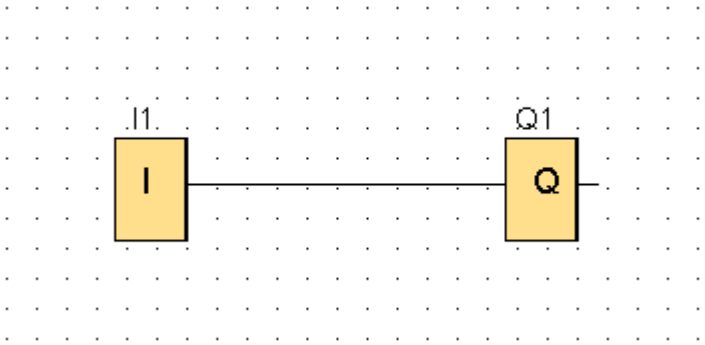


Slika 3. Sučelje programa LOGO!Soft

Slika 3. prikazuje sučelje programa u kojemu se formira logika po kojoj će PLC razvrstavati pakete. Sučelje se sastoji od instrukcija, prostora u kojemu se izrađuje sustav i alatne trake. S lijeve strane iz prostora koji se zove “Instructions” odabiru se funkcije te ulazi i izlazi s kojima se planira raditi. Dovoljno je kliknuti lijevom tipkom miša na bilo koju od funkcija i ponovno kliknuti lijevu tipku miša u radni prostor te će se funkcija uvrstiti u program.

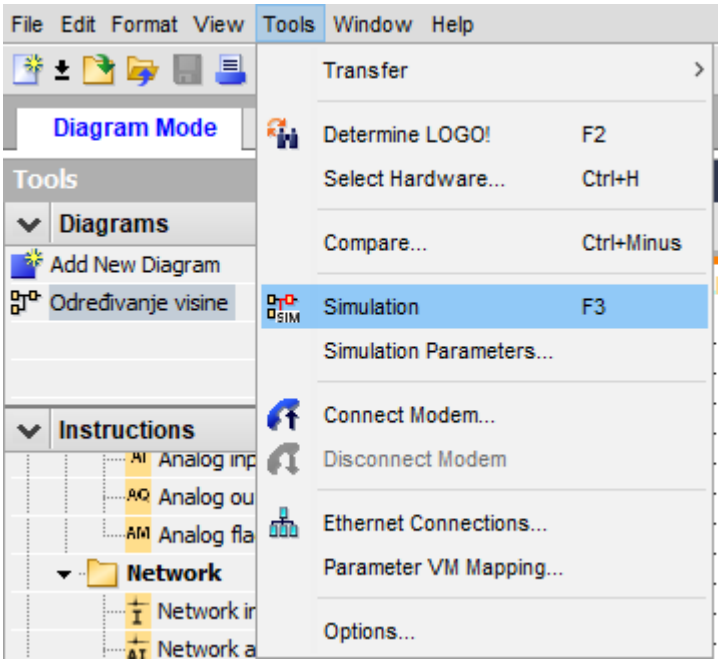


Slika 4. Prikaz raspoloživih funkcija

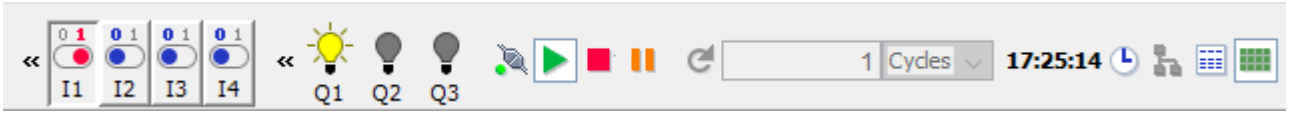


Slika 5. Prikaz funkcija unutar radnog prostora

Na slici 4. je prikazano gdje se nalazi i kako izgleda prostor Instructions iz kojeg se odabiru funkcije za formiranje logike. Na slici 5. je prikazano kako izgleda kada se odaberu neke od funkcija iz polja Instructions u radni prostor u kojemu se formira logika sustava. Na slici 5. s lijeve strane se nalazi funkcija I1 koja će predstavljati ulaz u sustav, a s desne strane je funkcija Q1 koja će predstavljati izlaz iz sustava.

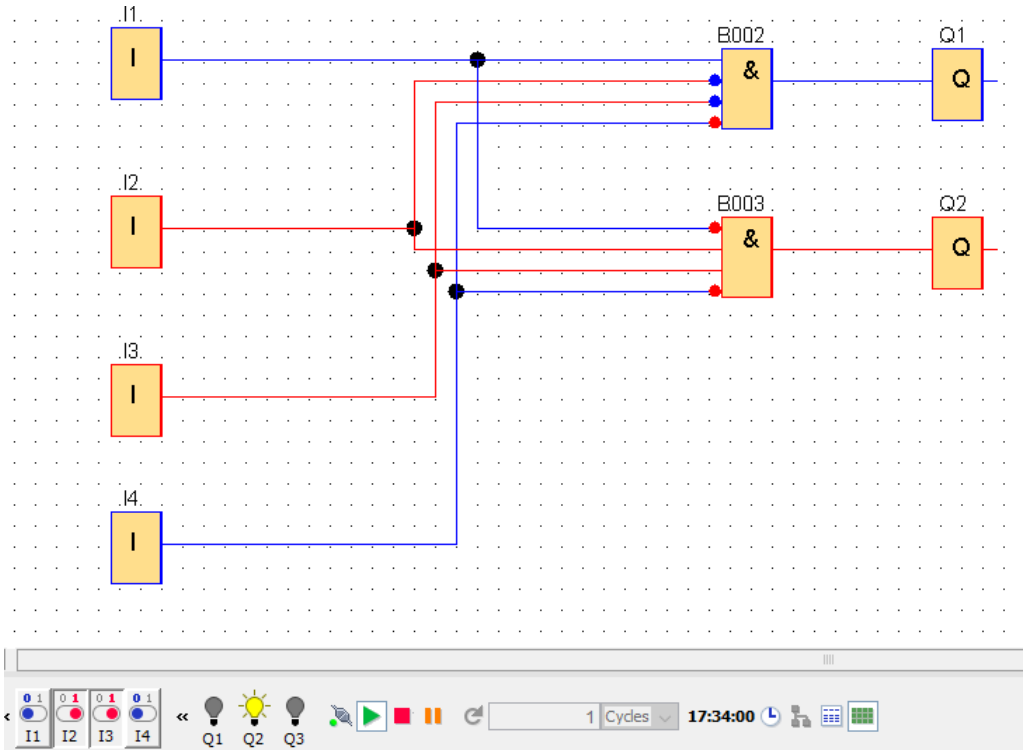


Slika 6. Pokretanje simulacije



Slika 7. Status simulacije

Na slici 6. je prikazano kako se pokreće simulacija programa koji je izrađen. Nakon odabira opcije za simuliranje, simulacija će se prikazati u donjem dijelu radnog prostora. Slika 7. prikazuje kako izgleda simulacija programa. U slučaju sa slike 7. može se vidjeti kako izgleda kada se na ulazu I1 pojavi signal. I1 prijeđe iz stanja 0 u stanje 1 te se promijeni boja iz plave u crvenu. Q1 je prikazan kao upaljena lampica što označava da kombinacija signala na ulazima u sustav zadovoljava uvjete za aktivaciju signala na izlazu Q1 iz sustava.



Slika 8. Primjer simulacije programa

Slika 8. prikazuje kako izgleda jednostavni program kod pokretanja simulacije. Lijevim klikom miša na pojedini ulaz (I1, I2, I3 ili I4) u prostoru simulacije moguće je mijenjati stanje ulaznih parametara sustava. Izlaz Q1 će biti uključen samo kada postoji signal na ulazu I1 i kada nema signala na ulazima I2, I3 i I4. Izlaz Q2 će biti uključen samo kada postoji signal na ulazima I2 i I3 i dok nema signala na ulazima I1 i I4. Logika, broj ulaza i izlaza sustava te ostale funkcije (neke od funkcija su brojač, analogni komparator, prosječna vrijednost) će se razlikovati od primjera do primjera ovisno o tome što se želi i na koji način postići.

3. ODREĐIVANJE VISINE I RAZVRSTAVANJE PAKETA

Kako bi se definirao logički model za PLC mora se utvrditi koliko je ulaza potrebno, koliko izlaza, koji su uvjeti aktiviranja pojedinog izlaza, što se događa sa signalima na izlazu i ako postoji potreba za prikazivanjem poruka na zaslon. Koristit će se 4 ulaza PLC-a. Prva 3 ulaza (I1, I2 i I3) će biti za određivanje visine predmeta, a četvrti ulaz (I4) će biti za signaliziranje visine predmeta koja nije dopustiva. Ako dođe do aktiviranja četvrtog ulaza takav paket će morati ići na dodatno razvrstavanje.

Izlazi iz PLC-a će biti signali koji u stvarnosti i u realnom pogonu mogu aktivirati cilindre ili bilo koje uređaje koji mogu manipulirati s paketima koji se nalaze na pokretnoj traci. Pri izradi programa također treba paziti da se u budućnosti program može nadograđivati prema uvjetima i zahtjevima krajnjih korisnika/klijenata.

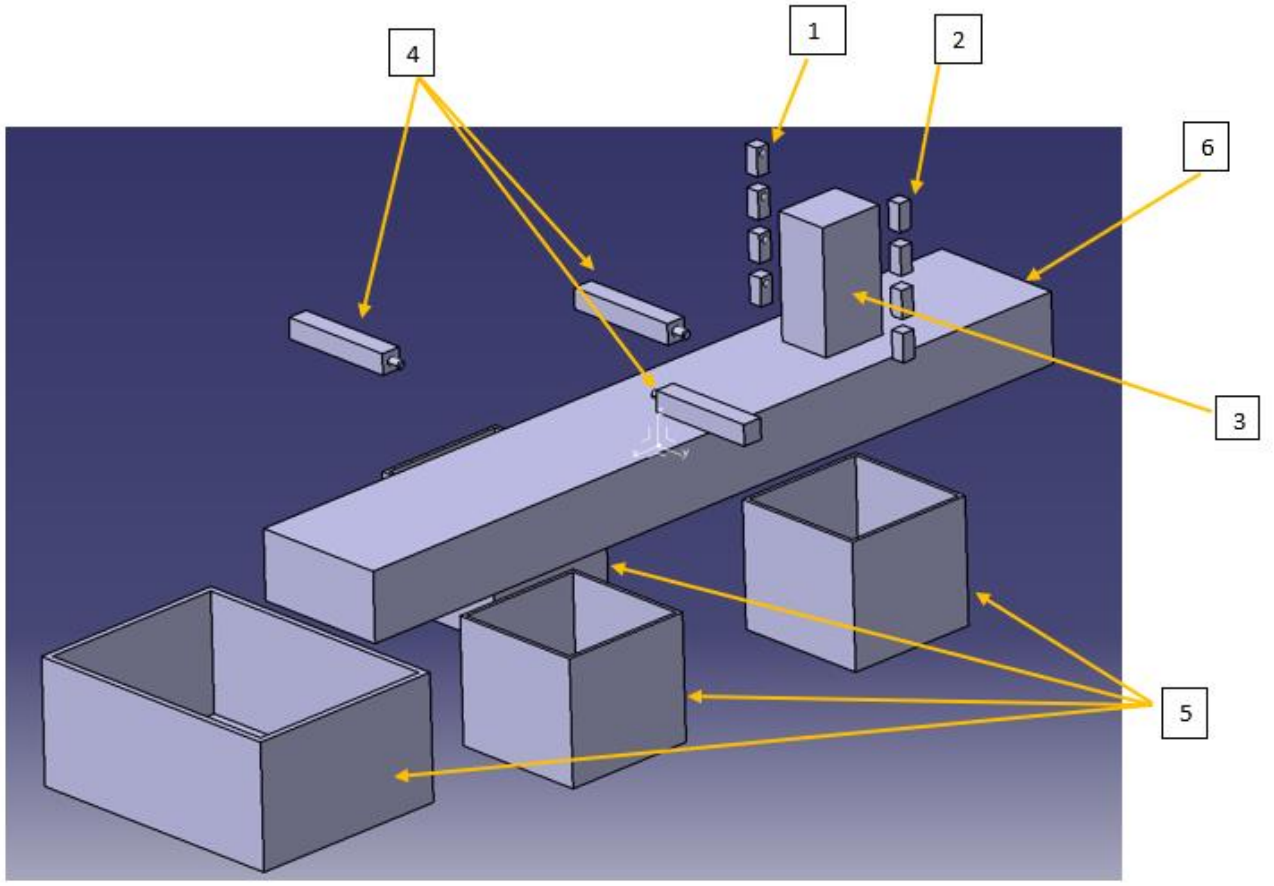
IR Break Beam senzori će biti postavljeni jedan iznad drugog te će tako činiti svjetlosnu infracrvenu zavjesu koja će služiti za detekciju paketa i njegove visine tijekom prolaska kroz istu.

Uvjet za aktivaciju prvog cilindra koji izbacuje pakete s pokretne trake u prvi spremnik će biti aktivan samo jedan ulaz I1. U tom slučaju će biti aktivan izlaz Q1 PLC-a. Senzor koji šalje signale na ulaz I1 PLC-a će se nalaziti na najnižoj poziciji. Za primjer će se odrediti da se nalazi na visini od 10 cm iznad pokretne trake. Svaki sljedeći postavljeni senzor će se nalaziti na 10 cm više od prethodnog. Što će značiti da će senzor koji šalje signal na ulaz I2 PLC-a biti na visini od 20 cm, a senzor koji šalje signal na ulaz I3 PLC-a biti na visini od 30 cm. Senzor koji signalizira da se radi o predmetu koji ne zadovoljava visinu bit će postavljen na visinu od 40 cm i on će slati signal na ulaz I4 PLC-a.

Aktivacija drugog cilindra koji će izbacivati pakete u drugi spremnik će biti moguća samo u slučaju ako postoji signal na ulazu I1 i ako postoji signal na ulazu I2. Ulazi I3 i I4 ne smiju biti aktivni. U bilo kojem drugom slučaju izlaz Q2 PLC-a koji daje signal za aktivaciju drugog cilindra neće biti aktiviran.

Treći cilindar će se aktivirati samo u slučaju kada postoji signal na ulazima I1 i I2 PLC-a i kada postoji signal na ulazu I3 PLC-a. U tom slučaju će biti aktivan izlaz Q3 PLC-a koji će aktivirati cilindar.

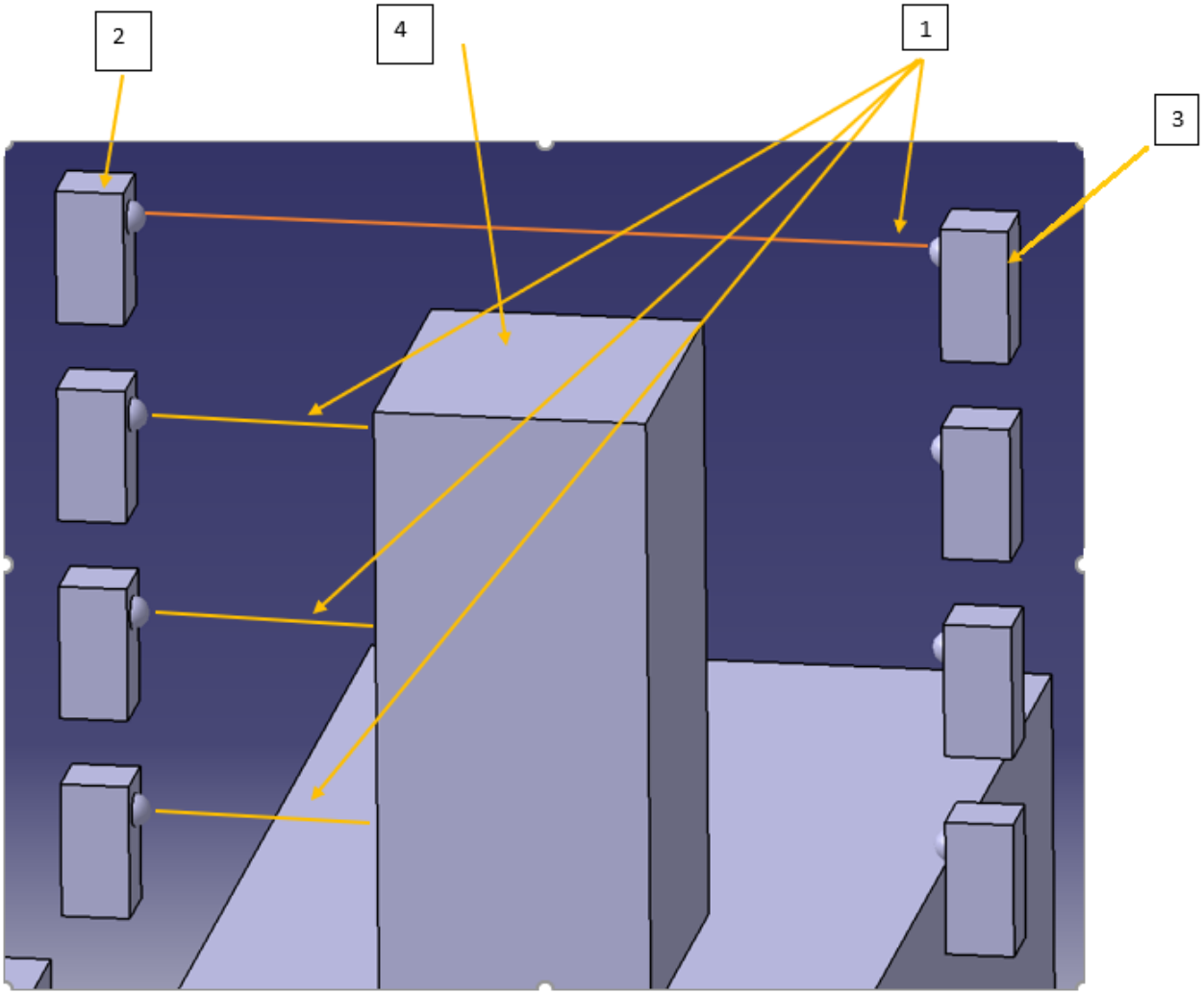
Zadnji slučaj će biti kada postoji signal na ulazima I1, I2 i I3, ali i na ulazu I4 PLC-a koji prima signale s četvrtog senzora koji se nalazi na visini od 40 cm. Uz aktivni četvrti izlaz PLC-a neće biti dodijeljen niti jedan cilindar nego će se na zaslonu PLC-a pojavljivati poruka u kojoj će biti naznačeno da se radi o neodgovarajućoj visini te da će predmet biti proslijeđen u spremnik za razvrstavanje.



Slika 9. 3D model sustava

Slika 9. prikazuje 3D model zamišljenog sustava izrađenog u programu Catia. Pod brojem 1 na slici iznad prikazani su predajnici infracrvenog snopa svjetlosti. Koristit će se 4 takva senzora i oni će slati signale na ulaze I1, I2, I3 i I4 PLC-a. Pod brojem 2 na slici iznad prikazani su prijemnici kojih će također biti 4. Kombinacija prijemnika i predajnika čini infracrvenu zavjesu kroz koju predmet prolazi. Pod brojem 3 na slici iznad je paket koji se kreće po pokretnoj traci. Njemu se određuje visina i razvrstava se po spremnicima. Pod brojem 4 na slici iznad su cilindri ili bilo koji elementi koji mogu manipulirati s paketom i

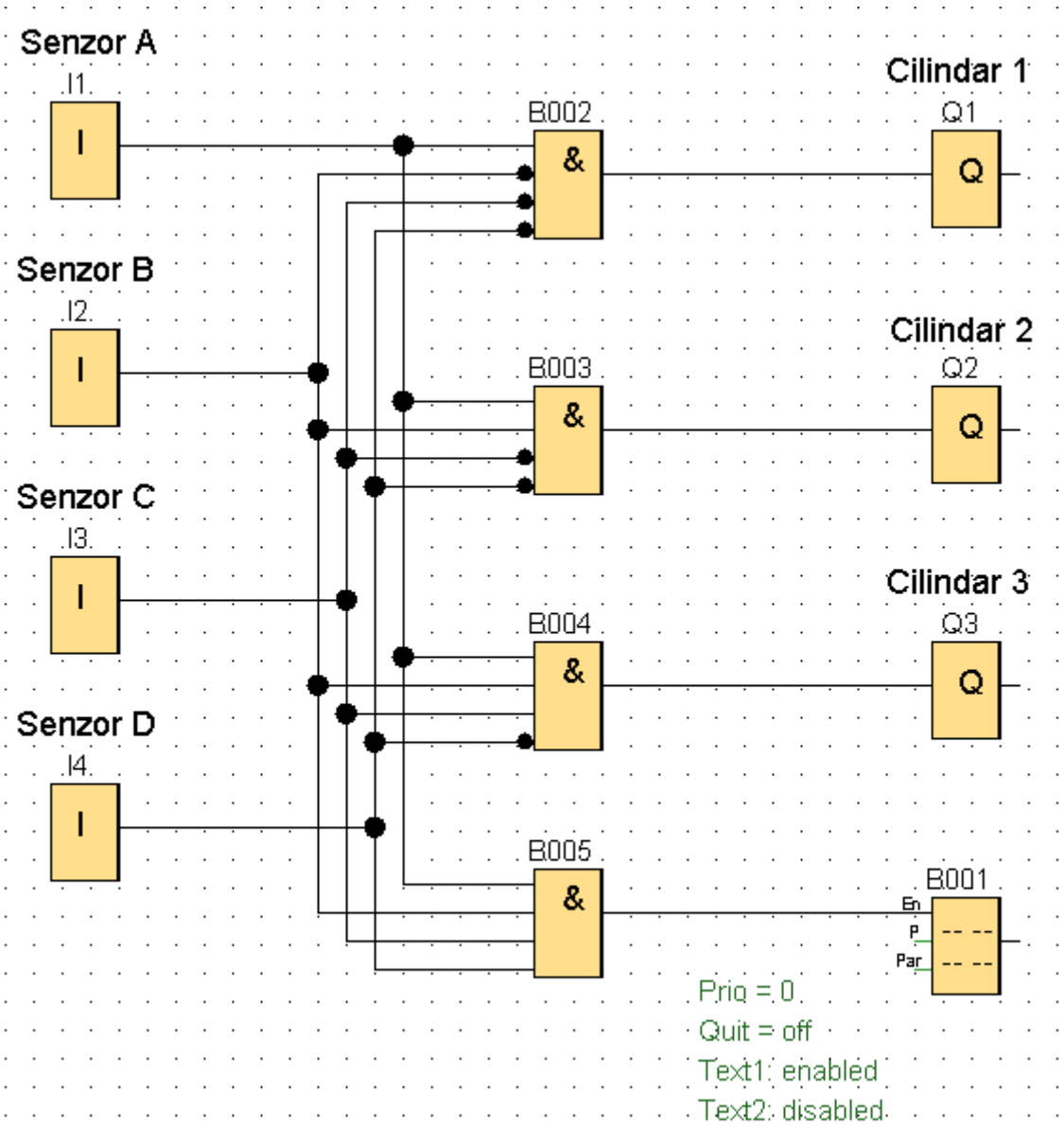
izbaciti ga s pokretne trake u odgovarajući spremnik. Pod brojem 5 na slici iznad su spremnici u kojima paketi završavaju.



Slika 10. Detekcija visine pomoću infracrvene zavjese

Na slici 10. je prikaz kako izgleda kada predmet/paket dođe u područje infracrvene zavjese. Ovisno o tome koliko je predmet visok, određen broj infracrvenih zraka će se prekinuti. Na slici iznad su prekinute 3 zrake dok je četvrta ostala netaknuta i snop je došao od predajnika do prijemnika. Pod brojem 1 na slici iznad su prikazane infracrvene zrake svjetlosti. Pod brojem 2 na slici iznad su prikazani predajnici. Pod brojem 3 na slici iznad su

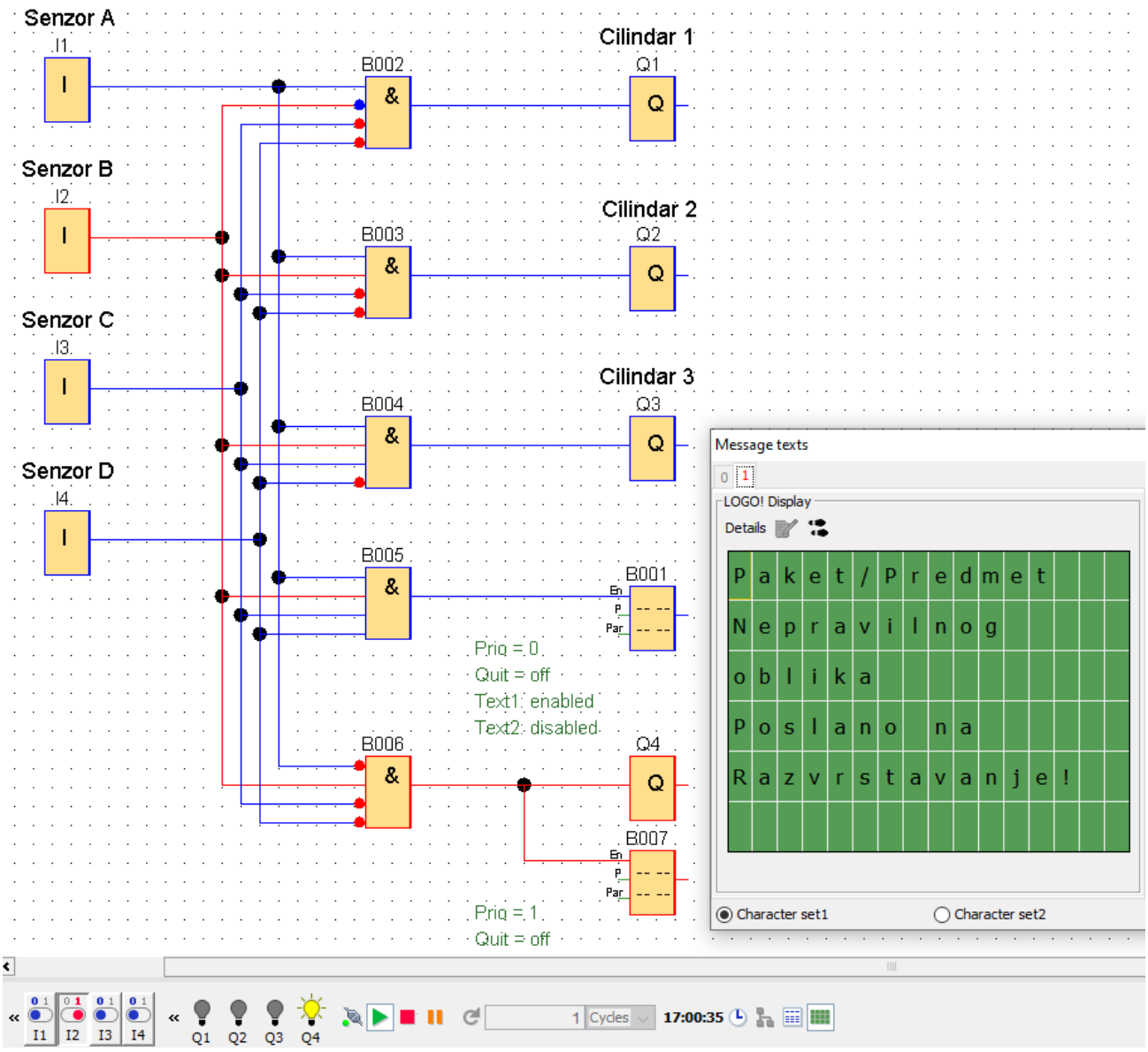
prikazani prijemnici. Pod brojem 4 na slici iznad je prikazan predmet koji prolazi kroz polje infracrvene svjetlosti.



Slika 11. Prikaz sheme PLC programa

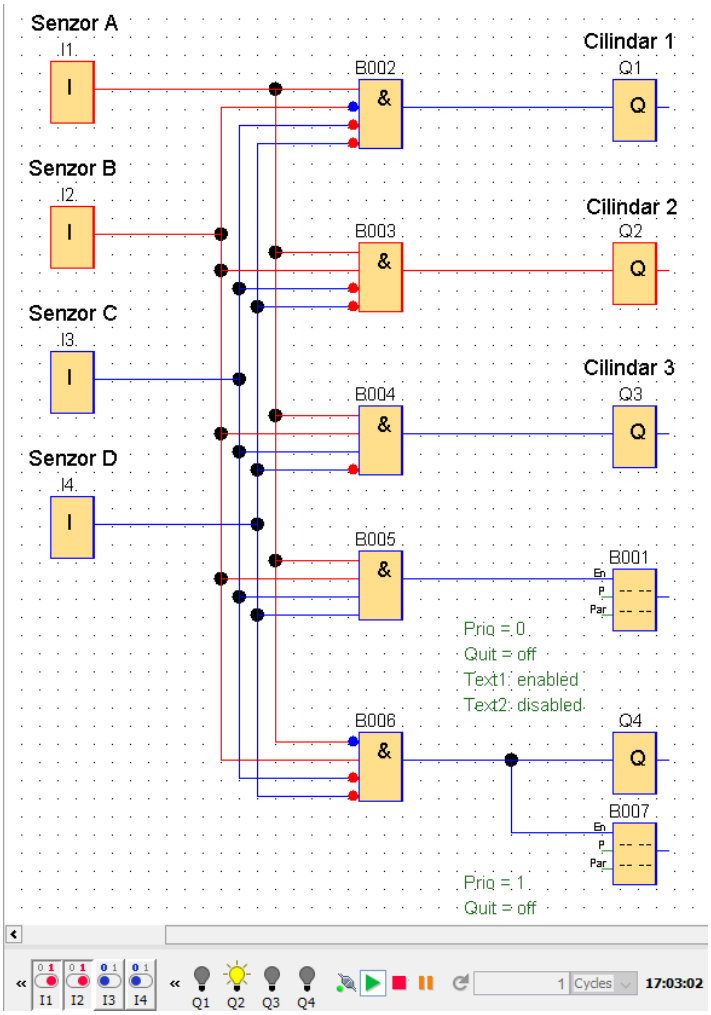
Na slici 11. je prikaz programa koji je prethodno opisan. Na lijevoj strani nalaze se 4 ulaza PLC-a (I1, I2, I3 i I4). Na svaki od ulaza dolaze signali sa senzora. Na ulaz I1 (u programu još označeno kao Senzor A) dolazi signal sa senzora koji je na najmanjoj visinskoj poziciji od 10 cm. Na ulaz I2 (u programu još označeno kao Senzor B) dolazi signal sa

senzora koji se nalazi na visi od 20 cm. Na ulaz I3 (u programu još označeno kao Senzor C) dolazi signal sa senzora koji se nalazi na visini od 30 cm. Na ulaz I4 (u programu još označeno kao Senzor D) dolazi signal sa senzora koji se nalazi na visini od 40 cm. U sredini se nalaze I odnosno AND sklopovi, a na desnoj strani programa su 3 izlaza (Q1, Q2 i Q3) s kojih se šalju signali za aktivaciju cilindara. Signal s izlaza Q1 aktivira cilindar koji paket izbacuje u spremnik 1. Signal s izlaza Q2 aktivira cilindar koji paket izbacuje u spremnik 2. Signal s izlaza Q3 aktivira cilindar koji paket izbacuje u spremnik 3. Zadnja funkcija s desne strane (funkcijski blok B001) služi za prikazivanje željene poruke na zaslonu PLC-a ili modula za prikaz poruke. Nadogradnja prezentiranog programa može biti slučaj kad je oštećen paket ili kada je paket nepravilnog oblika. Tada bi se dodalo još logičkih sklopova uz funkciju za prikaz poruke na zaslon te bi se opisalo što se događa i koji je uzrok nepravilnosti. To će biti prikazano na slici 12.



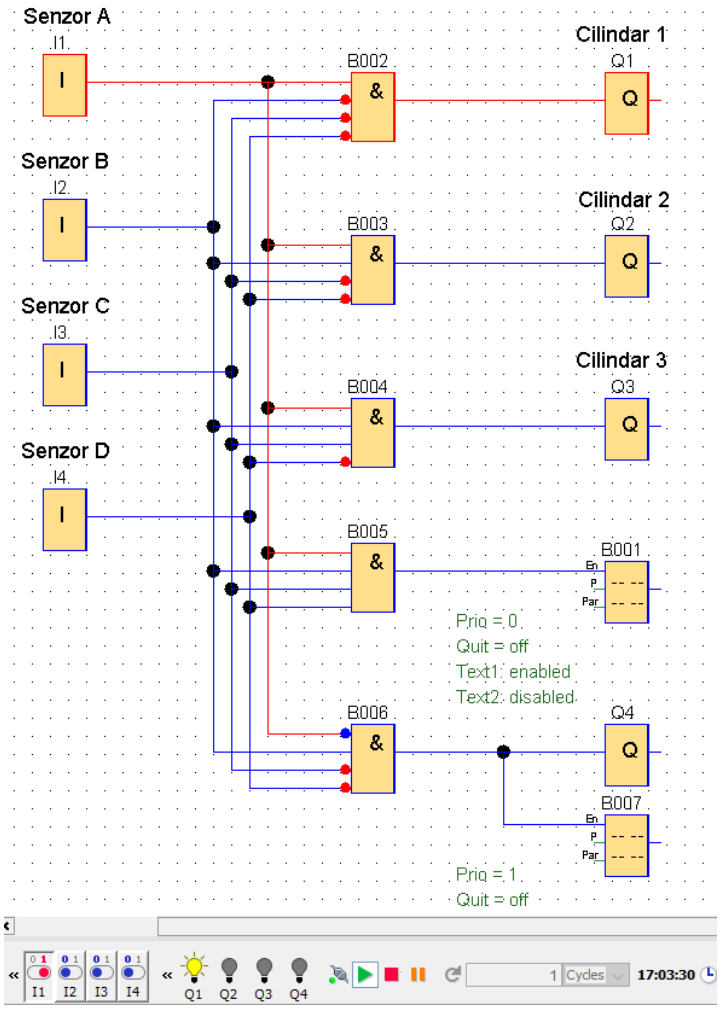
Slika 12. Poruka o nepravilnom obliku paketa

Na slici 12. je prikazano korištenje funkcije za prikaz poruka. Ovdje se radi o paketu nepravilnog oblika što se može vidjeti iz kombinacije aktivnih ulaza. Ulazi I1, I3 i I4 nisu aktivni dok je ulaz I2 aktivan. Paket odlazi na razvrstavanje.



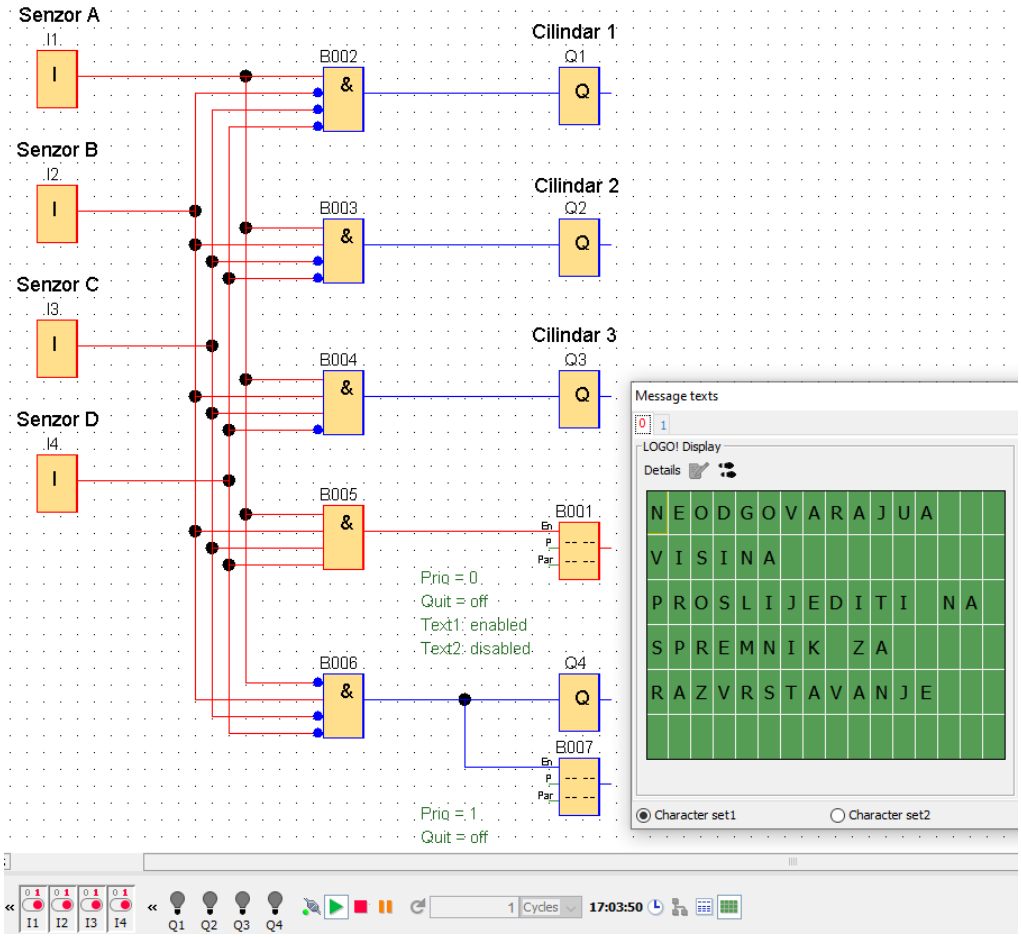
Slika 13. Prikaz aktivacije cilindra 2, slučaj 1

Na slici iznad je prikaz slučaja kada su ulazi I1 i I2 aktivni, a ulazi I3 i I4 nisu aktivni. Takva kombinacija ulaza zadovoljava uvjete aktivacije izlaza Q2 odnosno aktivaciju Cilindra 2 koji će izbaciti paket u spremnik 2.



Slika 14. Prikaz aktivacije cilindra 1, slučaj 2

Na slici iznad je prikaz kada je aktivan samo ulaz I1 dok ostali ulazi (I2, I3 i I4) nisu aktivni. Takva kombinacija stanja na ulazima zadovoljava uvjete aktivacije Q1 odnosno cilindra 1 koji će izbaciti paket u spremnik 1.

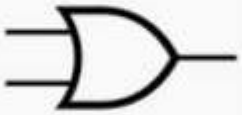
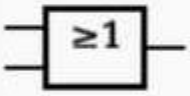
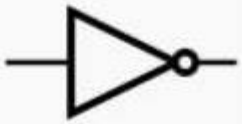


Slika 15. Poruka o neodgovarajućoj visini paketa

U primjeru sa slike 15. radi se o predmetu neodgovarajuće visine koji će nastaviti putovati po traci do kraja trake gdje će pasti u spremnik 4 te ići na posebno razvrstavanje, a pritom umjesto aktivacije nekog od cilindara će biti poruka na zaslonu koja će objasniti što se događa i gdje predmet odlazi.

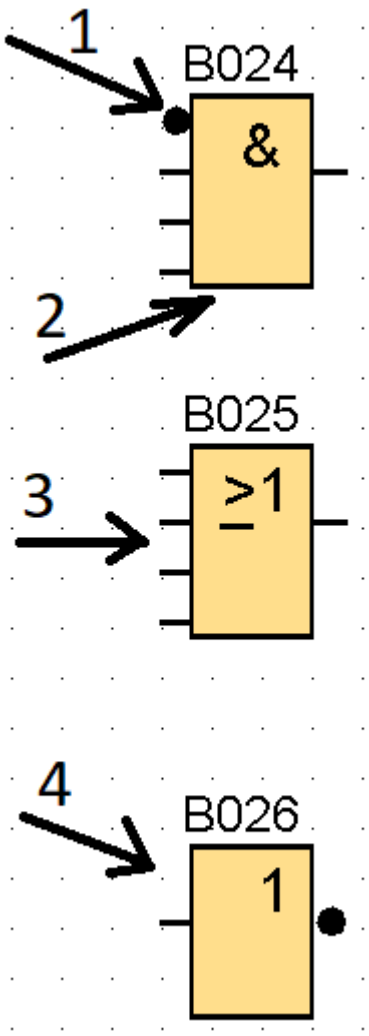
3.1 Tipovi korištenih funkcija

U završnom radu koriste se različite funkcije poput brojača, ispisa poruka na zaslon te funkcija koje služe za resetiranje pojedinih elemenata gdje nije dopuštena rekurzija odnosno da funkcija sama sebe resetira. Što se tiče logičkih funkcija koristit će se funkcije AND (I), OR (ILI) i logička funkcija NOT (NE) s time da će se logička funkcija NOT koristiti kao invertirani ulaz kod funkcije AND (I).

Operacija	Simbol (ANSI)	Simbol (IEC)	Booleov izraz	Tablica istine
I (AND)			$A \cdot B$	ULAZ: A, B — IZLAZ: A AND B 0 0 — 0 0 1 — 0 1 0 — 0 1 1 — 1
ILI (OR)			$A + B$	ULAZ: A, B — IZLAZ: A OR B 0 0 — 0 0 1 — 1 1 0 — 1 1 1 — 1
NE (NOT)			$\overline{A}$	ULAZ: A — IZLAZ: NOT A 0 — 1 1 — 0

Slika 16. Logičke funkcije [4]

Na slici 16. prikazane su 3 osnovne logičke funkcije, I, ILI, NE, njihovi simboli i tablice istine. Tablica istine je prikaz kombinacija ulaza koje daju određeni signal na izlazu. To će odrediti kakav će biti izlaz iz programa nakon što se obrade signali s ulaza logičkih funkcija. [4] Na ulazima logičkih funkcija će biti signali koje šalju ulazi PLC-a I1, I2, I3 i I4 koji su prethodno opisani.

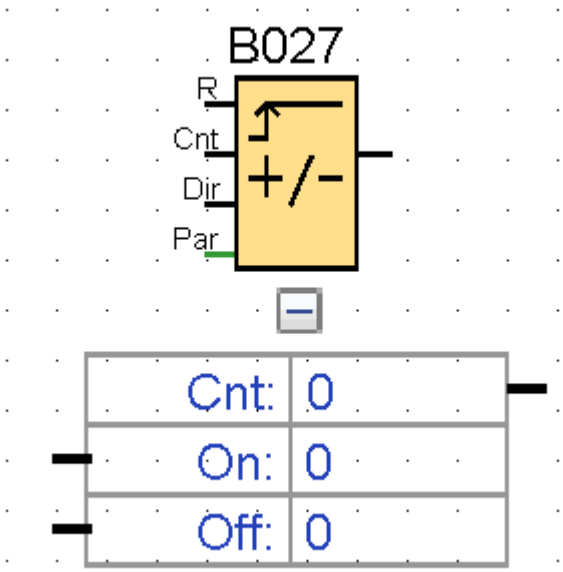


Slika 17. Logičke funkcije I, ILI, NE

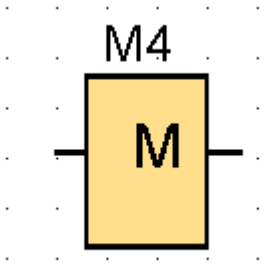
Na slici 17. broj 1 je oznaka invertiranog ulaza kod AND funkcije, a to je kraći prikaz NOT funkcije bez potrebe stavljanja dodatnog elementa NE ispred funkcije. Dvostrukim klikom lijeve tipke miša na ulaz AND funkcije dobiven je invertirani ulaz, a u ovom slučaju može ih biti maksimalno 4. Strelica 2 je funkcija AND. Strelica 3 je funkcija OR, a strelica 4

je funkcija NOT. Najčešće se ne stavlja u tom obliku osim ako se želi neki dio posebno istaknuti.

Brojač je jedna od funkcija koja služi za brojanje signala koji dolaze s nekog elementa. Bilo da se radilo o izlazima, logičkim funkcijama ili ulazima. Svaki brojač će imati pin koji će služiti za primanje signala koji se broji (Cnt) pin za resetiranja trenutnog stanja brojača (R) te pin za promjenu smjera brojanja (Dir). Pinom za promjenu smjera brojanja može se smanjiti trenutno stanje na brojaču. Zadnji pin je parameter koji služi za detaljnije upravljanje signalima iz sklopa za brojanje. Na On i Off pinove unutar parametara mogu se spojiti analogni izlazi iz drugih funkcija kao što su komparator, pojačalo, multipleksor i slično. Analognim izlazom Cnt unutar parametara može se spojiti na drugu funkciju koja prima analogni ulaz. [5] Na slici 18. prikazano je kako izgleda funkcija brojača u programu LOGO!Soft Comfort.

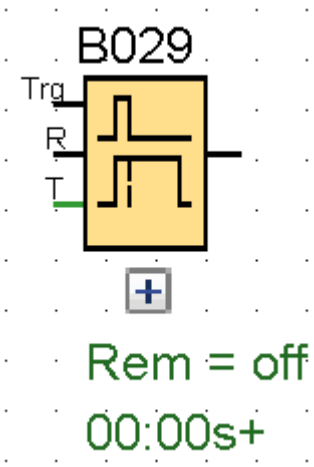


Slika 18. Prikaz funkcije brojača



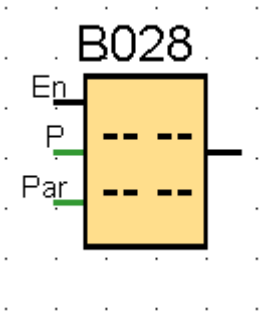
Slika 19. Prikaz funkcije flag

Na slici iznad je prikazana funkcija Flag koja omogućava resetiranje brojača tamo gdje nije moguće izvršiti resetiranje rekurzivnim putem. Prva funkcija Flag koja će se dodati u program počinje oznakom M8. Sljedeća funkcija Flag koja se doda u program imat će oznaku M1. Svakim sljedećim dodavanjem funkcije Flag u program broj oznake će se uvećati za 1 što će značiti da će treća funkcija Flag dodana u program imati oznaku M2. Četvrta funkcija Flag će imati oznaku M3. Najveći broj koji se može dodijeliti funkciji Flag je 64. [6]



Slika 20. Prikaz funkcije Off-Delay

Na slici 20. prikazana je funkcija Off-Delay. Ta se funkcija koristi na mjestu gdje je potrebno nešto isključiti nakon određenog vremena. Pin za aktivaciju odbrojavanja do isključivanja je označen s Trg na funkcijskom bloku B029 na slici iznad. [7] U prezentiranom slučaju koristit će se kako bi se izvršila zamjena punih spremnika u trajanju od 5 sekundi.

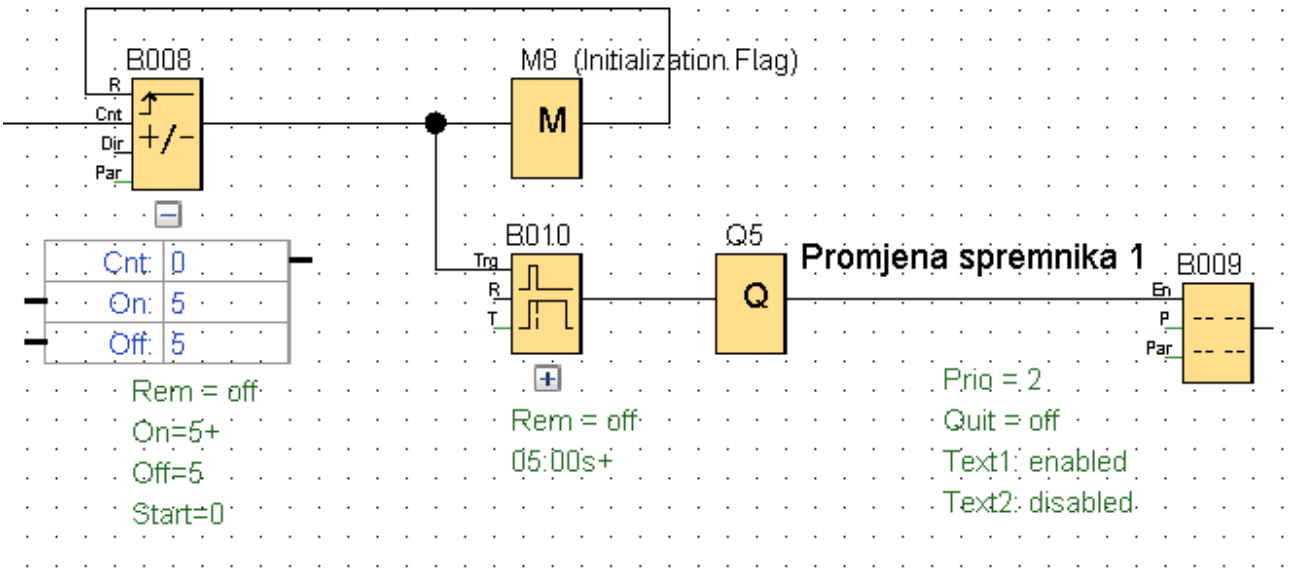


Slika 21. Prikaz funkcije Message Texts

Na slici 21. je prikaz funkcije Message texts. Ta se funkcija koristi za prikaz podatka ili poruke na zaslonu. Pin En na funkcijskom bloku B028 na slici iznad služi za aktivaciju prikaza željene poruke. U fizičkom obliku poruka može završiti na PLC-u koji ima ugrađeni zaslon na sebi. Postoje i moduli za nadogradnju koji služe za prikazivanje poruke. [8]

3.2 Brojanje predmeta u spremniku

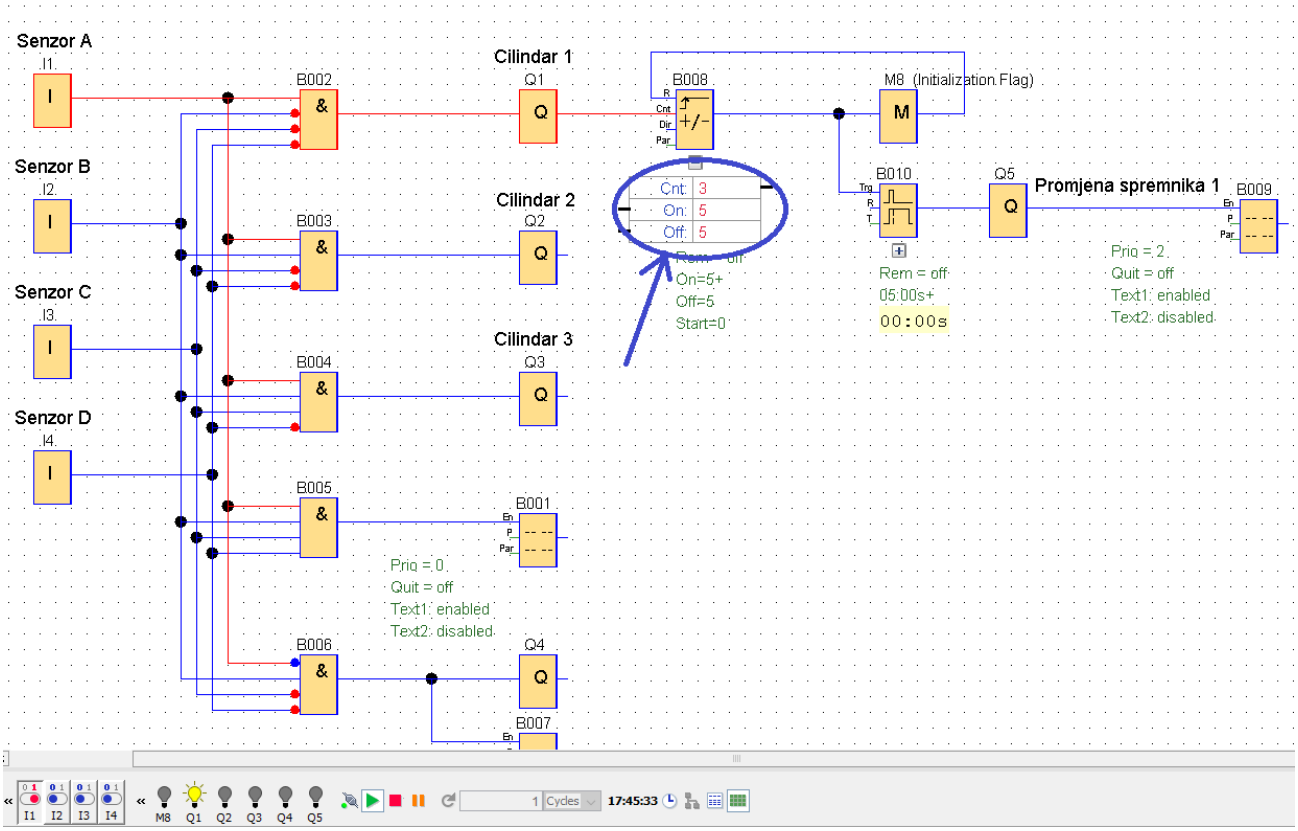
U prezentiranom primjeru na raspolaganju su 4 spremnika za pakete. U stvarnosti postoji potreba za praćenjem koliko je predmeta u pojedinom spremniku. Ako je spremnik pun i nema više mjesta, svaki sljedeći paket će završiti izvan spremnika. Postoji mogućnost da će se paket oštetiti ili da će se usporiti rad. Stoga treba paziti kakvo je trenutno stanje i koja je gornja granica koja se ne smije prijeći što se tiče broja paketa u spremnicima. Za to će se koristiti brojač odnosno funkcija koja će zabilježiti svaku aktivaciju cilindra. Podatak o tome koliko se puta cilindar aktivirao pohranit će se u memoriju. Svakom sljedećom aktivacijom odgovarajućeg cilindra prethodno stanje će se uvećati. Kada brojač dosegne vrijednost 5 paketa u spremniku isti će se resetirati preko funkcije Flag. Nakon dostignuća vrijednosti 5, vrši se zamjena spremnika koja traje određeno vrijeme. U prezentiranom primjeru zadano je vrijeme od 5 sekundi zbog simulacije.



Slika 22. Brojač paketa u spremniku

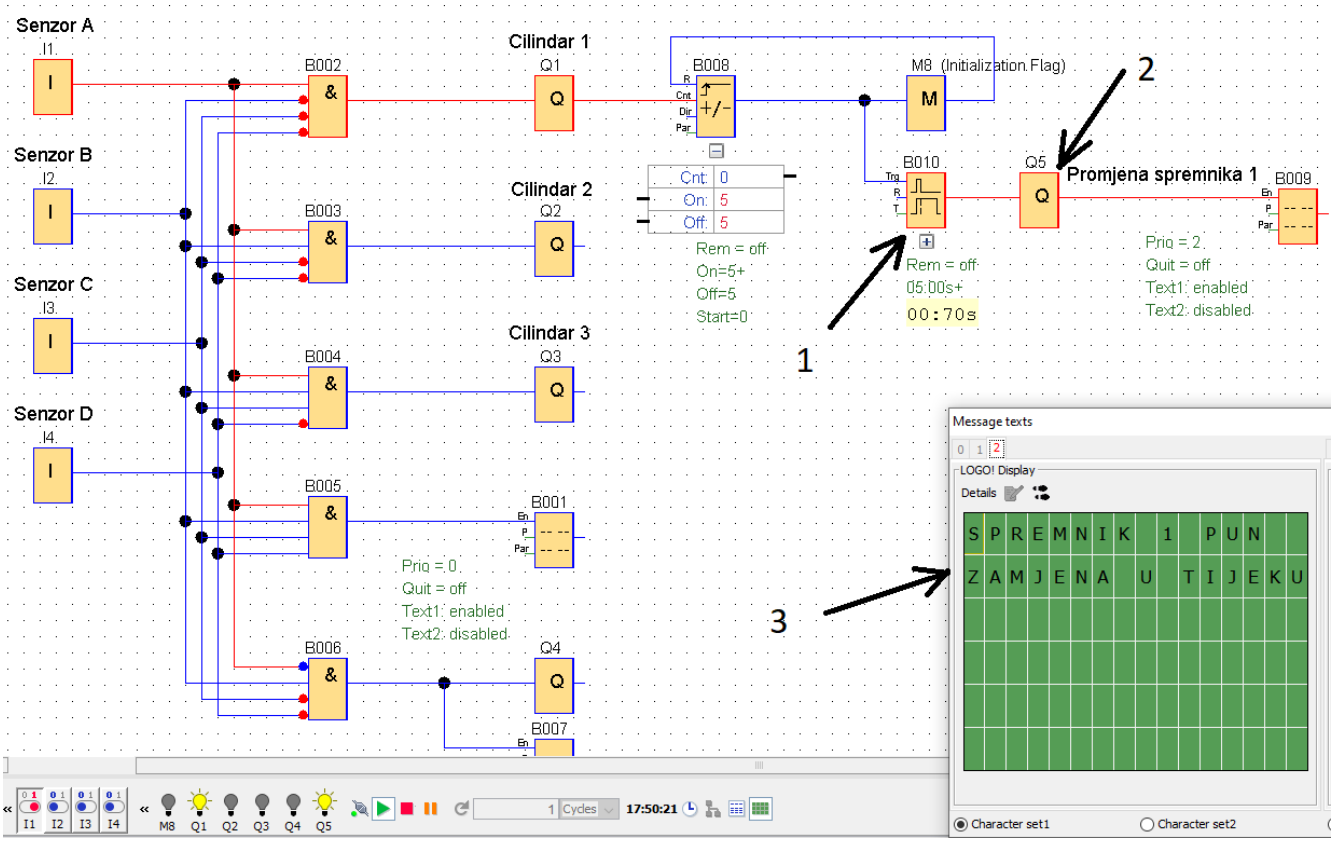
Na slici 22. prikazan je dio programa koji služi za brojanje paketa unutar svakog od spremnika. Prilikom aktivacije određenog cilindra funkcijski blok B008 (brojač) će uvećati svoje trenutno stanje. U trenutku kada ta vrijednost dostigne broj 5, na izlazu brojača će se pojaviti signal koji će preko funkcije Flag (funkcijski blok M8) resetirati brojač i postaviti

njegovu vrijednost na 0. To će omogućiti ponovno brojanje za sljedeći spremnik koji će doći na mjesto prethodno napunjenog. Također pri aktivaciji na izlazu brojača doći će do aktivacije funkcijskog bloka B010 i sustava za promjenu spremnika. Blok B010 ostat će aktivan 5 sekundi što označava potrebno vrijeme za zamjenu spremnika.



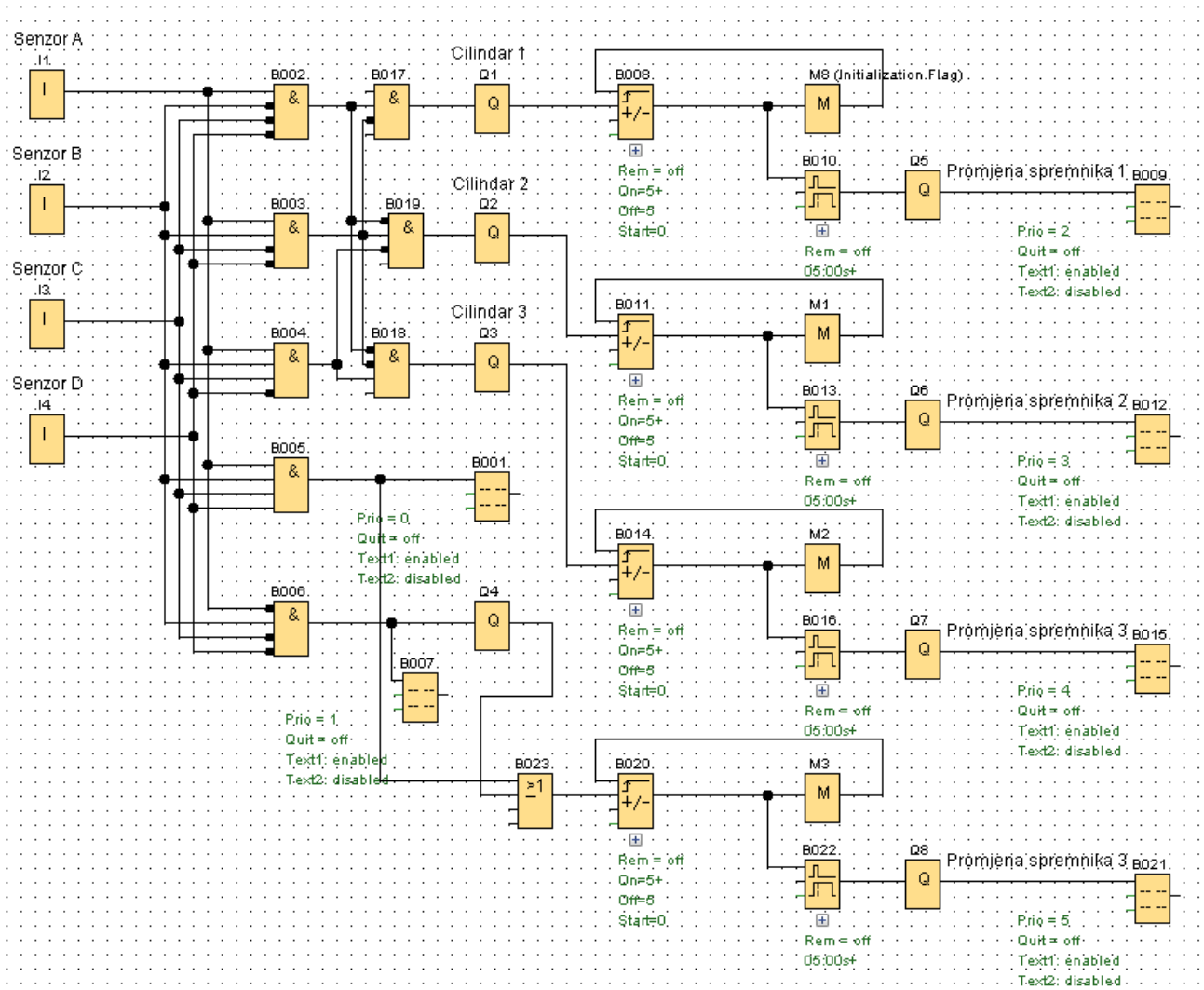
Slika 23. Trenutno stanje brojača

Slika 23. prikazuje gdje se očitava trenutno stanje brojača te koji su uvjeti za uključivanje odnosno isključivanje tj. resetiranje brojača. Na slici iznad označeni dio prikazuje otvorene parameter funkcije brojača. Ulazi s lijeve strane parametara su analogni. Izlaz s desne strane parametara je analogni.



Slika 24. Prikaz aktivacije mehanizma za promjenu spremnika

Na slici 24. je prikazana aktivacija mehanizma za promjenu spremnika. Broj 1 na slici iznad označava funkcijski blok koji služi za brojanje određenog vremena koje je potrebno za zamjenu spremnika. Broj 2 na slici iznad označava mehanizam koji vrši promjenu spremnika, a broj 3 na slici iznad je poruka koja je prikazana na zaslonu koja prikazuje informaciju o trenutnom zbivanju u procesu.



Slika 25. Izgled finalne verzije programa

Na slici 25. je prikaz gotovog programa za prepoznavanje i razvrstavanje predmeta po visini. Program se može nadograđivati uzimajući u obzir fizičke mogućnosti samog PLC-a i njegovih modula za proširenje. Mogu se dodati senzori za mjerenje težine, širine predmeta, ukupnog broja paketa koji su prošli razvrstavanje na traci, vrijeme koje je bilo potrebno za određeni broj predmeta u smjeni da se razvrsta te radni broj sati pojedinog cilindra ili drugog uređaja koji sudjeluje u proizvodnom procesu.

4. PRAKTIČNI DIO ZAVRŠNOG RADA

Što se tiče praktičnog dijela završnog rada koristit će se Arduino Uno R3 mikrokontroler, 3 infracrvena Break Beam senzora, LCD (engl. Liquid Crystal Display – hrv. zaslon temeljen na tekućim kristalima) 16x2 zaslon, testna pločica te ostale komponente koje će biti navedene i opisane u nastavku ovog rada.

Za simulaciju detektiranja visine paketa bit će potrebno napisati program za Arduino, prebaciti ga na mikrokontroler, napraviti shemu spajanja komponenata u funkcionalni sustav, razraditi uvjete po kojima će se izraditi simulacija i navesti što je sve potrebno od ulaza i izlaza za prikazati istu.

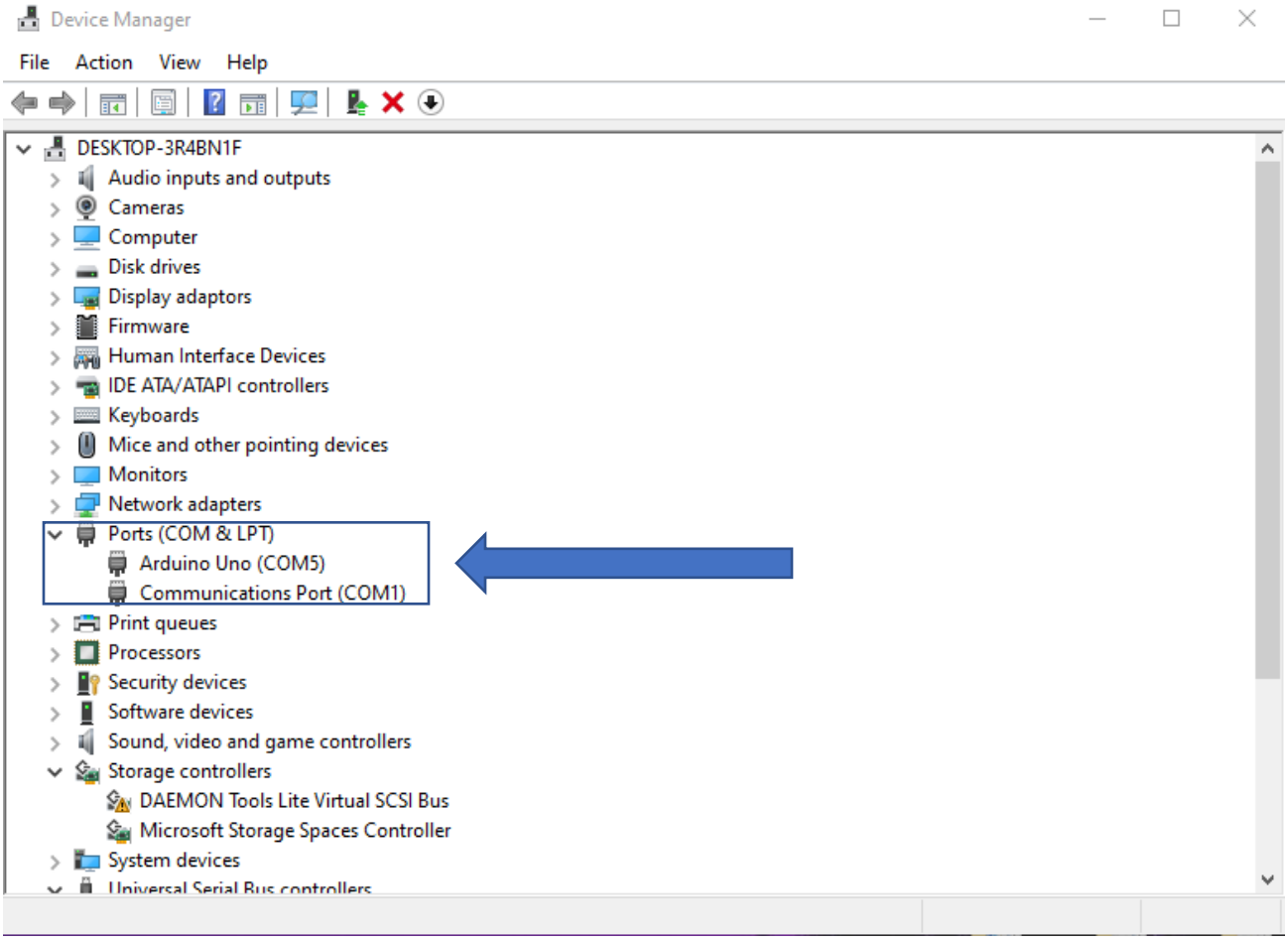
Trenutno stanje rada sustava će se pratiti na LCD 16x2 zaslonu. Nekoliko različitih slučajeva detekcije će biti snimljene u obliku videozapisa i fotografirane te će biti priložene na nekom od uređaja za pohranu podataka.

4.1 Povezivanje mikrokontrolera s računalom

Da bi se povezali mikrokontroler Arduino i računalo potreban je USB kabel B tipa koji je prikazan na slici 26. Preko USB kabela B tipa se napaja Arduino mikrokontroler i vrši se prijenos napisanog koda. Izvršavanje provjere konekcije će se vršiti na operativnom sustavu MS Windows 10. Potrebno je odabrati Device Manager te u prozoru koji se otvori pronaći Ports (COM & LPT). Pod dijelom Ports je potrebno pronaći Arduino Uno (COM5) kao što je prikazano na slici 27. [27]

U nekim slučajevima broj nakon COM može biti različit, sve zavisi o tome koji je port slobodan i kojemu je dopušteno korištenje. Ako je uspješno detektiran komunikacijski priključak (port) znači da je veza uspostavljena i da je uređaj spreman za komunikaciju s računalom odnosno s programom u kojemu se kreira kod.

Slika 26. USB B kabel [9]

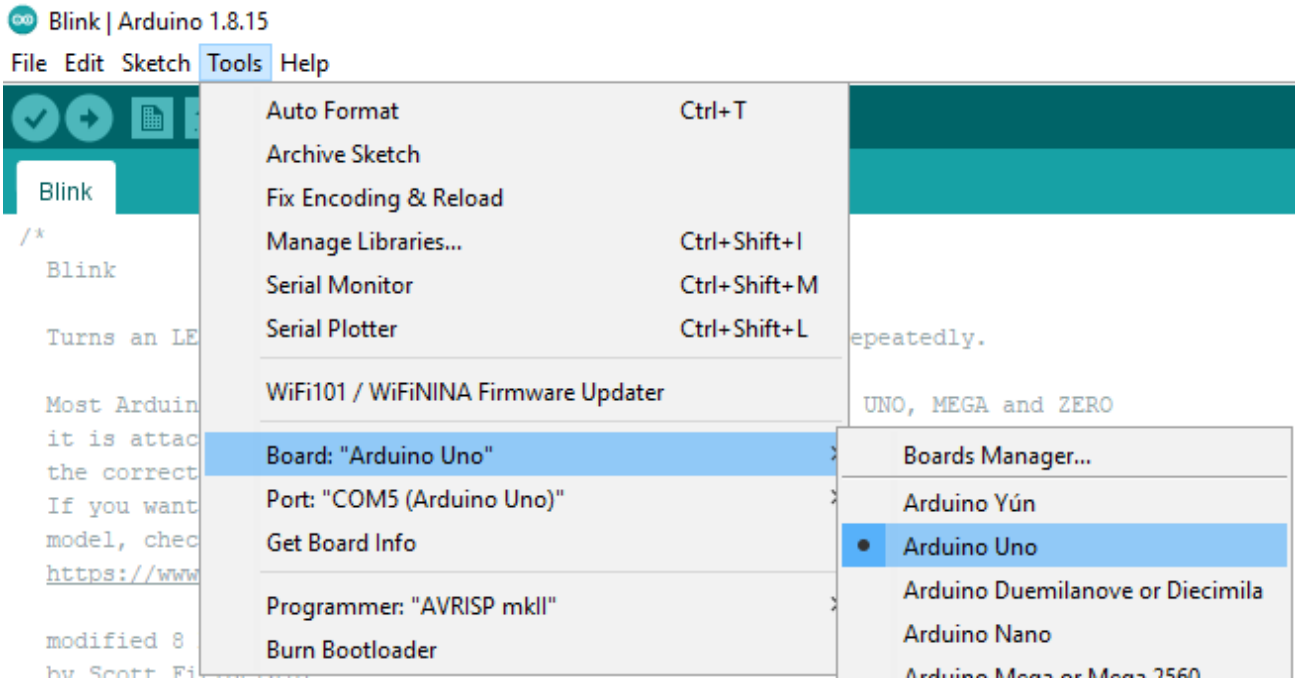


Slika 27. Provjera uspješnog povezivanja s mikrokontrolerom

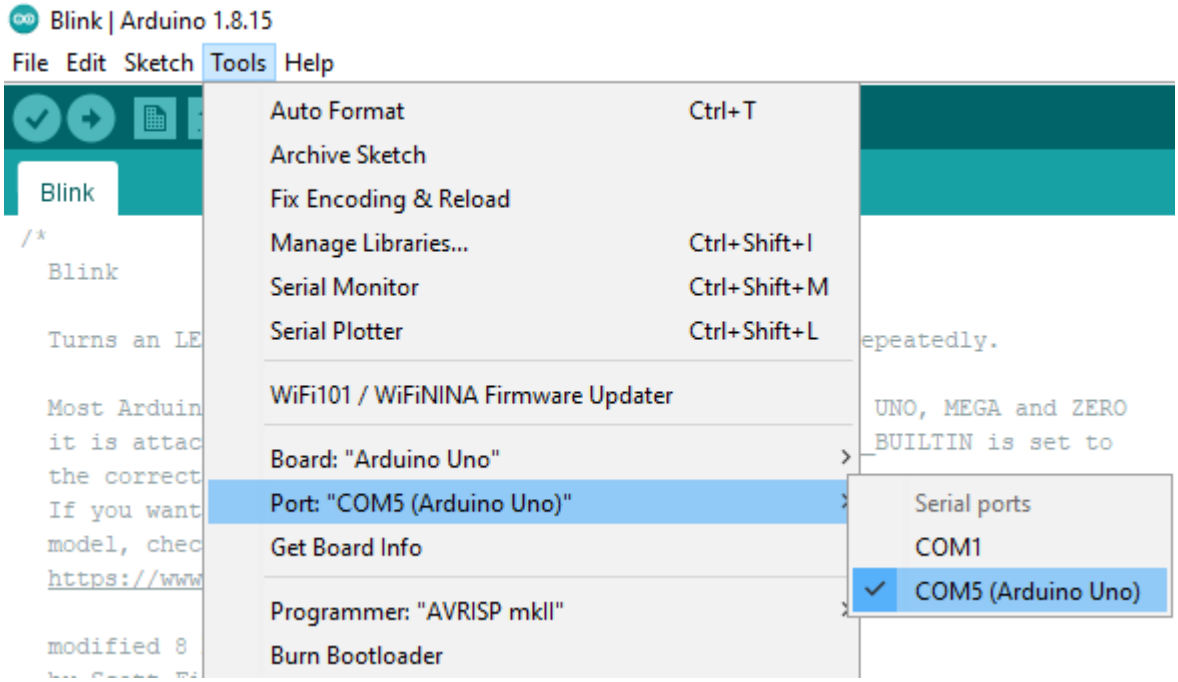
4.2 Software za rad s Arduinom

Da bi se mogla vršiti komunikacija s Arduinom potrebna je aplikacija za pisanje i provjeru programskog koda koja će nakon provjere prebaciti kod na Arduino mikrokontroler po kojemu će isti i raditi. Ime programa je Arduino IDE 1.8.15, a može se preuzeti s Arduino službene web stranice. [21] Prilikom prvog korištenja Arduino mikrokontrolera potrebno je pokrenuti testni program naziva Blink. Ako je sve u redu s mikrokontrolerom i ako funkcionira konekcija između programa i Arduina, lampica L će se upaliti, ostati upaljena 1 sekundu te će se nakon toga ugasiti i ostati ugašena 1 sekundu. [28]

Ovaj proces nema određeno vrijeme izvođenja te može trajati koliko god je potrebno i može se prekinuti bilo kada zato što se dio programa koji pali i gasi lampicu odvija u void loop() dijelu programa koji se beskonačno ponavlja. Prije samog prijenosa potrebno je odabrati pločicu mikrokontrolera koja se koristi, kao što je prikazano na slici 28. Nakon odabira pločice potrebno je provjeriti odnosno odabrati komunikacijski port na koji će se vršiti slanje programa, kao što je prikazano na slici 29. [28]

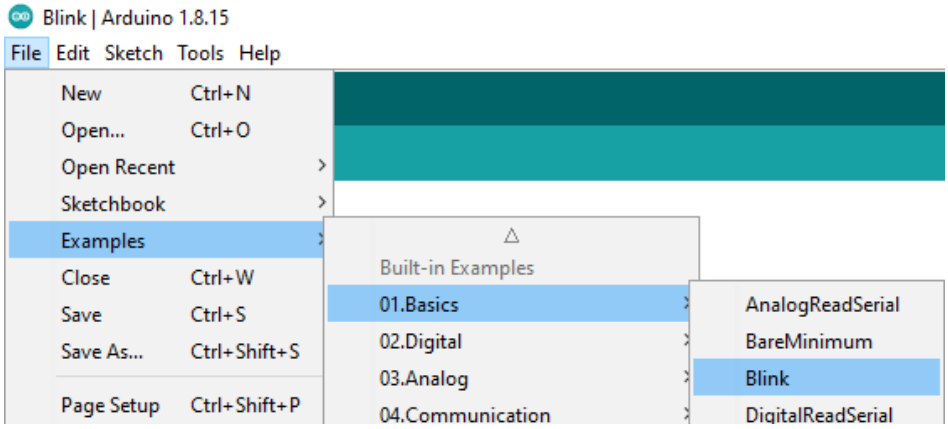


Slika 28. Odabir pločice mikrokontrolera



Slika 29. Testiranje veze između računala i mikrokontrolera

Nakon provjere ispravnosti mikrokontrolera i njegove povezanosti s računalom slijedi odabir programa i testiranje.

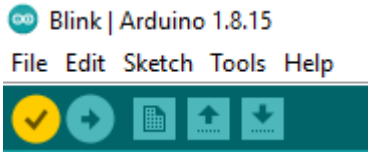


Slika 30. Odabir Blink testa u aplikaciji

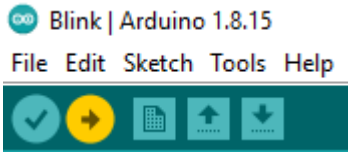
```
// the setup function runs once when you press reset or power the board
void setup() {
  // initialize digital pin LED_BUILTIN as an output.
  pinMode(LED_BUILTIN, OUTPUT);
}

// the loop function runs over and over again forever
void loop() {
  digitalWrite(LED_BUILTIN, HIGH);   // turn the LED on (HIGH is the voltage level)
  delay(1000);                       // wait for a second
  digitalWrite(LED_BUILTIN, LOW);    // turn the LED off by making the voltage LOW
  delay(1000);                       // wait for a second
}
```

Slika 31. Generirani Blink kod

Na slici 30. je prikaz kako generirati kod za Blink test, a slika 31. je kod Blink testa koji će se u sljedećem koraku prebaciti na Arduino. Nakon odabranog test primjera pokreće se provjera, kao što je prikazano na slici 32. Ako je komunikacija uspješno uspostavljena, vrši se slanje programa na mikrokontroler, kao što je prikazano na slici 33.

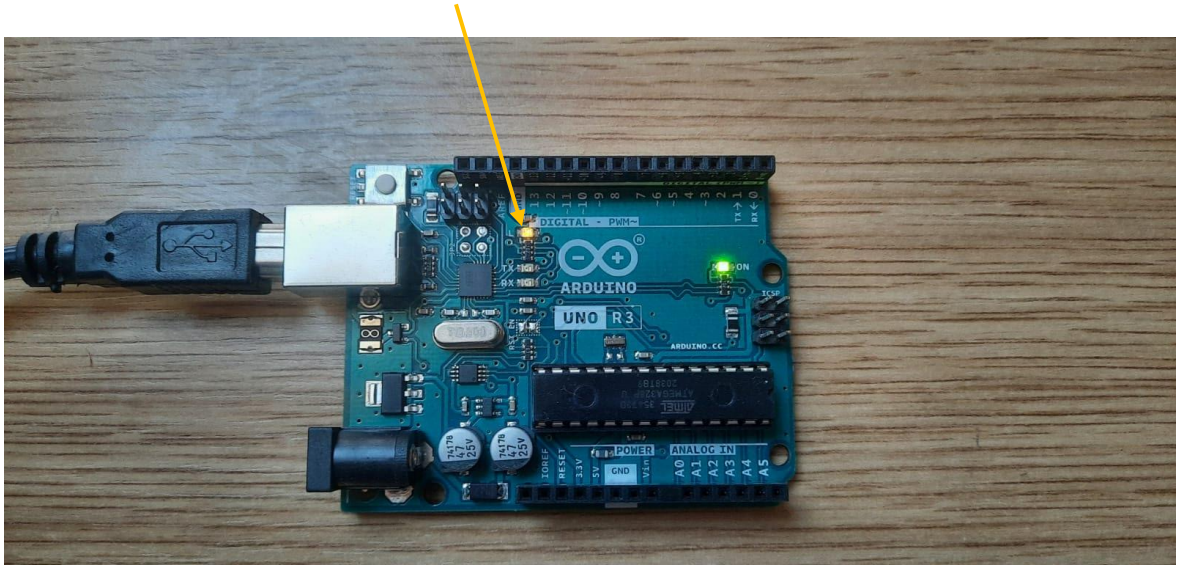


Slika 32. Prikaz opcije za verifikaciju programa

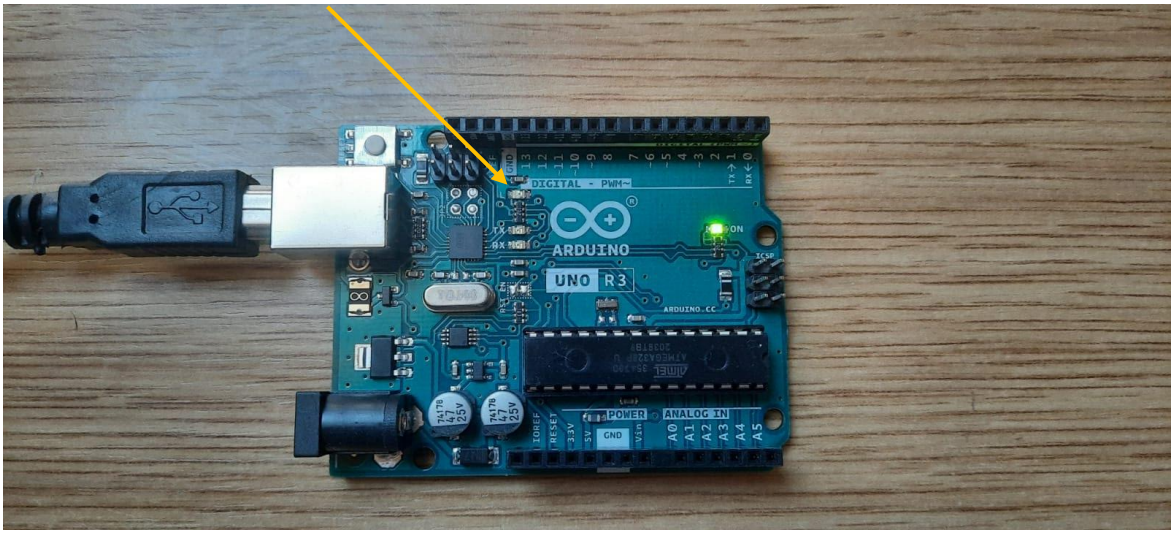


Slika 33. Slanje programa na mikrokontroler

Ako je program ispravan i uspješno prebačen na mikrokontroler, lampica će se upaliti i ugasiti svake sekunde. Na slici 34. je prikazan trenutak u kojemu je lampica upaljena.



Slika 34. Prikaz upaljene lampice na mikrokontroleru



Slika 35. Prikaz ugašene lampice na mikrokontroleru

Nakon uspješne provjere ispravnosti uređaja i komunikacije sa softwareom može se početi s izradom programa koji će detektirati predmet koji prolazi kroz svjetlosni snop infracrvenih senzora i spojiti komponente u funkcionalni sustav.

4.3 Izrada programa za detektiranje predmeta pomoću IR Break Beam senzora

Prije samog početka pisanja koda potrebno je vidjeti što je sve od komponenata na raspolaganju. Kako svaka od komponenata funkcionira. Ako postoji potreba za uključivanjem neke od biblioteka potrebno ju je uključiti kako bi komponente mogle raditi te kako bi se mogle koristiti specifične funkcije namijenjene za određene komponente (kao što je LCD zaslon). Na kraju je potrebno odrediti uvjete po kojima će program raditi nakon što primi signale od senzora.

4.3.1 LCD zaslon 16x2

Jedna od komponenti koja će služiti za praćenje trenutnog stanja sustava će biti LCD zaslon. Na njemu je moguće prikazati trenutnu vrijednost na nekom senzoru ili proizvoljnu poruku u obliku teksta ili znakova. Sastoji se od 16 mjesta za znakove i ima 2 reda. Prikazivati se na zaslonu mogu slova, brojevi i znakovi. Svaki znak se sastoji od polja **5x8** piksela kojeg je moguće proizvoljno zadati. [10]

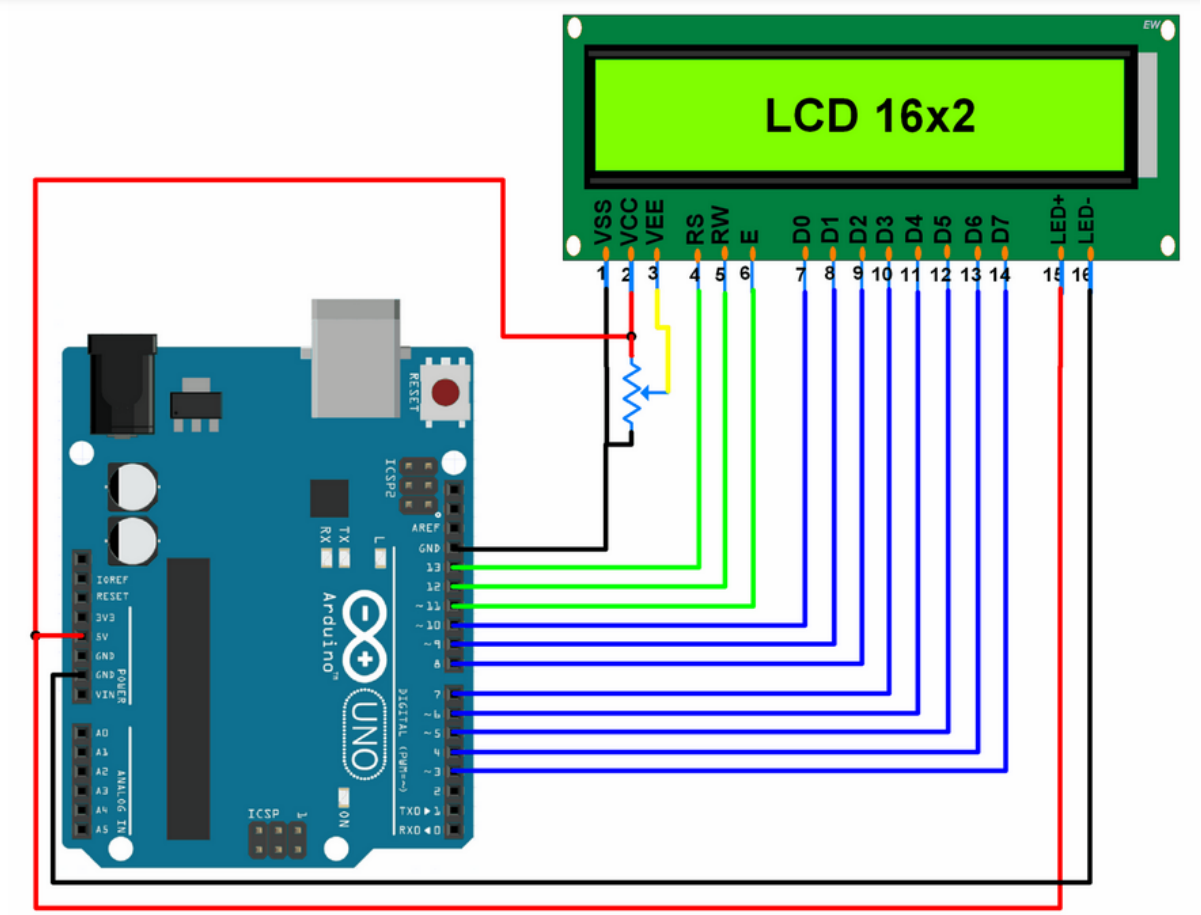
Da bi se zaslon mogao koristiti potrebno je unutar koda uključiti biblioteku LiquidCrystal.h. To je moguće postići na sljedeći način: **#include <LiquidCrystal.h>**. Nakon što je uključena biblioteka potrebno je stvoriti objekt za zaslon. Odmah ispod biblioteke piše se **LiquidCrystal lcd(12, 10, 8, 7, 6, 5)**; . Brojevi koji se nalaze unutar zagrade određuju koji se pinovi rezerviraju za komuniciranje Arduino mikrokontrolera i LCD zaslona. Broj 12 označava na koji pin na Arduino mikrokontroleru je spojen pin reset s LCD zaslona. Broj 10 je pin koji omogućava korištenje LCD zaslona (Enable pin), a ostala 4 pina (pin 8, pin 7, pin 6 i pin 5) su podatkovni pinovi koji su također spojeni na Arduino. [11]

Nakon toga potrebno je inicijalizirati LCD, a to se radi tako da unutar dijela programa koji se zove **void setup()** napiše funkcija **lcd.begin(16,2).** Brojevi koji se nalaze unutar zagrade su brojevi koji određuju koliko zaslon ima redaka i stupaca. [12] Postoje veći zasloni s kojima je moguće raditi, stoga treba pripaziti kod inicijalizacije.

Prilikom prikaza teksta na zaslonu treba voditi računa o tome koje je početno mjesto od kojeg će zaslon vršiti prikaz. Za to postoji funkcija **lcd.setCursor(0,0)**. (0,0) znači da se počinje od prvog stupca i prvog retka na zaslonu. [13] Prije svakog korištenja zaslona korisno

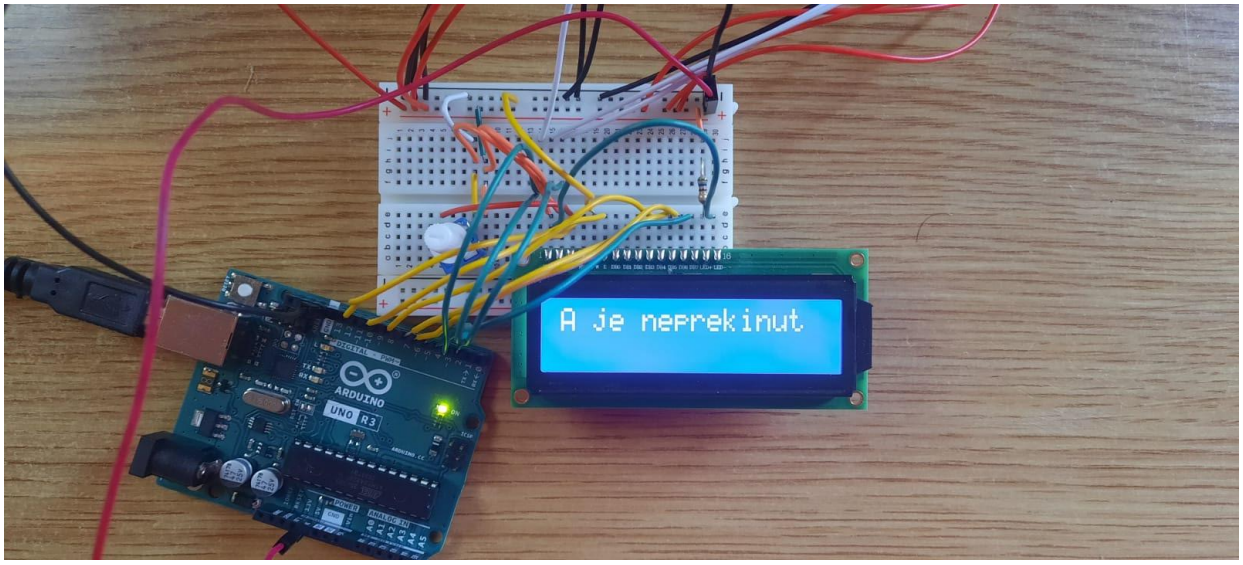
je napraviti čišćenje svega što se prethodno prikazivalo na LCD zaslonu (tekst, podatci sa senzora, prikaz znakova) da ne bi došlo do grešaka te da se nove poruke i podatci mogu prikazivati na zaslonu. Da bi se očistio zaslon potrebno je unutar dijela koda koji se zove void setup() napisati **lcd.clear()**. Nakon toga uređaj će biti spreman za korištenje. [14] Na slici 36. Je prikaz LCD zaslona koji će se koristiti u radu.

Slika 36. LCD zaslon 16x2



Slika 37. Shema spajanja LDC zaslona i Arduino mikrokontrolera [15]

Na slici 37. je prikazana shema spajanja LCD zaslona i Arduino mikrokontrolera. Na slici je spojeno svih 8 podatkovnih pinova, a za potrebe rada će se koristiti 4 podatkovna pina. Pinovi koji će biti korišteni su pin 5, pin 6, pin 7 i pin 8 što je određeno unutar funkcije **LiquidCrystal lcd(12, 10, 8, 7, 6, 5)**.

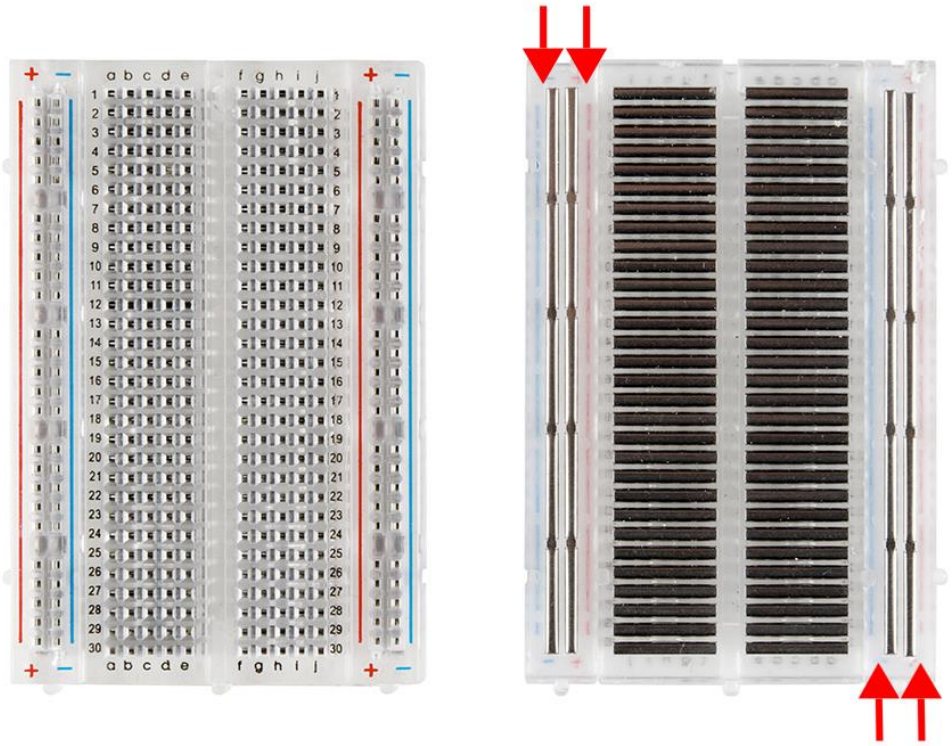


Slika 38. Prikaz spojenog LCD-a na testnu pločicu

Slika 38. prikazuje kako izgleda LCD zaslon kada je spojen na testnu pločicu. Intenzitet svjetlosti teksta na zaslonu moguće je namještati preko potenciometra. Između pina LED+ i napajanja je preporučljivo staviti otpornik zato što nemaju svi LCD-ovi ugrađene otpornike u sebi. Ako se ne bi stavio otpornik može doći do oštećenja pozadinskog osvjetljenja.

4.3.2 Testna pločica

Testna pločica je pločica na koju se spajaju elementi u funkcionalni sustav. Pločica koja se koristiti u praktičnom dijelu završnog rada je podijeljena u 4 dijela - prva 2 dijela se odnose na komponente, ostala 2 dijela su za napajanje i uzemljenje. Sastoji se od pinova tako da nema potrebe za lemljenjem, a komponente se lako spajaju i uklanjaju bez ikakvih oštećenja. Na sredini pločice je kanal koji služi za izoliranje jedne polovice pločice od druge. Razlog tome je standard za spajanje integriranih krugova koji iznosi 2.54 mm između pinova integriranih krugova. [16]



Slika 39. Unutrašnjost testne pločice [16]

Na slici 39. je prikazano kako izgleda unutrašnjost testne pločice. Unutrašnjost testne pločice sastoji se od niza metalnih trakica koje su međusobno odvojene i koje omogućavaju izradu odvojenih nezavisnih krugova od ostatka sustava. Na slici 40. je prikazano kako izgleda metalna traka kad se izvadi iz pločice. Služi za spajanje dijelova na pločici u koju se stavljaju komponente.

Slika 40. Metalna traka [16]

4.3.3 Arduino Uno R3 pločica

Arduino Uno R3 je pločica koja radi na bazi mikrokontrolera. Na sebi ima USB konektor, port za napajanje, mikrokontroler, analogne i digitalne I/O pinove (engl. Input/Output – hrv. Ulazno/Izlazni), gumb za resetiranje, TX i RX indikatore i kristalni oscilator. [17]

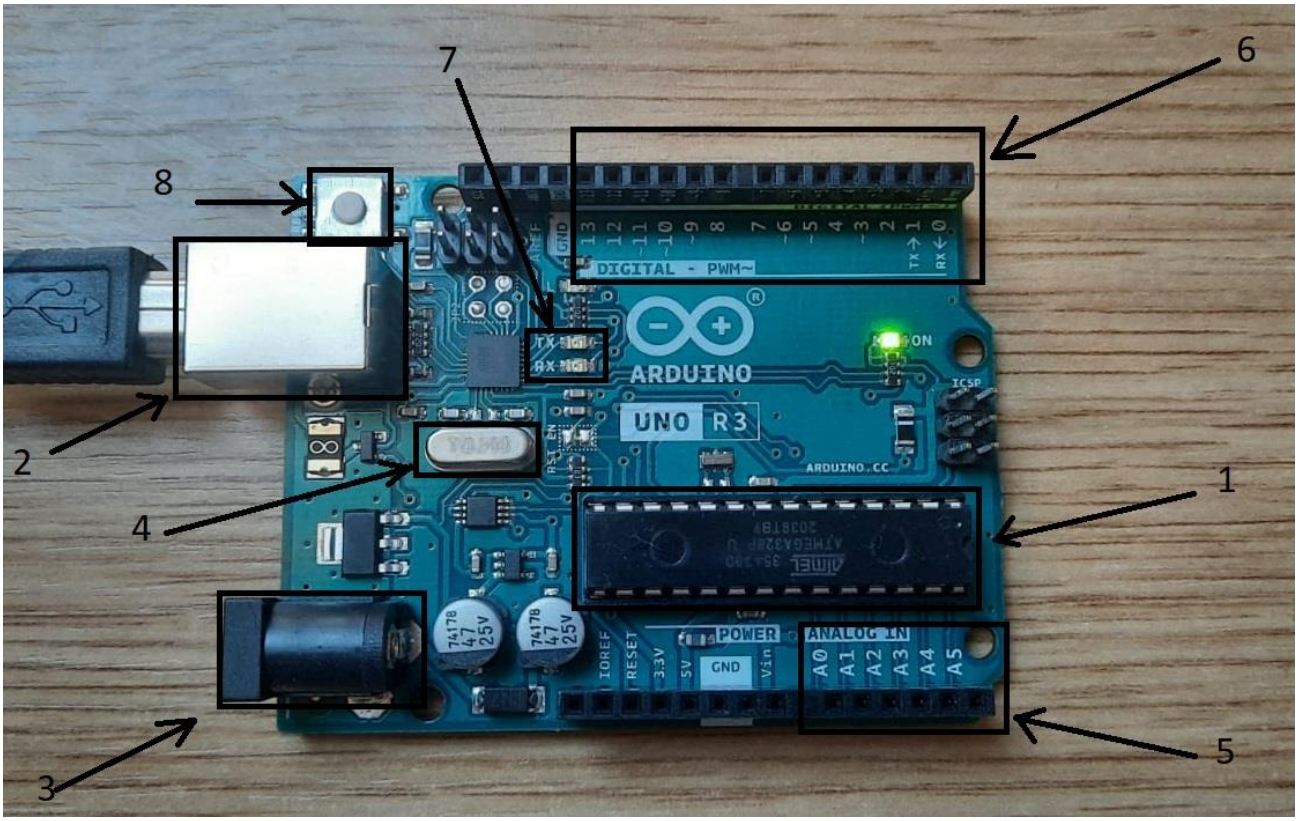
Preko USB konektora pločica se napaja i vrši se konekcija između računala i Arduino mikrokontrolera. Port za napajanje služi da se preko njega napaja pločica u slučaju da se napajanje ne vrši preko USB konektora, a napajati se može preko AC/DC ispravljača ili baterije. Maksimalno napajanje ne bi smjelo prelaziti 20 V. Ako bi napajanje prešlo 20 V postoji mogućnost oštećenja uređaja. Na pločici postoji regulator napajanja koji štiti pločicu od prevelikog napona ako je spojen veći izvor napajanja od 20 V. Preporučeno je napajanje između 7 i 12 V ako se uređaj ne napaja preko USB konektora. [18]

Mikrokontroler je mozak cijele pločice. Koristi se Atmega328P mikrokontroler. Sastoji se od flash memorije na koju se sprema program s računala. U sebi sadrži RAM od 2 kB, procesor i EEPROM od 1 kB. [19]

Pločica na sebi sadrži 6 analognih pinova. Analogni pinovi služe za čitanje signala s nekih od senzora (npr. temperaturnih) te ih pretvaraju u digitalnu vrijednost kako bi ih sustav mogao razumjeti. Na pločici su označeni s A0, A1, A2, A3, A4 i A5 (prikazano na slici 41. pod brojem 5). Analogni pinovi mogu također biti korišteni kao digitalni ulazi/izlazi. Postoji 14

digitalnih pinova (prikazano na slici 41. pod brojem 6) s time da pinovi 3, 5, 6, 9, 10 i 11 su pinovi koji mogu biti korišteni i za PWM (engl. Pulse-Width Modulation – hrv. Pulsno Širinska Modulacija) s kojim se mogu simulirati analogni izlazi. [17]

Gumb za reset može se koristiti za ponovno pokretanje programa ako se program ne ponavlja, a postoji potreba da se više puta testira program. Kristalni oscilator oscilira 16 milijuna puta u sekundi. Tijekom svakog otkucaja izvršava se jedna od operacija kao što je oduzimanje ili zbrajanje i slično. TX i RX indikatori služe za obavještavanje o prijenosu informacija između računala i Arduina. [17]



Slika 41. Dijelovi Arduino mikrokontrolera

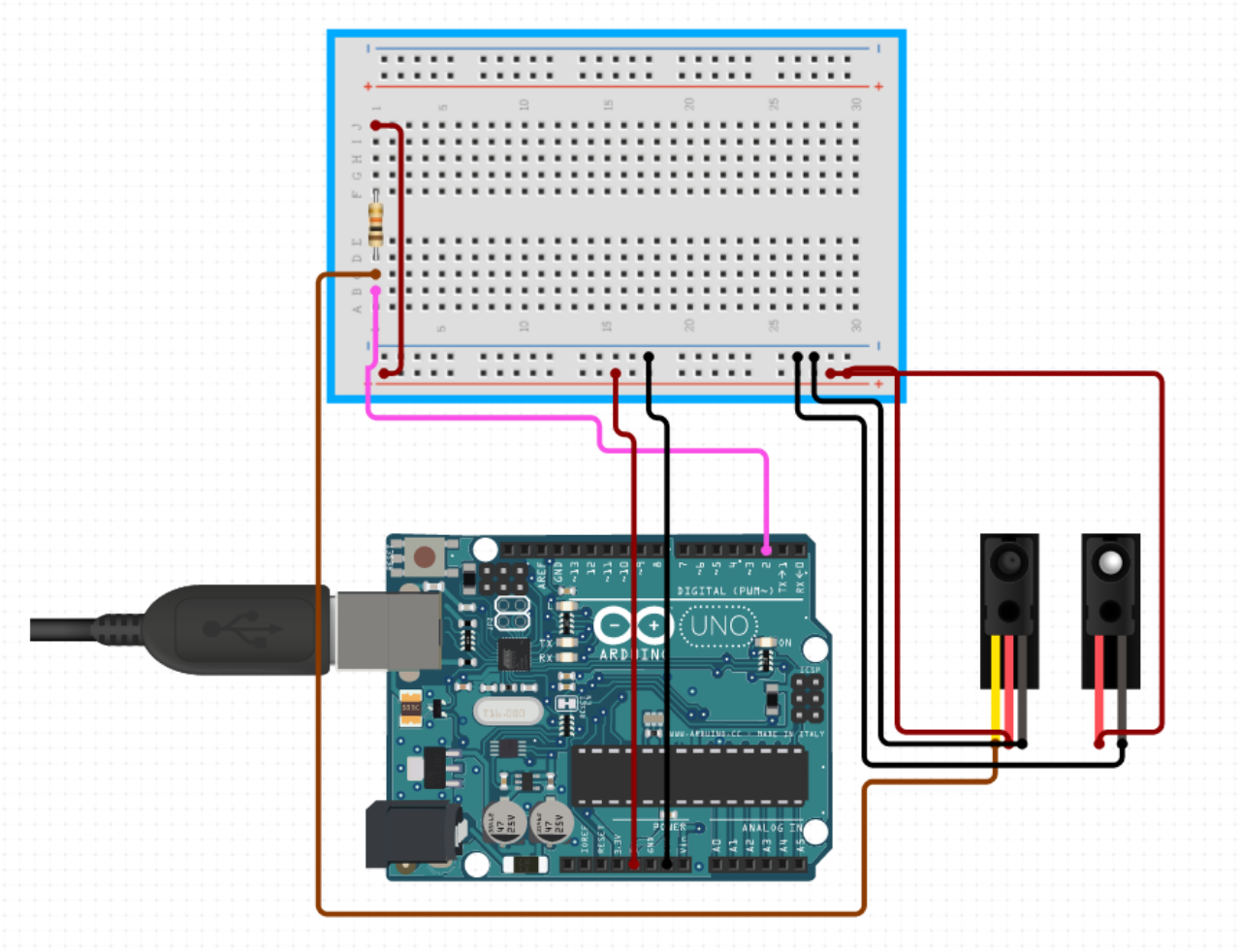
Na slici 41. je prikaz Arduino mikrokontrolera na kojemu su označeni pojedini dijelovi. Na slici iznad pod brojem 1 je označen Mikrokontroler ATmega328. Broj 2 na slici iznad prikazuje USB konektor. Pod brojem 3 je označen Port za vanjsko napajanje. Broj 4 prikazuje kristalni oscilator. Broj 5 na slici iznad prikazuje analogne ulaze. Broj 6 na slici iznad prikazuje digitalne ulaze. Broj 7 na slici iznad prikazuje TX i RX indikatore. Pod brojem 8 na slici iznad je gumb za resetiranje.

4.3.4 IR Break Beam senzori

Break Beam senzor se sastoji od dva dijela - jedan dio je predajnik, a drugi prijemnik. Iz predajnika se šalje snop svjetlosti koji je ljudskom oku nevidljiv. Taj snop odlazi do emitera ako se ništa između ta dva elementa ne nalazi. Ako se nešto nađe između ta dva elementa dolazi do prekida snopa svjetlosti što prijemnik detektira te šalje signal preko bijele žice. [1]

Slika 42. IR Break Beam senzor [19]

Na slici 42. je prikaz IR Break Beam senzora. Senzor koji ima 3 žice je prijemnik. Crvena žica se spaja na +5 V, a crna na GND. Bijela žica služi za slanje signala u trenutku kada senzor detektira promjenu. Prijemnikova crna boja služi za filtriranja svjetlosti koja nije IR spektra. Primjer takvog svjetla je dnevno svjetlo koje bi moglo ometati rad senzora u slučaju da prijemnik nema crni sloj boje na sebi. [19]

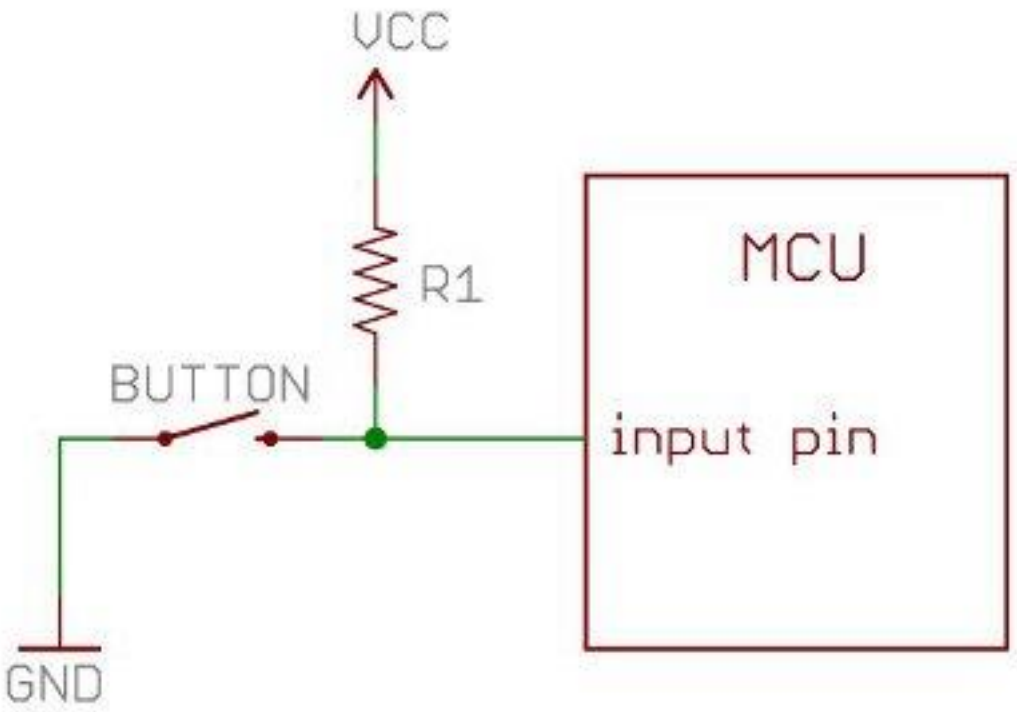


Slika 43. Shema spajanja IR Break Beam senzora

Na slici 43. je prikazano kako bi se IR Break Beam senzori trebali spojiti na testnu pločicu i Arduino mikrokontroler. U prezentiranom slučaju nema potrebe za postavljanjem otpora između žice koja detektira promjenu u svjetlosnom snopu i napajanja zato što je taj otpor već ugrađen u Arduino mikrokontroler.

Otpor se koristi kako bi se izbjeglo “plutajuće” stanje gdje nije moguće sa sigurnošću odrediti radi li se o stanju LOW (stanje 0 odnosno neaktivno stanje) ili stanju HIGH (stanje 1 odnosno aktivno stanje) pa pull-up otpornik zadržava aktivno stanje (HIGH odnosno 1) dok ne dođe do promjene. Ako su sklopka ili tipkalo postavljeni na mjesto u sustavu tako da prilikom njihovog aktiviranja propuštaju struju od napajanja (VCC) preko pull-up otpornika prema uzemljenju (GND) tada će doći do promjene iz aktivnog stanja (HIGH odnosno 1) u neaktivno stanje (LOW odnosno 0), kao što je prikazano na slici 44. [26] Pull-up otpornik je

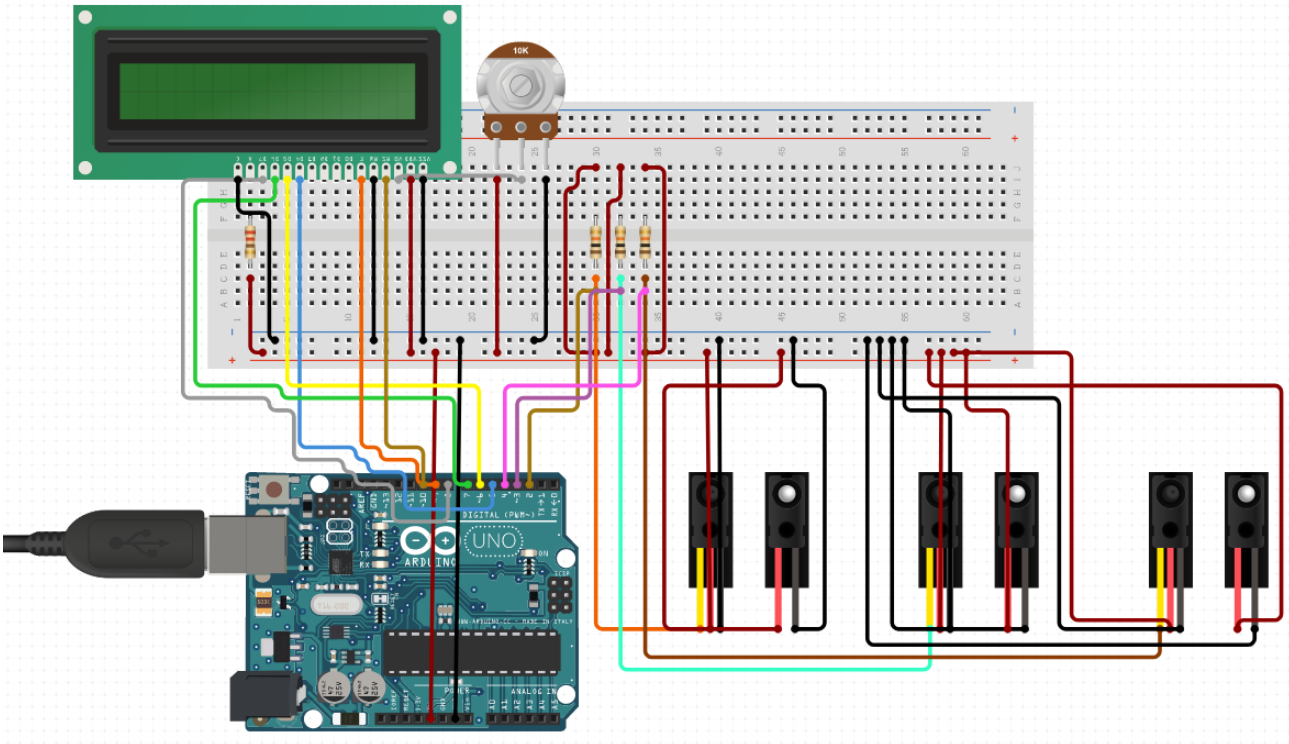
običnи otpornik koji može imati otpor između 20-50 kΩ. Spaja se između pina koji prima signale s prijemnika infracrvenog senzora i +5 V iz čega dobiva naziv pull-up (zadržava stanje HIGH odnosno aktivno stanje). Postoje još i pull-down otpornici koji se spajaju između pina i uzemljenja (GND) te zadržavaju neaktivno stanje (LOW odnosno 0). Shema spajanja IR Break Beam senzora i Arduino mikrokontrolera je izrađena na službenoj web stranici Circuit.io. [20]



Slika 44. Prikaz pull-up otpornika u sustavu [25]

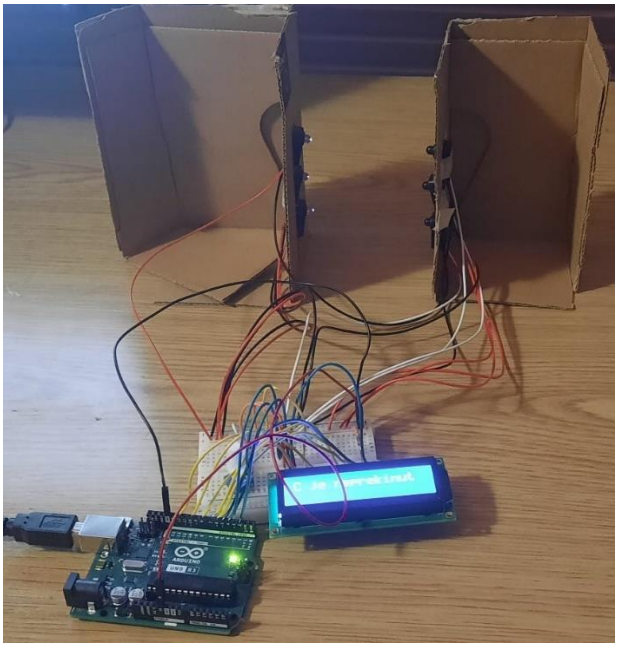
5. PROGRAM ZA DETEKCIJU VISINE PREDMETA

Kao što je u teoretskom dijelu zadatka opisano potrebno je nekoliko IR Break Beam senzora. U ovom slučaju koristit će se 3 IR Break Beam senzora zbog ograničenog prostora za spajanje na testnoj pločici i preglednosti. Cilj je detektirati na kojem dijelu je došlo do prekida snopa svjetlosti i tu informaciju prenijeti na LCD zaslon.



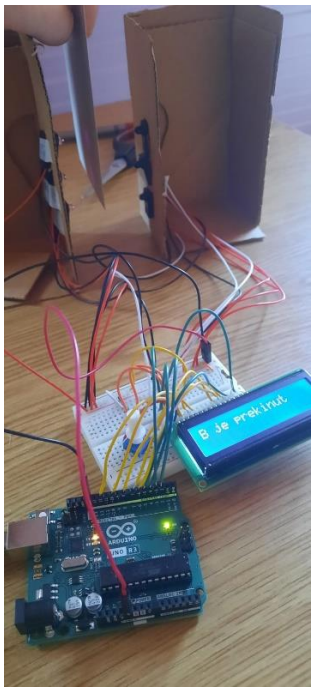
Slika 45. Shema spajanja dijelova

Na Slici 45. je prikazano kako treba spojiti pojedini dio da bi se dobila funkcionalna cjelina. Na sljedećim slikama će biti prikazano kako izgleda kada se spoje fizičke komponente u cjelinu. Shema spajanja IR Break Beam senzora i Arduino mikrokontrolera je izrađena na službenoj web stranici Circuit.io. [20]

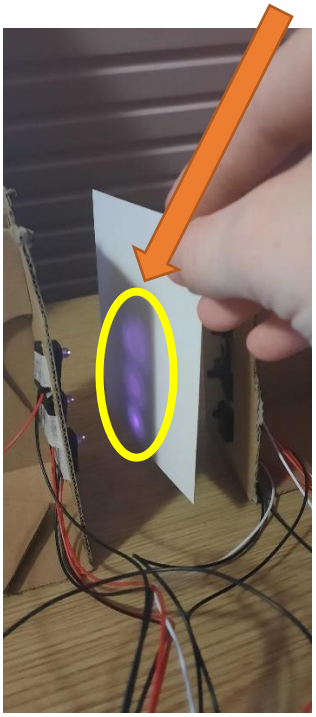


Slika 46. Izgled spojenog sustava

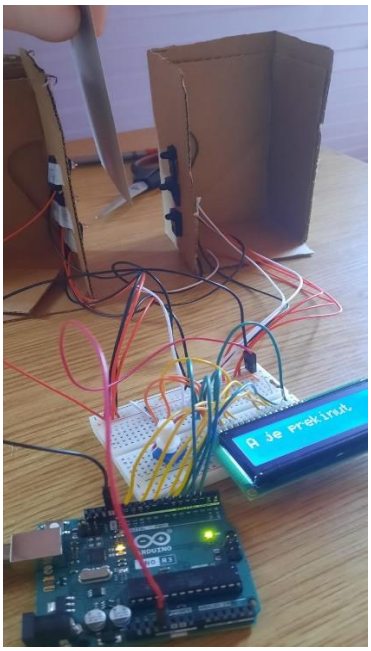
a)



b)



c)



Slika 47. Primjeri rada sustav

Na slici 46. prikazano je kako izgleda sustav kada su sve komponente spojene i kada je pokrenut program za detekciju predmeta. Sustav sa slike 46. nalazi se u stanju kada niti jedan snop svjetlosti nije prekinut što bi značilo da nema predmeta između senzora. Senzor na najvišoj poziciji je senzor koji će se u programu označavati kao senzor A. Senzor na srednjoj visinskoj poziciji je senzor koji će se u programu označavati kao senzor B, a posljednji senzor na najnižoj poziciji će imati naziv senzor C.

Na slici 47. nalaze se 3 slučaja. Slučaj a) je prikaz kada je sustav detektirao predmet s 2 senzora, a to su senzor A i senzor B. Zadnja prikazana poruka na zaslonu LCD-a je: “B je neprekinut” što znači da je sustav kao zadnju promjenu detektirao promjenu stanja na senzoru B.

Slučaj b) prikazuje kako izgledaju snopovi svjetlosti koji nisu vidljivi ljudskom oku, ali postanu vidljivi kada se fotografiraju s kamerom.

Slučaj c) prikazuje stanje sustava kada je prekinut svjetlosni snop od senzora A. Poruka na zaslonu LCD-a je: “A je neprekinut”.

5.1 KOD PROGRAMA

Kod programa po kojemu će Arduino mikrokontroler raditi je pisan u aplikaciji Arduino preuzete sa službene Arduino stranice. Aplikacija Arduino osim što služi za pisanje programa još ima i mogućnost provjere programskog koda i prebacivanja koda s aplikacije na Arduino mikrokontroler. Podatci koji se dobiju kao rezultat nakon izvršenja programa mogu se pratiti na zaslonima kao što je korišteni LCD 16x2 zaslon ili se mogu pratiti u Serial Monitoru. Serial Monitor je alat koji je sastavni dio Arduino aplikacije i služi za komuniciranje Arduino aplikacije i Arduino mikrokontrolera. Kada se pokrene, otvara se novi prozor koji će služiti za prikaz poruka koje se razmjenjuju između Arduino aplikacije i Arduino mikrokontrolera. [28] Programski jezik u kojemu se piše program je C++. U nastavku slijedi programski kod praktičnog dijela završnog rada.

```
#include <LiquidCrystal.h>

LiquidCrystal lcd(12, 10, 8, 7, 6, 5);

#define LEDPIN 13
#define SENSORPIN_A 2
#define SENSORPIN_B 3
#define SENSORPIN_C 4
//Deklaracija i inicijalizacija početnog stanja svakog od senzora
int sensorState_A = 0, lastState_A = 0;
int sensorState_B = 0, lastState_B = 0;
int sensorState_C = 0, lastState_C = 0;

void setup()
{
  lcd.begin(16, 2);
  lcd.clear();
  pinMode(LEDPIN, OUTPUT);
```

```
  pinMode(SENSORPIN_A, INPUT_PULLUP);
  pinMode(SENSORPIN_B, INPUT_PULLUP);
  pinMode(SENSORPIN_C, INPUT_PULLUP);
  Serial.begin(9600);
}

void loop()
{
  // Procitati stanje na senzorima:
  sensorState_A = digitalRead(SENSORPIN_A);
  sensorState_B = digitalRead(SENSORPIN_B);
  sensorState_C = digitalRead(SENSORPIN_C);

  // Provjera je li snop svjetlosti prekinut
  // Ako je prekinut, sensorState je LOW:
  if (sensorState_A == LOW)
  {
    // turn LED on:
    digitalWrite(LEDPIN, HIGH);
  }
  else
  {
    // iskljuciti LED:
    digitalWrite(LEDPIN, LOW);
  }
  if (sensorState_A && !lastState_A)
  {
    Serial.println("A je neprekinut");
    lcd.clear();
    lcd.setCursor(0, 0);
    lcd.print("A je neprekinut");
```

```
  delay(1000);
}
if (!sensorState_A && lastState_A)
{
  Serial.println("A je prekinut");
  lcd.clear();
  lcd.setCursor(0, 0);
  lcd.print("A je prekinut");
  delay(1000); // Vrijeme čekanja od 1 sekunde
}
lastState_A = sensorState_A;
if (sensorState_B == LOW)
{
  //Upaliti LED L:
  digitalWrite(LEDPIN, HIGH);
}
else
{
  // iskljuciti LED L:
  digitalWrite(LEDPIN, LOW);
}
if (sensorState_B && !lastState_B)
{
  Serial.println("B je neprekinut");
  lcd.clear();
  lcd.setCursor(0, 0);
  lcd.print("B je neprekinut");
  delay(1000);
}
if (!sensorState_B && lastState_B)
{
```

```
  Serial.println("B je prekinut");
  lcd.clear();
  lcd.setCursor(0, 0);
  lcd.print("B je prekinut");
  delay(1000);
}
lastState_B = sensorState_B;

if (sensorState_C == LOW)
{
  // Upaliti LED L:
  digitalWrite(LEDPIN, HIGH);
}
else
{
  // iskljuciti LED L:
  digitalWrite(LEDPIN, LOW);
}
if (sensorState_C && !lastState_C)
{
  Serial.println("C je neprekinut");
  lcd.clear();
  lcd.setCursor(0, 0);
  lcd.print("C je neprekinut");
  delay(1000);
}
if (!sensorState_C && lastState_C)
{
  Serial.println("C je prekinut");
  lcd.clear();
  lcd.setCursor(0, 0);
```

```
    lcd.print("C je prekinut");
    delay(1000);
  }
  lastState_C = sensorState_C;

  delay(500);  //Pauza prije ponovnog pokretanja petlje
}
```

Da bi se senzor definirao unutar programa potrebno je koristiti **#define SENSORPIN_A 2.** SENSORPIN_A je naziv varijable koja prima signale sa senzora A, a broj 2 označava mjesto na Arduino mikrokontroleru gdje se nalazi pin u koji se spaja bijela žica s prijemnika senzora. [26]

Ukoliko bi bilo potrebe za dodavanjem još senzora, u program je potrebno dodati **#define SENSORPIN_OznakaSenzora BrojPina**;. SENSORPIN_OznakaSenzora je ime varijable koja će primati signale sa senzora, a BrojPina je mjesto na Arduino mikrokontroleru gdje se spaja žica koja prenosi signale s prijemnika senzora. Umjesto "OznakaSenzora" stavlja se proizvoljno ime varijable. Treba uzeti u obzir koliko ima slobodnih pinova za korištenje i s koliko se slobodnih pinova na eksperimentalnoj pločici raspolaže. [26]

#define LEDPIN 13 označava L lampicu na Arduino mikrokontroleru. Lampica L se pali ili gasi ovisno o tome mijenja li se stanje na senzorima. Stanje na senzorima može biti LOW (0 odnosno neaktivno stanje) ili HIGH (1 odnosno aktivno stanje). [26]

Serial.begin(9600); je funkcija koja započinje komunikaciju Arduino mikrokontrolera i Serial Monitora u bitovima po sekundi. U prezentiranom slučaju radi se o 9600 bitova po sekundi. 9600 je broj koji se zove Baud rate (hrv. brzina prijenosa). Neke od ostalih podržanih vrijednosti su 19200, 57600, 115200. Broj koji je određen u funkciji Serial.begin(9600) mora biti isti kao broj odabran unutar prozora Serial Monitor. U suprotnom ne bi se dobio prikaz na LCD zaslonu i Serial Monitoru ili bi se prikazala greška. [28]

If(sensorState_A && !lastState_A) je funkcija koja provjerava trenutno stanje i prethodno stanje na senzoru. Ako je predmet došao između prijemnika i predajnika senzora A, promijenit će se vrijednost na sensorState_A u LOW (0 odnosno neaktivno stanje) te će se prikazati poruka na zaslonu LCD-a: "A je prekinut" nakon izvršenja dijela koda **lcd.print("A je prekinut");**. Funkcija lcd.print("A je prekinut") se nalazi unutar funkcije **If(sensorState_A && !lastState_A)**. U trenutku kada više nema predmeta između prijemnika i predajnika senzora A, vrijednost na sensorState_A prelazi u HIGH (1 odnosno aktivno stanje) te se prikazuje poruka na LCD zaslonu: "A je neprekinut" nakon izvršenja dijela koda **lcd.print("A je neprekinut");.** Funkcija lcd.print("A je neprekinut") se nalazi unutar funkcije **If(sensorState_A && !lastState_A)**. [26]

5.2 Mogućnost proširenja projekta

Ako postoji potreba za praćenjem koliko je paket dugačak odnosno širok moguće je dodati još jedan par ili nekoliko parova IR Break Beam senzora na određenim udaljenostima. Uvjeti bi bili da parovi IR Break Beam senzora detektiraju paket u isto vrijeme te nakon detekcije odrediti u koju kategoriju po širini odnosno dužini paket pripada te odgovara li uopće paket uvjetima po svojim dimenzijama. Osim što bi se s dodavanjem više IR Break Beam senzora dobila mogućnost mjerenja dužine/širine može se izračunati vrijeme koje je potrebno paketima da dođu od početka pokretne trake do kraja pokretne trake ili nekog od spremnika ukoliko je to potrebno za daljnje proračune.

6. ZAKLJUČAK

Cilj završnog rada je bio osmisliti i složiti sustav koji ima mogućnost detekcije predmeta i određivanja njegove visine. Da bi se to uspjelo ostvariti korišteni su IR Break Beam senzori, Arduino Uno R3 mikrokontroler, LCD 16x2 zaslon te program Arduino i LOGO!Soft Comfort.

Zamisao je bila da se 4 senzora postave na određene visine na kojima će detektirati predmet. Ovisno o tome koji su senzori detektirali predmet mogla se odrediti visina samog predmeta pomoću logičkih operacija u programu LOGO!Soft Comfort. Nakon određivanja visine predmeta cilindar bi izbacio predmet u spremnik, a brojači bi vodili brigu o tome koliko se predmeta nalazi u pojedinim spremnicima. Nakon određenog broja paketa u spremniku pojavila bi se obavijest na zaslonu da je promjena spremnika u tijeku.

U praktičnom dijelu je napravljena simulacija detekcije predmeta senzorima te je svaki put poslana poruka na LCD 16x2 zaslon kada bi došlo do promjene stanja sustava. Za prekid snopa svjetlosti je dovoljno bilo koristiti komad kartona ili bilo kojeg materijala koji ne propušta svjetlost. Bilo je potrebno izraditi program za Arduino mikrokontroler koji je upravljao s informacijama koje su dolazile sa senzora te prikazivati trenutno stanje sustava na zaslonu LCD-a.

Jedna od mogućih nadogradnji ovog sustava bi mogla biti određivanje dužine/širine predmeta tako što bi se dodalo još infracrvenih senzora na određenom razmaku od prethodno postavljenih senzora. Također je moguće implementirati mjerenje vremena prolaska paketa po pokretnoj traci ukoliko bi se vodila statistika koliko je vremena utrošeno da bi određeni broj paketa sustav obradio.

LITERATURA

Za internetske izvore:

[1.] IR Break Beam senzori, dostupno na: https://shop.pimoroni.com/products/ir-break-beam-sensor-5mm-leds , pristupljeno 23.07.2021

[2.] Siemens LOGO! PLC, dostupno na: https://www.indiamart.com/proddetail/siemens-logo-plc-system-20947457712.html, pristupljeno 23.07.2021

[3.] https://new.siemens.com/global/en/products/automation/systems/industrial/plc/logo/logo-basic-modules.html

[4.] Osnove logičke funkcije, dostupno na: https://razno.sveznadar.info/3_RS_digit/02_Bul/20-Operacije.htm , pristupljeno 29.07.2021

[5.] https://www.manualslib.com/manual/1642126/Siemens-Logo-Series.html?page=201

[6.] https://support.industry.siemens.com/cs/mdm/109741041?c=67016428811&lc=en-AZ

[7.] https://www.manualslib.com/manual/1382742/Siemens-Logo.html?page=150

[8.] https://www.electricalchile.cl/plclogosiemens7en.php

[9.] USB kabel B tipa, dostupno na: https://store.arduino.cc/usb-2-0-cable-type-a-b, pristupljeno 02.08.2021

[10.] https://www.elprocus.com/lcd-16x2-pin-configuration-and-its-working/

[11.] https://www.arduino.cc/en/Reference/LiquidCrystalConstructor

[12.] https://www.arduino.cc/en/Reference/LiquidCrystalBegin

[13.]https://www.arduino.cc/en/Reference/LiquidCrystalSetCursor

[14.] https://www.arduino.cc/en/Reference/LiquidCrystalClear

[15.] Shema spajanja Arduino mikrokontrolera i LCD 16x2 zaslona, dostupno na: https://www.electronicwings.com/arduino/lcd-16x2-interfacing-with-arduino-uno, pristupljeno 04.08.2021

[16.] Testna pločica, dostupno na: https://learn.sparkfun.com/tutorials/how-to-use-a-breadboard/all , pristupljeno 05.08.2021

[17.] https://store.arduino.cc/products/arduino-uno-rev3/

[18.] https://technobyte.org/2016/07/power-up-the-arduino-uno/

[19.] IR Break Beam senzori, dostupno na: https://www.adafruit.com/product/2167, pristupljeno 07.08.2021

[20.] https://www.circuito.io/app?components=512,11021

[21.] https://www.arduino.cc/

[22.] https://www.microchip.com/en-us/product/ATmega328P

[23.] https://www.chipoteka.hr/artikl/116826/arduino-starter-kit-engleski-k000007-8205000007

[24.] https://www.arduino.cc/en/software

[25.] Prikaz pull-up otpornika u sustavu, dostupno na: https://cdn.sparkfun.com/assets/6/f/b/c/7/511568b6ce395f1b40000000.jpg, pristupljeno 08.08.2021

Za knjige:

[26.] Learn to Program in Arduino™ C: 18 Lessons, from setup() to robots, William P. Osborne

[27.] Building Arduino PLCs, Pradeeka Seneviratne

[28.] Arduino Projects Book