Razvoj ugradbenih sustava

Projekt 

Ak. god. 2023./2024.

UPOTREBA UGRAĐENOG SUSTAVA ZA PRIPREMU NAPITAKA

*Nikola Platnjak*

Datum predaje: *17.04.2024.*

# Sadržaj

[Sadržaj 2](#_Toc164891747)

[Slike 4](#_Toc164891748)

[Tablice 5](#_Toc164891749)

[Dnevnik promjena dokumentacije 6](#_Toc164891750)

[Pojmovnik 8](#_Toc164891751)

[1 Opis projektnog zadatka 1](#_Toc164891752)

[2 Opis sustava 2](#_Toc164891753)

[2.1 Mehanička struktura 3](#_Toc164891754)

[2.1.1 Koračni motor 3](#_Toc164891755)

[2.1.2 Membranska pumpa 4](#_Toc164891756)

[2.1.3 MOTORNI POGON 4](#_Toc164891757)

[2.1.4 GRANIČNA SKLOPKA KRAJNJEG ZAUSTAVLJANJA 5](#_Toc164891758)

[2.1.5 NAPAJANJE 6](#_Toc164891759)

[2.1.6 BLUETOOTH MODUL HC-05 6](#_Toc164891760)

[2.1.7 RGB DIODA 7](#_Toc164891761)

[2.1.8 KONDENZATOR 7](#_Toc164891762)

[2.1.9 ISPRAVLJAČKA DIODA 8](#_Toc164891763)

[2.1.10 OTPORNIK 8](#_Toc164891764)

[2.1.11 8-KANALNI RELEJNI MODUL 9](#_Toc164891765)

[2.1.12 Arduino Uno mikrokontroler 10](#_Toc164891766)

[2.2 Resursi 11](#_Toc164891767)

[2.3 Minimalni zahtjevi 11](#_Toc164891768)

[3 Specifikacija zahtjeva 12](#_Toc164891769)

[3.1 Obrazac uporabe 12](#_Toc164891770)

[3.2 Sekvencijski dijagram 13](#_Toc164891771)

[3.3 Funkcijski zahtjevi 13](#_Toc164891772)

[3.3.1 Opća funkcionalnost 13](#_Toc164891773)

[3.3.2 Funkcionalnosti kretanja platforme 13](#_Toc164891774)

[3.3.3 Zahtjevi napajanja 14](#_Toc164891775)

[3.4 Mehanički zahtjevi 15](#_Toc164891776)

[3.4.1 Dimenzije i kućište 15](#_Toc164891777)

[3.5 Ostali zahtjevi 15](#_Toc164891778)

[3.6 Cijena 15](#_Toc164891779)

[3.7 Buduće značajke 16](#_Toc164891780)

[4 Detaljnije specifikacije funkcije 17](#_Toc164891781)

[4.1 Oblikovanje programske potpore 17](#_Toc164891782)

[4.1.1 Oblikovanje programske potpore za mobilnu Aplikaciju 17](#_Toc164891783)

[4.1.2 Oblikovanje programske potpore za Arduino 18](#_Toc164891784)

[4.1.3 Implementacija 22](#_Toc164891785)

[5 Korištene tehnologije i alati 23](#_Toc164891786)

[6 Zaključak i budući rad 24](#_Toc164891787)

[7 Reference 25](#_Toc164891788)

*Sadržaj se automatski osvježava prema tekstu (desni klik, „Update Field“) uz zadani formata poglavlja.*

# Slike

[Slika 1. Nema 17 koračni motor 3](#_Toc164891837)

[Slika 2. Membranska pumpa 12V 4](#_Toc164891838)

[Slika 3. Motorni pogon(DRV8825) 5](#_Toc164891839)

[Slika 4. Granična sklopka 5](#_Toc164891840)

[Slika 5. Napajanje 6](#_Toc164891841)

[Slika 6. Modul HC-05 7](#_Toc164891842)

[Slika 7. RGB dioda 7](#_Toc164891843)

[Slika 8. Kondenzator 8](#_Toc164891844)

[Slika 9. Ispravljačka dioda 8](#_Toc164891845)

[Slika 10. Otpornik 9](#_Toc164891846)

[Slika 11. 8-kanalni relejni modul 10](#_Toc164891847)

[Slika 12. Arduni Uno mikokontroler 10](#_Toc164891848)

[Slika 13. sekvencijski dijagram 13](#_Toc164891849)

[Slika 14. početni screen 17](#_Toc164891850)

[Slika 15. Slika recepata napitaka 17](#_Toc164891851)

# Tablice

[Table 1. Obrazac uporabe 12](#_Toc164892149)

[Table 2. opća funkcionalnost 13](#_Toc164892150)

[Table 3.funkcionalnost kretanje platforme 14](#_Toc164892151)

[Table 4. zahtjevi napajanja 14](#_Toc164892152)

[Table 5. dimenzije i kućište 15](#_Toc164892153)

[Table 6. cijene 15](#_Toc164892154)

[Table 7. buduće značajke 16](#_Toc164892155)

# Dnevnik promjena dokumentacije

|  |  |  |  |
| --- | --- | --- | --- |
| Rev. | Opis promjene/dodatka | Autor(i) | Datum |
| 0.1 | Popunjen dokument | Nikola Platnjak | 15.04.2024. |
| 0.2 | Dovršen dokument | Nikola Platnjak | 22.4.2024. |

# Pojmovnik

* FDM – Fused Deposition Modeling
* CAD – Computer Aided Design
* MIT –Massachusetts Institute of Technology
* IDE – Integrated Development Environment
* PWM – Pulse Width Modulation
* CAM – Computer-Aided Manufacturing
* CAE – Computer-Aided Engineering
* PLA - Polylactic Acid
* SPP - Serial port profile

# Opis projektnog zadatka

Cilj ovog projekta je projektiranje i izrada sustava za miješanje napitaka koja se upravlja preko mobilne aplikacije pomoću bluetooth modula za bežićno komuniciranje sa mobitelom.

Glavni fokus projekta je na izradi cijelog sustava sa arduino mikrokontrolerom za kontrolu cijelog sustava.

Kroz ovaj projekt, namjera je demonstrirati primjenu arduino mikrokontrolera u stvarnom svijetu te pokazati jedan od načina realizacije stvarnih problema. Također se želi demonstrirati princip bežičnog komuniciranja, kontrolu smjera koračnih motora kao i njihovu brzinu. Osim toga, ovaj projekt će omogućiti učenje i principe upravljanja arduino mikokontrolerom kao i samo njegovo programiranje te pružiti i uvid u područje ugradbenih sustava.

# Opis sustava

Sustav za pripremu napitaka je uređaj koji na zanimljiv i interaktivan način priprema razne napitke za svaku osobu koja poželi napitak. Upravo zbog toga njegova glavna funkcija je da pomogne osobama, a ponajprije organizatorima zabava. Zašto njima? Glavni razlog tome je da na zabavama domaćin ne stigne sve nadgledati kada je zaokupljen točenjem napitaka, a to je uvijek glavni razlog prolivenog napitka i kasnije čišćenja istog. Još jedan razlog tome je da mala djeca ne bi trebala sama sebi točiti pića, a odrasla osoba ne stigne svakome natočiti piće, a uz sve to i paziti na ostalu djecu. Zbog toga bi sustav za pripremu napitaka bio koristan jer bi to domaćine riješilo glavnih problema opisanih iznad. Sustav bi na jedan pritisak gumba točio od jednostavnih pa sve do kompleksnih napitaka bez prolivene kapi, a sam domaćin bi se mogao koncentrirati na bitnije stvari.

Ovaj rad je motiviran upravo tim problemima. Cilj je bio istražiti i napisati svu potrebnu dokumentaciju za izradu uređaja da pripomogne rješavanju navedenih problema. Uz samo dokumentaciju rad je namijenjen da uvede svaku osobu u izradu samog uređaja za kućnu primjenu, ali i za buduću profesionalnu primjenu. Ovaj rad opisuje svaku komponentu korištenu u izradi samog aparata, te način njegovog spajanja kako bi se lakše i jednostavnije objasnila uloga svakog elementa kao i uloga kompletnog uređaja.

Glavne komponente koje su se koristili za izradu samog uređaja su: arduino mikrokontroler, 3d printer i aplikacija za odabir napitaka.

U drugom poglavlju se govori o 3d printanju, na koji način se koristilo u radu i koji sve alati su bili potrebni za 3d printanje.

U trećem poglavlju obrađuje se način rada, uloga i programiranje same mobilne aplikacije.

U četvrtom poglavlju opisuju se komponente od kojih je sastavljen uređaj i njihova uloga, ali uz to objašnjava se način programiranja kao i sam način rada mikrokontrolera.

U petom poglavlju rad završava zaključkom u kojem se govori o prednostima i nedostacima i samoj ideji za stvaranje uređaja za miješanje napitaka.

## Mehanička struktura

### Koračni motor

Koračni motor je elektromotorni uređaj koji koristi koračne impulsne signale za rotaciju i kretanje. On se često koristi za precizno i kontrolirano kretanje u automatiziranim procesima, poput pomicanja dijelova u industrijskoj proizvodnji ili za rotaciju diskova u diskovima za čitanje podataka. Koračni motori su često pogodniji za precizne aplikacije zbog svoje točnosti i kontrole u odnosu na druge vrste elektromotora.

Uloga ovog motora u radu je pokretanje platforme po x osi tako da bi uređaj mogao natočiti više napitaka. Uz kretanje po x osi koristila su se još dva koračna motora za podizanje dijela platforme po z osi tako da bi mogao sam uređaj istisnuti neke od napitaka postavljenih iznad platforme. Da bi se dio platforme mogao podići dva motora su spojena serijski tako da se okreću u isto vrijeme.

Slika na kojoj se prikazuje elektronički, kamera

Opis je automatski generiran

Slika 1. Nema 17 koračni motor

### Membranska pumpa

Membranska pumpa je vrsta pumpe koja koristi membranu kao glavni funkcionalni dio koji omogućava prijenos tekućina. Njen radni princip funkcionira tako da se kroz kretanje dijelova pumpe komprimira i dekomprimira membrana, što uzrokuje promjenu volumena unutar pumpe i time stvara pritisak na tekućinu. Ova vrsta pumpe se često koristi u pogonima gdje je potrebna precizna kontrola volumena ili brzine protoka tekućina.

Uloga ove pumpe u radu je da prebacuje napitke iz njihovog spremnika u posudu u kojoj će se miješati napitak. Pumpa je zaštićena diodom i spojena na napon te na relej. Kada platforma stigne do određene pumpe, ta pumpa se uključuje i izbacuje napitak iz spremnika u posudu.



Slika 2. Membranska pumpa 12V

### MOTORNI POGON

Motorni pogon je integrirani upravljač za koračne motore, koji kontrolira brzinsko-torzijsko ponašanje motora pomoću arduina. Radi na načelu PWM, to jest omogućava precizno kontroliranje brzine i snage motora. Na sebi ima ugrađen hladnjak kao zaštitu od pregrijavanja i uz to ima i zaštitu od strujnog udara .

U radu su korištena dva motorna pogona DRV8825 da kontroliraju koračne motore. Jedan motorni pogon kontrolira motor za navođenje platforme po x osi, dok drugi kontrolira dva koračna motora za kretanje po z osi.

Slika na kojoj se prikazuje elektronički, strujni krug

Opis je automatski generiran

Slika 3. Motorni pogon(DRV8825)

### GRANIČNA SKLOPKA KRAJNJEG ZAUSTAVLJANJA

Granična sklopka krajnjeg zaustavljanja je komponenta koja služi za prekidanje električnog kruga u slučaju da mehanizam dosegne određenu poziciju ili kada se pojavi problem. Ova sklopka omogućava zaustavljanje uređaja u slučaju neispravnosti ili nenamjerne pogreške, što pomaže u očuvanju sigurnosti i štiti mehanizam i korisnike od ozljeda. Granična sklopka krajnjeg zaustavljanja može se koristiti u različitim industrijama, kao što su uređajevi za obradu drva, proizvodnja hrane i napitaka, i drugi slični.

Primjena u radu je za označavanje početne oznake platforme. Platforma bi se prije svakog točenja napitaka kretala prema sklopki i kada bi je dotakla, sklopka bi poslala signal mikrokontroleru i on bi zaustavio motor jer bi znao da je platforma došla na početak i da je spremna za izvođenje.

Slika na kojoj se prikazuje kabel, priključak

Opis je automatski generiran

Slika 4. Granična sklopka

### NAPAJANJE

Napajanje je električna komponenta koja pretvara ulaznu električnu energiju iz električne mreže u izlaznu energiju prikladnu za napajanje korištenih komponenti. U kontekstu računalne tehnologije, napajanje se odnosi na komponentu koja pretvara izmjeničnu struju električne mreže u istosmjernu struju potrebnu za napajanje raznih uređaja. Napajanje osigurava stabilnost naponskih razina i strujnih tokova potrebnih za pravilno funkcioniranje tih komponenata. Napajanje može imati različite snage, izražene u vatima, koje označavaju koliko energije može isporučiti nekom uređaju. Također, napajanja često sadrže zaštitne mehanizme poput zaštite od prevelikog napona i zaštite od preopterećenja kako bi se osigurala sigurnost uređaja i komponenti.

Napajanje koje se koristilo u ovom radu je kompjutersko napajanje od 600W. Uređaj ne iziskuje toliku snagu, ali napajanje je vrlo dobro poslužilo upravo zbog toga što nudi više izlaznih napona: 12V, 5V i 3.3V. Komponentama uređaja je potrebno 12V i 5V.

Slika na kojoj se prikazuje tekst, u dvorani, kućanski aparati, elektronika

Opis je automatski generiran

Slika 5. Napajanje

### BLUETOOTH MODUL HC-05

Modul HC-05 je bežična komponenta koja omogućava bluetooth protokol kao način povezivanja mikrokontrolera sa okolinom. Bluetooth protokol se koristi za dijeljenje podatka i komuniciraje među uređajima na kratkim udaljenostima bez potrebe za fizičkom vezom putem kabela. HC-05 koristi Bluetooth standard za komunikaciju i podržava SPP, što znači da se može ponašati kao serijski port, slično kao kada se koristi USB ili RS-232 kabel za komunikaciju. To omogućava jednostavno slanje i primanje podataka između uređaja putem standardnih serijskih komunikacijskih naredbi. Često se koristi u projektima elektronike, robotike gdje je potrebna bežična komunikacija između uređaja.

U ovom radu je modul korišten zbog lakšeg kontroliranja uređaja putem mobitela. Povezuje mobilnu aplikaciju i kod unutar mikrokontrolera bežičnim putem. Preko mobitela se odabere željeni napitak i modul pošalje niz brojeva prema mikrokontroleru koji onda zna koji koktel se radi i kako da napravi taj koktel.

Slika na kojoj se prikazuje komponenta strujnog kruga, elektronička komponenta, elektronika, pasivna komponenta strujnog kruga

Opis je automatski generiran

Slika 6. Modul HC-05

### RGB DIODA

Skraćeno od "Red, Green, Blue" (crvena, zelena, plava) je vrsta svjetlosne diode koja ima mogućnost emitiranja svjetlosti u raznim bojama kombiniranjem crvene, zelene i plave boje. Kontroliranjem intenziteta svake od tih boja moguće je proizvesti gotovo beskonačno mnogo različitih kombinacija boja. RGB diode često su upravljane mikrokontrolerima ili kontrolerima svjetlosnih efekata, omogućujući korisnicima da prilagode boju i intenzitet svjetla prema željama.



Slika 7. RGB dioda

### KONDENZATOR

Kondenzator je pasivna električna komponenta koja ima sposobnost pohranjivanja električnog naboja. Sastoji se od dvije elektroda koje su odvojene izolacijskim materijalom. Kada se kondenzator napuni, električni naboj se pohranjuje između elektroda i dielektrika. Kondenzatori se koriste u filtriranju, blokiranju istosmjernih i propuštanju signala. Ovisno o primjenama, postoje različite vrste kondenzatora, kao što su elektrolitski kondenzatori, keramički kondenzatori i mnogi drugi.

U ovom radu je korišten elektrolitski kondenzator i glavna uloga mu je da filtrira i održava stabilnost i pravilno funkcioniranje mikrokontrolera.



Slika 8. Kondenzator

### ISPRAVLJAČKA DIODA

Ispravljačka dioda je osnovna elektronička komponenta koja ima ključnu ulogu u pretvaranju izmjenične struje u istosmjernu struju. Njena svrha je omogućiti prolaz struje samo u jednom smjeru, blokirajući struju u suprotnom smjeru.

Dioda ima dva terminala - anodu i katodu. Kada se napon usmjeri od anode prema katodi, dioda postaje provodljiva i omogućava prolaz struje kroz sebe. Međutim, kada je napon usmjeren u suprotnom smjeru, tako da je katoda pozitivna u odnosu na anodu, dioda postaje ne provodljiva i sprječava prolaz struje.

Uloga ispravljačke diode u radu je stvaranje stabilnog napajanja na motornim pumpama, mogućnost rada na konstantnom istosmjernom napajanju i samu zaštitu ostalih dijelova rada u slučaju otkazivanja pumpi ili krivog rada same pumpe koja je po standardu u elektrotehnici spojena inverzno na relej.

Slika na kojoj se prikazuje tuba

Opis je automatski generiran uz srednju pouzdanost

Slika 9. Ispravljačka dioda

### OTPORNIK

Otpornik je pasivna elektronička komponenta koja se koristi za kontroliranje protoka električne struje. Otpornici su obično izrađeni od materijala koji imaju visoke otporne karakteristike, poput ugljika, metala ili keramike. Otpornici se često koriste za podešavanje struje, prilagodbu napona, zaštitu komponenata od preopterećenja i još mnogo toga.

U radu su se koristili otpornici različite vrijednosti kako bi se prilagodile performanse krugova, osigurala optimalna radna temperatura komponenata i smanjili preveliki protok struje.

Slika na kojoj se prikazuje tuba

Opis je automatski generiran

Slika 10. Otpornik

### 8-KANALNI RELEJNI MODUL

8-kanalni relejni modul je elektronički modul koji omogućava kontroliranje više električnih uređaja, korištenjem malog broja signala. Modul ima 8 releja koji su spojeni na ulaze i kada se dovede električni napon, relej se prebacuje iz jednog stanja u drugo, što omogućava prekidanje veze između dva električna kruga. Kada treba početi s radom relej dobije signal na određeni ulaz i na tom ulazu propušta signal. Ako je na tom ulazu spojen koračni motor za pomicanje platforme, platforma se kreće vračati na početak, kada je došla na početak kreće prema prvoj poziciji. Nakon što je došla do prve pozicije mikrokontroler prestaje slati signal na taj ulaz i relej gasi propusnost napajanja za taj koračni motor. Kada je platforma stala ovisno na kojoj je poziciji pali se pumpa ili koračni motori za podizanje platforme. To se ponavlja sve dok platforma ne dođe na kraj i informira da je napitak spreman. Ovaj modul se često koristi u automatizaciji, kontroliranju rasvjete, grijanja i drugih uređaja. Prednost releja je u odvajanju dva strujna kruga, odnosno razdvajanju njihovih točaka mase.

Uloga ovog releja je kontroliranje dijelova uređaja za izradu napitaka. Na njega su spojeni svi motori i sve pumpe koje se trebaju upaliti i ugasiti u određenom vremenskom periodu.

Slika na kojoj se prikazuje tekst, elektronički

Opis je automatski generiran

Slika 11. 8-kanalni relejni modul

### Arduino Uno mikrokontroler

Arduino koristi algoritme i program koji su mu dodijeljeni kako bi upravljao svim procesima pripreme napitka. Pomoću koda, Arduino kontrolira rad koračnog motora koji pomiče čašu na određeno mjesto gdje se napitak nalazi. Također, pomoću pumpi za tekućinu, Arduino može usisavati određene sastojke iz spremnika i dodavati ih u čašu. Arduino koristi informacije primljene iz aplikacije za odabir napitka kako bi znao koji sok treba poslužiti i koliko dugo mora osigurati rad pumpe pa time i količinu sastojaka u čaši. Uz rad koračnog motora za pomicanje platforme. Na njoj se nalaze dodatna dva koračna motora spojena serijski koja su također pod kontrolom Arduina. Omogućavaju precizno dodavanje glavnih sastojaka napitka tako da je na svaki zasebno spojena cijev s navojem te međusobno pomiču drugi dio platforme da aktiviraju dozator za doziranje tekućine. Svi ti procesi su savršeno sinkronizirani i upravljani od strane Arduina, što osigurava da se napitak priprema brzo i precizno.

Slika na kojoj se prikazuje elektronička komponenta, komponenta strujnog kruga, strujni krug, elektroničko inženjerstvo

Opis je automatski generiran

Slika 12. Arduni Uno mikokontroler

## Resursi

Potrebni resursi za izgradnju ovog ugrađenog sustava su:

* Računalo sa instaliranim arduinoIDE
* Sve potrebne komponente
* Znanje za programiranjem arduina
* Znanje upravljanjem motorima
* Znanje o funkcionalnostima komponenata
* 3d printer(platiti uslugu 3d printanja ili samostalno)
* Od ostalih komponenti(gumene cijevi, nosač boca, dozator napitaka, razvodna remenica, montažni nosač, zupčasti remen)

## Minimalni zahtjevi

Minimalni zahtjev projekta je kučište na koje će se sustav ugrađivati.

# Specifikacija zahtjeva

Aktor je korisnik aplikacije.

## Obrazac uporabe

|  |  |
| --- | --- |
| Naziv obrasca uporabe: | Daljinsko upravljanje vozilom |
| Namjena: | Miješanje napitaka |
| Uloga: | Korisnik |
| Opis: | Sustav omogućuje bežično miješanje napitaka |
| Slijed aktivnosti: | Korisnik uključuje sustav. Nakon uključenja je sustav spreman za korištenje |
| Sadržaj podataka: | Aplikacija za upravljanje |
| Prijenos podataka | Pritisak gumba na mobilnom uređaju |
| Preduvijeti: | Sustav je ugrađen i uključen te je aplikacija upaljena |
| Posljedice: | Korisnik upravlja sustavom |
| Kad počinje: | Nakon uključivanja sustava i spajanja na arduino. |
| Kad završava: | Nakon gašenja sustava |
| Izuzeci: | Nestanak struje |

Table 1. Obrazac uporabe

## Sekvencijski dijagram

Slika na kojoj se prikazuje tekst, dijagram, paralelno, snimka zaslona

Opis je automatski generiran

Slika 13. sekvencijski dijagram

## Funkcijski zahtjevi

### Opća funkcionalnost

|  |  |
| --- | --- |
| Zahtjev | Opis |
| F-1.1 | Uređaj mora nuditi uparivanje prilikom paljenja |
| F-1.2 | Uređaj mora imati prekidač za paljenje |

Table 2. opća funkcionalnost

### Funkcionalnosti kretanja platforme

|  |  |
| --- | --- |
| Zahtjev | Inicijalno RGB ledica svjetli crveno |
| F-2.1 | Odabirom napitka rgb dioda svjetli plavo |
| F-2 | Završetkom uređaja rgb dioda svjetli zeleno |
| F-2.1 | Prilikom odabira napitka platfomra se vraća u početnu poziciju |

Table 3.funkcionalnost kretanje platforme

### Zahtjevi napajanja

| Zahtjev | opis |
| --- | --- |
| F-3.1 | Uređaj koristi napajanje 5-12v |
| F-3.2 | Izvor napajanja je kompjutersko napajanje od 600W |

Table 4. zahtjevi napajanja

## Mehanički zahtjevi

### Dimenzije i kućište

|  |  |
| --- | --- |
| Zahtjev | Opis |
| F-4.1 | Kućište 70x50 cm |
| F-4.2 | Napajanje 15x15 cm |
| F-4.3 | Mikroupravljač 80x70 mm |

Table 5. dimenzije i kućište

## Ostali zahtjevi

Ovisno o količini napitaka koju jedan pomiješani napitak sadrži, platforma se zaustavlja na određenoj lokaciji. Ovisno o vrsti napitaka, mikrokontroler postavlja vrijeme prelijevanja napitaka u posudu.

## Cijena

|  |  |
| --- | --- |
| Zahtjev | Opis |
| F-4.1 | Arduino - 5€ |
| F-4.2 | Napajanje - 50€ |
| F-4.3 | HC05 Bluettoth modul - 2,44€ |
| F-4.4 | Nema 17 koračni motor - 5€ |
| F-4.5 | Membranska pumpa – 5€ |
| F-4.6 | DRV8825 motorni pogon - 3€ |
| F-4.7 | Granična sklopka - 3€ |
| F 4.8 | 8-kanalni relej - 10€ |

Table 6. cijene

## Buduće značajke

Ove dodatne značajke se ne planiraju implementirati u prototipu. To su ideje za budući razvoj uređaja.

|  |  |
| --- | --- |
| Zahtjev | Opis |
| O-8.1 | Tiskana pločica |
| O-8,2 | Profesionalnije kučište |
| O-8.3 | Zamjena remena za klizne ležajeve |

Table 7. buduće značajke

# Detaljnije specifikacije funkcije

## Oblikovanje programske potpore

### Oblikovanje programske potpore za mobilnu Aplikaciju

Slika na kojoj se prikazuje tekst, snimka zaslona, Font

Opis je automatski generiran

Slika 14. početni screen

Slika na kojoj se prikazuje tekst, snimka zaslona

Opis je automatski generiran

Slika 15. Slika recepata napitaka

### Oblikovanje programske potpore za Arduino

#include <AccelStepper.h>

const int opticCount = 8; // how many optic stations there are

const int parameterCount = 3; //number of seeting to be stored in array for each optic

// Define stepper motor connections and motor interface type. Motor interface type must be set to 1 when using a driver:

#define dirPin 2

#define stepPin 3

#define dirPinPlatform 4

#define stepPinPlatform 5

#define motorInterfaceType 1

//---------------------

// Create a new instance of the AccelStepper class:

AccelStepper stepper = AccelStepper(motorInterfaceType, stepPin, dirPin);

AccelStepper platformSteppers = AccelStepper(motorInterfaceType, stepPinPlatform, dirPinPlatform);

//---------------------

//matric has 3 number for each dispenser:

// first number is how far away is dispenser from its neighbour

// second is delay for how long mechanisam for dispensing shuold be pressed

// third number is how many times should value be pressed

// First number is always the same and numbers that go to serial are second and third

//(0,0,0, 460,0,0, 340,0,0, 0,0,0, 0,0,0, 0,0,0, 380,0,0, 470,0,0)

//u serial, ali bez razmaka ide npr.(10,1, 10,1, 10,1, 10,1, 10,1, 10,1, 10,1, 10,1)

int drinkMatrix[opticCount][parameterCount] = {

0, 0, 0,

460, 0, 0,

350, 0, 0,

0, 0, 0,

0, 0, 0,

0, 0, 0,

340, 0, 0,

460, 0, 0

};

//---------------------

//The below is for the contact switch.

const int buttonPin = 7; // the pin that the pushbutton is attached to

int buttonState = 0; // current state of the button

int lastButtonState = 0; // previous state of the button

//---------------------

// Below is for the RGB LED

const int redPin = A0;

const int greenPin = A1;

const int bluePin = A2;

//---------------------

//Below is for the water pumps

const int pump1 = 6;

const int pump2 = 8;

const int pump3 = 12;

const int pump4 = 13;

//---------------------

const int motorRelay = 9;

//The below are for running the machine:

int raftFound = false;

int drinkRequested = false;

//---------------------

void setup() {

// initialize serial communication:

Serial.begin(9600);

//--------------------------------

// for the contact switch

// initialize the button pin as a input:

pinMode(buttonPin, INPUT);

//--------------------------------

// For the stepper motors

stepper.setMaxSpeed(1000);

platformSteppers.setMaxSpeed(1000);

//--------------------------------

// For LED

pinMode(redPin, OUTPUT);

pinMode(greenPin, OUTPUT);

pinMode(bluePin, OUTPUT);

//-------------------------------

// For pumps

pinMode(pump1, OUTPUT);

pinMode(pump2, OUTPUT);

pinMode(pump3, OUTPUT);

pinMode(pump4, OUTPUT);

//-------------------------------

pinMode(motorRelay, OUTPUT);

}

void loop() {

//turn off all the pumps

//why on HIGH?

//ha ha, because I connect pump so they work on when signal is LOW and stop when signal is high.

digitalWrite(pump1, HIGH);

digitalWrite(pump2, HIGH);

digitalWrite(pump3, HIGH);

digitalWrite(pump4, HIGH);

digitalWrite(motorRelay, LOW);

//Lets find the location of the platform:

if (raftFound == false) {

Serial.println("Raft location not known yet");

setColor(255, 0, 0); // red

delay(1000);

Serial.print("Looking for the raft...");

//returning platform to starting position

returnToStart();

raftFound = true;

Serial.println("There it is! :)");

setColor(0, 0, 255); // blue

delay(500); // Delay a little bit to calm everything down

}

//working through the dispensing instructions until drink is complete and paddle is at last optic position:

if (drinkRequested == true) {

for (int optic = 0; optic < opticCount; optic++) {

//Move to pump

//platform is always at first dispenser so in matric first value to move is always 0

stepper.setCurrentPosition(0);

while (stepper.currentPosition() != -drinkMatrix[optic][0]) {

stepper.setSpeed(-300);

stepper.runSpeed();

}

delay(200);

if (drinkMatrix[optic][2] > 0 && raftFound == true) {

switch (optic) {

case 2:

//pump 1 is working

motorPumpstart(pump1, drinkMatrix[optic][1] \* 100);

break;

case 3:

//pump 2 is working

motorPumpstart(pump2, drinkMatrix[optic][1] \* 100);

break;

case 4:

//pump 3 is working

motorPumpstart(pump3, drinkMatrix[optic][1] \* 100);

break;

case 5:

//pump 4 is working

motorPumpstart(pump4, drinkMatrix[optic][1] \* 100);

break;

default:

//dispense what is required then resume moving to the next position:

while (drinkMatrix[optic][2] > 0) {

startDispensing(drinkMatrix[optic][1] \* 100);

drinkMatrix[optic][2] -= 1;

}

break;

}

}

}

//Drink complete

Serial.println("Drinks ready, enjoy.");

setColor(0, 255, 0); // green

drinkRequested = false;

raftFound = false;

}

while (drinkRequested == false) {

if (Serial.available()) {

for (int optic = 0; optic < opticCount; optic++) {

for (int parameter = 1; parameter < parameterCount; parameter++) {

readParameter(drinkMatrix[optic][parameter]);

}

}

CheckArray();

Serial.println("Done loading");

drinkRequested = true;

}

}

}

void CheckArray() {

//print out the array to check it:

for (int i = 0; i < opticCount; i++) {

for (int j = 0; j < parameterCount; j++) {

Serial.print(drinkMatrix[i][j]);

Serial.print(",");

}

Serial.println();

}

}

void setColor(int red, int green, int blue) {

analogWrite(redPin, red);

analogWrite(greenPin, green);

analogWrite(bluePin, blue);

}

void startDispensing(int vrijeme) {

platformSteppers.setCurrentPosition(0);

while (platformSteppers.currentPosition() != -1900) {

platformSteppers.setSpeed(-500);

platformSteppers.runSpeed();

}

//time of dispenser being pressed

delay(vrijeme);

platformSteppers.setCurrentPosition(0);

while (platformSteppers.currentPosition() != 1900) {

platformSteppers.setSpeed(500);

platformSteppers.runSpeed();

}

platformSteppers.setCurrentPosition(0);

while (platformSteppers.currentPosition() != -1100) {

platformSteppers.setSpeed(-500);

platformSteppers.runSpeed();

}

platformSteppers.setCurrentPosition(0);

while (platformSteppers.currentPosition() != 1100) {

platformSteppers.setSpeed(500);

platformSteppers.runSpeed();

}

delay(500);

}

void returnToStart() {

stepper.setCurrentPosition(0);

while ((buttonState = digitalRead(buttonPin)) != 0 && raftFound == false) {

stepper.setSpeed(300);

stepper.runSpeed();

}

//moving plaform to first dispenser

stepper.setCurrentPosition(0);

while (stepper.currentPosition() != -20) {

stepper.setSpeed(-300);

stepper.runSpeed();

}

}

void motorPumpstart(int pump, int vrijeme) {

digitalWrite(motorRelay, HIGH);

digitalWrite(pump, LOW);

//time of pump working

delay(vrijeme);

digitalWrite(pump, HIGH);

digitalWrite(motorRelay, LOW);

delay(500);

}

void readParameter(int& parameter) {

String inputString = Serial.readStringUntil(',');

parameter = inputString.toInt();

}

### Implementacija

Programski kod se direktno prenosi na arduino pomoću Arduino IDE. U Arduino IDE je potrebno odabrati COM port na računalu( port s kojim je Arduino spojen na kompjuter), te je potrebno odabrati verziju Arduina. U ovom slučaju to je bio Arduino UNO. Sam Arduino sustav, pločca i IDE uvelike olakšavaju izradu malih sustava.

# Korištene tehnologije i alati

Tehnologija korištena za prenošenje programskog koda arduina je Arduino IDE. Pomoću njega se mogao kod napisan u inačici programskog jezika C upisati i sačuvati na arduino pločici.

Za izradu mobilne aplikacije koristio se MIT APP INVENTOR.

Finalno, za izradu povećeg dizajna samog sustava korišten je 3d printer i tehnologije koje su omogućile dizajn i printanje su FUSION 360 i Cura slicer.

# Zaključak i budući rad

Današnji život je ispunjen s tehnologijom i postaje dio našeg života bez kojeg ljudi ne mogu živjeti. Zbog toga dodatna znanja u izradi raznih uređaja koji bi mogli uvelike pomoći ljudima uvijek su dobrodošla. Svakim novim izumom čovjek dobiva sve više znanja pa tako se i izradom ovog uređaja nešto novo uvijek može naučiti.

Za izradu ovog sustava bilo je potrebno znanje elektrotehnike i programskog inžinjerstva.  
Potencijalne dodatne funkcionalnosti uključuju dodavanje sustava za detekciju čaše, sustavom za detekciju pozicije platforme, te unaprjeđenje dizajna sustava.

Razvijeni sustav bi sigurno mogao biti komercijalno konkurentat zbog svoje unikatnosti i specifičnosti da zadovolji poseban krug korisnika, što omogućuje preciznije napretke i specifične dodatne zahtjeve.

# Reference

|  |  |
| --- | --- |
| [1] | Sculpteo, "The History of 3D Printing: 3D Printing Technologies from the 80s to Today", (2022). Dostupno: https://www.sculpteo.com/en/3d-learning-hub/basics-of-3d-printing/the-history-of-3d-printing/. [pristupano 20.1.2023.] |
| [2] | 3DJAKE, "Ender 3 V2 Neo", 3djake, (2023). Dostupno: https://www.3djake.hr/creality-3d-printeri-i-rezervni-dijelovi/ender-3-v2-neo. [pristupano 20.1.2023.] |
| [3] | G. Verma, "Starting with Autodesk", u *Autodesk Fusion 360 Book*, CADCAMCAE Works, pp. 1-2. |
| [4] | UltiMaker, "UltiMaker Cura", Ultimaker, (2023). Dostupno:https://ultimaker.com/software/ultimaker-cura. [pristupano 20.1.2023.] |
| [5] | D. THILAKANATHAN, "WHATIS 3D MODELING", u "*3D Modeling For Beginners"*, Thilakanathan Studios, 2016, pp. 12-13. |
| [6] | K. Patricija, "primjena 3d printera u modernom svijetu". (1.3. 2019.) Dostupno: https://www.3dprintaj.com/funkcionalnost-3d-printera/.[pristupano 20.1.2023.] |
| [7] | MIT app inventor, "About us", MIT, (2012). Dostupno: https://appinventor.mit.edu/about-us./ [pristupano 20.1.2023.] |
| [8] | M. S. Scott Fitzgerald," Arduino projects book, first edition ", M. S. T. I. Scott Fitzberald, Ur., Torino, 2015, pp. 4-5. |
| [9] | Vasudhendra Badami, "Arduino programming for beginners-1", hackerearth, (2016).Dostupno: https://www.hackerearth.com/blog/developers/arduino-programming-for-beginners/.[pristupano 20.1.2023.] |