

# RAZVOJ I IMPLEMENTACIJA DIGITALNOG FILTRA

---

**Zovko, Petar**

**Master's thesis / Specijalistički diplomski stručni**

**2022**

*Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj:* **University of Split / Sveučilište u Splitu**

*Permanent link / Trajna poveznica:* <https://um.nsk.hr/um:nbn:hr:228:344556>

*Rights / Prava:* [In copyright](#)/[Zaštićeno autorskim pravom.](#)

*Download date / Datum preuzimanja:* **2025-02-27**



*Repository / Repozitorij:*

[Repository of University Department of Professional Studies](#)



**SVEUČILIŠTE U SPLITU**  
**SVEUČILIŠNI ODJEL ZA STRUČNE STUDIJE**

Specijalistički diplomski stručni studij Elektrotehnika

**PETAR ZOVKO**  
**ZAVRŠNI RAD**  
**RAZVOJ I IMPLEMENTACIJA**  
**DIGITALNOG FILTRA**

Split, rujan 2022.

**SVEUČILIŠTE U SPLITU**  
**SVEUČILIŠNI ODJEL ZA STRUČNE STUDIJE**

Specijalistički diplomski stručni studij Elektrotehnika

**Predmet:** Digitalna obrada signala

**Z A V R Š N I R A D**

**Kandidat:** Petar Zovko

**Naslov rada:** Razvoj i implementacija digitalnog filtra

**Mentor:** dr.sc. Tonko Kovačević, prof. v.š.

Split, rujan 2022.

## SADRŽAJ

Sažetak.....	1
1. UVOD.....	2
2. DOMENA MATEMATIČKE FUNKCIJE .....	3
2.1. Kontinuirana funkcija .....	3
2.2. Diskretna funkcija.....	4
3. OPERACIJE NAD FUNKCIJAMA.....	5
3.1. Fourierov red.....	5
3.2. Konvolucija.....	7
4. FILTRI I FILTRACIJA .....	8
4.1. Filtri s konačnim impulsnim odzivom - FIR .....	11
4.2. Filtri s beskonačnim impulsnim odzivom - IIR .....	13
5. ANALOGNO DIGITALNA PRETVORBA.....	15
5.1. Uzorkovanje.....	15
5.2. Kvantizacija .....	16
6. TESTNI SUSTAV .....	17
6.1. Senzor .....	17
6.2. Analogno digitalni pretvarač .....	19
6.3. Mikroupravljač.....	20
7. RAZVOJ I IMPLEMENTACIJA.....	21
7.1. Programsko sučelje .....	21
7.2. Hardversko sučelje.....	26
7.3. Evaluacija.....	28
8. Zaključak .....	31
LITERATURA .....	32
POPIS SLIKA .....	33
POPIS TABLICA.....	34

## **Sažetak**

### **Razvoj i implementacija digitalnog filtra**

Ovaj rad objašnjava teoriju i postupke potrebne za implementaciju i razvoj digitalnog filtra na ugradbeni sustav. Sadržaj je postupno podijeljen od teze prema praktičnom primjeru tako da se stekne razumijevanje za projektiranje ovakvog sustava, evaluaciju te moguću implementaciju u određeni postojeću sustav koji ne zadovoljava ciljane tehničke karakteristike.

**Ključne riječi:** DSP, filter, razvoj, ESP32, C, MATLAB

## **Summary**

### **Development and implementation of a digital filter**

This paper describes the theory behind and steps which are involved with implementing a digital filter to a embedded system. The content starts from known facts and thesis from which it builds up to an practical example to explain closer what is needed to design such system, evaluate it and implement it on an existing system that does not meet design requirements.

**Key words:** DSP, filter, development, ESP32, C, MATLAB

## 1. UVOD

Digitalna obrada signala spada u granu znanosti o obradi i analizi signala. Osim digitalne obrade, također postoji i njena analogna varijanta. Glavna razlika te dvije grane je način reprezentacije samog signala, odnosno da li signal promatramo kao neprekidnu funkciju u vremenu ili kao diskretnu vremensku funkciju. Priroda oko nas je analogna, odnosno neprekidna, pa je tako intuitivno zaključiti da je signale prirodnije promatrati kao takve. Međutim vršiti efikasne i brze matematičke operacije nad takvim funkcijama je iznimno teško.

Razvojem digitalnog računala i digitalne elektronike sredinom prošlog stoljeća otvorila se potpuno nova grana analize i obrade signala. Analogno digitalni pretvarači su postajali sve efikasniji i jeftiniji kao i specijalizirani sklopovi za obrade signala.

Digitalni signal je egzaktno moguće reproducirati za birani vremenski period, što je jedna od njegovih najvećih prednosti. Također je njime lako manipulirati, odnosno vršiti operacije nad njim, jer je on jednostavno samo sekvenca nula i jedinica koju digitalno računalo može razumjeti. Nad takvom sekvencom moguće je vršiti razne operacije. Filtraciju, kao način odvajanja korisne informacije od šuma. Multipleksiranje, odnosno fuziju više različitih signala u jedan.

Jednostavnije je također vršiti kompresiju signala za transmisiju ili pohranjivanje te implementirati algoritam za korekciju grešaka u signalu koje se mogu dogoditi uslijed transmisije.

Grane primjene digitalne obrade signala su raznolike, od medicine, industrije, znanosti, telekomunikacija, vojske i mnogih drugih gdje je god potrebno prikupljanje i analiza podataka.

## 2. DOMENA MATEMATIČKE FUNKCIJE

Matematička funkcija je općenito definirana kao pravilo kojim se vrijednosti iz skupa  $X$  preslikavaju u skup  $Y$ , odnosno matematički napisano:

$$f: X \rightarrow Y \quad (1)$$

Svaka vrijednost iz jednog skupa može se pridružiti točno jednoj vrijednošću iz drugog skupa te po tome vrijedi da je:

$$f(x) = y \quad (2)$$

gdje je:

$$x \in X; y \in Y \quad (3)$$

Postoje razne podjele matematičkih funkcija prema različitim svojstvima, međutim nama je najzanimljivija podjela po domeni. Domena funkcije je skup brojeva  $X$  za koje funkcija kao ulaznu varijablu iz tog skupa može dati odgovarajući izlaz iz skupa  $Y$ . Skup  $X$  se također naziva domena funkcije, dok se skup  $Y$  naziva kodomena ili područje vrijednosti. Ovisno o definiciji skupa  $X$  funkciju možemo opisati kao diskretnu i kontinuiranu.

### 2.1. Kontinuirana funkcija

Kontinuirana funkcija kao ulaz može primiti sve brojeve iz intervala omeđenim definiranim skupom. Ako uzmemo za primjer funkciju  $f(x)$  gdje je:

$$f(x) = x \quad (4)$$

Područje validnosti te funkcije ili validnosti ulaza se nalazi u intervalu od  $-\infty$  do  $+\infty$  odnosno sadrži sve realne brojeve. Ako napravimo poveznicu matematičke funkcije sa signalom, možemo zaključiti da je analogni signal kontinuirana funkcija u vremenu jer za svaki trenutak  $t$  u intervalu od  $0$  do  $+\infty$  postoji karakteristika signala koja se može funkcijski odrediti. Najčešće je ta karakteristika amplituda signala, međutim ne mora nužno biti. Kako kontinuirana funkcija obuhvaća sve vrijednosti intervala, proračuni sa njima najčešće zahtijevaju korištenje integralnog i diferencijalnog proračuna jer ulazna vrijednost može poprimiti beskonačno mnogo vrijednosti.

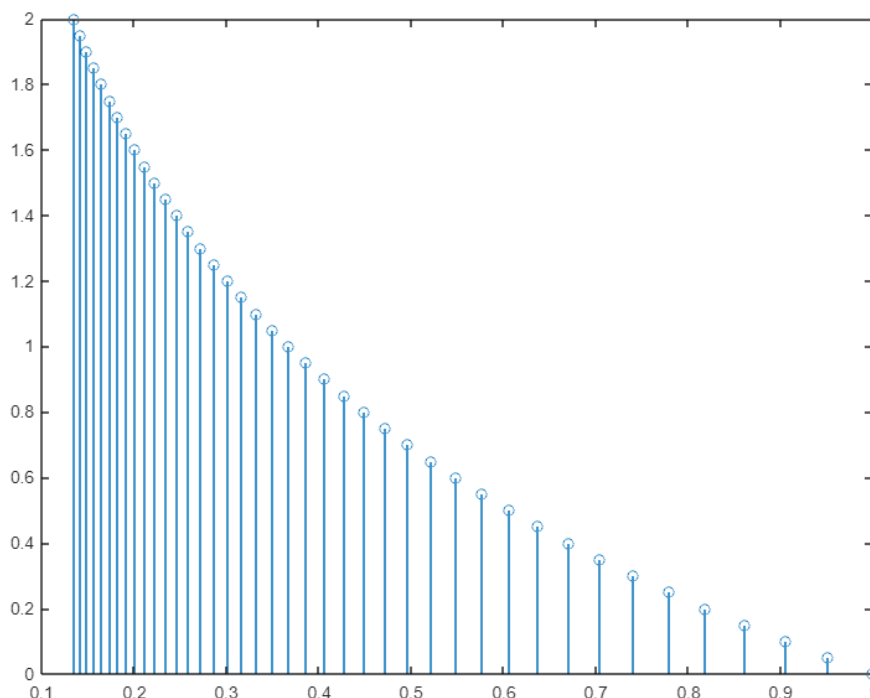
## 2.2. Diskretna funkcija

Diskretna matematička funkcija je definirana na način da joj domena sadrži egzaktne vrijednosti, odnosno interval te funkcije je određen sa točnim vrijednostima koje su validne kao ulaz i ne sadrže vrijednosti koje nisu definirane u skupu.

Ako za primjer uzmemo funkciju  $f[n]$  gdje je:

$$f[n] = \frac{1}{e^n} \quad (5)$$

U intervalu od  $n = 0, 0.05, 0.1, \dots, 1$  dobijemo:



Slika 1. Diskretna funkcija

Razmak između dvije točke funkcije je određen korakom intervala koji je definiran kao razlika između dvije susjedne točke po x-osi. Pretpostavkom da korak  $k$  teži u beskonačnost, diskretna funkcija postaje kontinuirana.

Ako napravimo analogiju sa signalima, digitalni signal je diskretna funkcija u vremenu s korakom određenim frekvencijom uzorkovanja, odnosno korak  $k$  iznosi:

$$k = \frac{1}{f} \quad (6)$$



### 3. OPERACIJE NAD FUNKCIJAMA

Definiranjem funkcije te njene domene kao i povezivanjem realnih signala u tu analogiju postavlja se pitanje kakva vrsta operacija se može upotrijebiti u obradi signala. Naime, matematičke operacije nad funkcijama koje su potrebne za analizu i obradu signala nerijetko su definirane samo za periodičke funkcije, pa se tako javlja problem kako vršiti proračune sa signalima koji nisu periodički jer su takvi signali najčešće prisutni u praksi kod senzorskih sustava. Odgovor je vrlo jednostavan, promatrati će se određeni interval datog signala (ili funkcije) i pretpostaviti će se da je taj signal periodičan tako da mu se definira period kao interval od  $-\infty$  do  $+\infty$ . Takvom pretpostavkom nad signalom mogu se vršiti operacije koje su najbitnije u obradi signala općenito, a to su Fourierova transformacija i konvolucija. Također, primjetno je da se analiza signala vrši na način da se promatra samo djelić signala istovremeno što uvelike smanjuje količinu podataka na kojoj se vrše radnje.

#### 3.1. Fourierov red

Fourierov red predstavlja periodičku funkciju kao zbroj sinusnih i kosinusnih funkcija raznih faza, amplituda i frekvencija. Svaka zasebna funkcija u toj sumi naziva se harmonik. Generalno gledano funkcija je prikazana kao beskonačna suma, dok se u praksi primjenjuju vrijednosti reda do određenog broja harmonika koji dovoljno dobro aproksimiraju početnu funkciju. Općenito Fourierov red možemo zapisati kao:

$$f(x) = a_0 + \sum_{n=1}^{\infty} a_n \cdot \cos\left(n \cdot \frac{\pi}{L}\right) + \sum_{n=1}^{\infty} b_n \cdot \sin\left(n \cdot \frac{\pi}{L}\right) \quad (7)$$

gdje je:

- $f(x)$  – funkcija koja se želi prikazati kao red
- $L$  – polovica perioda funkcije
- $a_0, a_n, b_n$  – Fourierovi koeficijenti

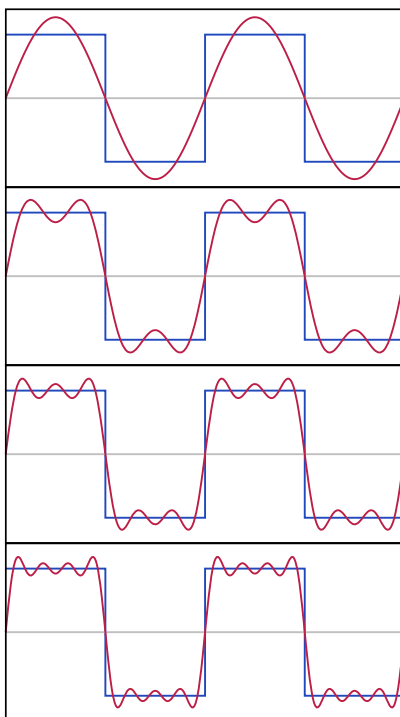
Koeficijente je potrebno izračunati za svaku funkciju koja se želi prikazati kao red.

Kako bi funkcija mogla biti prikazana kao red ona mora zadovoljiti nekoliko uvjeta:

- Mora biti apsolutno integrativna unutar perioda, odnosno mora zadovoljavati uvjet:

$$\int_{-L}^L |f(x)|^2 dx < \infty \quad (8)$$

- Mora imati konačan broj ekstrema u intervalu
- Mora imati konačan broj prekida u intervalu
- Mora biti ograničena



Slika 2. Aproksimacija funkcije Fourierovim redom

Kada se funkcija raspiše kao Fourierov red, tada se može promatrati u frekvencijskoj domeni što uvelike olakšava analizu signala kao i početne korake za razvoj adekvatnog filtra. Prebacivanje funkcije iz vremenske domene u frekvencijsku naziva se Fourierova transformacija. Kako je svaki harmonik reda određen fazom, amplitudom i frekvencijom pretvorba se vrši na način da se amplituda i frekvencija preslikaju na zasebni A-f graf. Sve prethodne teze definirane su na primjeru kontinuirane funkcije, međutim navodi vrijede i za diskretne funkcije.

### 3.2. Konvolucija

Konvolucija je matematička operacija nad dvjema funkcijama čiji je rezultat funkcija koja govori kako oblik jedne utječe ili mijenja oblik druge funkcije. Općenito se konvolucija može zapisati kao:

$$y(t) = f(t) * h(t) \quad (9)$$

Postupak konvolucije je ipak malo kompleksniji od prikazane jednačbe, pa tako jednačbu (9) valja promatrati u njenom proširenom obliku:

$$\int_{-\infty}^{+\infty} f(\tau) \cdot h(t - \tau) d\tau = y(t) \quad (10)$$

gdje je:

- $\tau$  – pomoćna varijabla
- $t$  – vremensko kašnjenje, odnosno pomak za računanje konvolucije u određenoj točki

Kako bi se izračunala funkcija konvolucije, potrebno je jednu funkciju zrcaliti po y-osi kako bi zadržali integritet proračuna. Konvolucija je komutativna operacija tako da nije bitno koja će se funkcija od dvije zrcaliti.

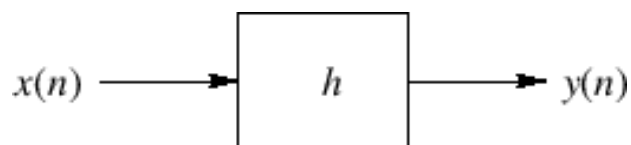
Intuitivniji način shvaćanja konvolucije je taj da se jedna funkcija zamisli kao „pravilo“ kojim će se mijenjati drugu funkciju. Takvom tvrdnjom može se zaključiti da je konvolucija funkcije s određenom funkcijom pravila zapravo oblikovanje izlaza prema definiranoj želji. Pravilo oblikovanja se još naziva i kernel.

U vremenskoj domeni za proračun konvolucije potrebno je vršiti integraciju uz dosta operacija množenja, međutim konvolucija u vremenskoj domeni predstavlja isto što i množenje u frekvencijskoj domeni, odnosno ako se nad funkcijama sa kojima se želi izračunati funkcija konvolucije napravi Fourierova transformacija, rezultante transformacije je potrebno pomnožiti te ako se želi rezultantu funkciju u vremenskoj domeni potrebno je napraviti inverznu Fourierovu transformaciju. Ova karakteristika je osnovi alat kod analize i obrade signala.

## 4. FILTRI I FILTRACIJA

Iz perspektive obrade signala, filtracija je postupak uklanjanja nepoželjnih komponenti signala. Filtar je uređaj ili proces s kojim se provodi postupak filtracije. Ako se želi generalizirati, filtri se mogu podijeliti na analogne i digitalne, odnosno diskretne filtre (sa konačnim impulsnim odzivom ili *FIR*) te kontinuirane (sa beskonačnim impulsnim odzivom ili *IIR*). U ovom smislu diskretni filtari znači da njihov impulsni odziv u određenom trenutku  $T$  dosegne vrijednost nula, dok kod kontinuiranog filtra vrijednost odziva zadržava stabilnu vrijednost beskonačno mnogo vremena.

Filtracija signala moguća je i sa diskretnim analognim komponentama, u tom slučaju se može konstruirati samo kontinuirani filtari, dok digitalni filtri mogu biti diskretni i kontinuirani. Razlika između te dvije vrste je prisustvo povratne veze, koja nije prisutna kod diskretnog filtra.

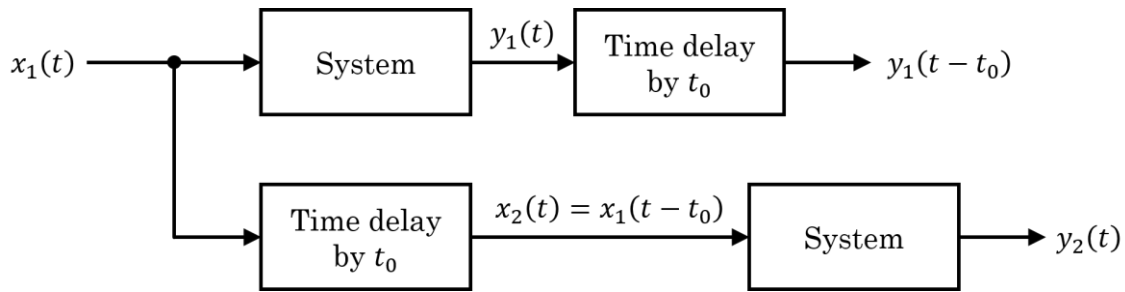


Slika 3. Blok shema filtracije [1]

Slika 3 pokazuje osnovnu blok shemu filtra, gdje je:

- $x[n]$  - ulazni signal
- $y[n]$  - filtrirani signal
- $h$  – funkcija filtracije

Kako bi filtracija bila moguća, sustav koji predstavlja filtari mora biti linearan vremenski neovisan (*linear time invariant - LTI*). Takav sustav proizvodi izlazni signal iz bilo kojeg ulaznog signala podložnog ograničenjima linearnosti i vremenske neovisnosti. Sustav je linearan ako promjena ulaza za određeni koeficijent proporcionalno djeluje na izlaz, dok vremenska neovisnost znači da sustav nije direktna funkcija vremena. Pod indirektna funkcija vremena se misli da je neovisan o relativnoj poziciji u vremenu to jest da sustav ima jednak učinak u trenutku  $t_0 = 0$  i  $t_0 = 100$ .



Slika 4. Prikaz vremenski neovisnog sustava [1]

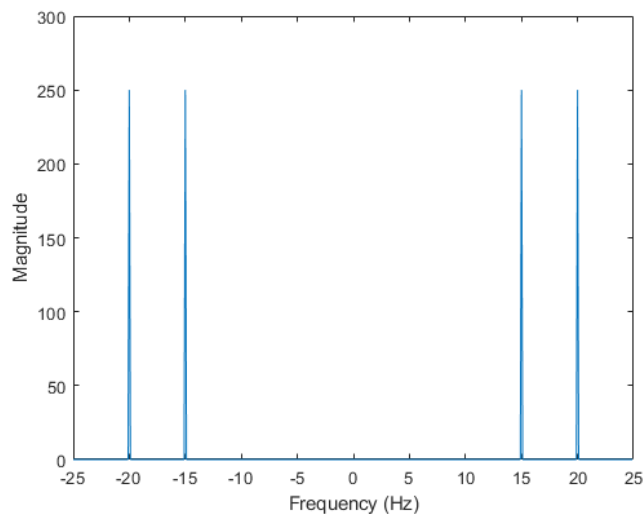
Važnost da sustav filtra bude LTI je u tome što je onda moguće definirati produkt, odnosno rezultantu filtracije pomoću konvolucije na način:

$$y[n] = h * x[n] \quad (11)$$

Ova teza proizlazi iz zaključka da je svaki LTI sistem potpuno definiran svojim impulsnim odzivom. Nadalje, ako se uzme određeni signal koji je potrebno filtrirati prvi korak će biti napraviti Fourierovu transformaciju ulaza i pogledat njegov frekvencijski spektar. Ako se za primjer pogleda signal koji je u vremenskoj domeni definiran kao:

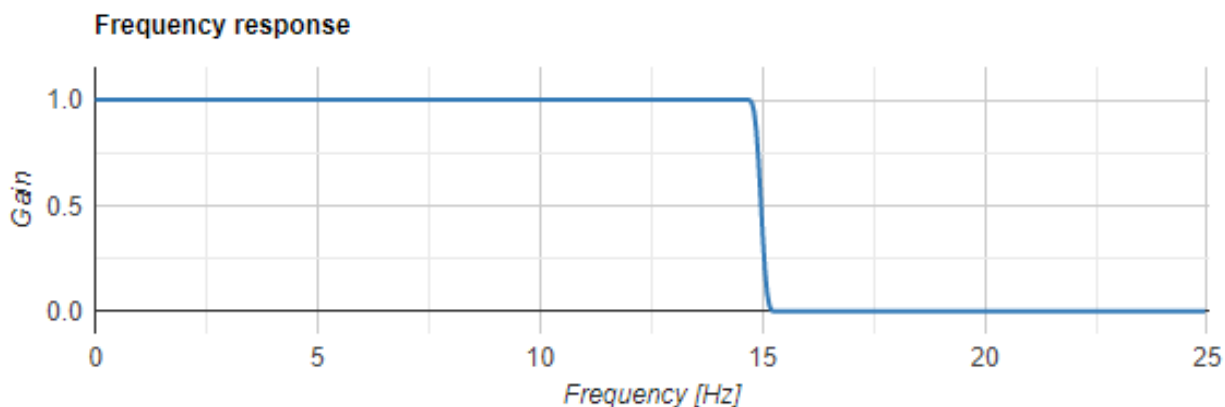
$$x[n] = \sin(2\pi 15n) + \sin(2\pi 20n) \quad (12)$$

Njegov frekvencijski spektar će imati vrijednosti na frekvencijama 15 Hz i 20 Hz.



Slika 5. Frekvencijski spektar funkcije

Kako je već rečeno da je konvolucija u vremenskoj domeni isto što je množenje u frekvencijskoj domeni, može se zaključiti da je dizajn filtra postupak gdje se odredi koeficijent pojačanja za svaku diskretnu frekvenciju. Da bi se odredio filter koji će propuštati samo frekvencije do 15Hz jednostavno se definira funkciju koja će imati frekvencijski odziv takav da frekvencije do 15 Hz imaju koeficijent 1, dok sve ostale imaju koeficijent 0. Takav frekvencijski odziv prikazan je na slici 6.



Slika 6. Frekvencijski odziv filtra

Koeficijenti koji se odrede za određenu frekvenciju nazivaju se pojačanje. Filtracijom je moguće prigušiti, pojačati ili ukloniti određene spektre signala. Ova značajka se najčešće koristi kod obrade audio signala.

Filtraciju je moguće vršiti transformacijom iz vremenske u frekvencijsku domenu te ponovnom transformacijom u vremensku domenu ili frekvencijski odziv filtra transformirati u impulsni odziv te vršiti konvoluciju u vremenskoj domeni. Koji način će se primijeniti u praksi uvelike ovisi o sustavu na koji se filter implementira. Praktično proračun i implementacija nisu ovako jednostavni, jer u realnom sustavu postoji kašnjenje između ulaza i izlaza koje s lošim filtrom može biti jako veliko. Također realni filter neće biti u mogućnosti ukloniti sve nepoželjne frekvencije iz spektra što će detaljnije biti prikazano u poglavlju implementacije.

#### 4.1. Filtri s konačnim impulsnim odzivom - FIR

Ako se filterski krug definira kroz prijenosnu funkciju, može se tvrditi da su izlazni i ulazni signal vezani relacijom:

$$H(z) = \frac{Y(z)}{X(z)} \quad (13)$$

gdje je:

- $Y(z)$  – z transformacija izlaznog signala
- $X(z)$  – z transformacija ulaznog signala
- $H(z)$  – prijenosna funkcija sustava

S obzirom da su praktično izvedivi filtri prijenosne funkcije koje su omjer dva polinoma sa realnim koeficijentima također se može napisati sljedeće:

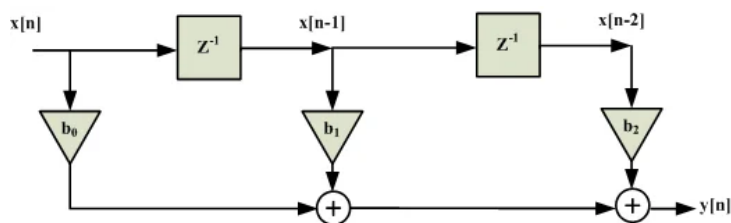
$$H(z) = \frac{\sum_{k=0}^{M-1} b_k z^{-k}}{1 + \sum_{k=1}^N a_k z^{-k}} \quad (14)$$

Filtri s beskonačnim odzivom su posebni slučaj ove prijenosne funkcije gdje je  $N = 0$ , odnosno nema povratne veze. Pa tako jednadžba prijenosne funkcije postaje:

$$H(z) = \sum_{k=0}^{M-1} b_k z^{-k} \quad (15)$$

što povratkom u diskretnu vremensku domenu postaje:

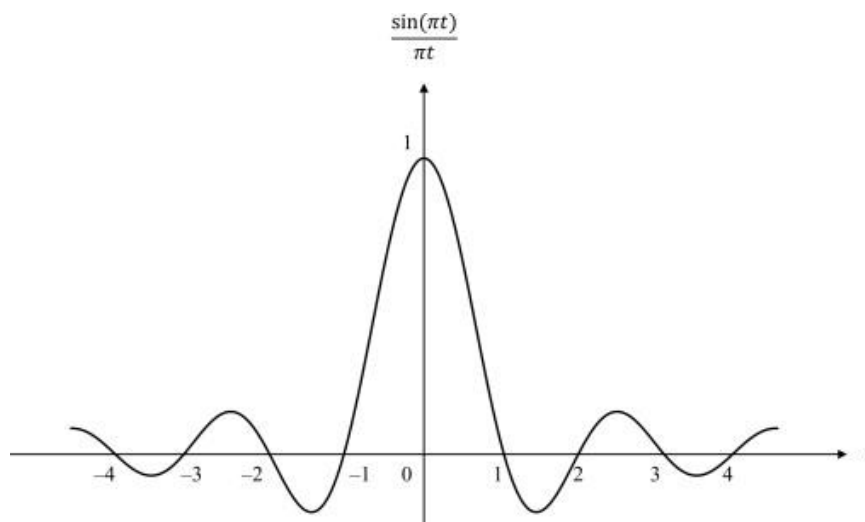
$$y[n] = \sum_{i=0}^{M-1} b_i x[n-i] \quad (16)$$



Slika 7. Blok shema FIR filtra

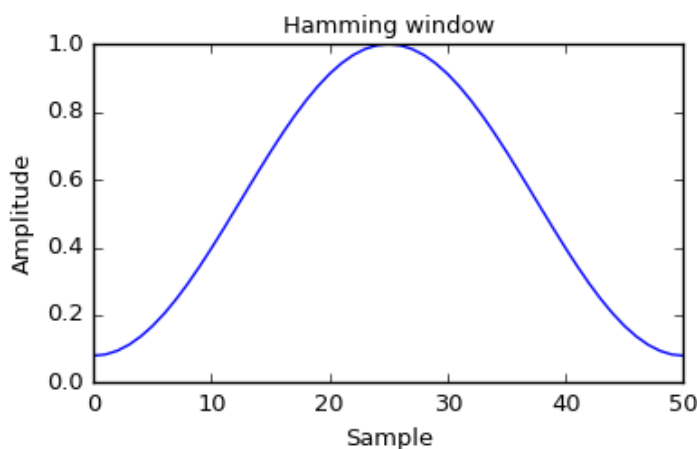
Može se primijetiti da ovakav filter za djelovanje koristi samo sadašnje i prošle vrijednosti ulaza bez ikakve povratne veze. Najveća značajka ovakvog filtra je što ima linearan fazni pomak, odnosno kašnjenje mu je predvidivo. Također zbog izostanka povratne veze ne treba brinuti o stabilnosti, međutim praktično ovakvi filtri zahtijevaju mnogo više matematičkih operacija nego IIR filtri sličnih karakteristika pa su time sporiji za izračunavanje. Osim toga ovakvi filtri nisu efikasni kada je vrijednost granične frekvencije velika s obzirom na frekvenciju uzorkovanja.

Kod praktičnog dizajna FIR filtra potrebno je odrediti propusni i nepropusni frekvencijski spektar. Funkcija koja se dobije takvim definiranjem je najčešće pravokutna funkcija u frekvencijskoj domeni. Inverznom Fourierovom transformacijom pravokutne funkcije iz frekvencijske domene u vremensku, može se vidjeti da je rezultanta te operacije sinc funkcija.



Slika 8. Sinc funkcija [2]

Međutim problem se javlja u tome što sinc funkcija ima beskonačno veliko trajanje što znači da bi signal bio filtriran idealnim pravokutnim filtrom treba imati beskonačno velik broj uzoraka što je praktično nemoguće. Kako bi se moglo praktično koristiti ovakav filter potrebno je odrediti prozor djelovanja, s obzirom da je dostupan samo konačan broj uzoraka nad kojim se vrši filtracija. Također umjesto pravokutne funkcije u frekvencijskoj domeni može se koristiti prozorske funkcije kako bi sinc funkcija u vremenskoj domeni sadržavala dovoljnu količinu informacije da se idealizirani filter praktično preslikava u dovoljno dobru aproksimaciju.



Slika 9. Hamming prozor [1]



## 4.2. Filtri s beskonačnim impulsnim odzivom - IIR

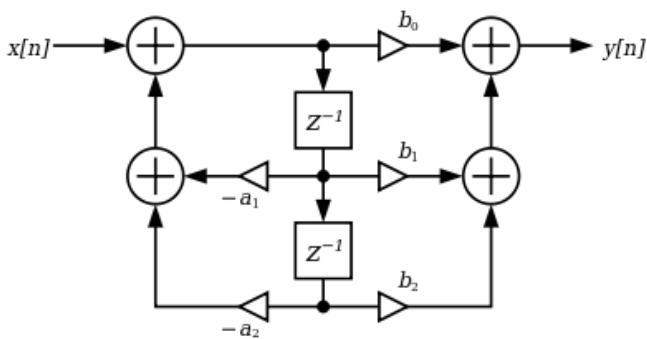
Prijenosna funkcija ovog filtra jednaka je onoj definiranoj za opće filtre, pa tako vremensku domenu filtra s beskonačnim odzivom moguće je definirati sljedećom formulom:

$$y[n] = \frac{1}{a_0} \left( \sum_{i=0}^P b_i \cdot x[n-i] - \sum_{j=1}^Q a_j \cdot y[n-j] \right) \quad (17)$$

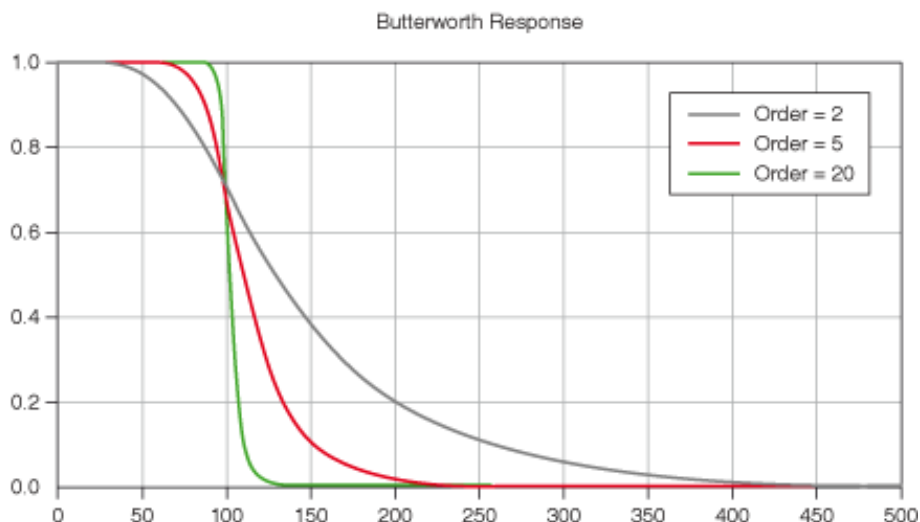
gdje je:

- $y[n]$  – izlazna vrijednost
- $x[n]$  - ulazna vrijednost
- $b_i$  koeficijente ulazne vrijednosti
- $a_j$  koeficijente povratne veze
- $P$  red ulaznog dijela filtra
- $Q$  red povratne veze

Filtar u principu koristi povratnu vezu koja pridonosi proračunu nove vrijednosti. Brojem povratnih veza definiran je red filtra što u teoriji gledano znači da veći broj povratnih veza, odnosno informacija o početnoj vrijednosti, daje precizniju i vjerniju reprezentaciju signala bez šuma. Dok je to u teoriji točno, u praksi veći broj povratnih veza znači veće kašnjenje koje svaka povratna veza dodatno uvrsti. Za pravilnu i funkcionalnu upotrebu filtra potrebno je napraviti kompromis između „čistoće podataka“ i kašnjenja istih.



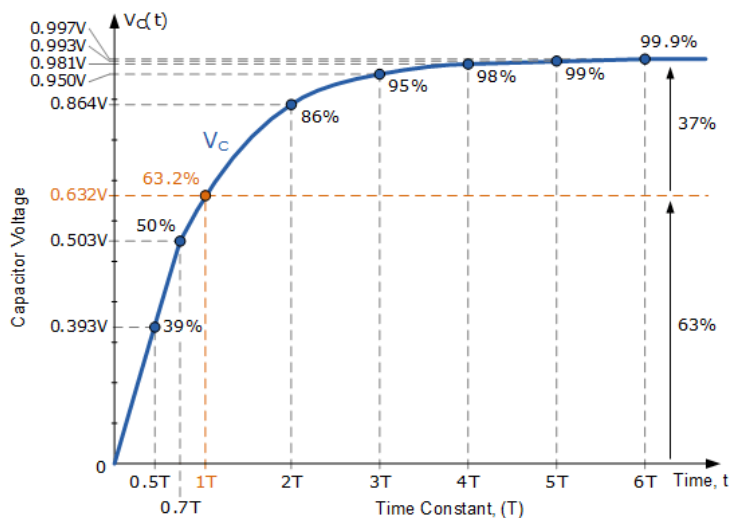
Slika 10. Blok shema IIR filtra [6]



Slika 11. Odziv Butterworth IIR filtra s različitim brojem povratnih veza [6]

Najjednostavniji primjer IIR filtra jest RC krug koji djeluje kao nisko propusni filter. Kako je iz teorije elektrotehnike poznato, kondenzator nikada neće dosegnuti stanje gdje je potpuno napunjen, pa samim time ima beskonačan impulсни odziv.

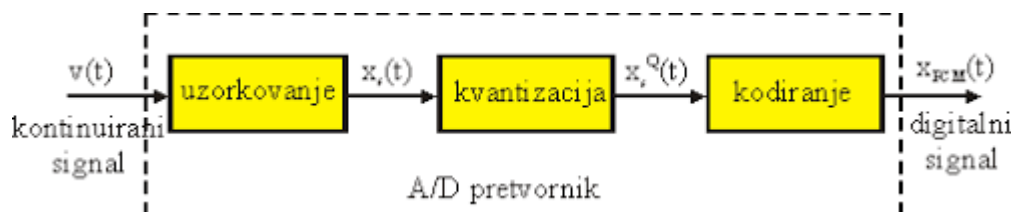
Kašnjenje ili tromost takvog filtra određena je vremenskom konstantom koja je umnožak otpora i kapaciteta.



Slika 12. Impulсни odziv RC kruga

## 5. ANALOGNO DIGITALNA PRETVORBA

Osnovni princip analogno digitalne pretvorbe se sastoji od tri koraka: diskretizacija po vremenu (uzorkovanje), diskretizacija po amplitudi (kvantizacija), pretvorba dekadске amplitude u binarnu (kodiranje).



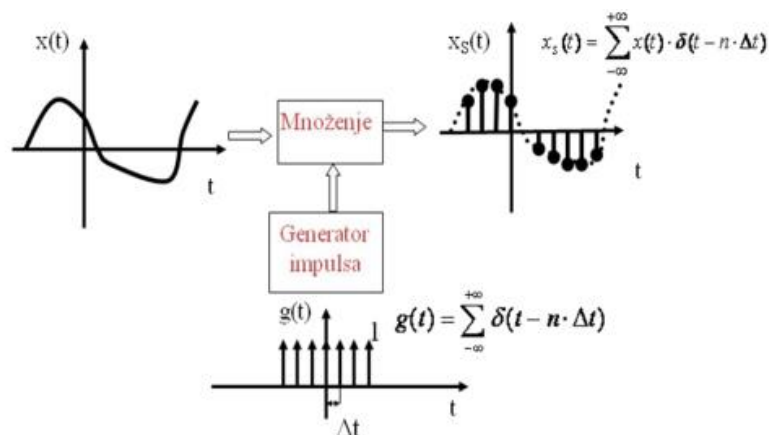
Slika 13. Blok shema A/D pretvarača [9]

### 5.1. Uzorkovanje

Kako su analogni signali kontinuirane funkcije po vremenu, da bi bilo moguće digitalno obraditi signal potrebno ga je prvo diskretizirati po vremenu. Postupak se izvodi na način da se ulazni (analogni signal) dovede u množitelj te se množi sa signalom uzorkovanja frekvencije  $f$ . Signal uzorkovanja je slijed impulsa jako kratkog trajanja. Na izlazu iz množitelja dobit će se signal sa diskretnim vrijednostima amplitude u vremenu.

Frekvencija uzorkovanja  $f$  je određena gornjom graničnom frekvencijom ulaznog signala  $f_c$  prema Nyquist-Shannon teoremu:

$$f \geq 2f_c \quad (18)$$

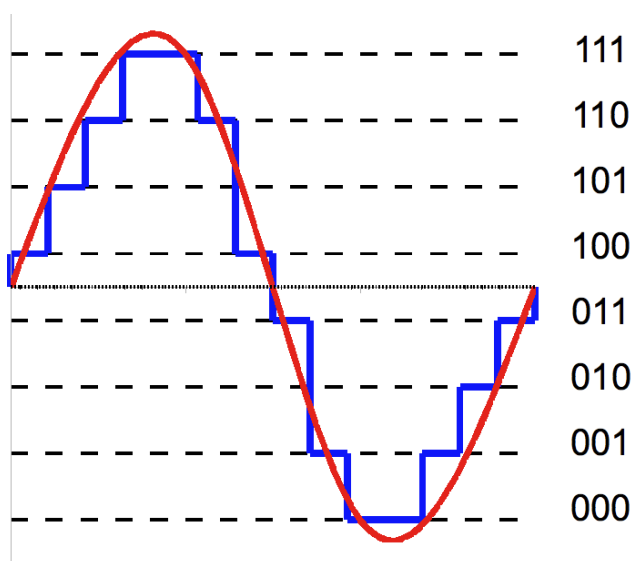


Slika 14. Uzorkovanje [10]

## 5.2. Kvantizacija

Kako realni analogni signal sadrži jako širok pojas frekvencija, uzorkovanjem signala se gubi informacija. Gornju graničnu frekvenciju potrebno je odrediti tako da se u izabranom spektru nalaze sve frekvencije koje sadrže bitnu informaciju signala.

Kvantizacijom se diskretiziraju vrijednosti amplitude dobivene uzorkovanjem. Postupak se provodi na način da se odredi minimalna i maksimalna vrijednost amplitude  $A_{min}$  i  $A_{max}$  te se područje podijeli na  $L$  broj razina. Svaka razina kodirana je određenom binarnom vrijednošću, a broj razina definira preciznost pretvorbe. Ako se za primjer promatra 8 bitni koder, znači da se može imati kodirano maksimalno 256 razina amplitude.



Slika 15. Kvantizacija

Amplituda se kodira na onu veću vrijednost koja zadovoljava uvjet da se nalazi u među pojasu dvije razine. Postupak kvantizacije utiskuje pogrešku u digitalni signal koju je nemoguće izbjeći, a ona ima maksimalnu vrijednost  $q$ :

$$q = \frac{A_{max} - A_{min}}{2L} \quad (19)$$

## 6. TESTNI SUSTAV

Testni sastoji se od senzora (tenzometar), mikroupravljača i analogno digitalnog pretvornika. Kako bi se simulirao signal sa širokim spektrom frekvencija, na tenzometar će se pričvrstiti šipku od opružnog čelika koja će titranjem stvarati prigušenu vibraciju. Tenzometar se najčešće koristi kao uređaj za mjerenje mase, gdje je mjerena vrijednost zapravo istosmjerni signal, pa tako bilo koja frekvencija superponiran na signal predstavlja šum.

### 6.1. Senzor

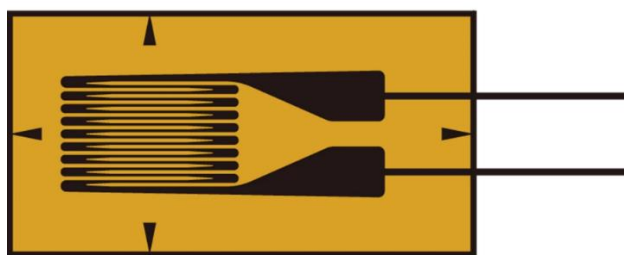
Tenzometar je uređaj koji mjeri naprezanje. Princip rada mu je vrlo jednostavan, na tanki fleksibilni film raspoređen je vodič u obliku rešetke kako bi se povećao ukupni otpor voda. Kad se na film djeluje silom naprezanja zbog promjene presjeka i duljine mijenja se ukupni otpor voda prema relaciji:

$$R = \rho \cdot \frac{l}{S} \quad (20)$$

gdje je:

- R – otpor voda [  $\Omega$  ]
- l – ukupna duljina voda [ m ]
- S – presjek voda [  $\text{mm}^2$  ]
- $\rho$  – električna provodnost [  $\Omega\text{m}$  ]

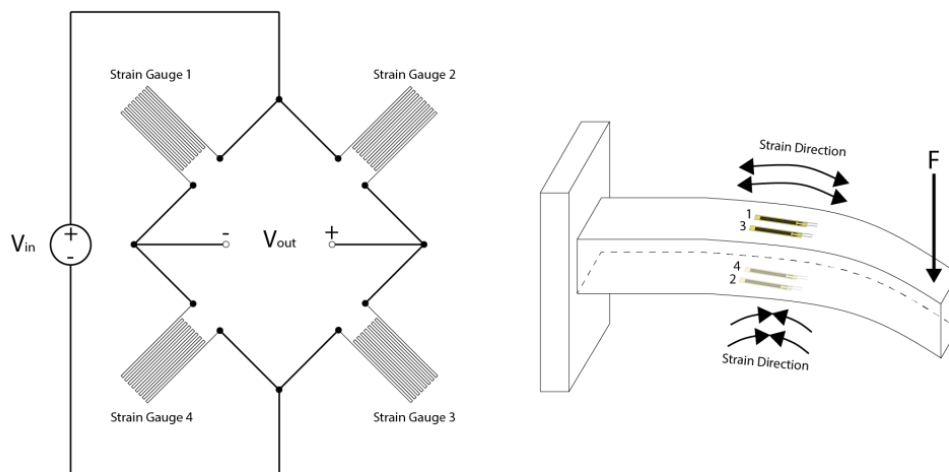
Promjenom otpora, mijenja se i pad napona na tenzometru koji je proporcionalan sili naprezanja na tenzometar. Ovaj uređaj se najčešće koristi spajanjem četiri jednaka tenzometra u Whietstone-ov most.



Slika 16. Tenzometar [6]

Valja naglasiti da su promjene otpora relativno male, a samim time i padovi napona, pa tako direktnim spajanjem na analogni ulaz mikroupravljača ili direktnim mjerenjem neće se moći očitati korisnu informaciju. Za ovakve uređaje postoje posebni integrirani krugovi prilagođeni pretvaranju iznimno malih napona u digitalni signal. Temperaturna stabilnost ovakvih uređaja je relativno loša, s obzirom da temperatura osim na dimenzije djeluje i na električki otpor. Međutim adekvatnim izborom materijala te temperaturnom kompenzacijom, utjecaji temperature mogu se svesti na prihvatljive vrijednosti.

Spajanjem više tenzometara u mjerni most kompenziraju se fizikalne greške koje se mogu pojaviti pri mjerenju, ali i smanjuje djelovanje šuma na cjelokupni sustav odnosno na mjerenu vrijednost.



Slika 17. Spoj tenzometra u Whietstone-ov most [6]

Kako je promjena otpora proporcionalna naprezanju koje djeluje na tenzometar, masa koja djeluje može se izračunati sljedećim relacijama:

$$F \propto k \times \Delta U \quad (21.1)$$

$$m \propto \frac{1}{g} \times F \quad (21.2)$$

$$m \propto K \times \Delta U \quad (21.3)$$

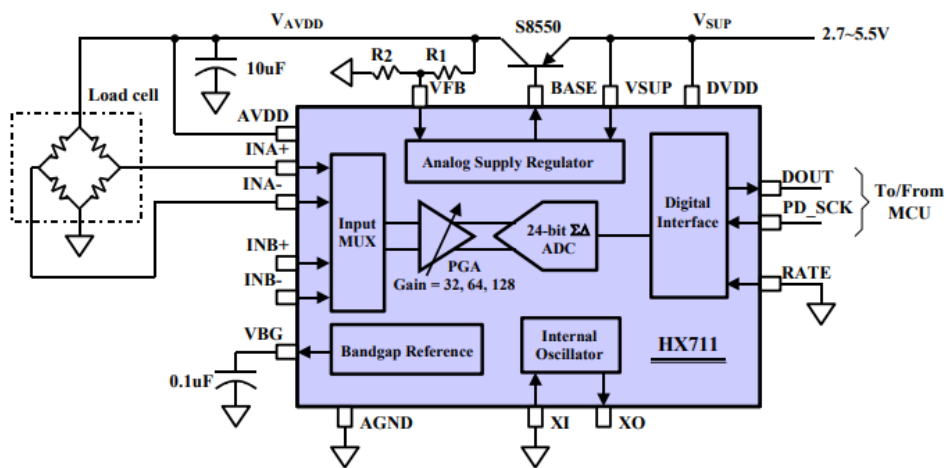
gdje je:

- $F$  – sila težine [ N ]
- $k$  – konstanta proporcionalnosti
- $\Delta U$  – razlika potencijala [ V ]
- $g$  – gravitacijsko ubrzanje [  $m/s^2$  ]
- $m$  – masa [ kg ]
- $K$  – konstanta proporcionalnosti -  $\frac{k}{g}$

Konstantu proporcionalnosti potrebno je odrediti eksperimentalno.

## 6.2. Analogno digitalni pretvarač

HX711 je integrirani krug osmišljen od strane AVIA Semiconductor, tvrtke koja se bavi proizvodnjom integriranih krugova. U principu ovaj čip je samo jako osjetljivi analogno digitalni pretvarač. Kod rada s ovakvim uređajem trebamo voditi računa o količini šuma u napajanju kao i dizajnu same tiskane ploče. Na slici 18 može se vidjeti funkcijski dijagram HX711.



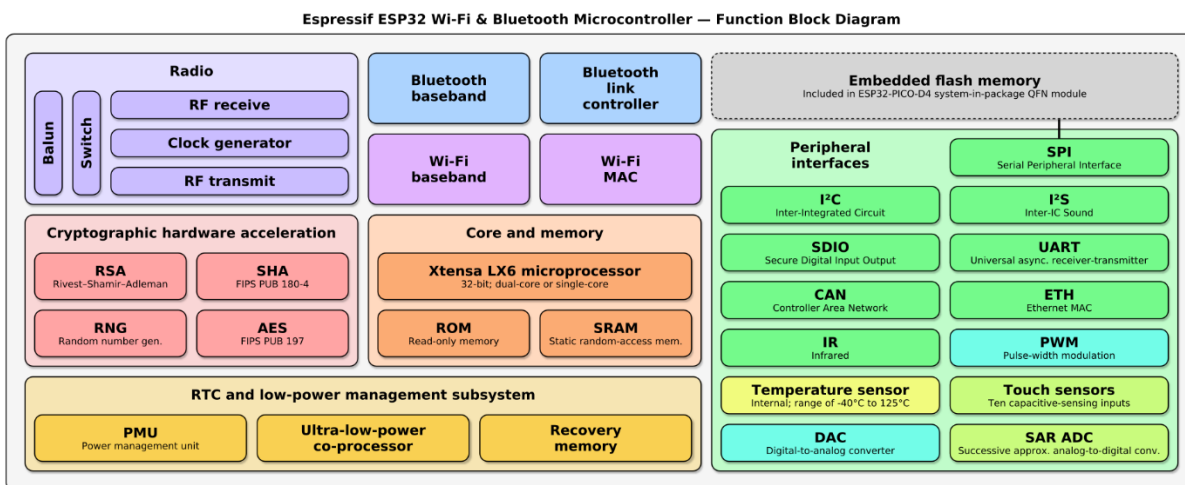
Slika 18. HX711 blok shema [11]

Može se primijetiti da su mjerne točke sa Whietstone-ovog mosta direktno spojene na nožice INA+ i INA-, što su ulazne točke prvog kanala multipleksera. Dalje se signal doveden na multiplekser pojačava što znači da se, osim korisnog signala, proporcionalno pojačava i šum u sustavu što dovodi do pogrešnog očitavanja. Kako bi se minimizirao taj problem, čip u sebi ima određeno sklopovlje za filtraciju napajanja za čiju pravilnu funkciju je potrebo dodati minimalno vanjskih komponenti.

Analogno digitalni pretvarač unutar integriranog kruga može raditi na frekvenciji uzorkovanja od 10 Hz ili 80 Hz te za komunikaciju korist dvije linije: DOUT i CLK. Kao signalizaciju da su podatci pretvoreni i spremni za slanje, HX711 DOUT liniju prebaci sa visoke na nisku logičku razinu te u tom trenutku mikroupravljač može početi slati signal na CLK liniju te očitavati DOUT liniju. Potrebno je poslati 24 CLK signala kako bi se primila potpuna informacija, a dodatnim CLK signalima se vrši konfiguriranje unutrašnjeg pojačala.

### 6.3. Mikroupravljač

ESP32 modul proizvod je kineske tvrtke *Espressif Systems* koja je specijalizirana za razvoj i proizvodnju sustava u čipu (engl. *System on Chip – SoC*). Ovaj modul sastoji se od procesora Xtensa LX6 s dvije 32-bitne jezgre i takta rada od 240 MHz. Posjeduje 320 KiB RAM-a te 448 KiB integriranog ROM-a (uz dodatnih 4MB vanjske *flash* memorije) kao i 40 ulaz/izlaza opće namjene (engl. *General-purpose input/output*). Popis svih mogućnosti može se vidjeti na slici 19.



Slika 19. Funkcijski dijagram ESP32 [3]

Najveća značajka ovog mikroupravljačkog modula je visoki stupanj integracije, odnosno na sebi sadrži sve potrebne komponente za korištenje bežične komunikacije. Moguće ga je koristiti kao zaseban komunikacijski modul ili kao međusklop za bežično povezivanje drugih sustava. Performansama je dovoljno snažan da na sebi može pokrenuti i operativni sistem. Kako bi se maksimalno iskoristilo efikasno korištenje njegovih mogućnosti i optimizirao paralelan rad, tvornički je preporučeno korištenje *freeRTOS* operativnog sistema koji radi u pravom vremenu.



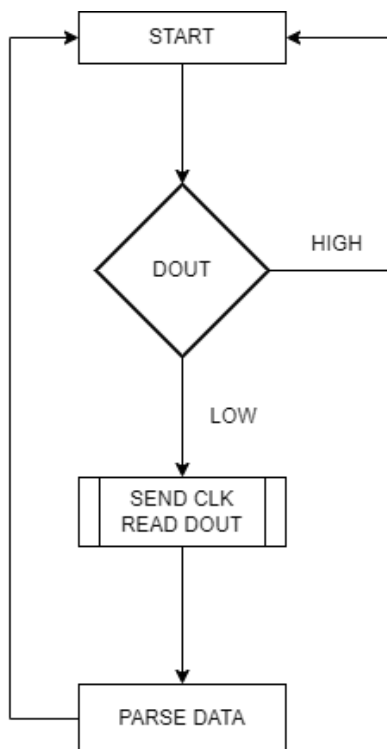
Slika 20. ESP32 modul [3]



## 7. RAZVOJ I IMPLEMENTACIJA

### 7.1. Programsko sučelje

Početna točka prije prikupljanja podataka je pisanje adekvatnog upravljača za HX711 analogno digitalni pretvarač. Iz tehničke dokumentacije spomenutog integriranog kruga zaključeno je da za komunikaciju koristi nestandardno serijsko sučelje sa dvije linije: CLK odnosno *clock* i DOUT koja služi za signalizaciju spremnosti podataka i slanje podataka. S obzirom da je sučelje nestandardno, rad drivera provjeren je osciloskopom.



Slika 21. Blok dijagram upravljača [6]

Kako nije poželjno da se sustav zaustavlja dok se čeka informacija o dostupnosti podataka, iskorištene su mogućnosti operativnog sistema koji upravlja radom ESP32 platforme. Upravljač je postavljen u zasebnu dretvu te udjeljuje procesorsko vrijeme ostatku programa dok čeka promjenu DOUT linije. Nakon registrirane promjene, čitanje se mora odvititi brzo i bez ometanja zbog čega u dijelu čitanja treba isključiti prekide i zauzeti procesorsko vrijeme u cijelom tijeku razmjene podataka. Iz razloga što ESP32 posjeduje dvije neovisne procesorske jezgre druga jezgra će raditi neometano. Nakon očitavanja cijele informacije, vrijednost treba pretvoriti u korisnu vrijednost i vršiti daljnju potrebnu obradu.

```

while(digitalRead(HX_DATA))
    TaskDelay(10);

portENTER_CRITICAL(&mux);

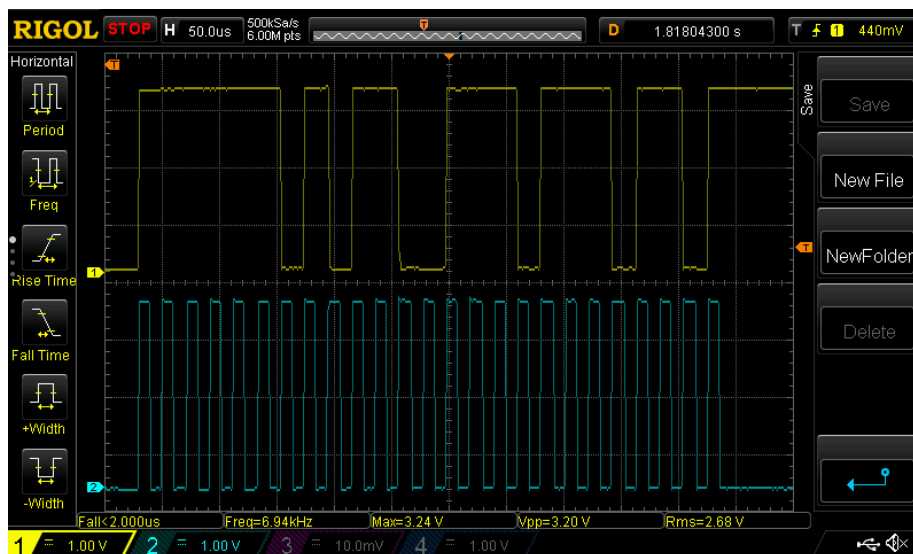
for(uint8_t i = 0; i < 24; i++)
{
    digitalWrite(HX_CLK, true);
    ets_delay_us(CLK_HIGH);
    tmp = tmp << 1;
    if(digitalRead(HX_DATA))
        tmp++;
    digitalWrite(HX_CLK, false);
    ets_delay_us(CLK_HIGH);
}
for(uint8_t i = 0; i < GAIN; i++)
{
    digitalWrite(HX_CLK, true);
    ets_delay_us(CLK_HIGH);
    digitalWrite(HX_CLK, false);
    ets_delay_us(CLK_HIGH);
}

portEXIT_CRITICAL(&mux);

```

Slika 22. HX711 upravljač [6]

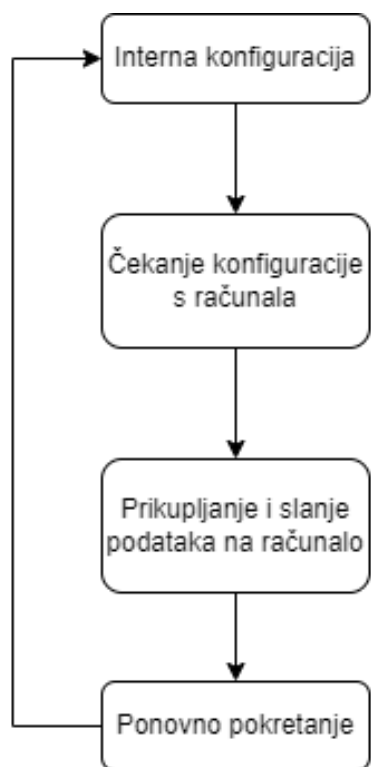
Pregledom snimki osciloskopa jasno se vidi funkcija opisana u dokumentaciji analogno digitalnog pretvarača. Okidač osciloskopa je postavljen na rastući brid prvog kanala kako bi se snimkom obuhvatila i signalizacija spremih informacija kao i preneseni podatci. Gornji signal na slici 23 predstavlja DOUT liniju sa HX711 integriranog kruga. Nakon što je informacija spremna za čitanje linija prelazi sa visoke na nisku naponsku razinu. Kada je mikroupravljač registrirao promjenu počinje proces očitavanja.



Slika 23. Funkcionalnost HX711 [6]

Slanjem 24 uzastopna impulsa (donji signal, slika 23) vrši se očitavanje pretvorene vrijednosti. Impulsi ne smiju trajati duže od 50 mikrosekundi, jer duže trajanje signala od toga analogno digitalni pretvarač stavlja u način rada za uštedu energije. Nakon slanja svih 24 impulsa za dohvat informacije, šalju se dodatni impulsi koji služe za konfiguriranje HX711 te se oni trebaju slati nakon svake konverzije. Ovdje se šalje jedan dodatni impuls koji predstavlja vrijednost pojačanja unutrašnjeg pojačala od 128.

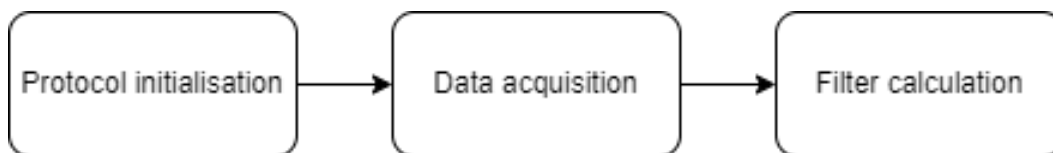
Kako bi se podatci mogli analizirati, potrebno je uz upravljač napisati i program koji će prikupljene podatke slati na računalo za daljnje procesiranje. Sljedeći dijagram toka prikazuje njegov rad:



Slika 24. Dijagram toka programa za prikupljanje podataka [6]

Nakon početne inicijalizacije programskih komponenti potrebnih mikroracunalu za normalan rad, program čeka podatak o vremenskom trajanju prikupljanja podataka s računala. Fiksni vremenski okvir prikupljanja podataka se koristi kako bi jednostavnije izolirali potreban slučaj za analizu. Nakon što je podatak primljen program, počinje očitavati podatke sa analogno digitalnog pretvornika te ih istovremeno šalje na računalo. Nakon što je završio zadani vremenski period mikroupravljač se ponovno pokreće i cijeli ciklus se iznova ponavlja.

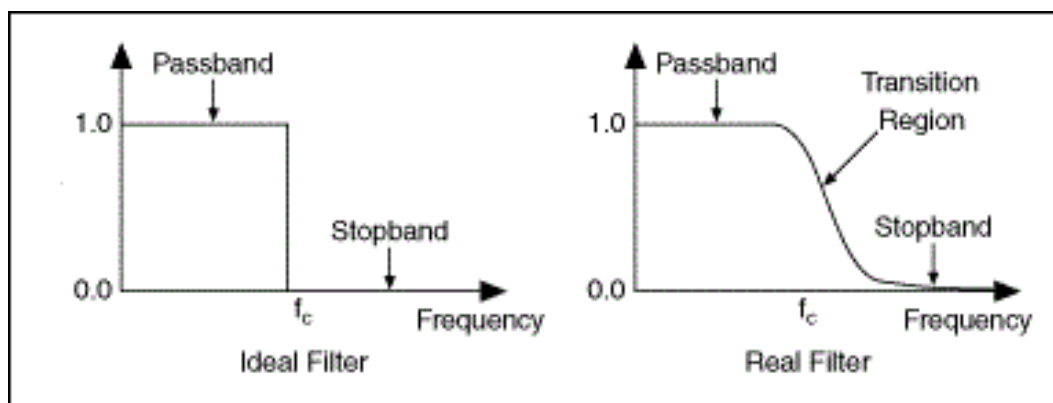
U svrhu prikupljanja i analize podataka te dizajniranja i evaluacije filtra koristi se programski paket MATLAB R2021b. Skripta se može podijeliti na tri osnovne funkcionalne jedinice: uspostava komunikacije, prikupljanje podataka i proračun filtra.



Slika 25. Dijagram toka MATLAB skripte [6]

Ovakvom organizacijom prikupljanja i analize može se jednostavno ponavljati postupak što je bitno kako bi se uz malen broj iteracija i utrošenog vremena došlo do zadovoljavajućih rezultata. Podatke je moguće spremirati kako bi se analizirali različiti parametri filtra ili možda primijetile dodatne anomalije koje se mogu pojaviti.

Komunikacija se uspostavlja pomoću serijske veze pa je tako prije početka potrebno odabrati port na koji je spojen mikroupravljač i brzinu prijenosa odnosno *baud* rate. Kada je komunikacija uspješno uspostavljena treba se odrediti trajanje prikupljanja podataka i te se informacije šalju prema mikroupravljaču. Nakon uspješnog primanja informacije mikroupravljač počinje prikupljati podatke slati ih na računalo. Kada se prikupi potreban broj podataka, oni se prikazuju u vremenskoj i frekvencijskoj domeni kako bi točan spektar frekvencija bio vidljiv. Analizom tih podataka procjenjuje se granična frekvencija propusnog spektra i spektar kojeg je potrebno eliminirati. Ova dva parametra su potrebna jer nije moguće potpuno ni idealno eliminirati sve frekvencije šuma. Nakon upisivanja svih potrebnih karakterističnih vrijednosti filtra, prikazat će se filtriran signal u vremenskoj i frekvencijskoj domeni kojeg se može usporediti sa izvornim signalom i procijeniti efikasnost filtra.



Slika 26. Idealni i realni filter [6]

MATLAB će proračunavati filter s beskonačnim impulsnim odzivom koji je izabran za ovu svrhu iz razloga što zahtjeva manje procesorske snage mikroupravljača za proračun izlazne vrijednosti pa se time minimizira kašnjenje između uzoraka.

```

%%%%%%%% Clear workspace and comand line
clear all;
clc;
%%%%%%%% Define global variables
portName = "COM7";
baudRate = uint32(115200);
sampleRate = double(80); %Hz

%%%%%%%% Preform initial steps and display script info
ret = initCmd(portName, baudRate);
%%%%%%%% Initiate serial communication
hSerial = serialport(portName, double(baudRate));
%%%%%%%% Define time to sample
secToSample = input(compose(...
                    "Define seconds to sample 1-20 (default: " ...
                    + secToSample + "): ");
%%%%%%%% Send settings to MCU
writeline(hSerial, compose("s"+ secToSample));
%%%%%%%% Prepare data buffer
dataSize = secToSample * sampleRate;
dataBuffer = double(zeros(1, dataSize));
%%%%%%%% Fill buffer with measurement data
for i = 1:dataSize
    dataBuffer(i) = str2double(readline(hSerial));
end
%%%%%%%% Plot the result
plotData(dataBuffer, sampleRate);
%%%%%%%% Enter passband frequency
passBandFreq = input("Enter passband frequency: ");
passBandFreq = passBandFreq / (sampleRate /2);
%%%%%%%% Enter stopband frequency
stopBandFreq = input("Enter stopband frequency: ");
stopBandFreq = stopBandFreq / (sampleRate /2);
%%%%%%%% Calculate filter coefficients
[order, cutoffFreq] = buttord(passBandFreq, stopBandFreq, 1, 20);
[bCoeff, aCoeff] = butter(order, cutoffFreq, "low");
filteredData = filtfilt(bCoeff, aCoeff, dataBuffer);
%%%%%%%% Plot result
plotData(filteredData, sampleRate);

```

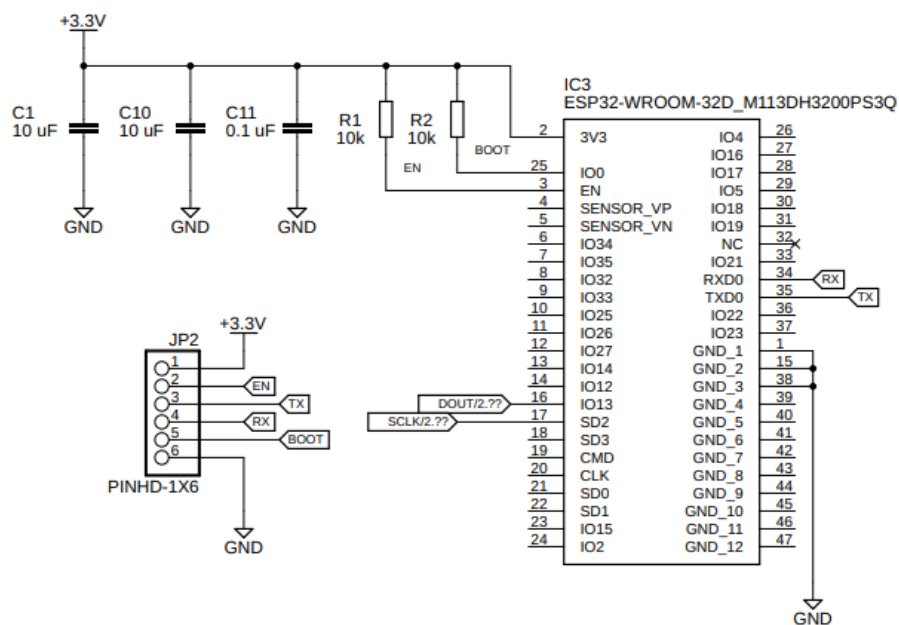
Slika 27. Isječak MATLAB skripte [6]

## 7.2. Hardversko sučelje

Schema sklopovlja oko mikroupravljača prikazana je na slici 28. Može se primijetiti da nije puno komponenti potrebno za njegov rad, međutim treba se uzeti u obzir tablicu 1. Kako bi se upravljački uređaj pravilno pokretao potrebno je osigurati da logičko stanje nožica IO0 i IO2 bude visoko. Ako to nije osigurano, pravilno pokretanje nije garantirano. Dodatno uz osiguravanje stanja nožica preporučeno je dodati filtarski kondenzator za visoko frekventne šumove, kao i kondenzatore većeg kapaciteta koji će osigurati pravilno napajanje i u slučaju kada mikroupravljač zahtjeva veću jakost struje. Naime, ESP32 prema dokumentaciji može zahtijevati jakosti struje u rasponu od 80 mA do 470 mA u normalnoj operaciji što je značajan skok, te ako ne osigura određenu akumulaciju energije može se dogoditi da se aktivira unutrašnja zaštita od niskog napona te da se upravljački uređaj ponovno pokrene.

Tablica 1. Stanja izlaza za podizanje [7]

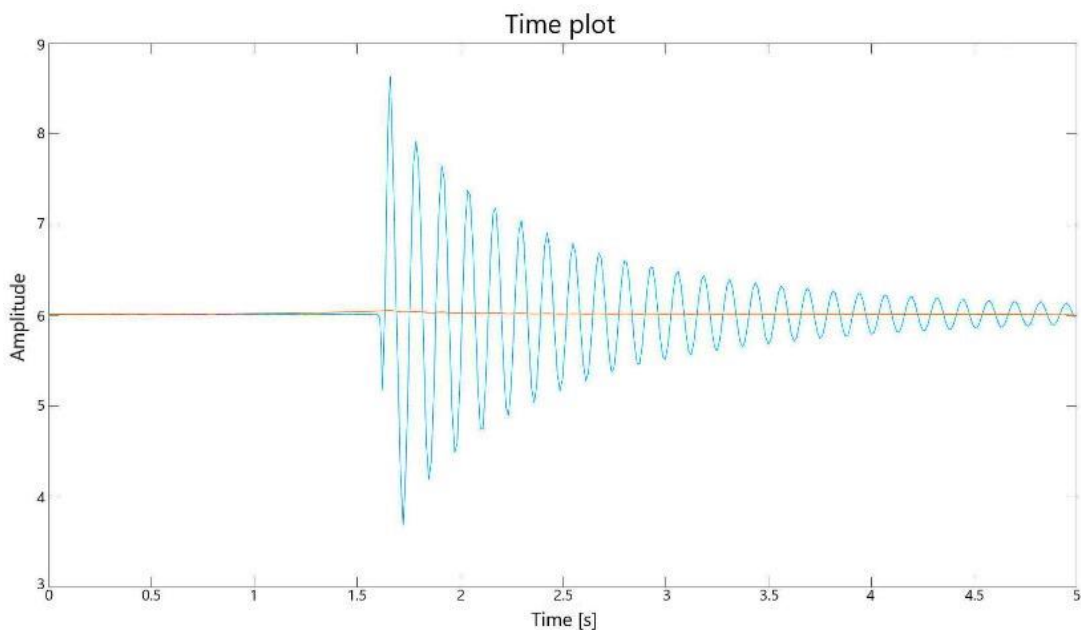
Način podizanja			
Nožica	Zadano stanje	SPI podizanje	Primanje programa
GPIO0	Visoko	Visoko	Visoko
GPIO2	Nisko	Nije bitno	Nisko



Slika 28. ESP32 shema [6]

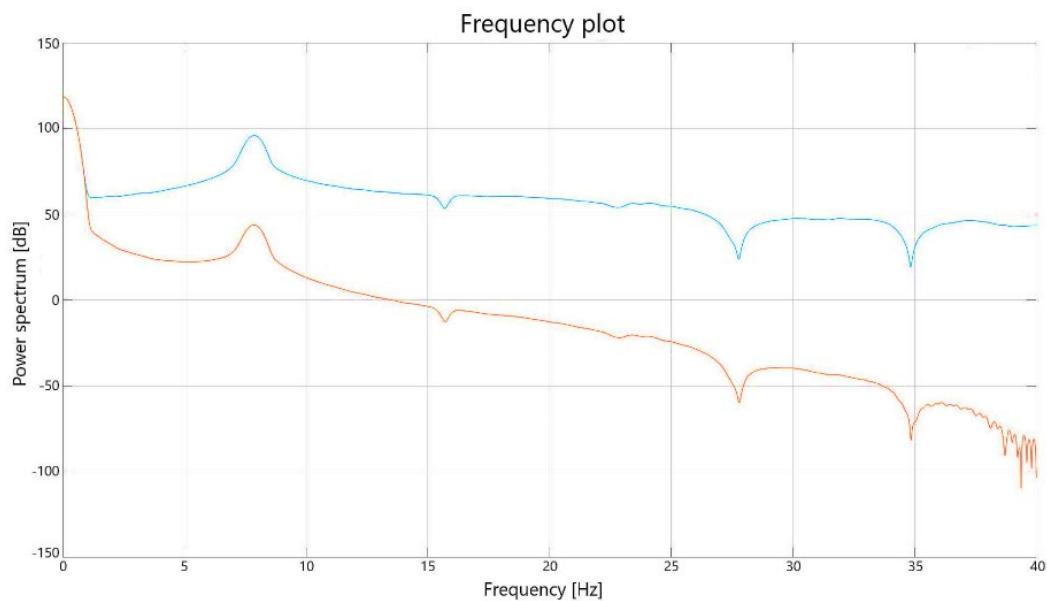


### 7.3. Evaluacija



Slika 30. Podatci u vremenskoj domeni [6]

Plava funkcija na slici 30 predstavlja ne filtrirane podatke prikupljene sa senzora. Titraj se ponaša kao sustav drugog reda, što je realan primjer iz prakse. U industrijskim okruženjima ovakva vibracija nerijetko stvara problem kod senzorskih sklopova u elektromehaničkim sustavima.



Slika 31 Podatci u frekvencijskoj domeni [6]



Kada se vibraciju promatra u frekvencijskoj domeni, vidljivo je da se dominantna frekvencija nalazi u spektru između 5 Hz i 10 Hz. Kao optimalna vrijednost za krajnju propusnu frekvenciju izabrana je vrijednost od 4 Hz. Sa svim uvrštenim podacima u MATLAB skriptu dobivaju se sljedeći koeficijenti:

Tablica 2. Koeficijenti filtra [6]

a	b
1	0.0157
-0.9687	0.0157

Uvrštavanjem dobivenog reda filtra od  $Q = P = 1$  i koeficijenata u formulu može se napisati:

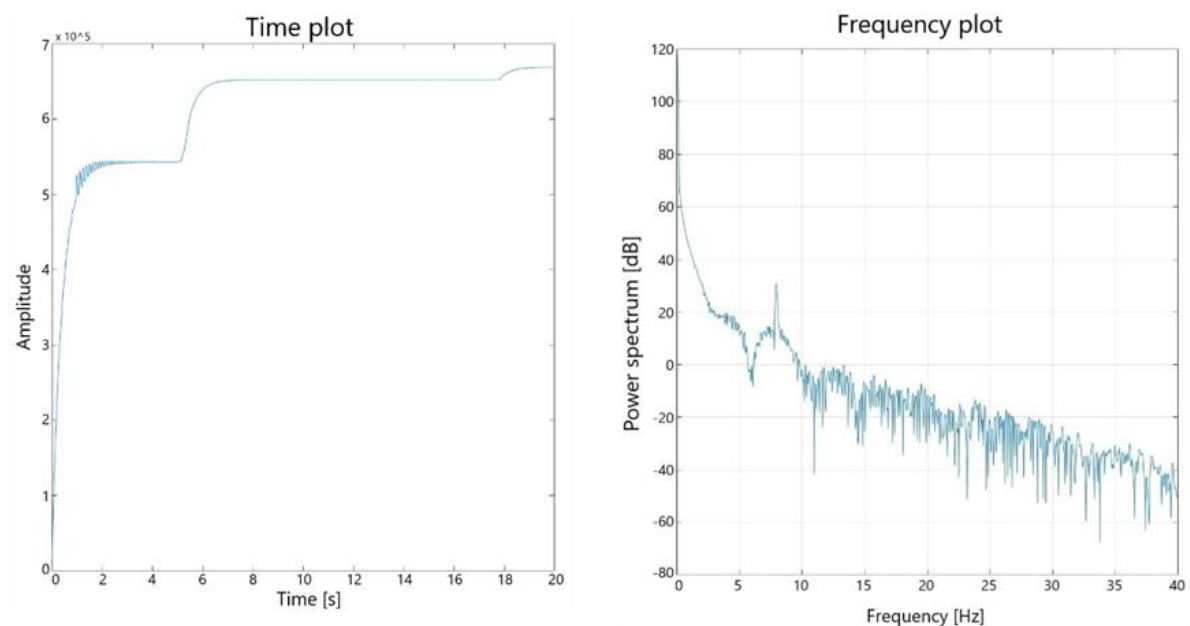
$$y[n] = 0.0157 \cdot (x[n] + x[n - 1]) + 0.9687 \cdot y[n - 1] \quad (23)$$

Te funkciju implementirati u C jeziku za mikroupravljač:

```
// IIR filter
// y[n] = b * (x[n] + x[n - 1]) + a * y[n - 1]
// x - input
// y - output
float a = 0.9687;
float b = 0.0157;
int32_t DoIIR(int32_t input)
{
    static float prevOut = 0;
    static float prevIn = 0;
    prevOut = (b * ((float) input + prevIn) + a * prevOut);
    prevIn = (float) input;
    return (int32_t)prevOut;
}
```

Slika 32. Implementacija filtra u C jeziku [6]

Nakon implementacije izračunatog filtra i ponovnog prikupljanja podataka dobiva se sljedeći odziv:



Slika 33. Filtrirani podatci [6]

Slika 33 prikazuje signal nakon implementacija filtra i ponovnog prikupljanja podataka u vremenskoj i frekvencijskoj domeni. U prvom dijelu se može primijetiti mala oscilacija, što je uzrokovano nedovoljnom količinom podataka koje je filter obradio. Većom količinom podataka filter poprima optimalnije vrijednosti i samim time konvergira prema boljim rezultatima. Pa se tako može primijetiti da s većim vremenskim odmakom ne dolazi više do oscilacija te da očitana vrijednost stabilno raste i poprima konstantne vrijednosti.

## 8. Zaključak

Na kraju valja naglasiti da je u ovdje samo prikazana osnova teorije o obradi signala. Tema je dosta opsežna i mnogostruko matematički kompleksnija nego što je opisano. Cilj ovog rada je provesti čitatelja kroz osnovne pojmove i postupke koji su potrebni da se pristupi rješavanju problema na postojećem sustavu.

Zbog dostupnosti širokog spektra alata za analizu praktično je potrebno raditi iznimno malo proračuna, ali je potrebno iznimno razumijevanje podataka. Nakon analize postupak se svodi na iteraciju i daljnju optimizaciju.

Praktičnim primjerom pokazano je kako filtracija može unaprijediti i poboljšati performanse sustava, međutim treba obratiti pažnju i na negativne učinke koji se time unose u sustav. Nepravilnim dizajnom i primjenom filtra moguće je sustav učiniti iznimno nestabilnim tako da je potrebno temeljito provesti testiranja. Također, svaki filter unosi kašnjenje pa treba posebno voditi računa o tome da li sustav ovisi sinkronizaciji i vremenskoj točnosti dostupnih podataka.

Digitalna obrada signala kao i slika, koje se praktično mogu promatrati kao signal, danas je nezamjenjiv dio tehničke znanosti. U vrijeme kad je brzina informacije bitan faktor i sve više prisutnih internetskih usluga, efikasno obraditi informaciju u digitalnom obliku postaje sve prisutnija potreba. Također zbog povećanja popularnosti i dostupnosti pametnih senzorskih komponenti na jednostavan i efikasan način moguće je smanjiti cijenu proizvoda uz zanemarive gubitke u performansama.

## LITERATURA

- [1] Smith S.W., *The Scientist and Engineer's Guide to Digital Signal Processing*, California Technical Publishing, 1999
- [2] Matijević D., Poljak S., *Fourierov red i Fourierova transformacija*, Hrvatski matematički elektronički časopis, 2011
- [3] Espressif Systems, *ESP32 Series Datasheet*, Version 3.9, 2022
- [4] Espressif Systems, *ESP32 Hardware Design Guidelines*, Version 3.2, 2021
- [5] Punskeya E., *Design of FIR Filters*, University of Cambridge, 2020
- [6] Zovko P., Tomasović L., Kovačević T., *FMS: Fluid metering system based on ESP platform*, CIET, 2022
- [7] Zovko P., *ESP32 tiskana ploča za razvoj digitalnog filtra*, Izvještaj specijalističke prakse, lipanj 2022.
- [8] National Instruments, *Measuring strain with strain gages*, <https://www.ni.com/en-us/innovations/white-papers/07/measuring-strain-with-strain-gages.html>
- [9] Šolić P., *Teorija informacija – 13 – Analogno-digitalna pretvorba*, FESB
- [10] Zavod za elektroakustiku, *Prijenos zvuka – Analogno digitalna pretvorba*, FER
- [11] AVIA Semiconductor, *24-Bit Analog-to-Digital Converter (ADC) for Weigh Scales*

## POPIS SLIKA

Slika 1. Diskretna funkcija .....	4
Slika 2. Aproksimacija funkcije Fourierovim redom .....	6
Slika 3. Blok shema filtracije [1].....	8
Slika 4. Prikaz vremenski neovisnog sustava [1] .....	9
Slika 5. Frekvencijski spektar funkcije .....	9
Slika 6. Frekvencijski odziv filtra .....	10
Slika 7. Blok shema FIR filtra.....	11
Slika 8. Sinc funkcija [2] .....	12
Slika 9. Hamming prozor [1].....	12
Slika 10. Blok shema IIR filtra [6] .....	13
Slika 11. Odziv Butterworth IIR filtra s različitim brojem povratnih veza [6] .....	14
Slika 12. Impulsni odziv RC kruga .....	14
Slika 13. Blok shema A/D pretvarača [9].....	15
Slika 14. Uzorkovanje [10].....	15
Slika 15. Kvantizacija.....	16
Slika 16. Tenzometar [6] .....	17
Slika 17. Spoj tenzometra u Whietstone-ov most [6].....	18
Slika 18. HX711 blok shema [11] .....	19
Slika 19. Funkcijski dijagram ESP32 [3] .....	20
Slika 20. ESP32 modul [3] .....	20
Slika 21. Blok dijagram upravljača [6].....	21
Slika 22. HX711 upravljač [6].....	22
Slika 23. Funkcionalnost HX711 [6].....	22
Slika 24. Dijagram toka programa za prikupljanje podataka [6].....	23
Slika 25. Dijagram toka MATLAB skripte [6].....	24
Slika 26. Idealni i realni filter [6] .....	24
Slika 27. Isječak MATLAB skripte [6] .....	25
Slika 28. ESP32 shema [6] .....	26
Slika 29. HX711 shema [6] .....	27
Slika 30. Podatci u vremenskoj domeni [6].....	28
Slika 31 Podatci u frekvencijskoj domeni [6] .....	28
Slika 32. Implementacija filtra u C jeziku [6] .....	29
Slika 33. Filtrirani podatci [6] .....	30

## **POPIS TABLICA**

Tablica 1. Stanja izlaza za podizanje [7] .....	26
Tablica 2. Koeficijenti filtra [6].....	29