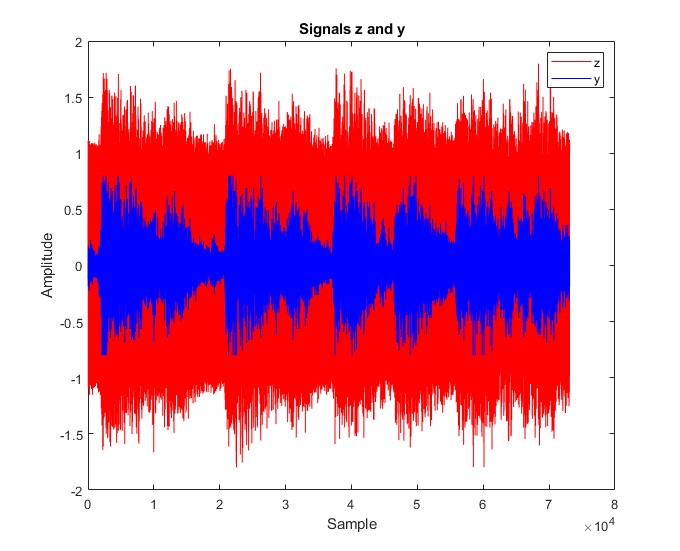
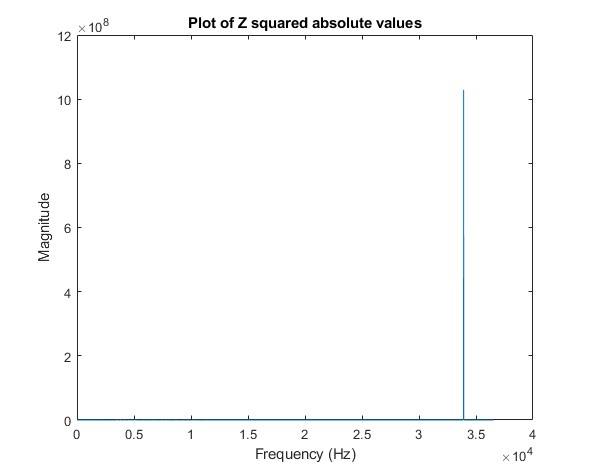
**תרגיל מחשב**

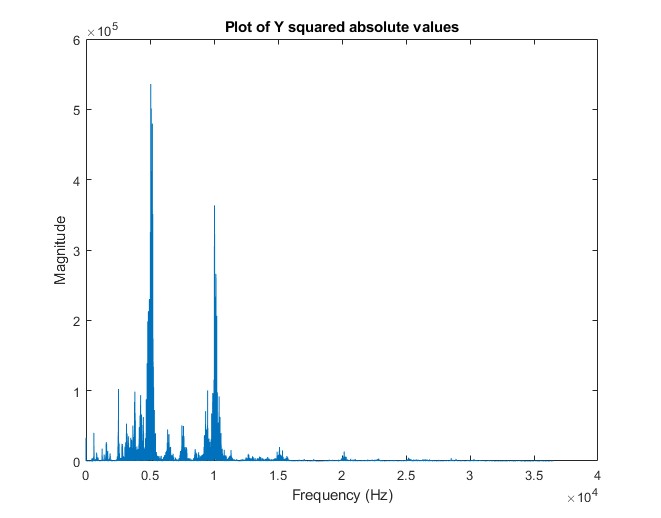
# חלק ב'

מטרת התרגיל: לתכנן מסנן ספרתי ע״י תכנון מסנן Butterworth אנלוגי והמרתו למסנן ספרתי באמצעות התמרה בי לינארית.

1. נתונים אותות ו- הדגומים בתדר (יש לפתוח את הקובץ sig\_2.mat במטלב או כל תוכנה אחרת).
   1. שרטט/י את הערך המוחלט בריבוע של התמרת פורייה האנלוגית של כל אחד מהאותות באמצעות מחשב ע״י שימוש ב-DFT.







* 1. מה ההבדל בין שני האותות?

תשובה:

**ניתוח בתחום הזמן:**

- האות 'y' (כחול): מופיע יותר במרכז, סביב הציר האופקי עם צורת גל צפופה.

- האות 'z' (אדום): בעל אמפליטודה גבוהה יותר, ונראה מפוזר יותר.

**ניתוח בתחום התדר:**

התמרת של פורייה של האות 'y':

- ספקטרום התדרים מציג מספר פיקים בולטים.

- רכיבים משמעותיים נצפים סביב 0.5 ו-1 הרץ, עם פיקים נמוכים יותר בתדרים אחרים.

התמרת פורייה של האות 'z':

- ספקטרום התדרים מראה פיק דומיננטי סביב 3.5 הרץ, המצביע על רכיב תדר חזק בערך זה.

- בהשוואה ל-'y', ל-'z' יש פחות רכיבי תדר, כאשר פיק עיקרי אחד שולט בספקטרום.

**סיכום:**

- האות y נראה עשיר יותר, עם מספר פיקים משמעותיים בהתמרת הפורייה שלו, מה שמרמז על כך שהוא מורכב ממספר רב של הרמוניות.

- לעומת זאת, האות z מאופיין בהרמוניה יחידה בסביבות 3.5 הרץ, מה שהופך אותו לפחות מורכב מבחינת תוכן התדר שלו.

* 1. האזן/י לאותות ו- . אם עובדים במטלב, אפשר להשתמש בשורות הבאות

playerObj = audioplayer(y,Fs);

start = 1;

stop = playerObj.SampleRate \* 3;

play(playerObj,[start,stop]);

* 1. תאר/י את ההבדל בין האותות.

תשובה:

האות y הוא אות שמע באורך 3 שניות בו נשמעת המילה "הללויה" בניגון, כאשר אין רעש רקע הנשמע לאוזן.

לעומת זאת, האות z זהה לאות y מלבד צפצוף נוסף אחיד לאורך כל האות.

הבדלים אלו ניכרים הן בשמיעת האותות בתחום הזמן, והן בניתוח תחום התדר, כאשר הצפצוף הנוסף ב-'z' מתבטא כרכיב תדר דומיננטי שאינו קיים ב-'y'.

מעוניינים לסנן את אחד מהאותות ( או ) כך שהאותות ישמעו דומה זה לזה ככל שניתן. את זאת יש לעשות ע״י מסנן ספרתי השקול למסנן אנלוגי מעביר נמוכים (low-pass) בעל המאפיינים הבאים:

1. מה הם המאפיינים של המסנן הספרתי (תדר מעבר, עצירה, ניחות וגליות) כך שהמערכת האנלוגית השקולה תעמוד בדרישות המפורטות מעלה?

תשובה:

הדרישות שניתנו עבור המסנן האנלוגי הם:

- גליות בתחום ההעברה (ripple) : פחות מ-

- ניחות בתחום הקיטעון :

- תדר תחום ההעברה :

- תדר תחום ההעברה :

כדי לתכנן מסנן דיגיטלי עם דרישות אלו, עלינו להמיר את התדרים האנלוגיים לתדרים דיגיטליים ע"י ההתמרה הבי-לינארית. התדרים הדיגיטליים יהיו תלויים בתדר הדגימה :

1. תדרי תחום המעבר ותדרי תחום הקיטעון בתחום הדיגיטלי:

שימוש בהתמרה הבי ליניארית:

1. גליות תחום ההעברה וגליות תחום הקיטעון :

גליות תחום המעבר והנחתת תחום הקיטעון במונחים של סקאלה לינארי:

מעוניינים לתכנן מסנן ספרתי IIR בעל פונקציית תמסורת באמצעות המרה של מסנן Butterworth ע״י ההתמרה הבי-לינארית:

1. חשב/י תדרים אנלוגיים מתאימים למסנן Butterworth  המבוקש. האם תדרים אלה צריכים להיות זהים לתדרים האנלוגיים הנדרשים ל- ?

תשובה:

1. התאמת התדרים:

כדי להבטיח שההתמרה הבי-ליניארית משקפת בצורה מדויקת את התדרים האנלוגיים הרצויים, נתאים את תדרי תחום ההעברה ותחום הקיטעון:

1. תדרים אנלוגיים למסנן Butterworth:

תדרים מותאמים אלו ישמשו לתכנון מסנן Butterworth (אנלוגי), מכיוון שהם משקפים את התדרים שיענו על הדרישות הרצויות לאחר ההתמרה הבי-לינארית.

1. תכנן/י מסנן אנלוגי מסוג Butterworth כתוב/י ביטוי כללי לאפסים של המסנן Butterworth ושרטט במחשב את מגניטודת תגובת התדר במחשב.

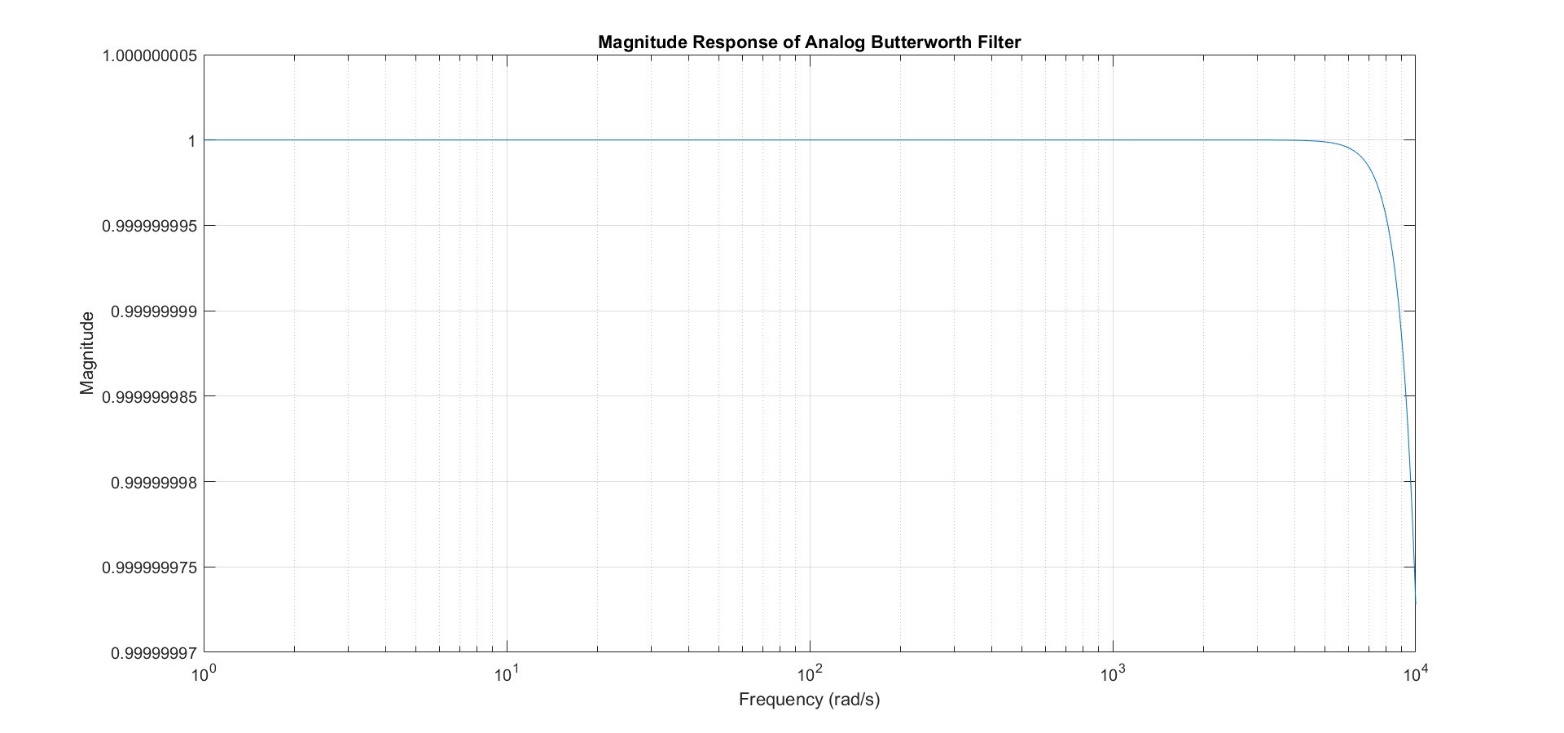
פתרון:

1. סדר המסנן:

סדר *n* של מסנן Butterworth ניתן לחשב באמצעות הנוסחה הבאה:

1. פונקציית התמסורת:

למסנן Butterworth אין אפסים ויש לו *n* קטבים הנמצאים על מעגל במישור השמאלי. הקטבים נתונים על ידי:

 כאשר הוא תדר הקיטעון.

A graph with a curve

Description automatically generated

1. שרטט/י את מגניטודת תגובת התדר של המסנן הספרתי
2. התמרה בי-לינארית

נפעיל את ההתמרה ההבי-לינארית כדי להמיר את מסנן Butterworth האנלוגי לתחום הדיגיטלי :

1. סקיצה של האמפליטודה:

% Step 1: Load the data from the file 'sig\_2.mat'

load('sig\_2.mat', 'Fs', 'z', 'y'); % Load sampling frequency, z, and y

% Step 2: Define filter specifications

A\_p = 5; % Passband attenuation in dB

A\_s = 20; % Stopband attenuation in dB

Omega\_p = 3600 \* 2 \* pi; % Passband frequency in rad/sec

Omega\_s = 3800 \* 2 \* pi; % Stopband frequency in rad/sec

% Pre-warped frequencies

F\_p = Omega\_p / (2 \* pi);

F\_s = Omega\_s / (2 \* pi);

% Pre-warp the frequencies

omega\_p = 2 \* Fs \* tan(Omega\_p / (2 \* Fs));

omega\_s = 2 \* Fs \* tan(Omega\_s / (2 \* Fs));

% Step 3: Calculate the order of the Butterworth filter

[n, Wn] = buttord(omega\_p, omega\_s, A\_p, A\_s, 's'); % Wn is the normalized cutoff frequency

% Step 4: Design the analog Butterworth filter

[b, a] = butter(n, Wn, 's'); % Wn is the normalized cutoff frequency

% Step 5: Convert to digital filter using bilinear transformation

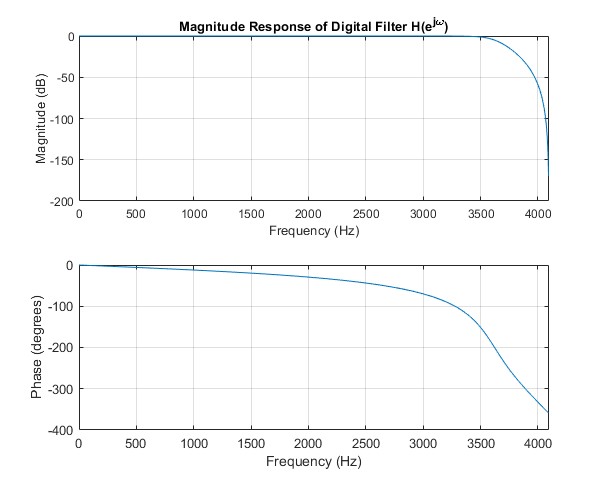
[bz, az] = bilinear(b, a, Fs);

% Step 6: Plot the magnitude response of the digital filter

figure;

freqz(bz, az, 1024, Fs);

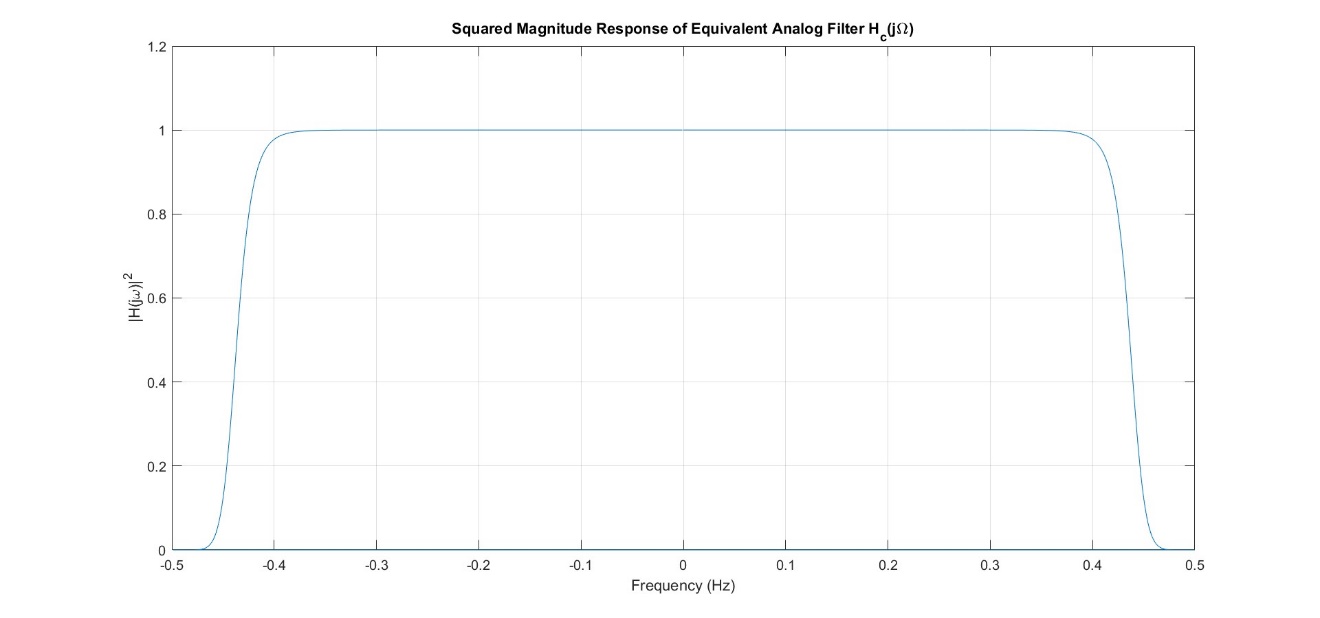
title('Magnitude Response of Digital Filter H(e^{j\omega})');



1. שרטט/י את תגובת התדר של המסנן הספרתי השקול .

פתרון:

1. כדי לשרטט את תגובת התדר של המסנן האנלוגי השקול , אנו צריכים להמיר את תגובת התדר הדיגיטלית חזרה לתחום האנלוגי. נעשה זאת באמצעות נוסחת ההמרה:

 נשרטט את באותו האופן בו שרטטנו את :

% Step 7: Plot the Frequency Response of the Equivalent Analog Filter

% Compute the frequency response

[H, w] = freqz(bz, az, 1024, 'whole');

% Adjust the frequency vector to include negative frequencies

w = w - 2\*pi\*(w > pi);

% Convert digital angular frequency to normalized angular frequency (omega/Fs)

omega\_normalized = w / Fs;

% Convert to Hz

F\_analog = omega\_normalized \* Fs / (2 \* pi);

% Compute the squared magnitude response

H\_squared = abs(H).^2;

% Plot the squared magnitude response

figure;

plot(F\_analog, H\_squared);

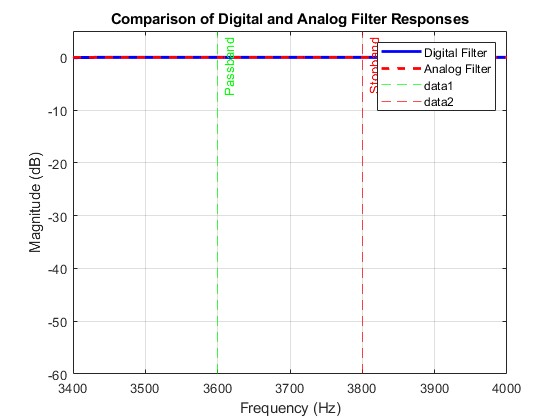
title('Squared Magnitude Response of Equivalent Analog Filter H\_c(j\Omega)');

xlabel('Frequency (Hz)');

ylabel('|H(j\omega)|^2');

grid on;

אכן ניתן לראות כי המסננים אכן דומים אחד לשני:



% Set up frequency range for detailed comparison

f = linspace(0, Fs/2, 1000); % Up to Nyquist frequency

w = 2\*pi\*f; % Angular frequency

% Digital filter response

[Hd, wd] = freqz(bz, az, w, Fs);

mag\_d = 20\*log10(abs(Hd));

% Analog filter response

[Ha, wa] = freqs(b, a, w);

mag\_a = 20\*log10(abs(Ha));

% Plotting comparison of Digital and Analog Filter Responses

figure;

plot(f, mag\_d, 'b', 'LineWidth', 2);

hold on;

plot(f, mag\_a, 'r--', 'LineWidth', 2);

grid on;

xlabel('Frequency (Hz)');

ylabel('Magnitude (dB)');

title('Comparison of Digital and Analog Filter Responses');

legend('Digital Filter', 'Analog Filter');

xlim([3400 4000]); % Focus on the region of interest

ylim([-60 5]); % Adjust as needed

xline(F\_p, 'g--', 'Passband');

xline(F\_s, 'r--', 'Stopband');

1. סנן את אחד מהאותות ( או ) כך שישמעו קרוב זה לזה ככל שניתן.

%% Section 5: Filter the Signal z and Play the Filtered Signal

% Filter the signal z using the designed digital filter

filtered\_z = filter(bz, az, z);

% Create an audioplayer object for the filtered z

playerObjFilteredZ = audioplayer(filtered\_z, Fs);

% Define start and stop samples for a 3-second playback

startZ = 1;

stopZ = playerObjFilteredZ.SampleRate \* 3;

% Play the filtered z signal for the first 3 seconds

play(playerObjFilteredZ, [startZ, stopZ]);

קיבלנו כאן סינון סביר:

% Compute SNR for the digital filter

% Original signal: z

% Filtered signal: filtered\_z

SNR\_digital = 10 \* log10(mean(z.^2) / mean((filtered\_z - z).^2));

fprintf('SNR of the digital filter (in dB): %.2f\n', SNR\_digital);

SNR of the digital filter (in dB): 0.27

אולם אם נשתמש לצורך העניין במסנן FIR, נקבל סינון טוב בהרבה:

%% perfect filtering (FIR)

% Define the parameters for FIR band-stop filter

N = 1000; % Filter length

n = -N:N; % Time index

B = pi/65; % Bandwidth of the notch

% Frequency to be removed (3800 Hz) converted to rad/s

w\_0 = 2\*pi\*3800 / Fs;

% Design the FIR filter

h\_1 = (2\*cos(w\_0\*n).\*sin(B\*n))./(pi\*n);

h\_1(N+1) = B/pi; % Correct the center value

% Filter the signal z using the FIR filter

z\_fir = z - conv(z, h\_1, 'same');

% Compute SNR for the FIR filter

% Original signal: z

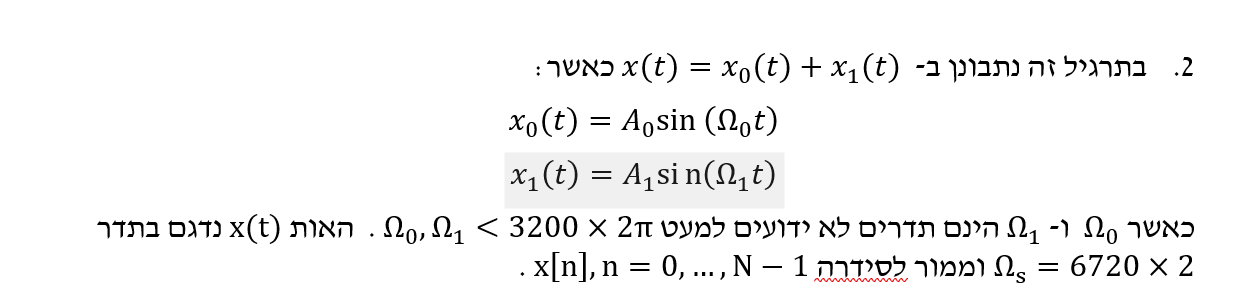
% Filtered signal: z\_fir

SNR\_FIR = 10 \* log10(mean(z.^2) / mean((z\_fir - z).^2));

fprintf('SNR of the FIR filter (in dB): %.2f\n', SNR\_FIR);

SNR of the FIR filter (in dB): 0.43



  
נתון לנו כי:

*אנחנו יודעים שבתדר אנחנו מבצע DFT על התמרת הפורייה בבדיד ונכפיל בחלון על מנת לקבל את הטווח הדרוש עבורנו-נזכור שסינוס בתדר(בערך מוחלט-מכיוון שנרצה את העצמה) הוא שתי דלתאות מוזזות וחלון בתדר הוא סינק-כלומר נקבל פעמיים-2 סינקים מוזזים:*

*בבדיד:*

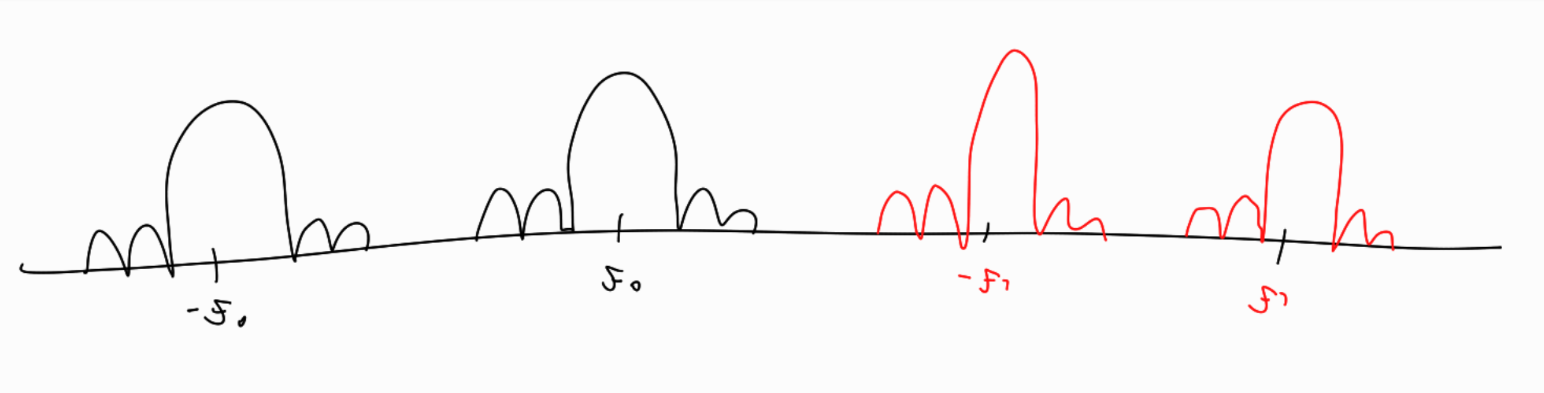
*והתמרת הפורייה תהיה*

ומכיוון שנכפיל בחלון-

=  
כאשר כאשר

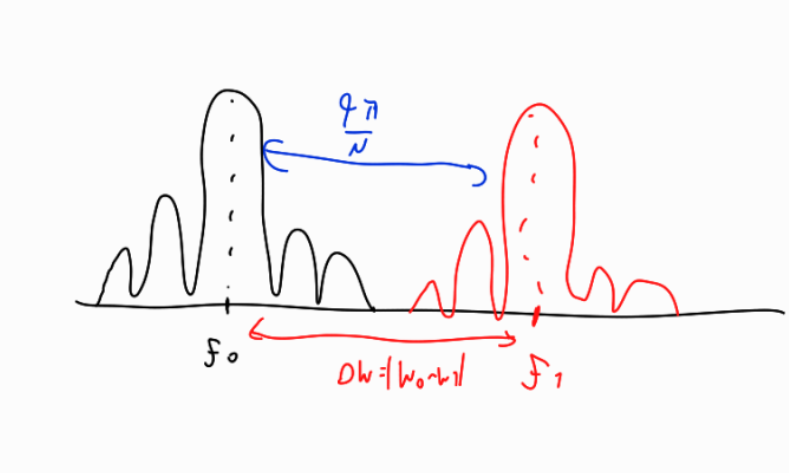
נשים לב שככל שהחלון שלנו יהיה גדול יותר הוא התוצאה תהיה יותר ויותר קרובה לסינק(נזכור שהאונה הראשית של סינק דועכת כמו *1/N ומכיוון שהתנאים אינם אידאלים והחלון הוא סופי-נקבל סינקים:*

כאשר מכיוון שנקח את העצמה-בעצם כשנבצע את הDFT נקבל את הערך המוחלט כלומר :



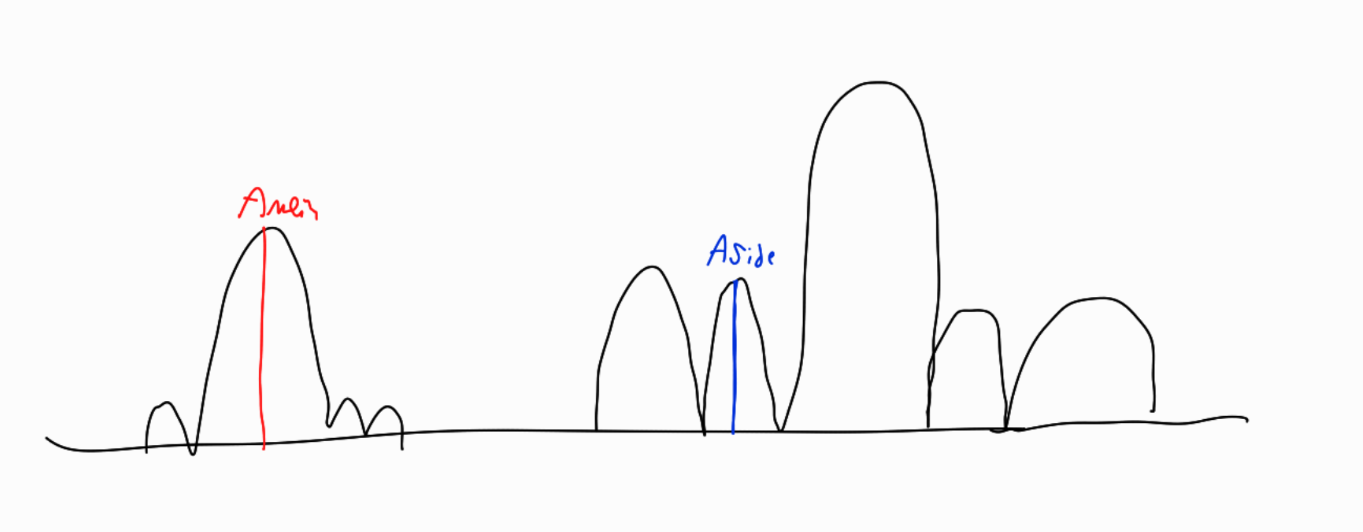
אם כן,על מנת לקבל הפרדה-נדרוש תחילה הפרדה בין האונות הראשיות כלומר: שהאונה הראשית של סינק אחד לא יעלה על האונה הראשית של סינק שני-כפי שלמדנו הדרישה היא

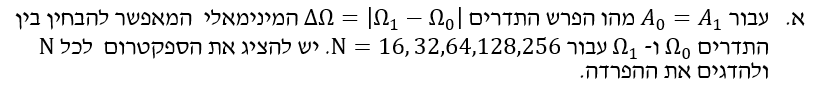
*כאשר הוא הרוחב של האונה הראשית*

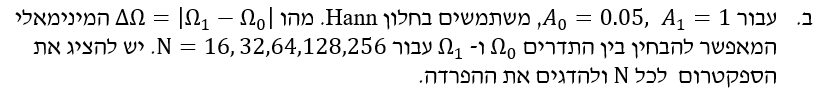
**

*וככל שהחלון יותר רחב האונות יותר צרות כלומר הדרישה על תהיה קטנה יותר*

*אם נרצה שהאונה הראשית של אחד לא תחתוך את האונה הראשית של אחר כלומר-אם נניח ש שהאונה המשנית של A1 לא תהיה גדולה יותר מהראשית של A0 כי לא נוכל להבדיל: כלומר נדרוש:*

**

**

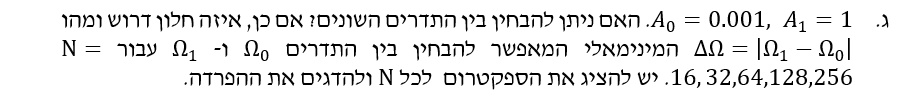
**

*בסעיף א-כש נוכל להשתמש בחלון רגיל ולקיים את הדרישה בקלות-בסעיף ב-הדרישה חמורה יותר:*

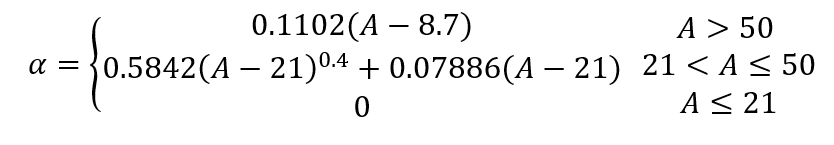
*שוב- נדרוש ואצלנו ולכן אם נשתמש בחלון hann (-32dB כפי שמופיע בטבלה) ונקבל את הדרוש כלומר אם נשתמש בחלון hann-נקיים את הדרישה עבור שני הסעיפים:*

*לכן נותר לנו רק לדרוש עבור האונות הראשית שלא יתנגשו:*

*נציב כל פעם מחדש עבור כל N ונקבל את הפרשי התדירויות::*

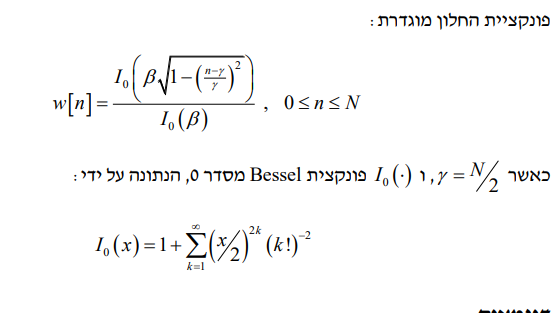
**

*כעת כלומר אף אחד מהחלונות שברשותנו לא עומדים בתנאי ונצטרך להשתמש בחלון kaiser:*

*חלון kaiser הוא סוג של מסנן fir שנבנה באופן מלאכותי על מנת לקיים בין היתר את הדרישות של הנחתה שלא יכולנו לקבל עם מסננים קלאסיים כמו hann שהשתמשנו בסעיפים קודמים:  
תחילה נמצא את התנאי על a: מכיוון שההנחה היא 60 כלומר A=60:  
*

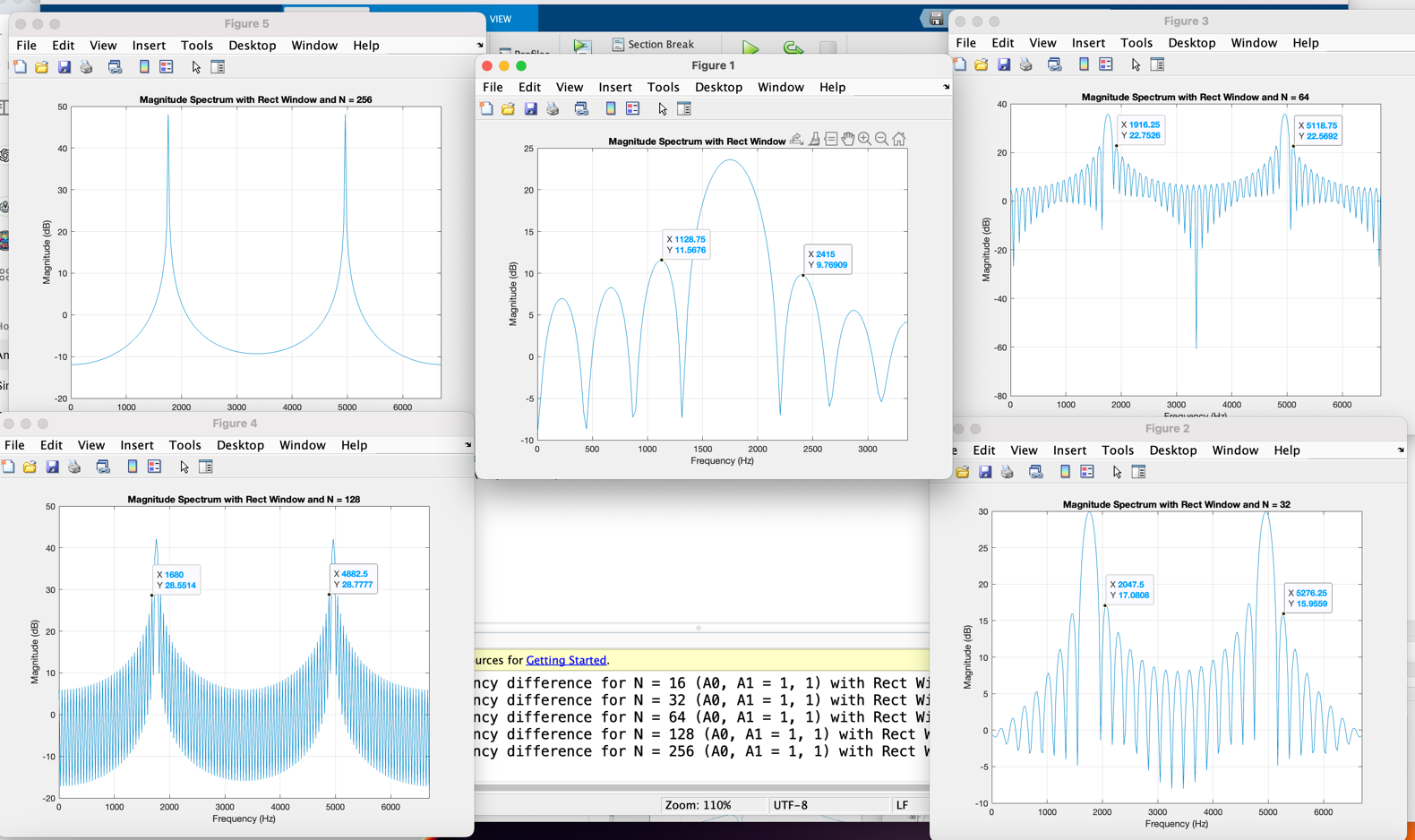
ולכן a=0.1102(A-8.7)=5.65

נמצא כעת את התנאי על : כפי שלמדנו

וכפי שלמדנו: 

ונציב...

*פתרון סעיף א:*



Minimum frequency difference for N = 16 (A0, A1 = 1, 1) with Rect Window is 0.125 Hz

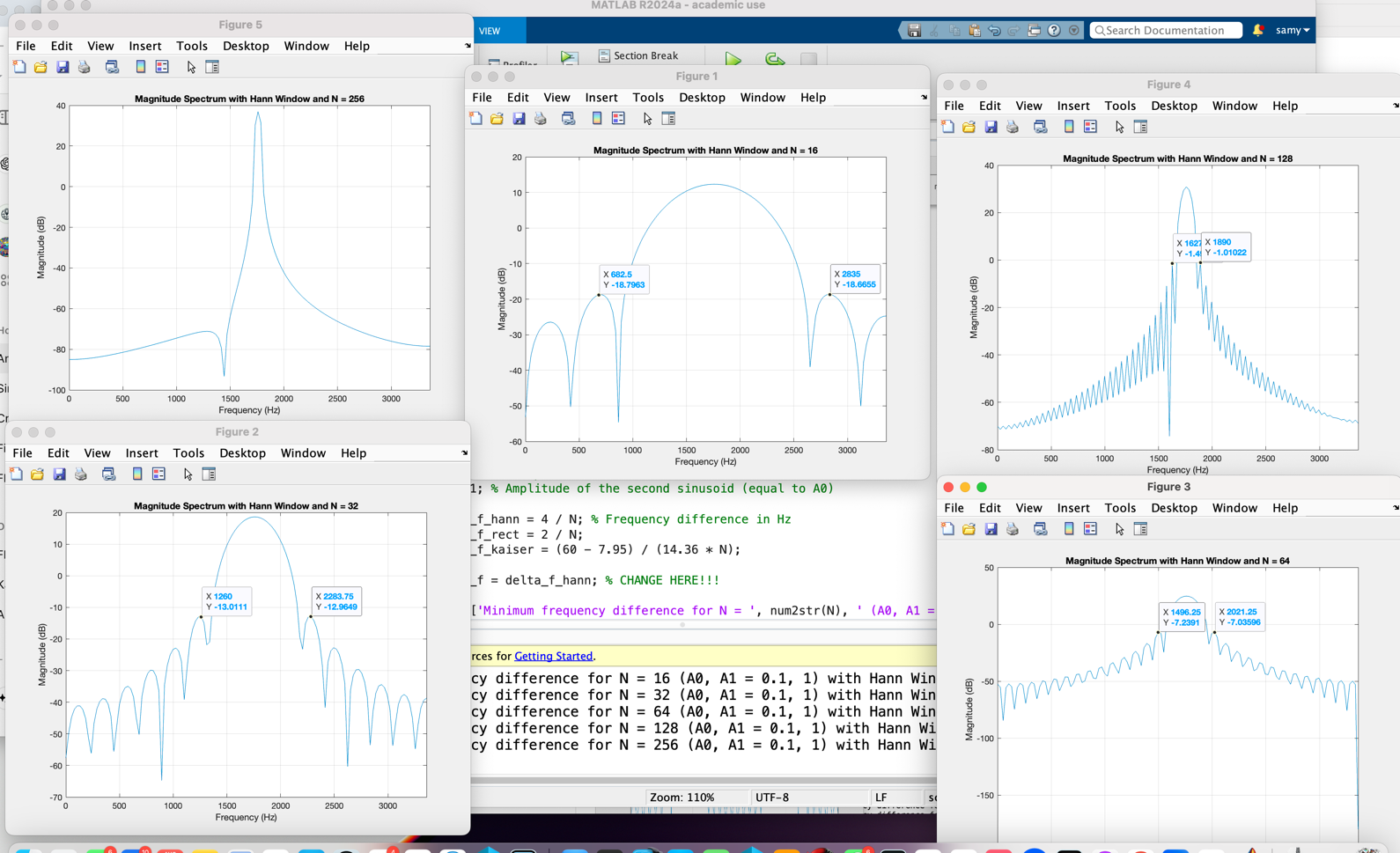
Minimum frequency difference for N = 32 (A0, A1 = 1, 1) with Rect Window is 0.0625 Hz

Minimum frequency difference for N = 64 (A0, A1 = 1, 1) with Rect Window is 0.03125 Hz

Minimum frequency difference for N = 128 (A0, A1 = 1, 1) with Rect Window is 0.015625 Hz

Minimum frequency difference for N = 256 (A0, A1 = 1, 1) with Rect Window is 0.0078125 Hz

*פתרון סעיף ב:*



Minimum frequency difference for N = 16 (A0, A1 = 0.05, 1) with Hann Window is 0.25 Hz

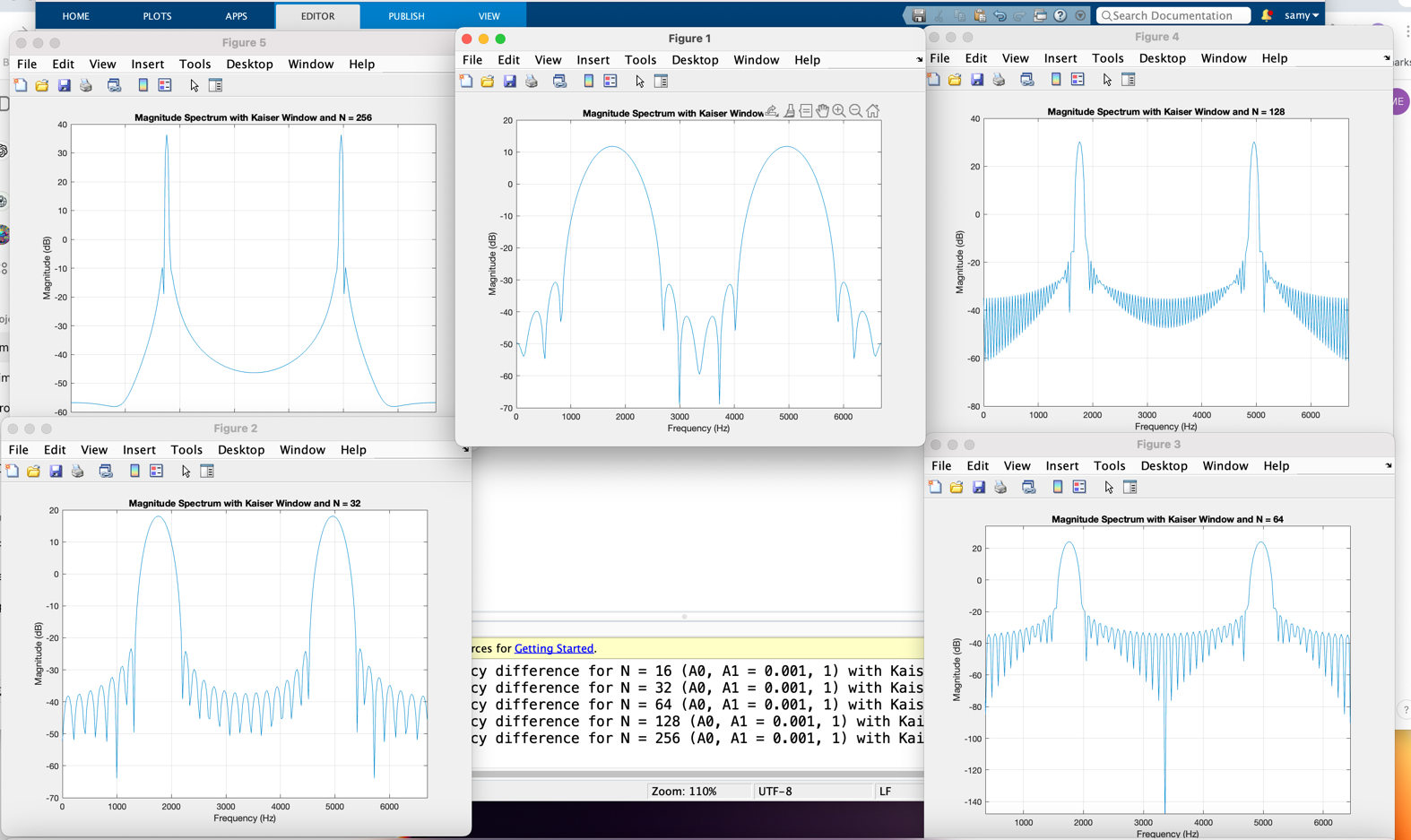
Minimum frequency difference for N = 32 (A0, A1 = 0.05, 1) with Hann Window is 0.125 Hz

Minimum frequency difference for N = 64 (A0, A1 = 0.05, 1) with Hann Window is 0.0625 Hz

Minimum frequency difference for N = 128 (A0, A1 = 0.05, 1) with Hann Window is 0.03125 Hz

Minimum frequency difference for N = 256 (A0, A1 = 0.05, 1) with Hann Window is 0.015625 Hz

*פתרון סעיף ג:*



Minimum frequency difference for N = 16 (A0, A1 = 0.001, 1) with Kaiser Window is 0.22654 Hz

Minimum frequency difference for N = 32 (A0, A1 = 0.001, 1) with Kaiser Window is 0.11327 Hz

Minimum frequency difference for N = 64 (A0, A1 = 0.001, 1) with Kaiser Window is 0.056635 Hz

Minimum frequency difference for N = 128 (A0, A1 = 0.001, 1) with Kaiser Window is 0.028318 Hz

Minimum frequency difference for N = 256 (A0, A1 = 0.001, 1) with Kaiser Window is 0.014159 Hz

As you can see on the graphs (I also added a measurement of it on some of the pictures), there is a peak which is not part of the signal X\_0 (is not symmetric with the other side of X\_0).  
This peak is X\_1 - Proving that it is recognizable from X\_0.

Matlab: :  
  
  
clc;

clear;

close all;

% Note that this script has the capacity to run for rect, hann and kaiser

% window. Just look for the comment CHANGE HERE and change accordingly.

% Define the different values of N

N\_values = [16, 32, 64, 128, 256];

name\_window = 'Rect Window'; % CHANGE HERE!!!

for N = N\_values

% Parameters

% N = 16; % Number of points in the window

fs = 6720; % Sampling frequency in Hz

beta = 5.65; % % beta parameter for the Kaiser window

% CHANGE HERE!!!

A0 = 1; % Amplitude of the first sinusoid

A1 = 1; % Amplitude of the second sinusoid (equal to A0)

delta\_f\_hann = 4 / N; % Frequency difference in Hz

delta\_f\_rect = 2 / N;

delta\_f\_kaiser = (60 - 7.95) / (14.36 \* N);

delta\_f = delta\_f\_rect; % CHANGE HERE!!!

disp(['Minimum frequency difference for N = ', num2str(N), ' (A0, A1 = ', num2str(A0), ', ', num2str(A1), ') with ', name\_window, ' is ', num2str(delta\_f), ' Hz']);

% Frequencies

f1 = 1600; % Frequency of the first sinusoid in Hz

f0 = f1 + delta\_f; % Frequency of the second sinusoid in Hz, ensuring |f0 - f1| = 2/N

% Time vector

t = (0:N-1)/fs;

% Signal components

x0 = A0 \* sin(2 \* pi \* f0 \* t);

x1 = A1 \* sin(2 \* pi \* f1 \* t);

% Combined signal

x = x0 + x1;

% Define the rectangular window

rect\_window = ones(N, 1);

% Apply the rectangular window to the signal

x\_windowed\_rect = x .\* rect\_window';

% Apply Hann window

hann\_window = 0.5 \* (1 - cos(2 \* pi \* (0:N-1)' / (N-1)));

x\_windowed\_hann = x .\* hann\_window';

% Generate the Kaiser window

kaiser\_window = kaiser(N, beta);

% Apply the Kaiser window to the signal

x\_windowed\_kaiser = x .\* kaiser\_window';

x\_windowed = x\_windowed\_rect; % CHANGE HERE!!!

N2 = 256; % Zero-padding length

x\_windowed\_padded = [x\_windowed, zeros(1, N2-N)];

% Compute the FFT

X = fft(x\_windowed\_padded, N2);

X = fftshift(X);

f = (0:N2-1)\*(fs/N2);

% Plot the magnitude spectrum

figure;

plot(f, 20\*log10(abs(X)));

xlabel('Frequency (Hz)');

ylabel('Magnitude (dB)');

title(['Magnitude Spectrum with ', name\_window, ' and N = ', num2str(N)]);

grid on;

% Zoom in on the frequencies of interest

xlim([0, fs/2]);

end;