**DSP-Project**

**Rom Hirsch Id: 313288763, Yrom Swisa Id: 203675814**

1. גרף של אות הכניסה בזמן:



גרף 1 - input signal vs. time

נמצא את התדר לפי הנקודות שסימנו כלומר:

האמפליטודה לפי הגרף :

1. ספקטרום אות הכניסה:

גרף 2 - spectrum of the input signal.



**ספקטרום האות :**

גרף 3 -absolute spectrum of the input signal



תדר האות:

תדר הרעש – כפי שחישבנו בסעיף 1:

1. תיכנון מסנן מסוג notch מסדר שני:

בתכנון נדרוש להוסיף אפסים בתדר הרעש כדי לאפסו, ולהוסיף קטבים במרחק של 0.9 לאותם אפסים לפי הנתון.

נגדיר את האפסים וקטבים לפי הנוסחאות הבאות:

𝛺c - תדר cut off

a = 0.9

1. Pole zero diagram :

**

גרף 4 - pole zero diagram

1. הגרף של התגובה לתדר של הפילטר:

גרף 5 - filter vs. normalized frequency form 0 to 2pi [rad/smp]

*גרף ספקטרום המסנן ביחס לתדר:*

**

גרף 6 - filter vs. frequency

ניתן לראות שהמסנן לא יעביר את תדר הרעש כפי שתיכננו.

1. המשוואה הדפראנציאלית של המערכת:

*נמיר את האות לזמן:*

*מערכת סיבתית כמו זו אינה תלויה בערכי העתיד לכן נבצע הזזה בזמן כלומר נציב*

*נקבל:*

*לכן המשוואה* הדפראנציאלית של המערכת שווה:

1. מימשנו את המערכת הדפראנציאלית בקוד matlab.
2. בוצע בקוד matlab בcell של question i.
3. גרף output signal vs. time.

**

1. תדר ואמפליטודה של האות



תדר האות לאחר הוספת הפילטר הוא :

*בנוסף ניתן לראות את התדר גם בעזרת הספקטרום להלן גרף של הספקטרום:*

**

ניתן לראות את התדר ב 250hz.

ספקטרום האות בdb :



לסיכום ניתן לראות שלאחר העברת האות דרך ה- notch filter כבר אין לנו את הרעש שהיה ב 50hz ועכשיו ניתן לראות את האות המתוכנן ב 250hz. בזמן ניתן לראות בתחילה תופעת מעבר של המסנן ולאחר מכן את האות הנדרש.

**קוד מטלב:**

% Id: 313288763,203675814

clear all ;

clc;

close all;

clear figure;

% init vars

data = load("input\_signal.mat").s;

N = length(data) ;% samples number

fs = 5e3 ;% sampled rate

Ts = 1/fs;

T0 = N\*Ts ;% total sampling time

df = 1/T0; % resolution we can get in the spectrum figure()

t = (0:length(data)-1).\*Ts; % time domain

%% Question a: plot Input signal time domain

figure()

plot(t,data);

title("Input signal time vs. time")

xlabel("Time[s]")

ylabel("Amplitude")

%% Question b: plot Input signal spectrum frequency Domain

spectrum = fft(data,N)/(N\*0.5);

freq\_axis = 0:length(spectrum)-1; % calc frequency array

freq\_axis = freq\_axis .\*df; % calc frequency array

omega = 2.\* pi .\*(freq\_axis/fs); % Omega = 2pi\*(f/fs)

% plot Input signal spectrum db

figure()

plot(freq\_axis,db(abs(spectrum)));

title("spectrum of the input signal db")

xlabel("frequency[Hz]")

ylabel("amplitude[DB]")

% plot Input signal spectrum absolute

figure()

plot(freq\_axis,abs(spectrum));

title("spectrum of the input signal absolute")

xlabel("frequency[Hz]")

ylabel("amplitude")

%% Question d: notch filter design

noiseFreq = 50; % noise frequince we found

N = 1000; % samples number

a = 0.9; % notch magnitude

z = exp(1j.\*omega); %z Transpose param z=e^-jomega

omegaCutOff = 2 \* pi\*(noiseFreq/fs); %calc frequency cut off

%Calc zeros and poles

zero1 = exp(1j\*omegaCutOff);

zero2 = exp(-1j\*omegaCutOff);

pole1 = a \* exp(1j\*omegaCutOff);

pole2 = a \* exp(-1j\*omegaCutOff);

notchFilter = ((z - zero1) .\* (z-zero2) )./((z-pole1) .\* (z-pole2)); % notch filter equation

%% Question e: plot zero and pole map

zplane([zero1;zero2],[pole1;pole2]);% plot zero and pole map

%% Question f: plot notch filter spectrum

% plot notch filter spectrum between 0 - 2pi

figure()

plot(omega,abs(notchFilter))

title("plot notch filter spectrum rad/samp")

xlabel("rad/samp")

ylabel("Magnitude")

% plot notch filter spectrum in the frequncy domain

figure()

plot(freq\_axis,abs(notchFilter))

title("plot notch filter spectrum in the frequncy")

xlabel("Frequency[Hz]")

ylabel("Magnitude")

%% Question h and i: y[n] difference calc and pass the input through the filter.

y = zeros (1,N); %init y[n]

x=data;

%calc difference

for n = 3 : N

y(n) = x(n)- x(n-1).\*(zero1+zero2)+x(n-2).\*(zero1.\*zero2)+y(n-1).\*(pole1+pole2)-y(n-2).\*(pole1.\*pole2);

end

Y = fft(y)/(N\*0.5);

%% Question j and k: plot y[n] output signal vs. time.

plot(t,y)

title("plot y[n] output signal vs. time")

xlabel("Time[s]")

ylabel("Amplitude")

% plot y[n] spectrum (Y[n])

figure()

plot(freq\_axis,abs(Y))

title("plot y[n] absulote spectrum")

xlabel("Frequency[Hz]")

ylabel("Magnitude")

% plot y[n] spectrum db (Y[n])

figure()

plot(freq\_axis,db(abs(Y)))

title("plot y[n] db spectrum")

xlabel("Frequency[Hz]")

ylabel("Magnitude [db]")