

# REGULACIJA VISINE LEVITIRANE LOPTICE ZA STOLNI TENIS POMOĆU PID REGULATORA NA ARDUINO PLATFORMI

---

**Kosić, Adrian**

**Undergraduate thesis / Završni rad**

**2024**

*Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj:* **University of Rijeka, Faculty of Engineering / Sveučilište u Rijeci, Tehnički fakultet**

*Permanent link / Trajna poveznica:* <https://um.nsk.hr/um:nbn:hr:190:038573>

*Rights / Prava:* [Attribution 4.0 International](#) / [Imenovanje 4.0 međunarodna](#)

*Download date / Datum preuzimanja:* **2025-01-27**



*Repository / Repozitorij:*

[Repository of the University of Rijeka, Faculty of Engineering](#)



SVEUČILIŠTE U RIJECI  
TEHNIČKI FAKULTET

Preddiplomski sveučilišni studij elektrotehnike

Završni rad

**REGULACIJA VISINE LEVITIRANE LOPTICE ZA STOLNI  
TENIS POMOĆU PID REGULATORA NA ARDUINO  
PLATFORMI**

Rijeka, rujan 2024.

Adrian Kosić

0069090181

SVEUČILIŠTE U RIJECI  
TEHNIČKI FAKULTET

Preddiplomski sveučilišni studij elektrotehnike

Završni rad

**REGULACIJA VISINE LEVITIRANE LOPTICE ZA STOLNI  
TENIS POMOĆU PID REGULATORA NA ARDUINO  
PLATFORMI**

Mentor: izv. prof. dr. sc. Ivan Volarić

Rijeka, rujan 2024.

Adrian Kosić

0069090181

Rijeka, 20. ožujka 2023.

Zavod: **Zavod za automatiku i elektroniku**  
Predmet: **Elementi automatizacije postrojenja**  
Grana: **2.03.03 elektronika**

## ZADATAK ZA ZAVRŠNI RAD

Pristupnik: **Adrian Kosić (0069090181)**  
Studij: Sveučilišni prijediplomski studij elektrotehnike

Zadatak: **Regulacija visine levitirane loptice za stolni tenis pomoću PID regulatora na Arduino platformi / Arduino based ping-pong ball levitation control via PID regulator**

### Opis zadatka:

Potrebno je izraditi sustav za regulaciju visine levitirane loptice za stolni tenis pomoću PID regulatora implementiranog na Arduino platformi. Loptica se nalazi u uspravno postavljenoj cijevi, na čijem se dnu nalazi ventilator čijom se brzinom vrtnje upravlja (npr. ventilator od računala, čijom se brzinom upravlja preko PWM signala). Na vrhu cijevi nalazi se ultrazvučni senzor udaljenosti (npr. HC-SR04 modul) koji mjeri visinu (položaj) loptice. Korisnik željeni položaj loptice regulira preko potenciometra spojenog na analogni ulaz mikrokontrolera.

Rad mora biti napisan prema Uputama za pisanje diplomskih / završnih radova koje su objavljene na mrežnim stranicama studija.

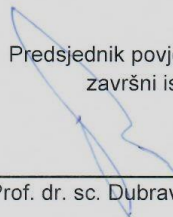
Zadatak uručen pristupniku: 20. ožujka 2023.

Mentor:



Doc. dr. sc. Ivan Volarić

Predsjednik povjerenstva za  
završni ispit:



Prof. dr. sc. Dubravko Franković

## **IZJAVA**

Za rad pod nazivom „Regulacija visine levitirane loptice za stolni tenis pomoću PID regulatora na Arduino platformi“ izjavljujem da sam samostalno izradio uz navedenu literaturu, stečeno znanje tijekom školovanja i stručnim vodstvom mentora izv. prof. dr. sc. Ivana Volarića.

Rijeka, rujan 2024.

Adrian Kosić

---



## Sadržaj

<b>1. UVOD</b> .....	1
<b>2. ARDUINO</b> .....	2
<b>2.1. Arduino Uno</b> .....	2
<b>2.2. Mikrokontroler ATmega328</b> .....	5
<b>2.3. PWM modulacija</b> .....	6
<b>3. ULTRAZVUČNI SENZOR HR-SC04</b> .....	8
<b>3.1. Specifikacije i primjena senzora</b> .....	10
<b>3.2. Izračunavanje udaljenosti i ograničenja senzora</b> .....	11
<b>4. OSTALE KOMPONENTE SPOJA</b> .....	13
<b>4.1. Nidec ventilator</b> .....	13
<b>4.2. Linearni potencijometar</b> .....	16
<b>4.3. Izvor napajanja</b> .....	17
<b>5. PID REGULACIJA</b> .....	18
<b>5.1. Primjena PID regulacije</b> .....	23
<b>5.2. Prednosti i nedostaci PID regulacije</b> .....	23
<b>7. PROGRAMSKI KOD</b> .....	28
<b>8. ZAKLJUČAK</b> .....	32
<b>9. LITERATURA</b> .....	33
<b>10. POPIS OZNAKA I KARTICA</b> .....	35
<b>11. SAŽETAK I KLJUČNE RIJEČI</b> .....	36
<b>11.1. Sažetak i ključne riječi</b> .....	36
<b>11.2. Summary and keywords</b> .....	36

## 1. UVOD

Regulacija visine levitirane loptice koju je potrebno izraditi u ovom završnom radu, implementirana je pomoću PID (proporcionalnog, integracijskog i derivacijskog) regulatora koji zahtijeva teorijska znanja iz područja automatike i elektronike. Primjena PID regulatora u ovom projektu pokazuje učinkovitost održavanja stabilnosti sustava. Arduino Uno koristi se kao osnova za realizaciju sustava zbog svoje fleksibilnosti, jednostavnosti korištenja i široke dostupnosti komponenti, što ju čini idealnom za slične sustave. U ovom radu fokus je na upravljanju visine levitirane loptice unutar uspravno postavljene cijevi, gdje ventilator na dnu cijevi generira strujanje zraka koje podiže lopticu. Da bi se postigla stabilnost levitacije loptice, potrebno je precizno kontrolirati brzinu ventilatora na temelju trenutnog položaja loptice. Kvalitetno podešavanje PID regulator, omogućuje fino upravljanje sustava kako bi se minimalizirale oscilacije i postigla stabilna visina loptice uz minimalnu pogrešku. U ovom projektu, Arduino Uno R3 razvojna pločica koristi se kao centralni dio sustava koji prikuplja podatke s ultrazvučnog senzora postavljenog na vrhu cijevi, obrađuje te podatke koristeći PID regulaciju i na temelju rezultata upravlja brzinom ventilatora putem PWM (eng. *Pulse Width Modulation*) signala. Pomoću potencijometra povezanog na analogni ulaz Arduina, može se jednostavno postaviti željena visina loptice. Također, osim izrade sustava, opisać će se svi korišteni elementi u projektu.

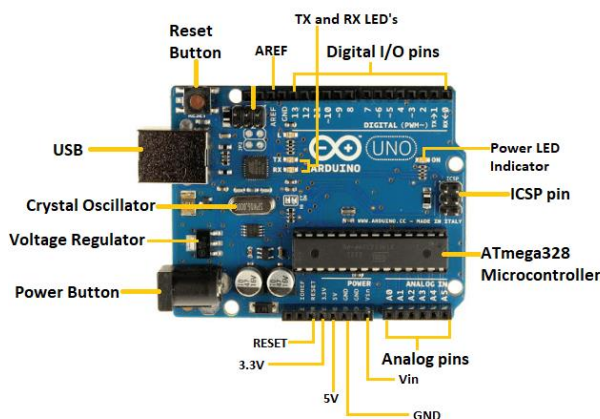


## 2. ARDUINO

Arduino je elektronička komponenta otvorenog koda temeljena na hardveru i softveru, čime je postala vrlo rasprostranjena u svijetu kreatora kao što su hobisti, edukatori, inženjeri i profesionalci za razvoj elektroničkih projekata. Tijekom godina, Arduino je bio mozak tisućama projekata, a prvi put se pojavio u Italiji sredinom 2000-ih. Njegov dizajn stvara brz i jednostavan razvoj uređaja i naprava, kako bi omogućio korisnicima da lakše povežu senzore i druge elektroničke komponente, te ih potom programiraju putem programskog jezika temeljenog na C/C++. Podržava tri uobičajena operacijska sustava Windows, Linux i Mac. Dizajneri su ga nazvali Arduino prema kafiću u kojem su se sastajali i radili na ovom projektu [5].

### 2.1. Arduino Uno

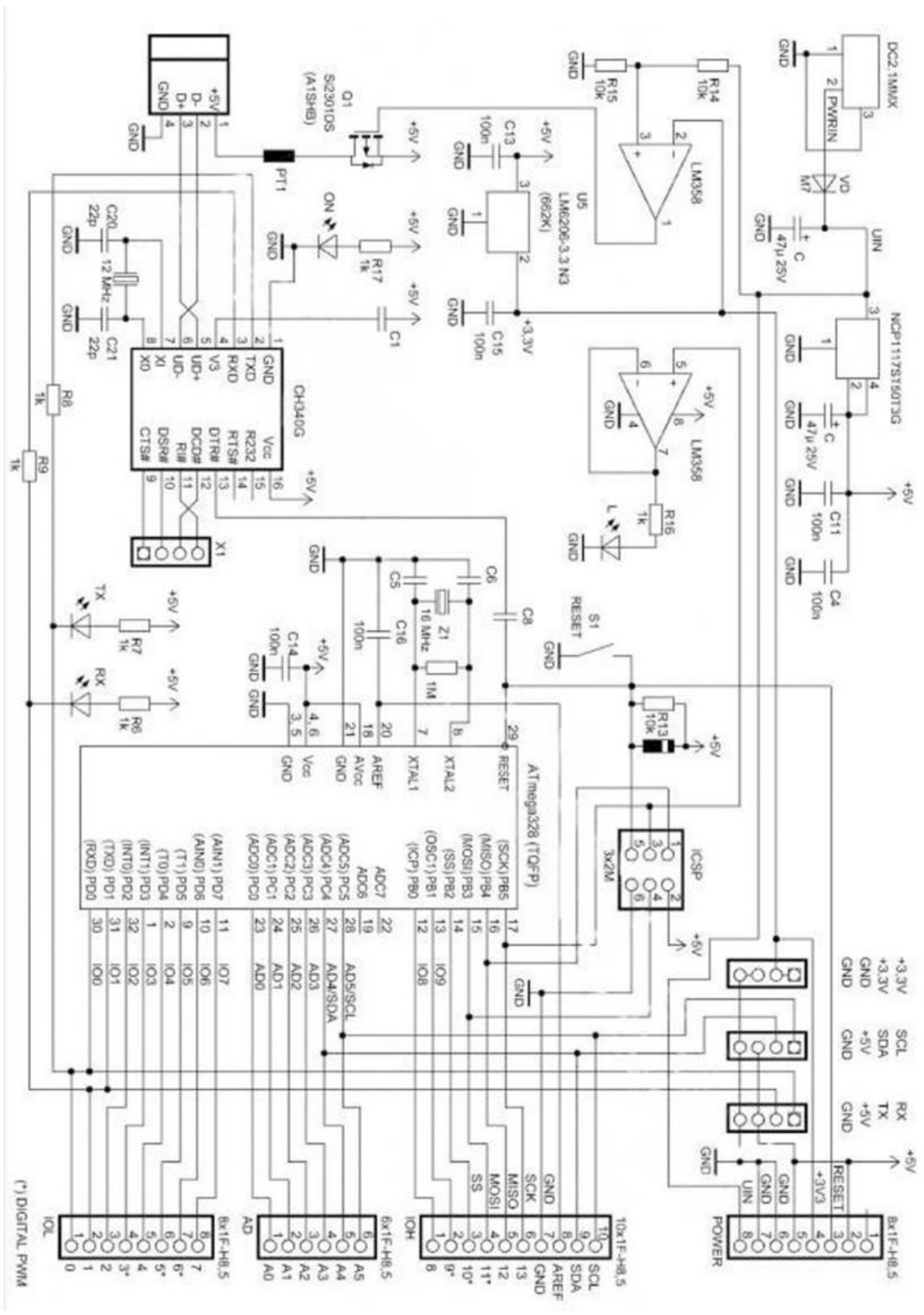
Arduino Uno, prikazan na slici 2.1., najčešća je razvojna pločica iz palete drugih Arduino razvojnih pločica zbog svoje jednostavnosti, učinkovitosti i cijene. Pokreće ga ATmega328P mikrokontroler, a sastoji se od 14 digitalnih I/O (eng. *Input/Output*) pinova, od kojih su 6 pinova 8-bitni PWM pinovi, 6 pinova su 10-bitni analogni ulazi i osnovnih komunikacijskih priključaka kao što su SPI (eng. *Serial Peripheral Interface*), I2C (eng. *Inter-integrated circuit*) i UART (eng. *Universal Asynchronous Receiver/Transmitter*). Zbog USB (eng. *Universal Serial Bus*) priključka na Arduino pločici, kabel koji se spaja s Arduino-om, djeluje kao serijski priključak i kao izvor napajanja za sučelje pločice. Takva dvostruka funkcionalnost čini ga jedinstvenim za preporuku i lakim za korištenje početnicima. Dijelovi prikazani na slici 2.1., objašnjeni su u tablici 2.1. Na slici 2.2 prikazana je shema Arduino razvojne pločice [2].



Slika 2.1. *Arduino Uno pločica* [1].

Tablica 2.1. *Dijelovi Arduino pločica* [1].

Mikrokontroler ATmega328	Kombinira memoriju (SRAM, EEPROM i Flash), analogno-digitalni pretvornik, SPI serijske priključke, I/O linije, registre, mjerače vremena, vanjske i unutarnje prekide i oscilator.
ICSP konektor	(eng. <i>In-Circuit Serial Programming</i> ) konektor omogućuje korisniku programiranje mikrokontrolera bez <i>bootloader</i> -a.
LED indikator napajanja	Uključeno stanje LED-a pokazuje da je napajanje aktivno.
Digitalni I/O pinovi	Digitalni pinovi imaju vrijednost HIGH ili LOW. Pinovi označeni brojevima od D0 do D13 su digitalni pinovi.
TX i RX LED diode	Uspješan protok podataka predstavljen je treperenjem ovih LED dioda.
AREF	Analogni referentni pin koristi se kao referentan napon AD pretvorbe.
Reset tipkalo	Resetira Arduino.
USB	Omogućuje povezivanje pločice s računalom i programiranje mikrokontrolera.
Kristalni oscilator	Koristi kristale da osigura ulaz takta u mikrokontroler, tj. određuje broj operacija koje će se izvršiti u sekundi.
Regulator napona	Smanjuje ulazni napon (7 – 12 V) koji se dovodi na Vin pin i pretvara ga u 5 V koji služi za mikrokontroler, a drugi je regulator od 3.3 V kojeg možemo koristiti za napajanje vanjskih modula.
GND	Pin za uzemljenje
Vin	Ulazni napon.
Analogni pinovi	Pinovi označeni brojevima od A0 do A5.



Slika 2.2. Prikaz sheme Arduino pločice [4].

## 2.2. Mikrokontroler ATmega328

ATMega328P je jedan od najpoznatijih i najčešće korištenih mikrokontrolera u svijetu elektronike i ugradbenih sustava, razvijen od strane tvrtke Atmel. ATMega328P je 8-bitni mikrokontroler temeljen na AVR arhitekturi, koja koristi RISC (eng. *Reduced Instruction Set Computing*) pristup za bržu i učinkovitiju obradu podataka. Mikrokontroler radi na maksimalnoj frekvenciji od 20 MHz i ima 32 KB *flash* memorije za pohranu programa [3, 6].

ATMega328P podržava do 23 višenamjenske I/O linije, što omogućuje povezivanje i upravljanje velikim brojem senzora, aktuatora i drugih perifernih uređaja. Mikrokontroler također ima ugrađeni AD (analogno-digitalni) pretvornik s 10-bitnom rezolucijom, omogućujući precizno čitanje analognih signala. Sadrži tri 8-bitna i jedan 16-bitni timer, koji omogućuju implementaciju složenih funkcija kao što su generiranje PWM signala, mjerenje trajanja impulsa i brojanje događaja. ATMega328P podržava nekoliko komunikacijskih protokola, uključujući UART, SPI i I2C. Ovi protokoli omogućuju jednostavnu i brzu razmjenu podataka između mikrokontrolera i drugih uređaja, kao što su senzori, memorijski moduli, zaslone, ili drugi mikrokontroleri. UART je posebno koristan za serijsku komunikaciju s računalom ili drugim serijskim uređajima, dok SPI i I2C omogućuju povezivanje više perifernih uređaja na zajedničku sabirnicu, čime se štedi broj potrebnih pinova. Popularnost ATMega328P mikrokontrolera proizlazi iz njegove primjene u Arduino platformi, posebno na modelima kao što su Arduino Uno. Arduino Uno je postao sinonim za jednostavan ulazak u svijet mikrokontrolera zahvaljujući jednostavnom programiranju, dostupnosti brojnih biblioteka i aktivnoj zajednici koja pruža podršku i resurse. ATMega328P na Arduino omogućuje korisnicima da lako programiraju mikrokontroler koristeći Arduino IDE (eng. *Integrated Development Environment*), koji pojednostavljuje rad sa složenim funkcijama kao što su upravljanje LED-icama, sensorima, motorima, i drugim komponentama. Uz Arduino, mikrokontroler ATMega328P se koristi i u brojnim samostalnim projektima gdje je potrebno pouzdano i fleksibilno rješenje za upravljanje. Njegova jednostavna arhitektura, niska potrošnja energije i široka podrška čine ga idealnim za primjene u automatizaciji doma, robotici, prenosivim uređajima, i mnogim drugim područjima [3, 6].

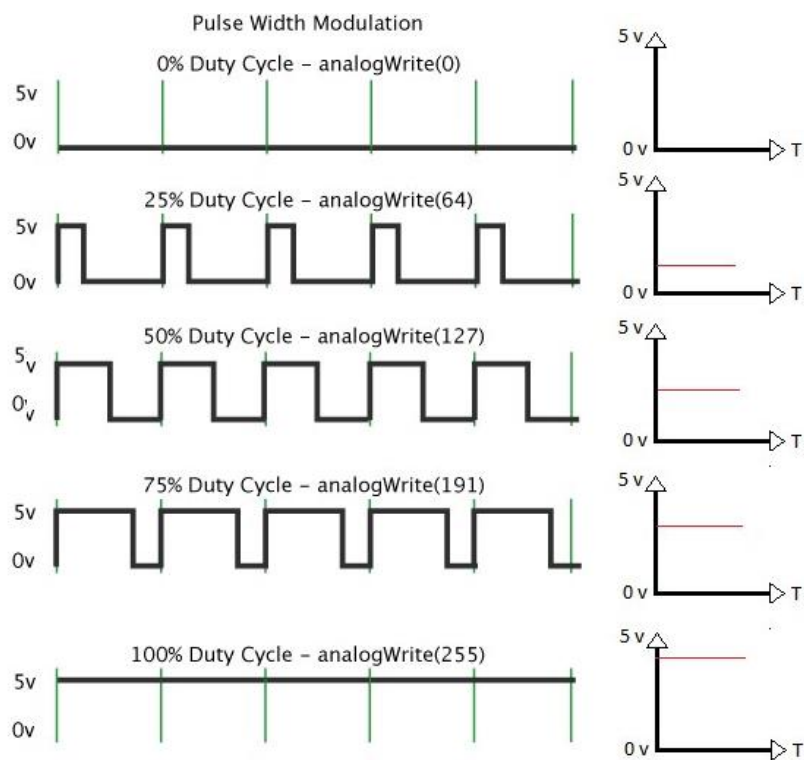
Mikrokontroler ATMega328P je poznat po svojoj pouzdanosti, stabilnosti i jednostavnosti korištenja. Međutim, ATMega328P ima i svojih ograničenja. Kao 8-bitni mikrokontroler, nije prikladan za vrlo složene ili računalno intenzivne aplikacije koje zahtijevaju veću procesorsku

snagu ili više memorije. U takvim slučajevima, korisnici se često okreću jačim mikrokontrolerima ili mikroprocesorima.

### 2.3. PWM modulacija

PWM ili modulacija širine impulsa je tehnika koja se koristi za dobivanje analognih vrijednosti pomoću digitalnih impulsa konstante amplitude. Osnovna ideja PWM-a je brzo uključivanje i isključivanje izvora napajanja, pri čemu se mijenja omjer vremena uključenja i isključenja unutar jednog ciklusa. Kod 8-bitnih PWM signala, upisivanjem različitih vrijednosti između 0 i 255, možemo dobiti vrijednosti izlaznog napona između 0 i ulaznog napona [7].

Na slici 2.3, okomite zelene linije predstavljaju jednu vremensku periodu PWM signala. Prvi graf prikazuje potpuno isključen, dok peti graf prikazuje potpuno uključen izvod mikrokontrolera. Na drugom, trećem i četvrtom grafu izvod je povremeno uključen. Vrijeme u kojem je izvod uključen nazivamo širina pulsa, a vrijeme između dva ponavljanja pulsa nazivamo periodom.



Slika 2.3. Grafički prikaz PWM impulsa [7].

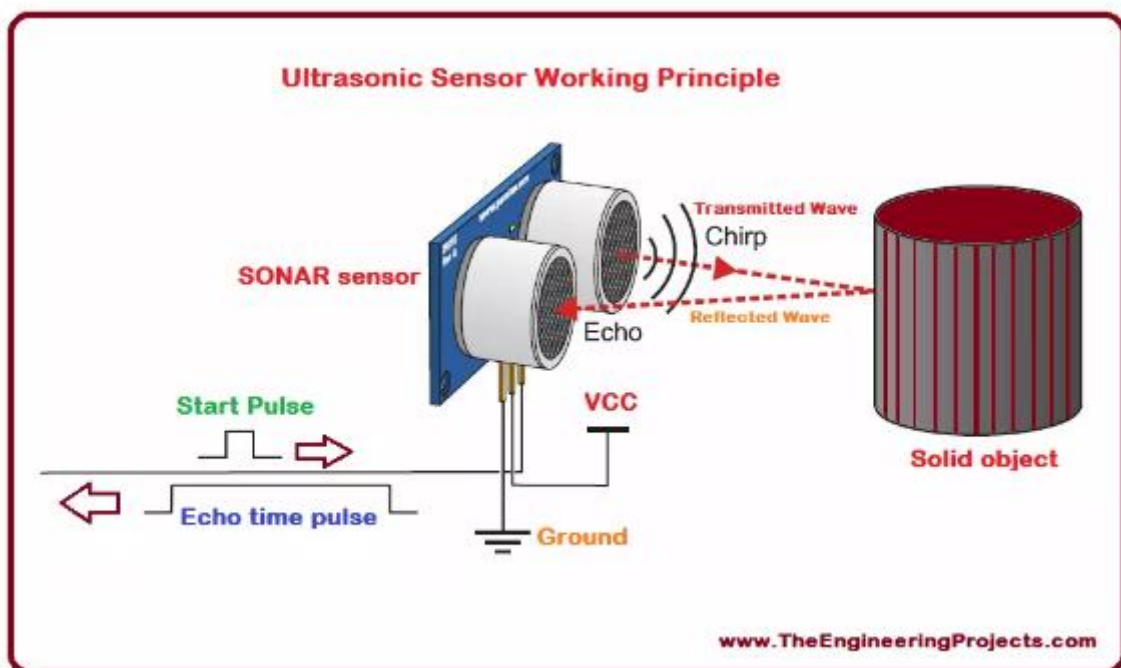
Tablica 2.2 prikazuje PWM frekvencije i pinove Arduino pločica koje ga podržavaju.

Tablica 2.2. *Pinovi i frekvencije različitih Arduino pločica [7].*

PLOČICA	PWM pinovi	PWM frekvencija
Uno, Nano, Mini	3, 5, 6, 9, 10, 11	490 Hz (pinovi 5 i 6: 980 Hz)
Mega	2-13, 44-46	490 Hz (pinovi 4 i 13: 980 Hz)
Leonardo, Micro, Yun	5, 6, 9, 10, 13 3, 11	490 Hz 980 Hz
Uno WiFi Rev.2	3, 5, 6, 9, 10	976 Hz
MKR pločice	0-8, 10, A3(18), A4(19)	732 Hz
MKR1000 WiFi	0-8, 10, 11, A3(18), A4(19)	732 Hz
Due	2-13	1000 Hz
Zero	3-13, A0(14), A1(15)	732 Hz
101	3, 9 5, 6,	490 Hz 980 Hz

### 3. ULTRAZVUČNI SENZOR HR-SC04

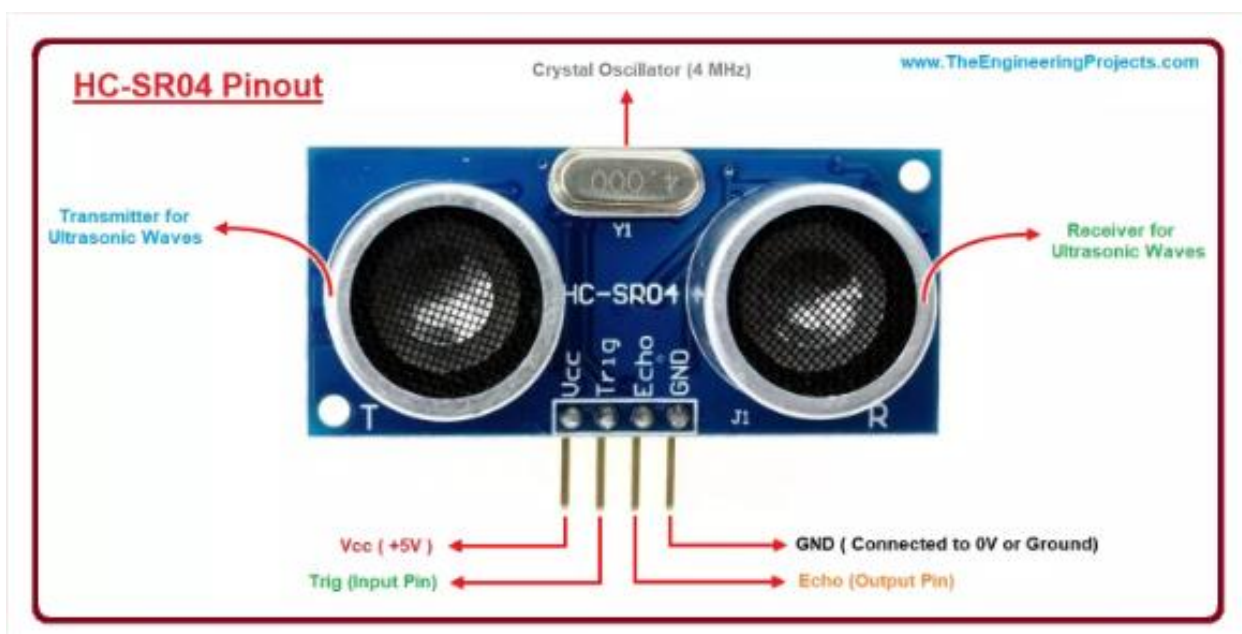
Ultrazvučni senzor HC-SR04 je uređaj za mjerenje udaljenosti objekata koristeći ultrazvučne valove. Posebno se koristi se u projektima koji uključuju mikrokontrolere, poput Arduina, ali i u drugim područjima robotike, sigurnosnih sustava, pametnih uređaja i drugih automatiziranih sustava. Zbog svoje jednostavnosti, pristupačne cijene i točnosti, korišten je diljem svijeta zbog načina rada sličnom sonaru koji se koristi na morskim plovilima. Ovaj uređaj radi na principu emitiranja ultrazvučnih valova (slika 3.1), tj. zvučnih valova frekvencije od 40 kHz, a koji su iznad granice ljudskog sluha. Senzor radi na naponu od 5 V i sastoji od dvije ultrazvučne sonde koje djeluju kao odašiljač i prijemnik. Kada je senzor aktiviran, odašiljač emitira kratki impuls ultrazvučnih valova koji se šire u zraku. Kada ti valovi naiđu na prepreku, reflektiraju se natrag prema senzoru, gdje ih prijemnik detektira. Kada prijemnik primi te impulse, generira se izlazni impuls čija je trajanje proporcionalno udaljenosti objekta ispred. Nadalje, HC-SR04 se pokazao učinkovitim u različitim primjenama kada je riječ o pouzdanosti, preciznosti i širokom rasponu mjerenja [10, 13].



Slika 3.1. Način rada ultrazvučnog senzora [13].

Prikazani na slici 3.2, pinovi modula su sljedeći:

- VCC – napaja senzor i spaja se na izlazni napon Arduina od 5 V.
- Trig – koristi se za početak mjerenja. Nakon što na njega dovedemo impuls visokog stanja u trajanju 10 $\mu$ s, senzor emitira 8 ultrazvučnih impulsa na frekvenciji od 40 kHz.
- Echo – kada senzor počne emitirati ultrazvučne valove, ovaj pin se postavlja u 1, a kada primi reflektirane valove postavlja ga natrag na 0. Prema tome, trajanje impulsa na ovom izlaznom pinu proporcionalno je udaljenosti.
- GND – nulti potencijal spojen na Arduino.



Slika 3.2. Pinovi ultrazvučnog HC-SR04 senzora [11].



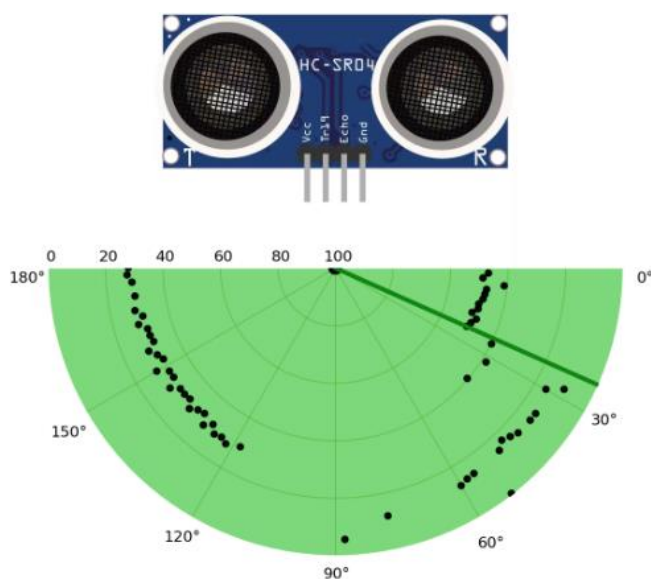
### 3.1. Specifikacije i primjena senzora

U tablici 3.1 nalaze se tehničke specifikacije za HR-SC04 ultrazvučni senzor udaljenosti.

Tablica 3.1. *Specifikacije HC-SR04 senzora* [10].

Radni napon	DC 5V
Radna struja	<15 mA
Radna frekvencija	40 kHz
Domet	2-400 cm
Točnost mjerenja	3 mm
Kut mjerenja	15 stupnjeva
Dimenzija	45 x 20 x 15 mm

Senzor ima uski kut mjerenja, obično oko 15 stupnjeva, što znači da dobro detektira objekte izravno ispred sebe, ali manje reagira na objekte izvan tog kuta. Na slici 3.3 prikazan je kut detekcije ultrazvučnog senzora.



Slika 3.3. *Kut detekcije ultrazvučnog senzora* [12].

Senzor HR-SC04 ima širok raspon primjena u industriji koji su usmjereni na mjerenje udaljenosti i smjera. Neki od primjena su: mjerenje dubine, mjerenje brzine i smjera, protuprovalni alarmi, ugradbeni sustavi, medicinski ultrazvuk i ovlaživači zraka. Osim spomenutih primjena, i u prirodi imamo životinje, kao što šišmiši, koje koriste principe slične ovom senzoru [11, 12].

### 3.2. Izračunavanje udaljenosti i ograničenja senzora

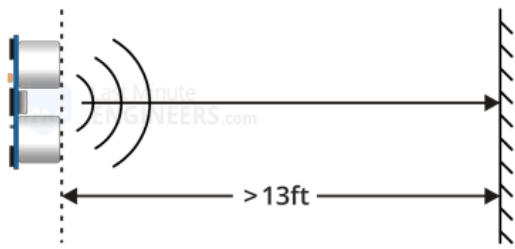
Udaljenost je izravno proporcionalna vremenu koje je ultrazvučnim valovima potrebno da se vrate na prijammnik. Udaljenost objekta od senzora možemo izračunati prema sljedećem izrazu:

$$S = \frac{t \cdot v}{2}, \quad (3.1)$$

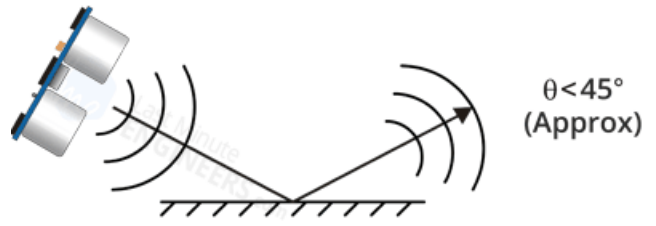
gdje je  $S$  mjerena udaljenost,  $v$  je brzina zvuka, dok je  $t$  proteklo vrijeme između odašiljanja i primitka vala. Pošto valovi moraju preći dvostruku udaljenost (prema predmetu, te natrag prema prijammniku) u (3.1) je izračunata udaljenost podijeljena s dva. Ova metoda mjerenja udaljenosti ima i svoje nedostatke. Ukoliko senzor nije kvalitetno pričvršćen te je u doticaju s predmetom kojem mjeri udaljenosti, dolazi do krivog očitavanja (jer senzor ima minimalnu udaljenost koju može izmjeriti), što se i događalo u praktičnoj izvedbi ovog sustava, kada je loptica došla u kontakt sa senzorom.

Problemi nastaju kada:

- je mjerena udaljenosti veća od 4 m (prikazano na slici 3.4),
- objekt ima reflektirajuću površinu, a mjerenje se vrši pod kutom, pa se valovi ne reflektiraju natrag prema senzoru (prikazano na slici 3.5),
- su dimenzije mjenog objekta premalene, pa ne dolazi do dovoljne refleksije (prikazano na slici 3.6),
- je mjereni objekt izrađen od mekog materijala koji apsorbira valove (prikazano na slici 3.7),
- dođe do promjene u okolišu, tj. mediju kroz kojeg ultrazvučni valovi prolaze. Temperatura, prašina, vlaga, te slični uvjeti mijenjaju brzinu propagacije vala kroz zrak.



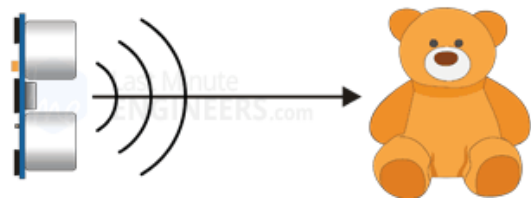
Slika 3.4. *Prevelika udaljenost.*



Slika 3.5. *Nemogućnost očitavanja zvuka pod kutom.*



Slika 3.6. *Premali objekt.*



Slika 3.7. *Nepravilni objekt.*

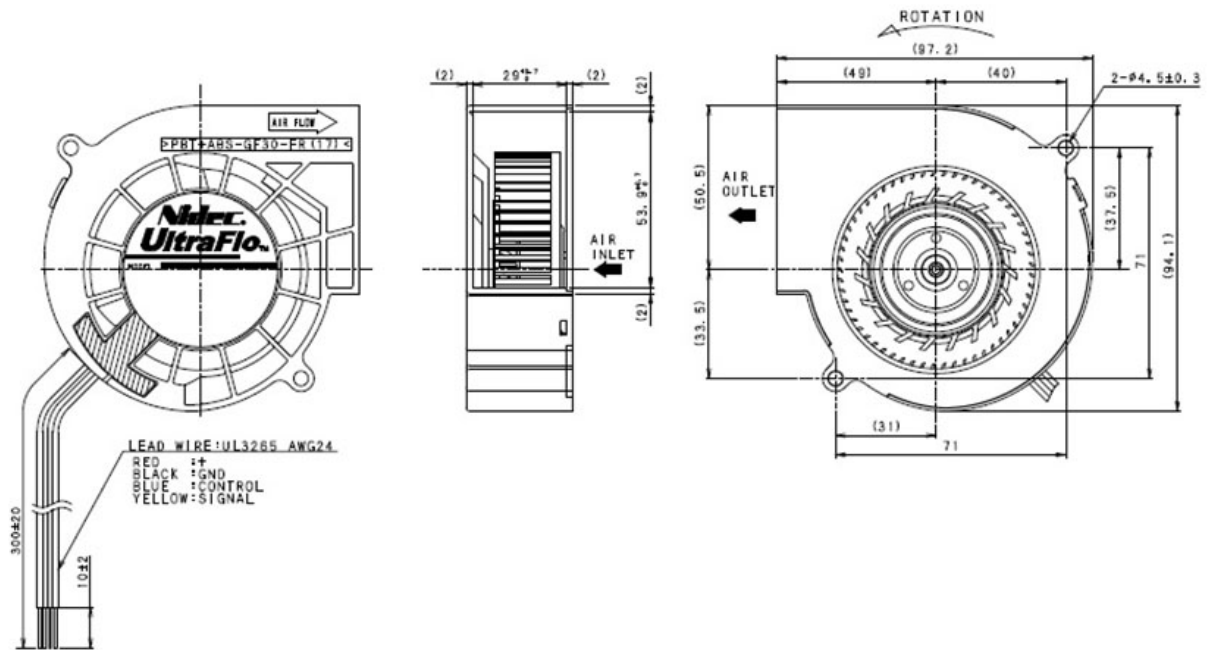
## 4. OSTALE KOMPONENTE SPOJA

### 4.1. Nidec ventilator

Nidec je jedan od vodećih proizvođača ventilatora i motora, poznat po visokokvalitetnim proizvodima koji se koriste u raznim industrijama. Ventilatori s 4 žice često se koriste u aplikacijama koje zahtijevaju preciznu kontrolu brzine, pouzdanost i nisku razinu buke, kao što su računala, serverski sustavi i općenito u elektronici za hlađenje komponenti. Boje žica kod takvih ventilatora su prilično standardizirane. Crvena i crna žica služe za dovođenje napona napajanja od 12V. Pomoću plave žice upravljamo brzinom vrtnje ventilatora koristeći PWM signal koji generira mikrokontroler. Pri radnom ciklusu od 100% ventilator radi na maksimalnoj brzini, dok pri radnom ciklusu od 50% brzina je upola manja zbog izmjeničnog uključivanja i isključivanja signala. Žuta žica služi kao povratna informacija o brzini vrtnje motora, te se na njoj generiraju dva impulsa po jednom okretaju ventilatora. Izgled centrifugalnog ventilatora može se vidjeti na slici 4.1, a njegove dimenzije na slici 4.2.



Slika 4.1. Nidec ventilator.

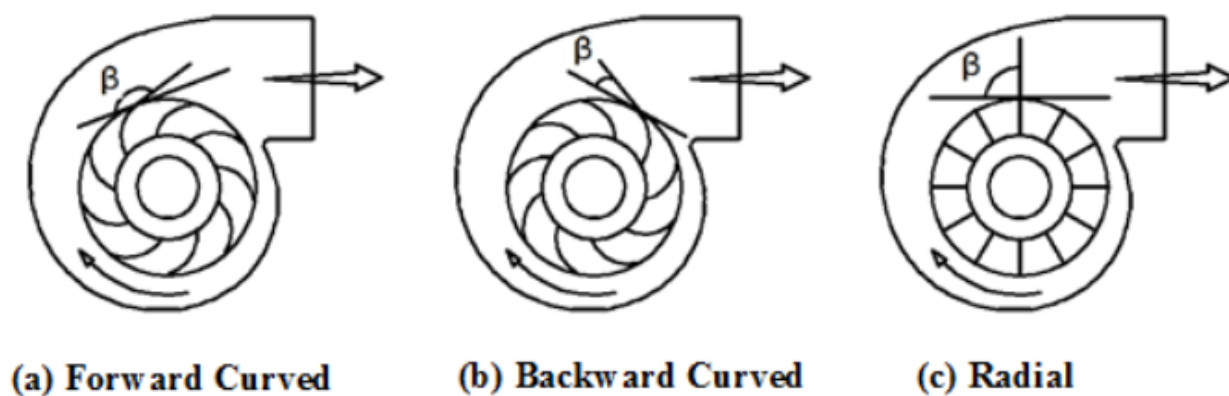


Slika 4.2. Dimenzije Nidec ventilatora [23].

Rad centrifugalnog ventilatora ovisi o rotoru koji se sastoji od niza zakrivljenih lopatica povezanih na rotirajuću osovinu koju pokreće motor. Rotacijom rotora, stvara se centrifugalna sila koja uzrokuje radijalno kretanje zraka prema van. Ovakva vrsta ventilatora ima dvostruku funkcionalnost zbog mogućnosti reverzibilnog načina rada, što se koristi u odušnim sustavima.

Postoje tri vrste lopatica:

- zakrivljene lopatice prema naprijed u smjeru rotacije koje ostvaruju veći protok zraka pri nižim brzinama, ali s manjim tlakom, pa samim time i veću potrošnju energije (prikazano na slici 4.3),
- zakrivljene lopatice prema natrag u smjeru suprotnom od rotacije, što omogućuje veću učinkovitost te su prikladna za rad pri većim brzinama (prikazano na slici 4.3),
- ravne lopatice koje stoje radijalno i jednostavnog su oblika (prikazano na slici 4.3).



Slika 4.3. Vrste lopatica [24].

Ventilatore možemo podijeliti na aksijalne i centrifugalne. Razlika je u konstrukciji, tlaku i načinu usmjeravanja zraka. Aksijalni ventilator (lijevo na slici 4.4) ima motorom pogonjeno vratilo s propelerom koje paralelno uvlači i gura zrak velikog volumena, ali niskog tlaka, prema van u smjeru osovine. Centrifugalni ventilator (desno na slici 4.4) ima motorom pogonjeno vratilo složenijeg dizajna, pri čemu zrak ulazi okomito, a izlazi pod kutom od  $90^\circ$  u odnosu na smjer ulaska. U usporedbi s aksijalnim ventilatorom, centrifugalni ventilator stvara znatno veći tlak te se koristi za elemente kojima je potrebno učinkovitije hlađenje [14].



Slika 4.4. Vrste ventilatora [14].

## 4.2. Linearni potencijometar

Linearni potencijometar, prikazan na slici 4.5, je komponenta koja se koristi u elektroničkim krugovima za varijabilnu kontrolu otpora. Ova vrsta potencijometra je linearna, što znači da je otpor proporcionalan poziciji klizača duž otporne staze. Ako je klizač na početku otpornog elementa, izlazni otpor između klizača i jednog krajnjeg terminala bit će minimalan ili jednak 0, dok će između klizača i suprotnog terminala biti maksimalan.

Potencijometar se sastoji od dvije komponente:

- Otporni element - fiksna traka materijala s fiksnim otporom koja se proteže između dva krajnja terminala potencijometra.
- Klizač - pokretni kontakt koji se može pomicati duž otpornog elementa. Klizač je spojen na srednji izvod potencijometra, a položaj klizača određuje otpor između srednjeg izvoda i preostala dva krajnja izvoda.

Potrebno je paziti da se ne prekorači maksimalna snaga da ne bi došlo do opterećenja i uništenja potencijometra. Prema modelu potencijometra može se iščitati toleranciju otpora, obično  $\pm 20\%$ , i radnu temperaturu, u prosjeku od  $-55^{\circ}\text{C}$  do  $+125^{\circ}\text{C}$ . Kao i ostali elementi, posebice oni mehanički, i potencijometar ima vijek trajanja kojeg možemo primijetiti na habanju otporne staze, što smanjuje točnost i pouzdanost potencijometra.



Slika 4.5. Linearni potencijometar od  $10\text{k}\Omega$ .

### 4.3. Izvor napajanja

Akumulator Yuasa YB4L-B 12V, prikazan na slici 4.6, je korišten kao izvor napajanja u završnom radu specifikacija prikazanih u tablici 4.1. To je olovno-kiselinski akumulator sa šest ćelija, koje zajednički daju napon od 12 V s kapacitetom od 4 Ah. Zahtjeva povremeno održavanje, uključujući provjeru i dopunjavanje elektrolita destiliranom vodom, što je tipično za standardne olovno-kiselinske akumulatore. Opremljen je ventilima za ispuštanje plinova, što sprječava nakupljanje pritiska unutar akumulatora. Najčešće se koristi u motorima i skuterima.



Slika 4.6. Yuasa akumulator [15].

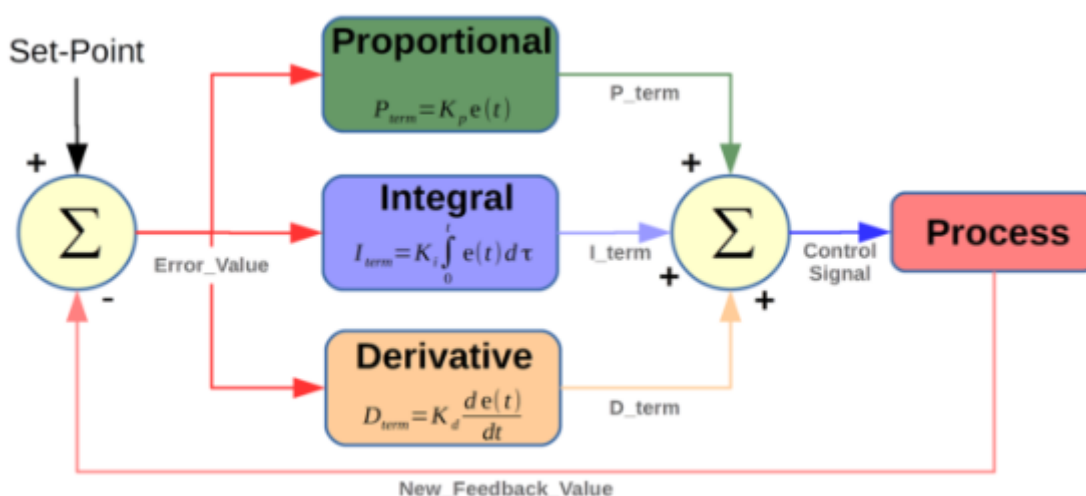
Tablica 4.1. Specifikacije Yuasa akumulatora [15].

Kapacitet	4 Ah
Vršna struja	45 A
Polaritet	Desni +
Dimenzije (D x Š x V)	120 x 70 x 92 mm
Težina	1.5 kg



## 5. PID REGULACIJA

PID regulacija (slika 5.1) je jedna od najčešće korištenih metoda za upravljanje procesima u industriji, robotici i automatskom upravljanju. PID regulator omogućuje precizno i stabilno održavanje željene vrijednosti neke fizikalne veličine (npr. temperature, brzine, položaja, itd.) unatoč smetnjama i promjenama u sustavu. Sastoji se od tri osnovna koeficijenta: proporcionalnog, integracijskog i derivacijskog člana.



Slika 5.1. Blok dijagram PID regulatora [16].

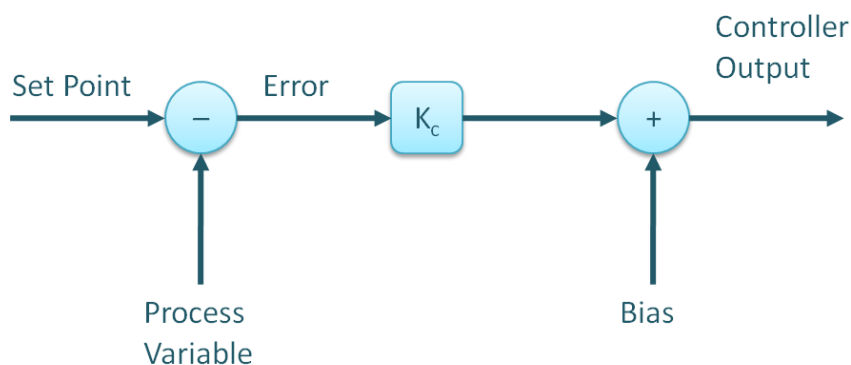
Promjenom proporcionalnog, integracijskog te derivacijskog koeficijenta, možemo postići različita djelovanja s ciljem postizanja željenog odziva sustava. Ulaz PID regulatora je signal greške, a izlazni signal se generira na način da se postigne željeno podešavanje sustava. Djelovanje PID kontrolera možemo izraziti kao:

$$e_a(t) = e(t) + T_d \frac{de(t)}{dt} + K_i \int e(t) dt, \quad (5.1)$$

gdje je  $e(t)$  trenutna pogreška,  $T_d$  je derivacijski koeficijent,  $\frac{de(t)}{dt}$  je brzina promjene pogreške,  $K_i$  je integracijski koeficijent,  $\int e(t) dt$  je integral pogreške, dok je  $e_a(t)$  ukupni izlaz PID regulatora.

Upravljanje sustavom je moguće i od strane čovjeka, tako što operater povremeno očitava varijablu procesa, ovisno koja je potrebna (npr. temperature, brzine, položaja) i prilagodi varijablu upravljanja. Radi na principu ON-OFF regulatora, gdje su moguća dva kontrolna stanja, uključeno ili isključeno. Takva ograničena primjena nije toliko rasprostranjena zbog inherentnih oscilacija i same kvalitete PID regulatora pa se rijetko koristi. PID regulator održava izlaz tako da postoji što manja greška između procesne i referentne varijable [17].

Proporcionalni ili P član reagira proporcionalno pogrešci  $e(t)$ , tj. razlici između željene (referentne) vrijednosti i stvarne vrijednosti (procesne varijable). Rezultirajuća pogreška se množi s proporcionalnom konstantom da bi se dobio izlaz iz regulatora. Ako je vrijednost pogreške jednaka nuli, tada je izlaz regulatora nula. Prema slici 5.2. pomoću  $K_c$  člana se povećava brzina odziva, ali je uvijek prisutna greška u stacionarnom stanju, pa često zahtijeva ručno resetiranje ako se koristi bez ostalih članova [16].



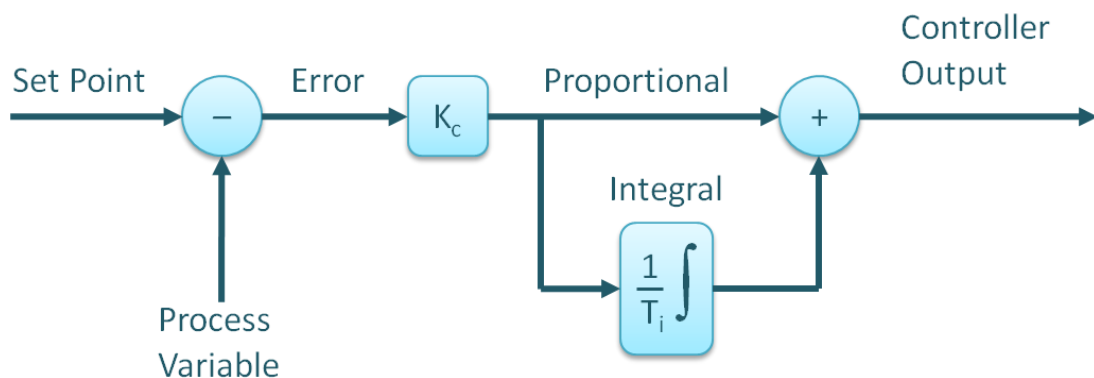
Slika 5.2. P regulator [16].

Integracijski član ili I član osigurava uklanjanje greške u stacionarnom stanju koja postoji kad se koristi samo P član. Integrira pogrešku tijekom vremenskog razdoblja dok se vrijednost pogreške ne izjednači s nulom prema izrazu:

$$e_a(t) = e(t) + K_i \int e(t)dt, K_i = const. \quad (5.2)$$

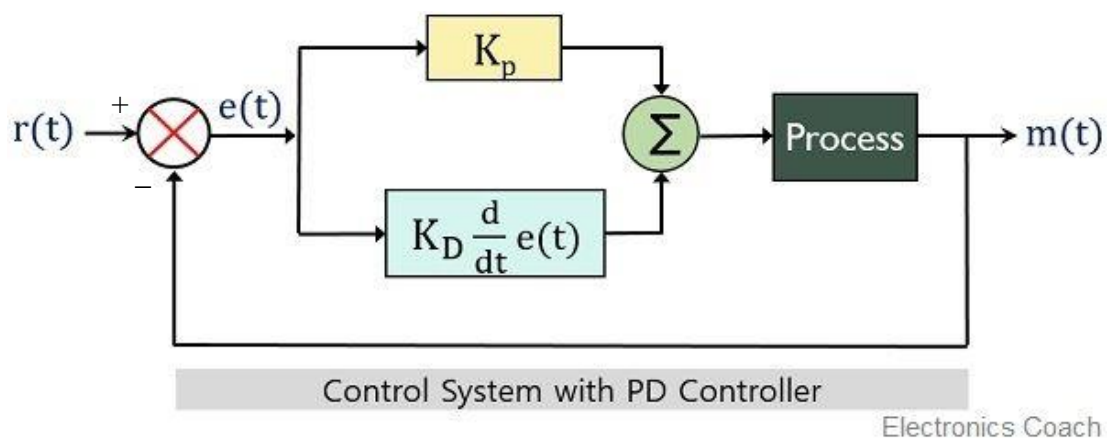
Izlaz integracijskog člana smanjuje se kada se pojavi negativna pogreška, te ograničava brzinu i stabilnost sustava. Smanjenjem integracijskog pojačanja  $K_i$ , brzina odziva se povećava. Kako se

poveća pojačanje I-regulatora, smanjuje se i pogreška u stacionarnom stanju. Zato se PI regulator koristi gdje nije potrebna velika brzina odziva. Na slici 5.3 prikazana je blokovska shema PI regulatora [17].



Slika 5.3. *PI regulator* [16].

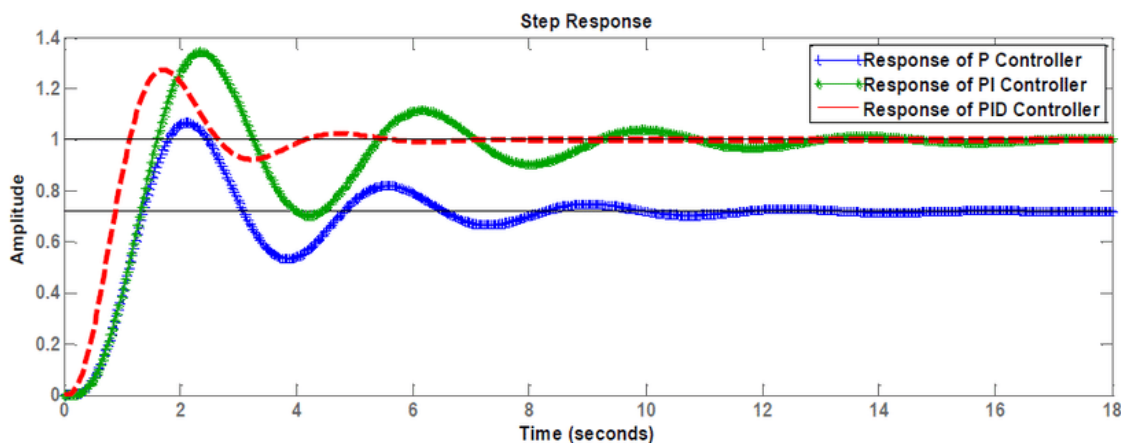
Derivacijski ili D član smanjuje oscilacije i povećava stabilnost sustava, ali ima problem predviđanja budućeg ponašanja pogreške. Na slici 5.4 prikazan je PD regulator.



Slika 5.4. *PD regulator* [21].

Kombinacijom svih članova, dobiva se PID regulacija. Jedan od načina podešavanja  $K_p$ ,  $K_i$  i  $K_d$  je igra pogađanja jer iako u teoriji postoje metode za podešavanje parametara, u praksi su često te metode neprimjenjive. Najlakši i najbrži način je postavljanjem  $K_i$  i  $K_d$  članova jednakim nuli. Metodom pokušaja povećamo  $K_p$  član dok sustav ne dostigne oscilirajuće ponašanje. Nakon toga, počinje se mijenjati  $K_i$  član sve dok sustav ne prestane oscilirati. Mijenjanjem  $K_d$  člana se onda pokuša ubrzati odziv sustava. Sljedeći način podešavanja je tehnika otvorene petlje gdje se promatra krivulja odziva i ručno podešava izlaz kontrolera. Zatim se izračuna nagib, mrtvo vrijeme

i vrijeme uspona krivulje zbog zamjene tih vrijednosti sa P, I i D jednadžbama da se dobiju vrijednosti pojačanja PID regulatora. Grafičkim prikazom na slici 5.5 ukazane su razlike svakog regulatora [17].



Slika 5.5. Grafički prikaz odziva regulatora [22].

Jedna od najpoznatijih i najpreciznijih metoda podešenja parametara PID regulatora je Zeigler-Nicholsova metoda. Kao i kod metode pokušaja,  $K_i$  i  $K_d$  su jednake nuli, a  $K_p$  se očitava kad se sustav dovede do granice osciliranja. U tom trenutku  $K_p$  se izjednači sa  $K_u$  (eng. *Ultimate Gain*), te je moguće očitavanje periode oscilacija ( $T_u$ ). Pomoću izraza u tablici 5.1 moguće je izračunati parametre P, PI, te PID regulatora [16, 17, 20].

Tablica 5.1. Ziegler-Nicholsova metoda [20].

Tip regulatora	$K_p$	$K_i$	$K_d$
P	$0.5K_u$	/	/
PI	$0.45K_u$	$\frac{0.54K_u}{T_u}$	/
PID	$0.6K_u$	$\frac{1.2K_u}{T_u}$	$\frac{3K_u \cdot T_u}{40}$

Tyrus-Luybenova tehnika ugađanja, čiji su izrazi za izračun parametara dani u tablici 5.2, izvedena je iz Ziegler-Nicholsove metode. Jedina razlika je u izrazima za izračun koeficijenata.

Tablica 5.2. Tyreus-Luybenova metoda [20].

Tip regulatora	$K_p$	$K_i$	$K_d$
PI	$\frac{Ku}{3.2}$	$2.2T_u$	/
PID	$\frac{Ku}{2.2}$	$2.2T_u$	$\frac{T_u}{6.3}$

Cohen-Coonova metoda, čiji su izrazi za izračun parametara dani u tablici 5.3, također je izvedena iz Ziegler-Nicholsove metode, ali nije prikladna za svaku primjenu. Parametri metode su sljedeći:

- $g_p$  je dobitak procesa koji se računa dijeljenjem postotaka promjene u procesnoj varijabli (PV) i promjene u izlazu regulatora (CO).
- $\tau$  je vremenska konstanta koja se računa vremenskom razlikom između presjeka na kraju mrtvog vremena i PV-a koji doseže 63% svoje ukupne promjene.
- $t_d$  je mrtvo vrijeme koje se računa vremenskom razlikom između promjene CO i sjecišta tangencijalne linije i izvorne PV razine.

Tablica 5.3. Cohen-Coonova metoda [20].

Tip regulatora	$K_c$	$T_i$	$T_d$
P	$\frac{1.03}{g_p} \left( \frac{\tau}{t_d} + 0.34 \right)$	/	/
PI	$\frac{0.9}{g_p} \left( \frac{\tau}{t_d} + 0.092 \right)$	$3.33t_d \left( \frac{\tau}{t_d} + 0.34 \right)$	/
PD	$\frac{1.24}{g_p} \left( \frac{\tau}{t_d} + 0.129 \right)$	/	$0.27t_d \left( \frac{\tau - 0.32t_d}{\tau + 0.129t_d} \right)$
PID	$\frac{1.35}{g_p} \left( \frac{\tau}{t_d} + 0.185 \right)$	$2.5t_d \left( \frac{\tau + 0.18t_d}{\tau + 0.611t_d} \right)$	$0.37t_d \left( \frac{\tau}{\tau + 0.185t_d} \right)$

## 5.1. Primjena PID regulacije

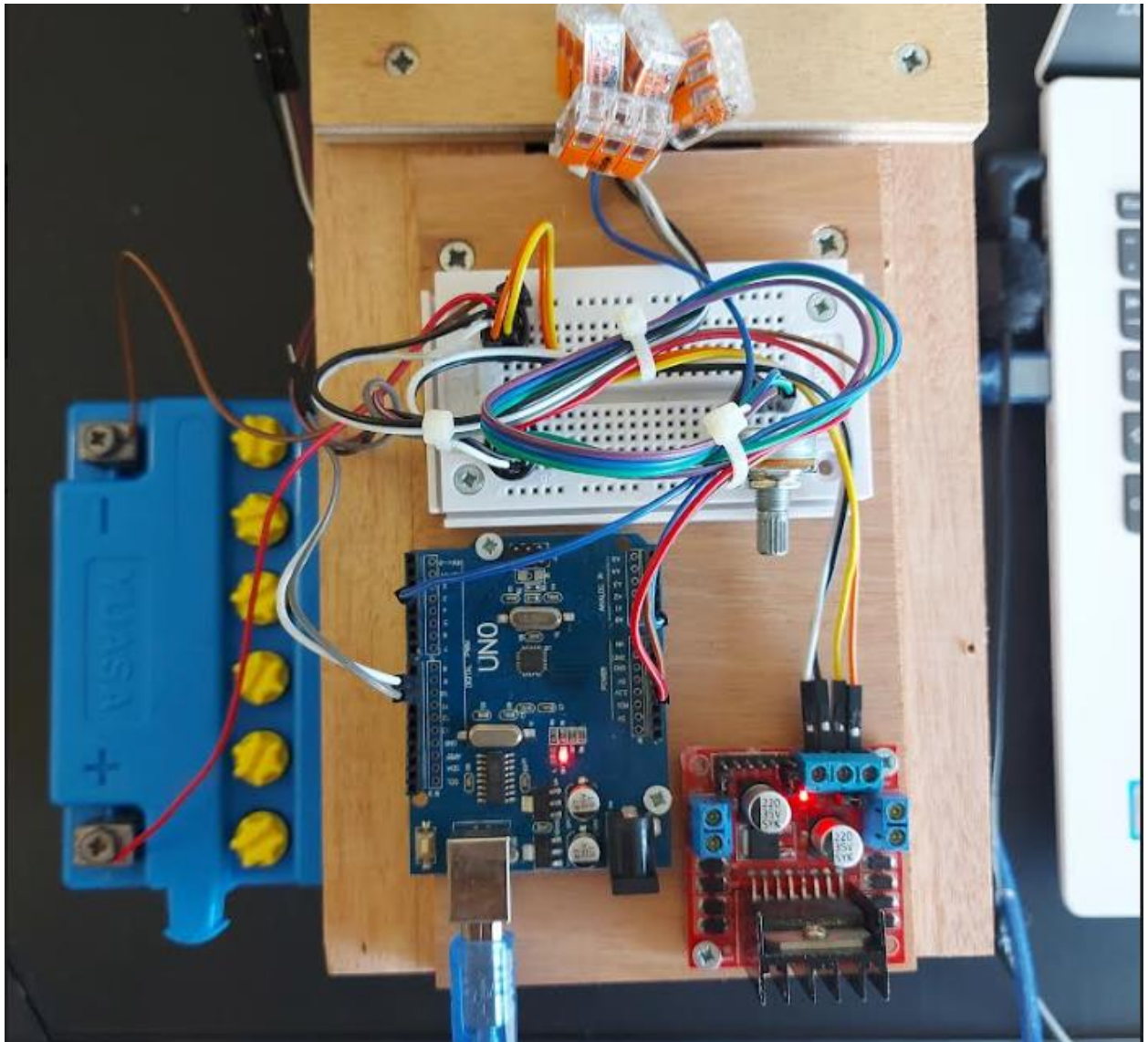
Primjenu PID regulacije koristimo pri kontroli temperatura ili razine, kada je potrebno zadržavanje temperature (grijanje, klimatizacija i ventilacija itd.) odnosno kontinuirano mjerenje razine. Može se koristiti pri kontroli brzine i položaja prema nekom položaju ili pomaku. Također, pri kontroli protoka, gdje je potrebna ON-OFF kontrola za upravljanje ventila ili pumpa. Moguća je i kontrola tlaka u sustavu, ali kod ove kontrole dovoljan je P regulator zbog potrebe brzog odziva. PID regulaciju pronalazimo u zrakoplovstvu i kontroli motora zbog stabilizacije leta kako bi održali visinu i smjer unatoč vjetru.

## 5.2. Prednosti i nedostaci PID regulacije

Prednosti PID regulatora su njegova pouzdanost i jednostavnost za razumijevanje i implementaciju. Podržavan je u svim softverima za programiranje PLC-a (eng. *Programmable Logic Controller*). Također, ima širok raspon primjena, tj. manja je mogućnost unošenja pogrešaka tijekom implementacije, za razliku od mnogih drugih regulatora koji su prikladni samo za određene zadatke, što naposljetku rezultira pouzdanijim sustavom. PID regulatore mnogo je lakše ugoditi nego druge metode. Ipak, nije uvijek lako odrediti kako će se ponašati u sustavu [20].

Jedan od nedostataka je što performanse PID regulatora jako ovise o odabranim parametrima regulatora. Ako parametri nisu dobro podešeni, neće raditi kako se očekuje i neće dati optimalne rezultate potrebne za njegovu učinkovitost. Čak i s podešenim PID regulatorom, još uvijek postoje kompromisi koji se moraju uzeti u obzir. Nadalje, PID se može koristiti bez puno problema kada je dinamika procesa linearna i vremenski nepromjenjiva. Međutim, ako ti uvjeti nisu ispunjeni, performanse regulatora će se značajno smanjiti. PID regulator ne može razlikovati željeni ulaz od smetnji, što može uzrokovati probleme izbacivanjem procesa iz normalnog raspona rada. Kao rezultat toga, proces postaje zahtjevniji za kontrolu. Budući da PID regulator ima poznato kašnjenje u vremenu odziva, često prekasno reagira na promjene u nestabilnom procesu. To uzrokuje probleme s regulacijom konačnih varijabli jer ako postoje bilo kakve promjene unutar tog kašnjenja zbog oscilacija, PID regulator neće reagirati na njih do sljedećeg oscilacijskog ciklusa [20].

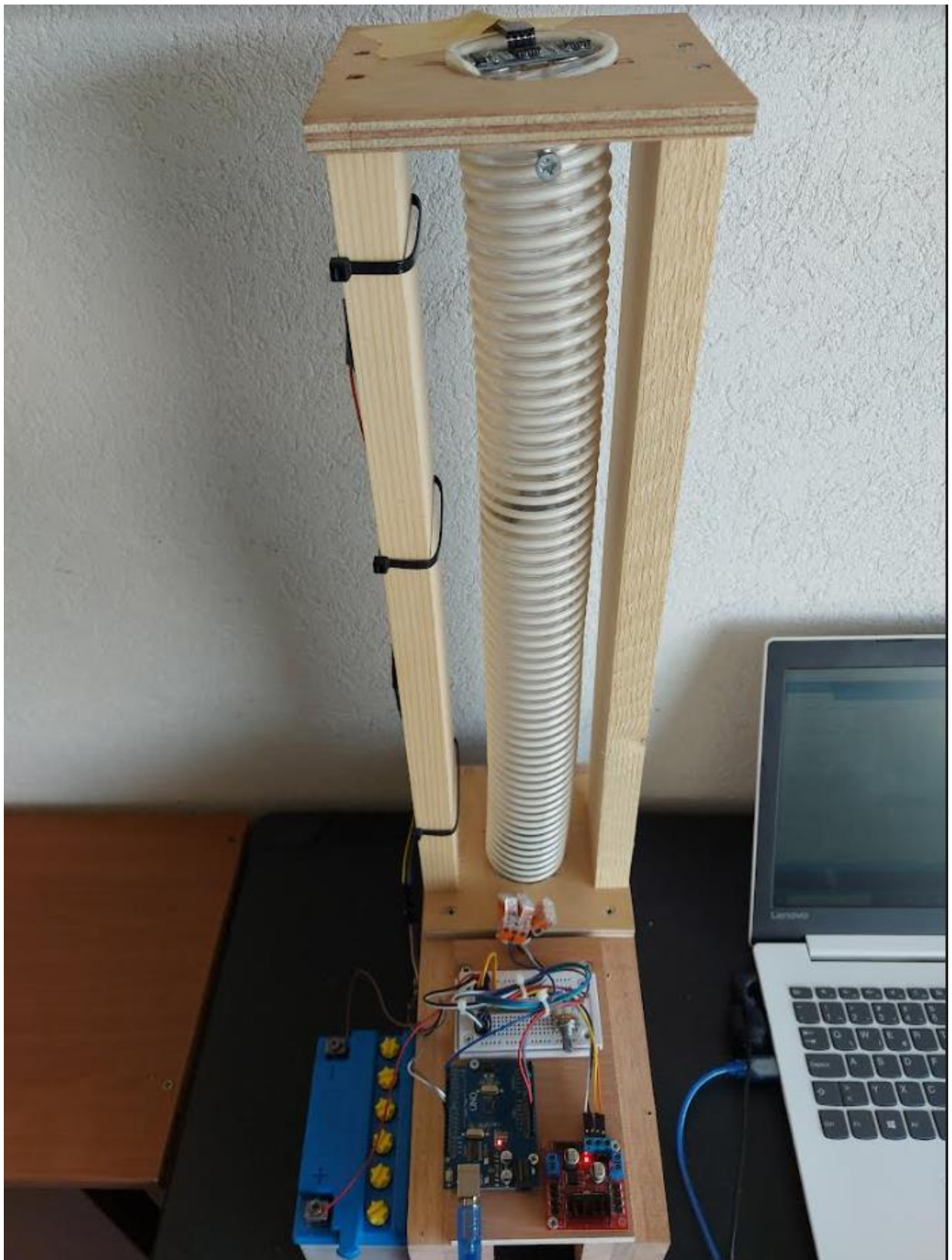




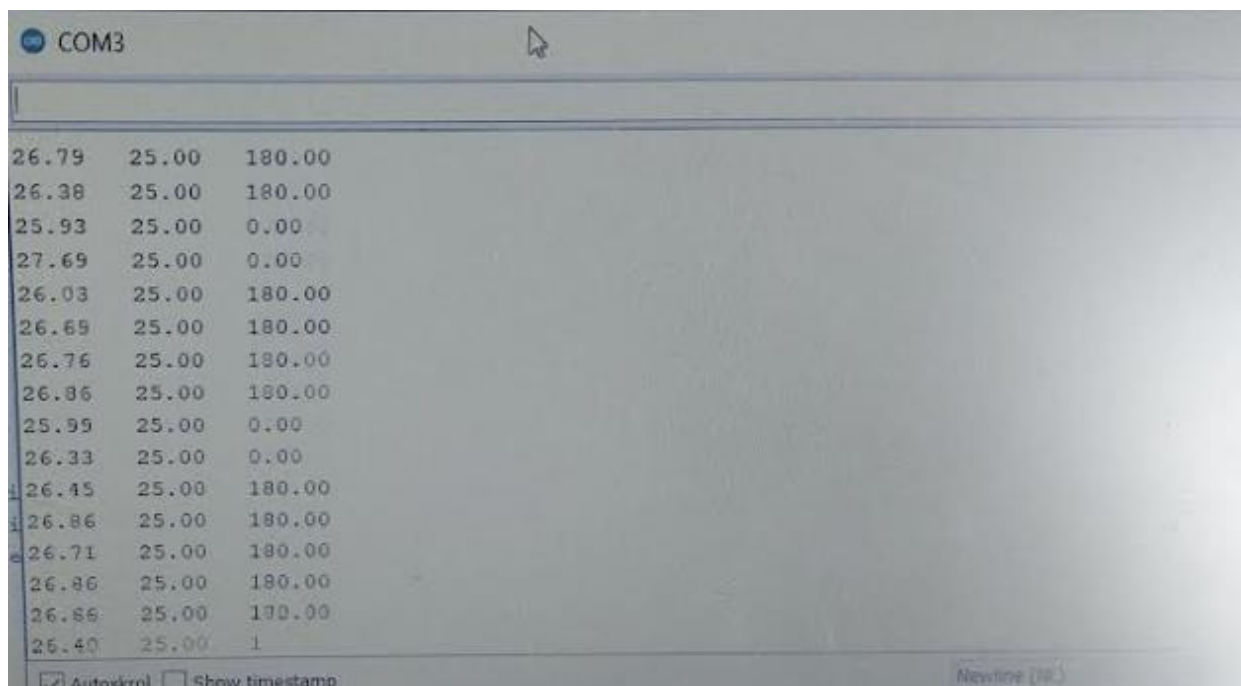
Slika 6.2. *Praktični dio projekta.*

Nakon pokretanja programa, potrebno je prilagoditi PID parametre, zbog male mase loptice i velike snage ventilatora nastale su poteškoće podešavanja. Maksimalni radni ciklus PWM ulaza ventilatora, prikazan na slici 6.4, postavljen je na 180 od maksimalnih 255 kako bi se osigurala optimalna stabilizacija. Funkcija `SetSampleTime` postavlja vrijeme uzrokovanja u milisekundama, što znači da PID regulator ažurira svakih 20 ms. Na slici 6.3 prikazana je levitacija loptice kada su podešeni svi parametri, te je potenciometrom određena željena visina loptice.





Slika 6.3. Levitacija loptice pomoću PID regulacije.



Slika 6.4. Prikaz izlaznih vrijednosti.

## 7. PROGRAMSKI KOD

Pomoću softvera Arduino IDE koji služi kao uređivač koda i kompajler, napisan je kod za levitaciju loptice te programiran na Arduino Uno pločicu. U prvom dijelu koda, pozivamo biblioteku `<PID_v1.h>`, u kojoj je implementirano PID upravljanje pomoću naredbe `#include`.

```
#include <PID_v1.h>
```

Zatim su definirane globalne varijable u kojima su spremljene oznake pinova na koje su spojeni elementi završnog rada. Tipom varijable `const int` definirane su konstante cijelog broja. Varijablom `double` postiže se preciznost sustava zbog mogućnosti pohrane decimalnih brojeva. Setpoint je ulazna vrijednost ili željena visina loptice, pomoću input-a dobiju se podaci trenutne izmjerene visine loptice, a output je izlaz PID regulacije koji upravlja brzinom ventilatora. Sljedeći korak je definiranje parametra PID regulatora s tri varijable naziva Kp, Ki i Kd

```
const int trigPin = 9;

const int echoPin = 10;

const int fanPin = 3;

const int potPin = A0;

double setpoint, input, output;

double Kp = 1.5, Ki = 5, Kd = 1.5;
```

Zatim se vrši instanciranje klase u kojoj je implementirana funkcionalnost PID regulatora pozivanjem njenog konstruktora `myPID` koji kao argumente prima tri pokazivača tj. adrese na kojima se nalaze varijable s izmjerenom visine loptice, izlaznom varijablom u koju se sprema brzina ventilatora, te željenom visinom loptice. Osim toga, konstruktoru je potrebno proslijediti i tri parametra PID regulatora: Kp, Ki i Kd. Zatim je definirana `void setup()` funkcija koja se pokreće samo jednom, kada se Arduino ploča uključi ili kada se resetira. Pomoću funkcije `pinMODE` definirano je koji su pinovi ulazni, a koji izlazni. Zatim je uspostavljena serijska komunikacija s računalom brzine 9600 bitova po sekundi. Nakon toga pozvane su tri metode prethodno

instancirane klase koje podešavaju neke od parametra PID kontrolera: ograničenja izlazne varijable, te brzine uzrokovanja.

```
// Inicijalizacija PID kontrolera
PID myPID(&input, &output, &setpoint, Kp, Ki, Kd, REVERSE);

void setup() {
    // Inicijalizacija pinova
    pinMode(trigPin, OUTPUT);
    pinMode(echoPin, INPUT);
    pinMode(fanPin, OUTPUT);

    Serial.begin(9600);

    // Podešavanje granica za izlaz (PWM ventilatora)
    myPID.SetOutputLimits(40, 180);
    myPID.SetSampleTime(20);

    // Aktiviranje PID kontrolera
    myPID.SetMode(AUTOMATIC);
}
```

Najvažniji dio svakog programa je funkcija `void loop()`, koja kao što joj ime sugerira, se iterativno ponavlja sve dok Arduino ima napajanja. Pomoću ove funkcije Arduino neprekidno obavlja zadatke koji su mu potrebni za rad ili očitavanje parametra. Unutar funkcije `loop()`, prvo u varijablu `input` spremljena je trenutna visina loptice izmjerena pomoću ultrazvučnog senzora. Zatim se očitava i vrijednost potencijometra. Analogni ulazi Arduino pločice spojeni su na 10-bitni AD pretvornik, stoga vrijednost potencijometra je u opsegu 0-1023, što nam ne odgovara, jer visina koju želimo regulirati je u opsegu 0-50. Stoga, očitane vrijednosti potencijometra potrebno je preskalirati na željeni raspon, što je učinjeno pomoću funkcija `map()`. Zatim se poziva metoda `Compute`, koja vrši proračun PID regulatora, te u varijablu `output` zapisuje izračunatu vrijednost brzine ventilatora. Naposljetku, funkcijom `analogWrite()` podešavamo brzinu ventilatora.

```
void loop() {
```

```

// Mjerenje trenutne visine loptice

input = measureDistance();

Serial.print(input);

setpoint = map(analogRead(potPin), 0, 1023, 0, 50);

// Izračunavanje PID izlaza

myPID.Compute();

Serial.print('\t');

Serial.print(setpoint);

Serial.print('\t');

Serial.println(output);

analogWrite(fanPin, output);

}

```

Mjerenje udaljenosti loptice implementirano je u funkciji `double measureDistance()`. Prvo generiramo nisku razinu impulsa na *trig* pinu ultrazvučnog senzora pomoću `digitalWrite()`, što osigurava neaktivnost senzora. Nakon pauze od 2  $\mu$ s, *trig* pin se postavlja na visoku razinu, čime inicira slanje ultrazvučnih impulsa u trajanju 10  $\mu$ s. Zatim mjerimo trajanje impulsa dok je *echo* pin na visokoj razini, a nakon toga računamo udaljenost loptice prema (3.1). Pošto je senzor montiran na vrh cijevi, maksimalna visina koju može izmjeriti je 50 cm, pa na kraju funkcije, ako je izmjerena vrijednost veća, postavljamo je na 50. Naredbom `return distance` vraća se izračunata udaljenost.

```

// Funkcija za mjerenje udaljenosti pomoću ultrazvučnog senzora

double measureDistance() {

    long duration;

    double distance;

```

```
// Slanje ultrazvučnog impulsa
digitalWrite(trigPin, LOW);
delayMicroseconds(2);
digitalWrite(trigPin, HIGH);
delayMicroseconds(10);
digitalWrite(trigPin, LOW);

// Mjerenje trajanja eha
duration = pulseIn(echoPin, HIGH);

distance = duration * 0.034 / 2;

// Ograničavanje maks udaljenosti na visinu tube
if (distance > 50) {
    distance = 50;
}

return distance;
}
```

## 8. ZAKLJUČAK

U ovom završnom radu postignuta je kontrola visine levitirane loptice pomoću PID regulacije koristeći Arduino platformu. Na stabilnost i preciznost sustava utječemo pomoću parametra  $K_p$ ,  $K_i$ ,  $K_d$ . PID regulator omogućuje učinkovitu kontrolu nad položajem loptice, ali je otkrio određene izazove i poteškoće. Jedna od glavnih poteškoća bila je precizno podešavanje PID parametara. Iako teoretski modeli i simulacije sugeriraju određene optimalne vrijednosti, u praksi je potrebno mnogo eksperimentiranja i prilagođavanja kako bi se postigla željena stabilnost sustava. Sustav je bio osjetljiv na promjene u vanjskim uvjetima, poput malih odstupanja parametra, što je dodatno otežalo kalibraciju. Također, tijekom rada primijećeno je da PID regulator, unatoč svojoj fleksibilnosti, zahtijevao dublje razumijevanje kako bi se izbjegle situacije poput prekomjernog osciliranja loptice ili sporog odziva sustava. Ove poteškoće pokazale su važnost analiziranja i prilagodbe sustava u stvarnim uvjetima, što je bilo ključno u postizanju zadovoljavajućih rezultata. Uz PID regulaciju trebalo se posvetiti jačini samog ventilatora, koja ima značajniji utjecaj na projekt. Nakon nekoliko isprobanih ventilatora dolazimo do zaključka kako za ovaj projekt potreban centrifugalni ventilator, vrlo malih snaga. Povezivanje, a zatim i prilagođavanje samih elementa za održivost levitacije loptice, bilo je kompliciranije nego li je u početku zamišljeno. Preciznim podešavanjem PID parametara, projekt je uspješno demonstrirao kontrolu nad položajem loptice unutar uspravno postavljene cijevi.

## 9. LITERATURA

- [1] „Arduino Uno“, s interneta, <https://www.javatpoint.com/arduino-uno>, 16. kolovoza 2024.
- [2] „Arduino Boards“, s interneta, <https://circuitdigest.com/article/different-types-of-arduino-boards>, 17. kolovoza 2024.
- [3] „Arduino sk.rs“, s interneta, <https://www.sk.rs/2016/10/sklp04.html>, 17. kolovoza 2024..
- [4] „Arduino Schematic“, s interneta, [https://electronoobs.com/eng\\_arduino\\_tut31\\_sch3.php](https://electronoobs.com/eng_arduino_tut31_sch3.php), 17. kolovoza 2024.
- [5] „Arduino Uno“, s interneta, [https://en.wikipedia.org/wiki/Arduino\\_Uno](https://en.wikipedia.org/wiki/Arduino_Uno), 17. kolovoza 2024.
- [6] „ATmega328p“, s interneta, <https://en.wikipedia.org/wiki/ATmega328>, 18. kolovoza 2024.
- [7] „PWM modulacija“, s interneta, <https://starmotech.com/pwm-pulse-width-modulation-i-for-petlja/>, 18. kolovoza 2024.
- [8] Hirzel, T.: „Basics of PWM (Pulse Width Modulation)“, s interneta, <https://docs.arduino.cc/learn/microcontrollers/analog-output>, 19. kolovoza 2024.
- [9] „PWM modulacija“, s interneta, [https://en.wikipedia.org/wiki/Pulse-width\\_modulation](https://en.wikipedia.org/wiki/Pulse-width_modulation), 18. rujna 2024.
- [10] „HR-SC04“, s interneta, <https://lastminuteengineers.com/arduino-sr04-ultrasonic-sensor-tutorial/>, 21. kolovoza 2024.
- [11] „HR-SC04“, s interneta, <https://www.theengineeringprojects.com/2018/10/introduction-to-hc-sr04-ultrasonic-sensor.html>, 21. kolovoza 2024.
- [12] „HR-SC04“, s interneta, <https://starmotech.com/kako-se-koristi-hc-sr04-ultrasonichni-arduino-senzor/>, 21. kolovoza 2024.
- [13] „HR-SC04“, s interneta, <https://www.theengineeringprojects.com/2017/08/ultrasonic-sensor-arduino-interfacing.html>, 21. kolovoza 2024.
- [14] „Nidec fan“, s interneta, <https://www.nidec.com/en/technology/capability/fan-motor/>, 25. kolovoza 2024.
- [15] „Battery Yuasa“, s interneta, <https://www.bblbatteries.co.uk/product/yb4l-bcp-yuasa-battery-12v-4ah-with-acid-pack/?cn-reloaded=1>, 25. kolovoza 2024.



- [16] „PID regulacija“, s interneta, <https://www.theengineeringconcepts.com/pid-controller/>, 31.kolovoza 2024.
- [17] „PID regulacija“, s interneta, <https://www.elprocus.com/the-working-of-a-pid-controller/>, 31. kolovoza 2024.
- [18] „PID regulacija“, s interneta, [https://en.wikipedia.org/wiki/Proportional%E2%80%93integral%E2%80%93derivative\\_controller](https://en.wikipedia.org/wiki/Proportional%E2%80%93integral%E2%80%93derivative_controller), 1.rujna 2024.
- [19] „PID regulacija“, s interneta, <https://www.automatika.rs/baza-znanja/teorija-upravljanja/osnove-pid-regulatora.html>, 1. rujna 2024.
- [20] „PID regulacija“, s interneta, <https://plcynergy.com/pid-controller/>, 2. rujna 2024.
- [21] „PD regulator“, s interneta, <https://electronicscoach.com/proportional-derivative-controller.html>, 2. rujna 2024.
- [22] „PD regulator“, s interneta, [https://www.researchgate.net/figure/System-Response-of-P-PI-PID-Controller-tuned-with-Process-Reaction-Curve-Method\\_fig1\\_327816155](https://www.researchgate.net/figure/System-Response-of-P-PI-PID-Controller-tuned-with-Process-Reaction-Curve-Method_fig1_327816155), 2. rujna 2024.
- [23] „Centrifugalni Nidec ventilator“, s interneta, <https://www.nidec.com/en/product/search/category/B101/M111/S103/NCJ-G10D12BS8AZ-5/>, 7. rujna 2024.
- [24] „Krilca ventilatora“, s interneta, [https://www.researchgate.net/figure/Centrifugal-Fans-Blades-16-8\\_fig7\\_332606723](https://www.researchgate.net/figure/Centrifugal-Fans-Blades-16-8_fig7_332606723), 7. rujna 2024.
- [25] „Centrifugalni ventilator“, s interneta, [https://en.wikipedia.org/wiki/Centrifugal\\_fan](https://en.wikipedia.org/wiki/Centrifugal_fan), 7. rujna 2024.

## 10. POPIS OZNAKA I KARTICA

PID	Proportional-Integral-Derivative
PWM	Pulse width modulation
I/O	Input/Output
SPI	Serial Peripheral Interface
UART	Universal Asynchronous Receiver/Transmitter
I2C	Inter-integrated circuit
USB	Universal Serial Bus
CPU	Central Processing Unit
V	Volt
A	Amper
EEPROM	Electrically erasable programmable read-only memory
SRAM	Static random-access memory
PLC	Programmable Logic Controller
HZ	Hertz
ADC	Analog-to-digital conversion
RISC	Reduced Instruction Set Computing

## 11. SAŽETAK I KLJUČNE RIJEČI

### 11.1. Sažetak i ključne riječi

Ovaj završni rad prikazuje sustav za regulaciju visine levitirane loptice na Arduino platformi. Cilj rada je osmisлити sustav koji koristi ultrazvučni senzor, smješten na vrhu cijevi za mjerenje visine loptice, dok se potenciometrom upravlja željenom visinom loptice. Poseban naglasak stavljen je na podešavanje PID regulatora i prilagodbu snage ventilatora, čime se povećava stabilnost sustava. Rad detaljno opisuje sve korištene elemente, uključujući Arduino Uno, ultrazvučni senzor HC-SR04, Nidec ventilator i linearni potenciometar.

Ključne riječi: Arduino Uno, PID regulacija, upravljanje, stabilnost

### 11.2. Summary and keywords

This bachelor thesis will introduce a system for regulating the height of a levitating ball with the help of Arduino. The primary aim of the thesis was to design and develop a system which employs a tube mounted ultrasonic sensor at the top and a potentiometer to set and measure the height of the ball. Special emphasis was put on how to tune the PID controller. In this regard experimentation and measurements were undertaken for the purposes of making adjustments prior to system stabilization. The thesis describes in detail all the components used in the design and construction of the system including Arduino Uno, HC-SR04 ultrasonic sensor, Nidec fan and linear potentiometer.

Keywords: Arduino Uno, PID regulation, control, stability