Tehnički fakultet Rijeka

Izborni projekt – Operacijski sustavi

Drone Time Attack



Deni Klen, Mateo Srića

Mentor: izv. prof. dr. sc. Kristijan Lenac

Rijeka, veljača 2021

Sadržaj

[Uvod 3](#_Toc63786262)

[Upute za korištenje 3](#_Toc63786263)

[Fizička konstrukcija 4](#_Toc63786264)

[Arduino 6](#_Toc63786265)

[Python 7](#_Toc63786266)

[Serial Monitor 7](#_Toc63786267)

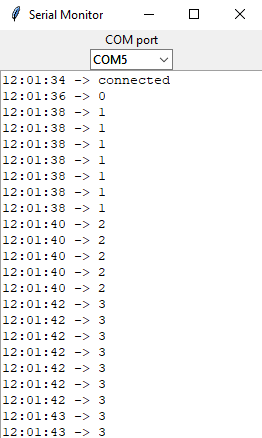
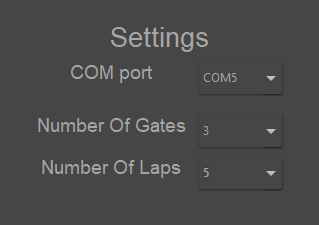
[Drome Time Attack 7](#_Toc63786268)

# Uvod

Tema ovog projekta bila je izrada poligona za trkaće bespilotne letjelice i izrada programske podrške koja prikazuje odrađena vremena. Projekt je podijeljen u tri cjeline - fizičku konstrukciju poligona, programsku podršku napisanu u programskom jeziku Arduino koja upravlja Arduino Nano mikročipovima te programsku podršku pisanu u programskom jeziku Python. Za rad s Arduino čipovima je bio zadužen Deni Klen u čemu je uključena programska podrška i spajanje senzora i žica na mikročip, dok je za svu programsku podršku u Pythonu kao i fizičku konstrukciju vratiju izuzev spajanja mikročipova bio zadužen Mateo Srića. Cilj projekta je kreiranje potpuno nezavisne bežične mreže vrata koje međusobno komuniciraju s jednim glavnim mikrokontrolerom stvarajući trkaći poligon kao i implementiranje praćenja vremena prolaska te mogućnost ispisivanja najboljih rezultata.

# Upute za korištenje

Prije otvaranja programa, uključite sva vrata koja će se koristiti spajajući ih na prijenosne baterije kao i prijamnik na USB priključak na računalu. Prije pokretanja glavnog programa, preporučuje se pokretanje Serial Monitor programa da se osigura da sva vrata kao i prijamnik ispravno rade.

*Slika 1. Serijski output Slika 2. Postavke*

Zatim, ako se nisu pojavila kriva očitanja, se otvara Drone Time Attack. Nakon pokretanja programa odaberite priključak na koji je spojen prijamnik i željeni broj vratiju te broj krugova koji želite odvoziti. Kada su postavke podešene, započeti praćenje vremena pritiskom na Start. Kada se prijamnik spoji na računalo, generira se tablica i daje se zvučni signal da je spojen. Potrebno je postaviti bespilotnu letjelicu na stanicu za uzlijetanje. Praćenje vremena se pokreće nakon što bespilotna letjelica napusti stanicu za uzlijetanje. Prilikom prolaska kroz vrata računalo daje zvučni signal kako bi osoba koja upravlja bespilotnom letjelicom bila sigurna da ga je prepoznala. Vrijeme svakog prolaska kroz vrata, kao i cijelog kruga se prikazuje nakon odvoženog kruga, dok se završno vrijeme ispisuje automatski na kraju odvožene staze.

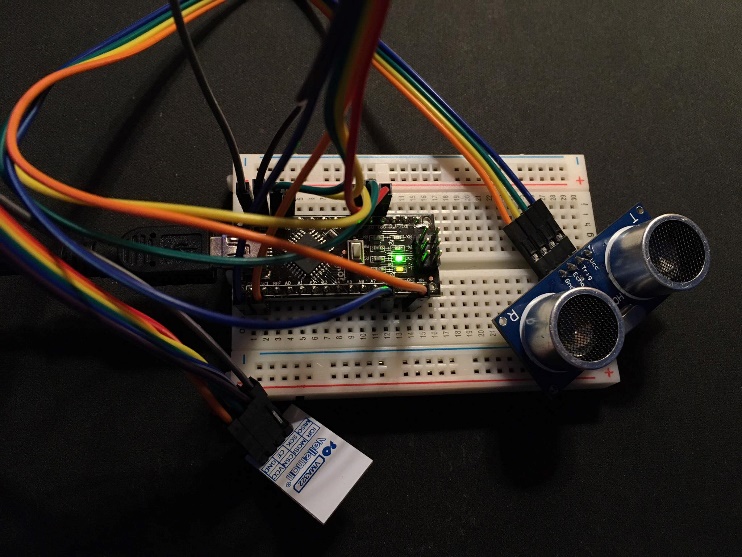


*Slika 3. Tablica sa vremenima*

Ukoliko je potrebno resetirati vožnju to se radi pomoću tipke Reset nakon čega se bespilotna letjelica mora opet vratiti na stanicu za uzlijetanje.

# Fizička konstrukcija

Konstrukcija ovog projekta sastoji se od stanice za uzlijetanje te vratiju. Stanica za uzlijetanje sastoji se od jedne ATmega328 NANO razvojne pločice, uparenog sa nRF24L01 bežičnog komunikacijskog uređaja i HC-SR04 ultrazvučnog senzora.



*Slika 4. Prijamnik*

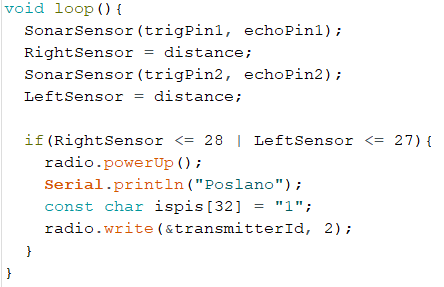
Vrata se sastoje od visinom prilagodljive PPH konstrukcije u kojoj su smještena dva HC-SR04 ultrazvučna senzora, nRF24L01 bežični komunikacijski uređaj te ATmega328 NANO razvojna pločica. Cijela konstrukcija napaja se pomoću baterije koja se nalazi na dnu konstrukcije.

*Slika 5. Okvir vratiju Slika 6. Postolje i napajanje*

# Arduino

Arduino sučelje je korišteno kao spona između fizičke konstrukcije i programske podrške napisane u programskom jeziku Python. Kôd potreban za izvršavanje cijele logičke operacije je bio u dva dijela, „*Odašiljač*“ koji predstavlja svaka vrata posebno i „*Prijemnik*“. Odašiljačev dio kôda sastoji se od funkcije koja pomoću ultrazvučnih senzora provjerava ako je udaljenost koju jedan od dva ugrađena senzora šalju postala manja od 30tak centimetara što je udaljenost od vrha do dna okvira kroz koji bespilotne letjelice prolaze. Ako je došlo do zadovoljavanja uvjeta, aktivira se bežični radio kanal koji šalje unikatni broj vratiju kroz koje je bespilotna letjelica prošla.

**

*Slika 7. Odašiljačev kôd*

Primateljev dio koda sastoji se od start funkcije i primanja radio poruka. Start funkcija provjerava ako je bespilotna letjelica postavljena na stanicu za uzlijetanje, ako nije čeka se da se postavi. Nakon toga slijedi provjera koja gleda ako se letjelica uzdigla s platforme, te ako je ta tvrdnja istinita na serijski izlaz ispisuje broj 0 što unutar Python koda označava početak mjerenja vremena. Unutar dijela za primanje poruka pokreće se kanal preko kojeg se šalju podaci sa svakih vratiju na glavnu stanicu. Ako se dobije povratnu informaciju s nekih od vrata prijemnik na serijski izlaz ispisuje unikatni ID vratiju kroz koje je bespilotna letjelica prošla.

|  |  |
| --- | --- |
| *Slika 8. Start funkcija prijemnika* | *Slika 9. Primanje radio poruka* |

# Python

## Serial Monitor

Cilj Seral Monitor-a je izravno čitanje sa serial port-a kao što je na primjer u Arduinovom IDE-u. Korisno ga je koristiti u slučajevima kada se sumnja na nekakve greške kod senzora, slanja podataka itd. Knjižnice koje su korištene su Tkinter, Serial, Threading i Time. Prozor se sastoji od jedne oznake(Label), jednog kombiniranog okvira pomoću kojeg se odabire komunikacijski priključak i jednog Text objekta u kojemu se ispisuje izlaz(Slika 1.). Čitanje s komunikacijskog priključka počinje odabirom iz liste kombiniranog okvira nakon čeka se pokreće main funkcija koja ispisuje podatke i vrijeme kada su primljeni na kraj Text objekta.

|  |
| --- |
| *Slika 10. Serijsko spajanje* |
| *Slika 11. Glavni program* |

## Drome Time Attack

Za grafičko sučelje su se koristile knjižnice Tkinter i ttkThemes, za čitanje sa serijskog ulaza knjižnica Serial. Za reprodukciju zvuka se koristio Playsound, za dretve Threading i Sys. Početak programa se događa na kraju kôd pozivom root.mainloop() koji otvara prozor. Nakon toga program čeka ulazne podatke od korisnika – pritiske na tipke i odabir raznih izbornika.

|  |
| --- |
|  |
| *Slika 12. Pokretanje programa* |

|  |  |
| --- | --- |
| *Slika 13. Namještanje postavki* | |
| *Slika 14. Funkcije tipke "Start"* | *Slika 16. Funkcije tipke "Reset"* |
| *Slika 15. Funkcije tipke "Show Highscores"* | *Slika 17. Funkcije tipke "Reset Highscores"* |

Serijsko spajanje započinje pritiskom na tipku "Start" na isti način kao i kod Serial Monitora. Kada se Arduino spoji, šalje poruku "connected" nakon čega se pušta zvuk i generira tablica. Kada bespilotna letjelica uzleti, pokreće se štoperica čija dretva je već u izvršavanju mijenjajući varijablu state i isStarted u True. Svaki put kada prijemnik dobije podatak, uz uvjet da je broj dobiven za jedan veći od prethodnog(1, 2, 3 - može; 1, 2, 2 – drugu dvojku prepoznaje, ali ignorira), vrijeme dobivanja stavlja u 2d polje times. Kada prođe kroz zadnja vrata na poligonu, generira ponovno cijelu tablicu pokazujući vremena za prethodni krug. Kada prođe kroz zadnja vrata zadnjeg kruga, provjerava se ako je neko vrijeme bilo najbolje(jedan krug ili cijela utrka) nakon čeka zatvara čitanje s komunikacijskog priključka i resetiranje podataka.

|  |  |
| --- | --- |
| *Slika 18. Popis funkcija za čitanje sa komunikacijskog priključka i main* | |
| *Slika 19. Popis funkcija za stvaranje tablice i najboljih vremena* | *Slika 20. Popis funkcija za štopericu* |

|  |  |
| --- | --- |
| *Slika 21. Prozor za upis imena nakon utrke* | *Slika 22. Prozor prikaza najboljih vremena* |
| *Slika 23. Provjera sigurnosti pri resetiranju najboljih vremena* | *Slika 24. Provjera sigurnosti pri izlaženju iz programa* |