

ISTARSKO VELEUČILIŠTE  
UNIVERSITÀ ISTRIANA DI SCIENZE APPLICATE  
Stručni studij politehnike

Izrada snimača podataka, obrada i vizualizacija  
prikupljenih podataka bazirano na principima  
slobodnog i otvorednog koda

Završni rad

Kristijan Cetina  
JMBAG: 2424011721  
kcetina@iv.hr

Pula, 21. kolovoza 2019.

## Sažetak

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetur id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris.

**Ključne riječi** *riječ, dva, tri . . .*

Kolegij: Elektronika

Mentorica: Sanja Grbac Babić, mag. računarstva, v.predavač

## Abstract

Abstract in English

Nulla malesuada porttitor diam. Donec felis erat, congue non, volutpat at, tincidunt tristique, libero. Vivamus viverra fermentum felis. Donec nonummy pellentesque ante. Phasellus adipiscing semper elit. Proin fermentum massa ac quam. Sed diam turpis, molestie vitae, placerat a, molestie nec, leo. Maecenas lacinia. Nam ipsum ligula, eleifend at, accumsan nec, suscipit a, ipsum. Morbi blandit ligula feugiat magna. Nunc eleifend consequat lorem. Sed lacinia nulla vitae enim. Pellentesque tincidunt purus vel magna. Integer non enim. Praesent euismod nunc eu purus. Donec bibendum quam in tellus. Nullam cursus pulvinar lectus. Donec et mi. Nam vulputate metus eu enim. Vestibulum pellentesque felis eu massa.

Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetur adipiscing elit. In hac habitasse platea dictumst. Integer tempus convallis augue. Etiam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula.

**Keywords:** *word, two, three . . .*

*Posveta*

## Zahvala

Zahvala svima koji zaslužuju

”A good scientist is a person with original ideas. A good engineer is a person who makes a design that works with as few original ideas as possible. There are no prima donnas in engineering” - Freeman Dyson

## **Izjava o samostalnosti izrade završnog rada**

Izjavljujem da sam završni rad na temu "GPS datalogger" samostalno izradio uz pomoć mentorice Sanje Grbac Babić mag. računarstva, koristeći navedenu stručnu literaturu i znanje stečeno tijekom studiranja. Završni rad je pisan u duhu hrvatskog jezika.

Student: Kristijan Cetina

# Sadržaj

<b>0</b>	<b>Opis zadatka i ograničenja</b>	<b>2</b>
0.1	Uvod . . . . .	2
0.2	Slobodan i otvoreni kod . . . . .	2
0.3	Arduino platforma . . . . .	4
0.4	Jupyter Notebook . . . . .	4
0.5	Git . . . . .	5
<b>1</b>	<b>Prikupljanje podataka - hardware</b>	<b>7</b>
1.1	GPS logging shield . . . . .	7
1.2	Prikupljanje podataka o temperaturi . . . . .	10
1.2.1	Softwareski filter . . . . .	10
1.2.2	Hardwareski filter . . . . .	10
<b>2</b>	<b>Obrada podataka - software</b>	<b>13</b>
2.1	Prikupljanja podataka: software na Arduino . . . . .	13
<b>3</b>	<b>Zaključak</b>	<b>14</b>
	<b>Literatura</b>	<b>15</b>

# Popis tablica

# Popis slika

1.1	Shema spoja TMP36 senzora . . . . .	8
1.2	Shema Adafruit GPS Logger Shield . . . . .	9
1.3	Vrijednosti senzora bez filtriranja . . . . .	11
1.4	Vrijednosti senzora sa softwareskim filtriranja . . . . .	11



## Poglavlje 0

# Opis zadatka i ograničenja

### 0.1 Uvod

Cilj ovog rada je bio izraditi jednostavan snimač podataka (*datalogger*) koji će spremati GPS podatke zajedno s podacima prikupljenim sa instaliranih senzora za kasniju analizu. Izrađeni uređaj je namjenjen kao snimač podataka u kompleksnijem sklopu koji se može koristiti kada god postoji potreba za loggiranje podataka. Uređaj je namjenjen da zadovolji široki spektar potreba koje se mogu javiti bilo u industriji npr. prilikom praćenja pošiljki ili pak prilikom skupljanja podataka u istraživačke svrhe kako bi se razumio širi problem.

Sklop je baziran na Arduino platformi koja omogućava lak razvoj prototipova uz široku dostupnost gotovih dodatnih modula (*shields*) koji se jednostavno spajaju na bazno mikroračunalo.

Prikupljeni podaci se spremaju na SD karticu na uređaju u datoteku za kasniju obradu i analizu. Prikupljeni podaci se uz pomoć programskog jezika Python i dodatnih modula za statističku i numeričku analizu kao što su Pandas i Matplotlib obrađuju kroz sučelje interaktivne bilježnice Jupyter Notebook. Pristup obrade putem interaktivne bilježnice uz korištenje raznih tipova ćelija kao što su *Code Cells*, *Markdown Cells* i *Raw Cells* omogućava lakšu vizualizaciju i pregled samog rada koji je pogodan za kasnije dijeljenje svim zainteresiranim stranama koji žele pregledati ili nastaviti rad na analizi.

Kompletan Git repozitorij ovog rada javno je dostupan na <https://github.com/KristijanCetina/BachelorThesis>

### 0.2 Slobodan i otvoreni kod

Izraz otvoreni kod *open source* odnosi se na nešto što ljudi mogu slobodno mijenjati i dijeliti jer je dizajn javno dostupan[1]. Izraz je nastao u kontekstu razvoja računalnog softwarea dok se danas odnosi na pristup radu bio on software, hardware ili bilo kakav drugi tip projekta. Spomenimo kako

postoje razne licence pod kojima se objavljuju open source radovi, a u praksi se razlikuju u načinu na koji izjeleni i izvorni rad mora biti distribuiran svim ostalim zainteresiranim stranama.

Razlozi i prednosti primjene open source pristupa projektima su višestruke, a neke od njih su:

- Kontrola proizvoda
- Učenje i trening
- Sigurnost
- Stabilnost

**Kontrola proizvoda:** Kada je izvorni kod i ostala dokumentacija nekog proizvoda otvorena onda se može pogledati kako točno radi taj proizvod i na koji način je izgrađen. Na taj način svaki korisnik može imati kontrolu nad onime što koristi jer ne postoji koncept crne kutije (*BlackBox concept*) te omogućava korisniku da uz dostupni kod i sheme popravi ili unaprijedi proizvod. Zapičajmo se koliko puta smo se osobno susreli sa situacijom kada zbog kvara nekog uređaja smo bili primorani posjetiti i platiti ovlaštenog servisera koji ima specijalni alat za diagnostiku i popravke?

**Učenje i trening:** Uvidom u otvorenu dokumentaciju možemo vidjeti kako je neki stručnjak riješio određeni problem te to rješenje u potpunosti ili modificirano može se primjeniti na vlastiti problem. Otvorena dokumentacija omogućava proučavanje rješenja određenih problema i na taj način se skraćuje vrijeme i pojeftinjuje razvoj novih proizvoda koji imaju slične zahtjeve. Znanstvenici objavljuju svoja otkrića kako bi ih drugi iza njih mogli koristiti. Inženjeri u svakodnevnom radu ne izvode i dokazuju npr Ohmov ili Newtonove zakone već ih samo primjenjuju.

**Sigurnost:** Proučavanje objavljene dokumentacije nekog projekta drugi stručnjaci iz područja mogu uvidjeti neke propuste koje autori zbog kompleksnosti proizvoda ili drugih razloga nisu primjetili te dojaviti autorima grešku kao bi se ista mogla ispraviti. Neke greške se mogu pojaviti samo u iznimno malom broju slučajeva ili kada se poslože veliki broj faktora te nije realno očekivati da se prilikom testiranja proizvoda simulira svaki mogući scenarij korištenja. Zainteresirane strane mogu dodatno testirati proizvod u specifičnim uvjetima i na taj način otkriti inače skrivenu grešku u proizvodu te nakon otklanjanja greške sam proizvod postaje sigurniji.

**Stabilnost:** Mnogi proizvodi se koriste za vrlo bitne aspekte rada nekog većeg sustava te njihova zamjena iziskuje velike promjene i investicije, a ponekada nije niti moguća. Korištenjem proizvoda otvorenog koda i dokumentacije omogućava se korištenje nastavak podrške i korištenja tog proizvoda i nakon eventualnog nestanka kompanije koja je napravila proizvoda te isti više nije dobaljiv od proizvođača. Ako se koriste open source proizvodi moguće je samostalno rekreirati proizvod ukoliko se ukaže takva potreba.

## 0.3 Arduino platforma

Arduino je elektronička platforma otvorenog koda<sup>1</sup> baziran na hardwareu i softwareu koje je lako za koristiti. Arduino platforma obuhvaća mikrokontrolerske pločice bazirane na AVR arhitekturi s integriranim digitalnim, alalognim ulazima i izlazima kao i PWM<sup>2</sup> izlazima. Platforma omogućava lagano spajanje dodatnih vanjskih uređaja kao što su razni senzori, releji, servo i motori putem dodatnog upravljačkog modula i ostale elektroničke i elektromehaničke komponente. Sheme svih mikrokontrolera objavljene pod Creative Commons<sup>3</sup> licencom i javno dostupne svim zainteresiranim stranimama.

Adruino pločice su relativno povoljne u usporedbi s ostalim platformama i kao takve omogućavaju pristupačnije učenje svim zainteresiranima. Potrebno je ponekad malo spretnosti s lemilicom dok se često mogu slagati moduli na prototipnoj pločici bez lemljenja sa izradom spojeva putem spojnih žica.

Jednostavno korisničko sučelje (IDE<sup>4</sup>) za izradu korisničkih programa (*sketch*) je jednostavno za korištenje za početnike dok istovremeno omogućava izradu vrlo kompleksnih programa iskusnim korisnicima. IDE je kompatibilan s većinom danas raspostranjenih operacijskih sustava (GNU/Linux, MacOS i Windows). Programski jezik za izradu programa je baziran na C/C++ te omogućava daljnje proširivanje kroz C++ biblioteke ili koristiti AVR-C programski jezik.

## 0.4 Jupyter Notebook

U ovom radu za obradu i prikazivanje podataka korišten je programski jezik Python<sup>5</sup> uz dodatke Pandas<sup>6</sup> i Matplotlib<sup>7</sup>. Pandas omogućava lakšu

---

<sup>1</sup><https://www.arduino.cc/en/Guide/Introduction>

<sup>2</sup>Pulse Width Modulation - Pulsno širinska modulacija

<sup>3</sup><https://creativecommons.org/>

<sup>4</sup>Integrated development environment

<sup>5</sup><https://www.python.org/>

<sup>6</sup><https://pandas.pydata.org/>

<sup>7</sup><https://matplotlib.org/>

manipulaciju podacima dok Matplotlib omogućava izradu kvaitetnih grafova s velikom mogućnošću prilagodbe raznim željama i potrebama. Sve zajedno je implementirano kroz sustav *interaktivne bilježnice* Jupyter<sup>8</sup> koja omogućava brzu i jednostavnu obradu podataka kao i njeno dijeljenje sa svim zainteresiranim stranama. Jupyter notebook je web aplikacija otvorenog koda koja se može izvršavati na lokalnom računalu ili koristeći resurse računalstva u oblaku. Podržava razne programske jezike poput Julia, Ruby, R, C++ i mnoge druge te u ovom radu korišten Python. Jupyter notebook omogućava kreiranje i djeljenje dokumenata koji sadržavaju izvršivi programski kod, jednadžbe, grafove i vizualizacije te popratni tekst u jednoj cijelini koju trenutno drugim načinima nije moguće ili je vrlo nepregledno za postići. Područja primjene su najčešće obrada i transformacija podataka, numeričke analize, statistički modeli, vizualizacija podataka, strojno učenje i još mnogo toga.

## 0.5 Git

Git<sup>9</sup> je distribuirani sustav za verzioniranje koda i ostalog rada kojeg želimo djeliti sa suradnicima. Git sa svojim jednostavnim i brzim granama omogućava lakši razvoj proizvoda kao i ispitivanje mogućnosti i funkcija. Kada se želi ispitati neka funkcionalnost bez da se ugrozi ono što do sada radi kako treba nema potrebe kopirati cijeli projekt u novi folder i onda u njemu testirati već se jednostavno kreira nova grana u kojoj se radi razvoj i kada smo sigurni da sve radi kako želimo onda se ta grana ujedini s glavnom granom projekta koja prihvati dodatne funkcionalnosti koje su razvijene za proizvod. Kako je Git lagan za resurse onda se može kreirati vrlo veliki broj grana za razne potrebe bez značajnog utjecaja na performanse razvojnog računala ili potrošnje spremišnog prostora.

S obzirom na distribuiranu narav Gita svaki suradnik koji radi na projektu ima svoju kopiju na kojoj radi te nije vezan za neki server i stalnu komunikaciju s ostatkom tima već je ista potrebna samo kada se povlače i šalju učinjene promjene.

Git je nastao 2005 godine za potrebe razvoja Linux jezgre i od tada je poprimio mnoge simpatije unutar inženjerske zajednice koja ga koristi kako bi zajednički razvija projekte.

Kako bi se olakšalo djeljenje i suradnja na projektima 2008. godine je pokrenut GitHub - centralno mjesto za usluge poslužitelja<sup>10</sup> (*hosting*) putem kojeg je moguće pratiti životni ciklus i povijet projekta. Svatko može pronaći projekt koji ga zanima te ukoliko ima dovoljno vremena i znanja može i pridonjeti njegovom razvoju. Brojne kompanije koriste GitHub kao

---

<sup>8</sup><https://jupyter.org/>

<sup>9</sup><https://git-scm.com/>

<sup>10</sup><https://github.com/features>

bi podjelile svoje projekte. Podatak od travnja 2019. godine kaže kako više od 2.1 milijuna kompanija i organizacija koristi GitHub. Jedna od njih je i Adafruit - kompanija koja proizvodi elektroničke dodatke za Arduino i druge platforme i fokusirana je na edukacija mladih (i onih koji se tako osjećaju), a njihov GitHub sadrži više od 1100 repozitorija<sup>11</sup>. Upravo je njihov GPS Logger Shield korišten u ovom projektu, a dostupnos dokumentacije i dostupna podrška je jedan od glavnih razloga zašto je odlučeno koristiti upravo taj proizvod.

---

<sup>11</sup><https://github.com/adafruit/>

## Poglavlje 1

# Prikupljanje podataka - hardware

### 1.1 GPS logging shield

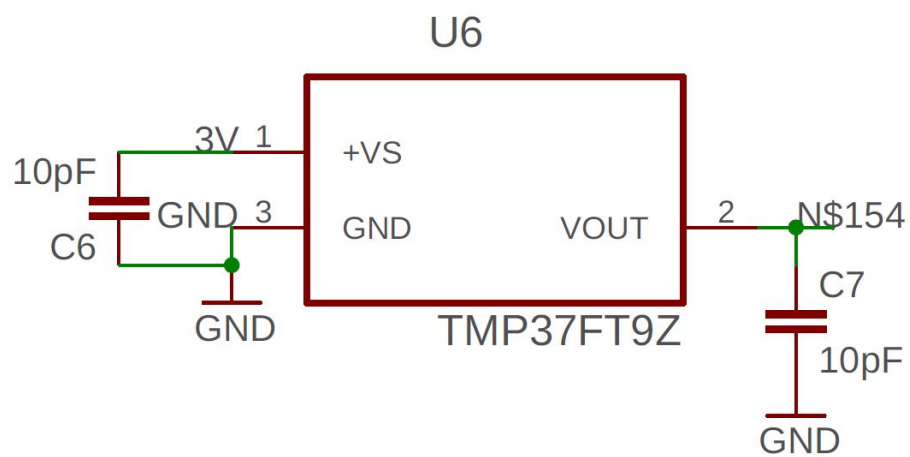
Na shemi 1.2 nalazi se shema gotovog elektroničkog sklopa kako dolazi iz tvornice<sup>12</sup>. Na samoj tiskanoj pločici postoji tkz. prototipno područje za dodavanje vanjskih elemenata čiji je raster 2.54 mm koji odgovara standardu *true-hole* elemenata. Na to područje je dodan temperaturni senzor TMP36<sup>3</sup> zajedno s dodatnim pasivnim elementima koji služe kao filter smetnji koje se javljaju u radu zbog okoline. Shema spoja je prikazana na slici 1.1.

---

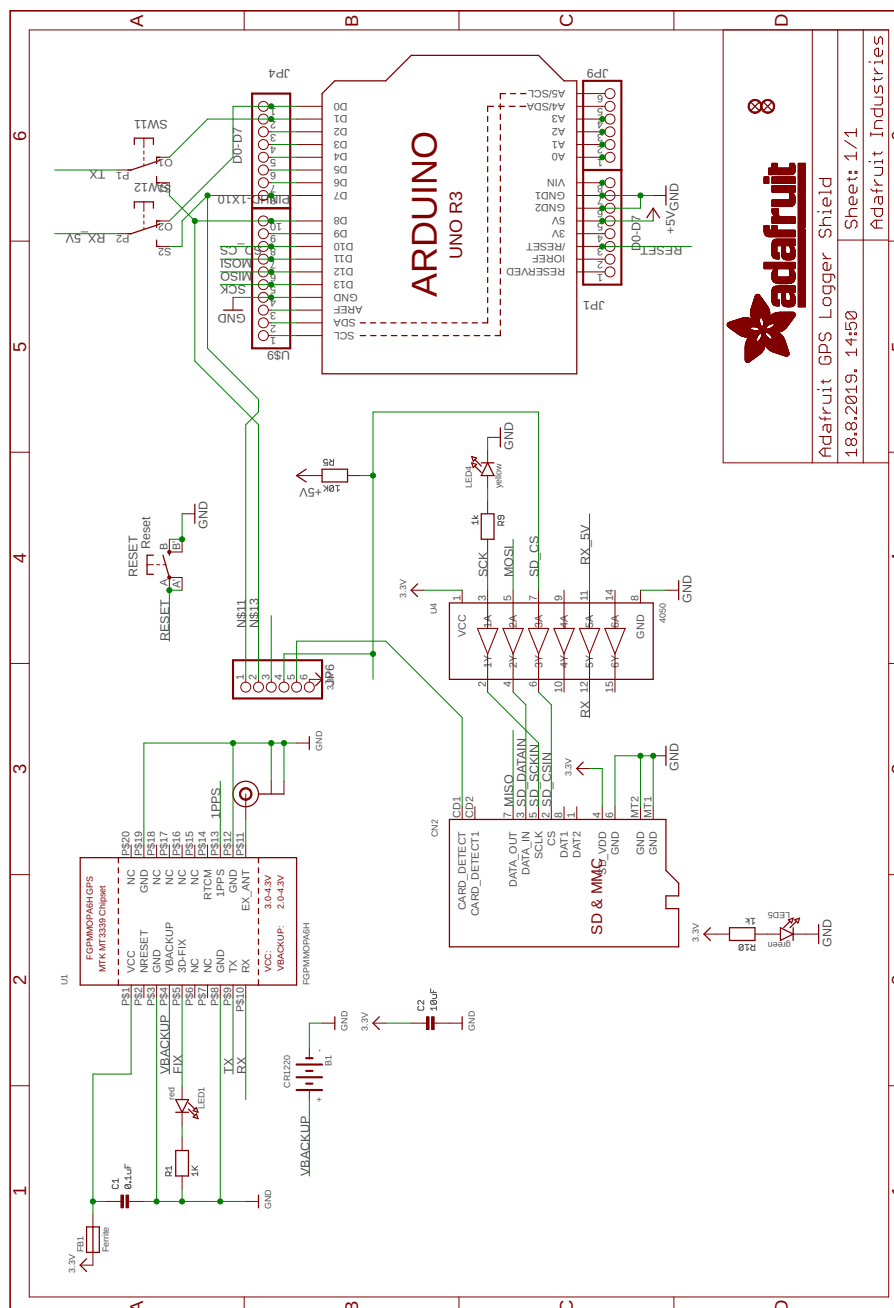
<sup>1</sup>Kompletna dokumentacija dostupna je na <https://learn.adafruit.com/adafruit-data-logger-shield?view=all>

<sup>2</sup>GitHub repozitorij korištene verzije dostupan na <https://github.com/adafruit/Adafruit-GPS-Logger-Shield-PCB>

<sup>3</sup>Datasheet dostupan na [https://github.com/KristijanCetina/BachelorThesis/blob/master/resources/TMP35\\_36\\_37.pdf](https://github.com/KristijanCetina/BachelorThesis/blob/master/resources/TMP35_36_37.pdf)



Slika 1.1: Shema spoja TMP36 senzora



Slika 1.2: Shema Adafruit GPS Logger Shield



## 1.2 Prikupljanje podataka o temperaturi

Kako svaki elektronički sklop ima definirani raspon radne temperature bitno je znati u kojim uvjetima isti se nalazi. Ukoliko je temperatura previsoka može se uključiti aktivno hlađenje ili ako se unaprijed zna da će se sklop nalaziti pod povišenom radnom temperaturom onda se može konstruirati adekvatan sustav hlađenja. Isto vrijedi za prenisku temperaturu. Prema ranije spomenutoj shemi 1.1 dodatn je temperaturni senzor koji mjeri radnu temperaturu okoline uređaja. Pri testiranju ova vrsta senzora se pokazala veoma pouzdana, uz minimalno samozagrijavanje koje bi utjecalo na točnost mjerene veličine ali je isto tako pokazala vrlo brze promjene izlazne vrijednosti koja može biti do vanjskih smetnji. Kako bi se otklonio taj problem primjenjena su dva rješenja. Prvi je hardwareski filter - kondenzatori koji je prikazan na shemi 1.1, a drugi je softwareski filter. Tvornički podaci o izlaznom naponu šuma mogu se pronaći u datasheetu uređaja, slika 20. Na slici 1.3 prikazane su izlazne vrijednosti senzora bez ikakvog filtriranja i obrade. frekvencija uzorkovanja je 10Hz (10 očitavanja u sekundi) Svakako nije realno za očekivati da se temperatura mjenja sukladno očitanim vrijednostima.

### 1.2.1 Softwareski filter

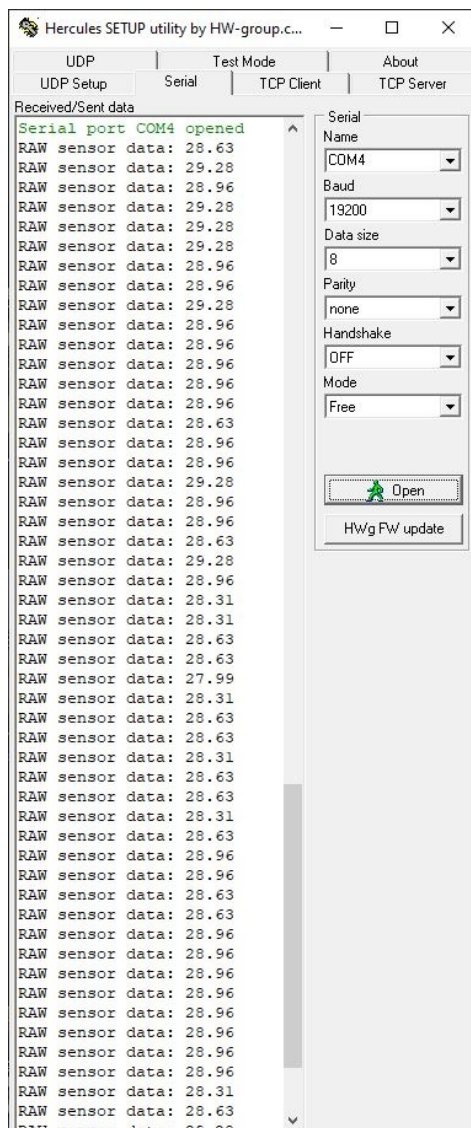
Softwareski filter radi na principu da očitava 10 vrijedosti sa senzora te ih sprema u polje. Potom ih sortira po veličini i uzima medijan<sup>4</sup> vrijednost kao točnu temperaturu. Na taj način se postiže da se eliminiraju sve vrlo visoke i vrlo niske vrijednosti koje se mogu pojaviti zbog šuma u signalu. Vrijednosti se čitaju svakih 100 ms te uz računanje na bazi 10 vrijednosti daje frekvenciju od 1 očitavanja u sekundi koja odgovara i frekvenciji uzorkovanja podataka sa GPS senzora. Prilikom testiranja utvrđeno je da veći broj uzoraka ne doprinosi kvaliteti izmjerenih vrijednosti, a pro manjem broju uzoraka može se potkrasti poneka nerealna vrijednost. Kako je očekivano vrijeme promjene temperature značajno duže od 1 sekunde onda su prihvaćene navedene vrijednosti i metoda filtriranja. Na slici 1.4 prikazane su izlazne vrijednosti senzora nakon primjene opisanog softwareskog filtera. Primjeti se značajno manje skokova od nečega što se može smatrati stvarna vrijednost.

### 1.2.2 Hardwareski filter

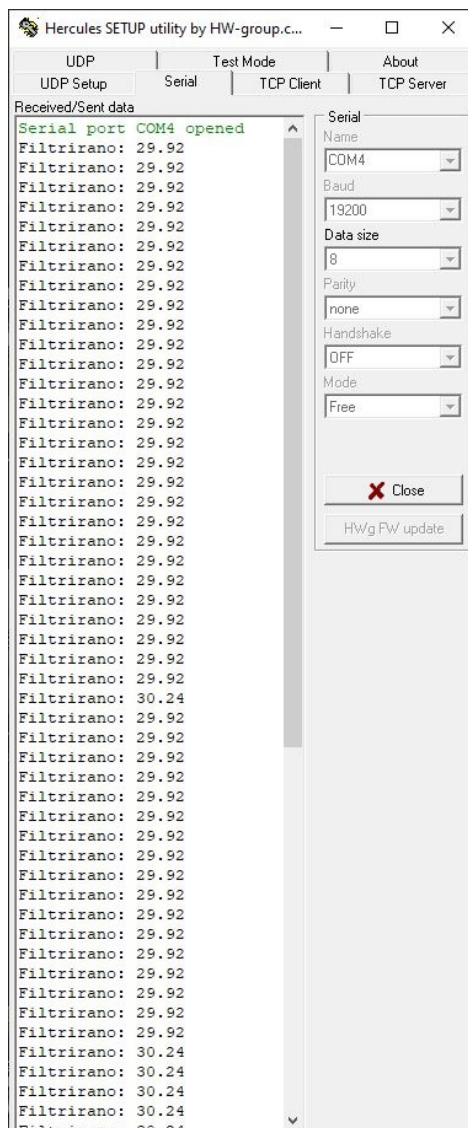
Filter je jednostavna mreža keramičkih kondenzatora vrijednosti  $10pF$  koji su spojeni što bliže senzoru između izvoda za napajanje i izlaza senzora prema točki nultog potencijala (*GND*, *masa*) kako bi apsorbirali eventualne smetnje. Iako je softwareski filter u nekim situacijama dovoljno dobar ovo

---

<sup>4</sup>Medijan (mediana, centralna vrijednost) je pojam iz statistike koji određuje sredinu distribucije. Pola vrijednosti skupa (distribucije) nalazi se iznad mediane, a pola ispod



Slika 1.3: Vrijednosti senzora bez filtriranja



Slika 1.4: Vrijednosti senzora sa softwareskim filtriranjem

jednostavno i jeftino rješenje daje dodatan sloj filtriranja koji za posljedicu ima vrlo glatko očitavanje temperature bez skokova u vrijednostima.

## Poglavlje 2

# Obrada podataka - software

U ovom poglavlju biti će opisana softwareska komponenta za obradu i vizualizaciju prikupljenih podataka.

### 2.1 Prikupljanja podataka: software na Arduinu

Sustav skuplja i sprema podatke sa GPS receivera kao i s temperaturnog senzora koje sprema u .csv <sup>1</sup> formatu koji je pogodan za kasniju obradu bilo putem Excel programskog alata ili drugih alata za obradu i vizualizaciju podataka.

---

<sup>1</sup>Comma Separated Values

## Poglavlje 3

# Zaključak

Ovdje dode zakljucak

# Literatura

- [1] R. H. Inc., “What is open source?.” <https://opensource.com/resources/what-open-source>. (3.8.2019.).