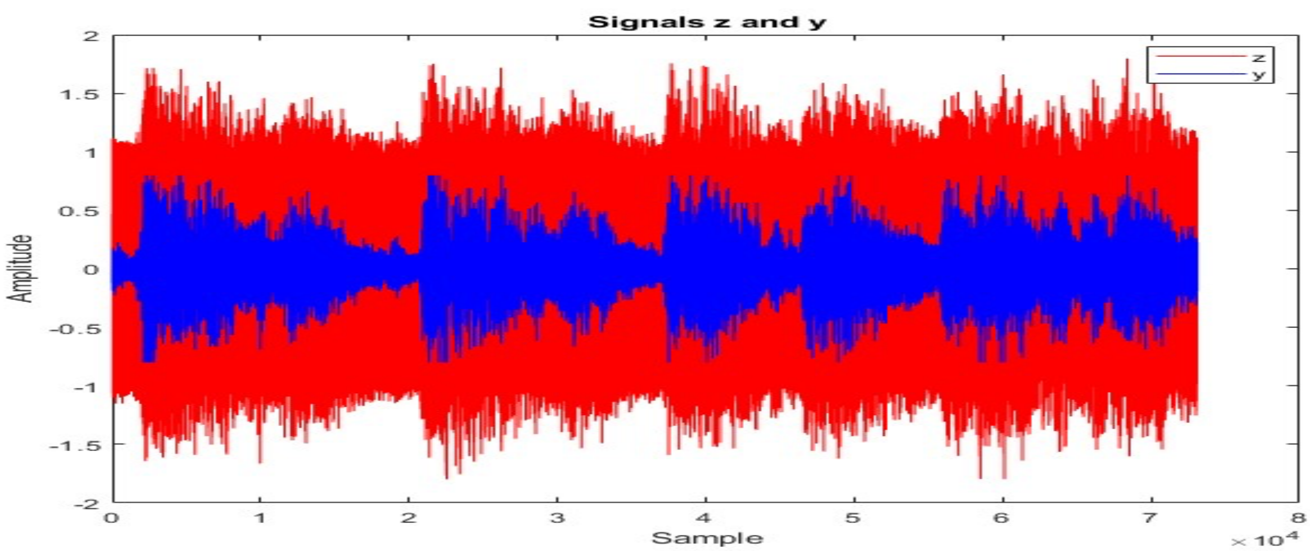
תרגיל מחשב

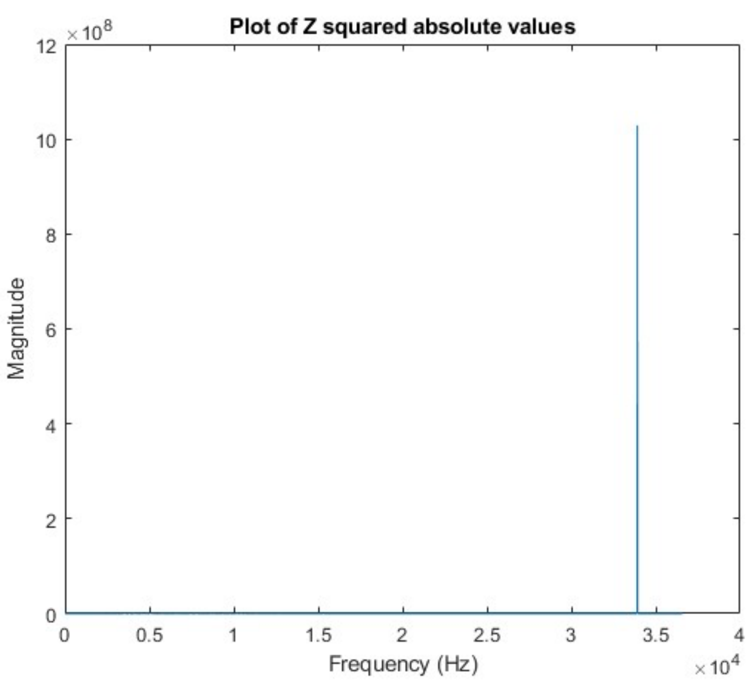
חלק ב׳

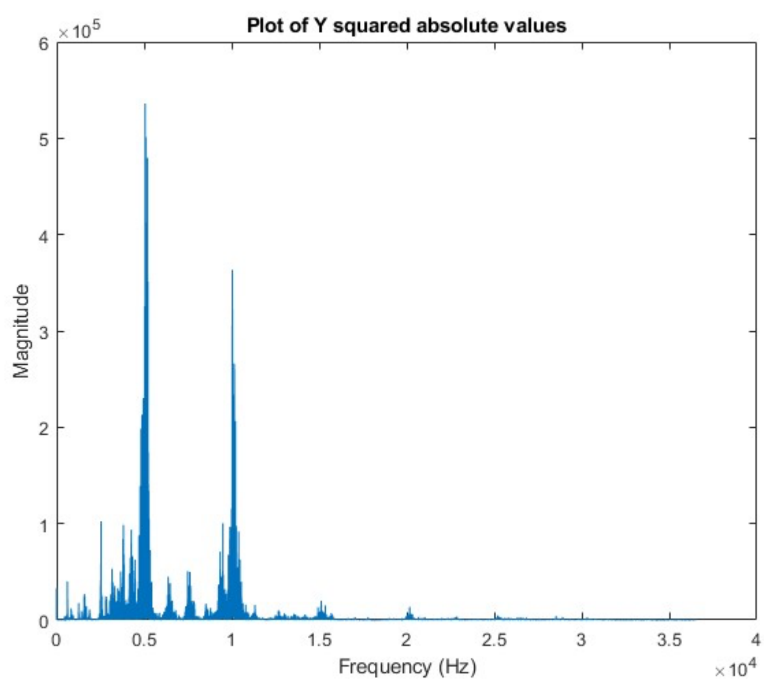
מטרת התרגיל: לתכנן מסנן ספרתי ע״י תכנון מסנן Butterworth אנלוגי והמרתו למסנן ספרתי באמצעות התמרה בי לינארית.

1. נתונים אותות $z(t)$ ו- $y(t)$ הדגומים בתדר $\text{Fs} = 8192\ \text{Hz}$ (יש לפתוח את הקובץ sig_2.mat במטלב או כל תוכנה אחרת).

   a. שרטט/י את הערך המוחלט בריבוע של התמרת פורייה האנלוגית של כל אחד מהאותות באמצעות מחשב ע״י שימוש ב-DFT.







b. 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.

c. האזן/י לאותות y ו- z. אם עובדים במטלב, אפשר להשתמש בשורות הבאות

```
playerObj = audioplayer(y,Fs);
```

```
start = 1;
stop = playerObj.SampleRate * 3;
play(playerObj,[start,stop]);
```

d. 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 ($z(t)$ או $y(t)$) כך שהאותות ישמעו דומה זה לזה ככל שניתן. את זאת יש לעשות ע״י מסנן ספרתי $H(z)$ השקול למסנן אנלוגי $H_c(s)$ מעביר נמוכים (low-pass) בעל המאפיינים הבאים:

$$A_s = 20\ dB$$

$$A_p = -20log_{10}(1-\delta_p) < 5\ dB$$

$$\Omega_p = 3600\ \times 2\pi\ \text{K rad/sec}$$

$$\Omega_s = 3800\ \times 2\pi\ \text{K rad/sec}$$

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

תשובה:

הדרישות שניתנו עבור המסנן האנלוגי הם:

- גליות בתחום ההעברה A_p (ripple): פחות מ-$5\ dB$
- ניחות בתחום הקיטעון A_s: $20\ dB$
- תדר תחום ההעברה Ω_p: $3600 \cdot 2\pi \left[\frac{rad}{sec}\right]$
- תדר תחום ההעברה Ω_s: $3800 \cdot 2\pi \left[\frac{rad}{sec}\right]$

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 F_s:

1. תדרי תחום המעבר Ω_p ותדרי תחום הקיטעון Ω_s בתחום הדיגיטלי:

שימוש בהתמרה הבי ליניארית:

$$\omega_p = 2F_s \tan\left(\frac{\Omega_p}{2F_s}\right)$$

$$\omega_s = 2F_s \tan\left(\frac{\Omega_s}{2F_s}\right)$$

2. גליות תחום ההעברה δ_p וגליות תחום הקיטעון δ_s:

גליות תחום המעבר והנחתת תחום הקיטעון במונחים של סקאלה לינארי:

$$\delta_p = 1 - 10^{\frac{-A_p}{20}}$$

$$\delta_s = 10^{\frac{-A_s}{20}}$$

מעוניינים לתכנן מסנן ספרתי IIR בעל פונקציית תמסורת $\mathrm{H}(z)$ באמצעות המרה של מסנן Butterworth $\widetilde{H}(s)$ ע״י ההתמרה הבי-לינארית:

$$\mathrm{H}(z) = \widetilde{\mathrm{H}}(s)\Big|_{s=\frac{z-1}{z+1}}$$

ב. חשב/י תדרים אנלוגיים מתאימים למסנן Butterworth $\tilde{H}(j\Omega)$ המבוקש. האם תדרים אלה צריכים להיות זהים לתדרים האנלוגיים הנדרשים ל- $H_c(j\Omega)$?

תשובה:

1. התאמת התדרים:

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

$$\Omega_p' = \frac{2F_s \tan\left(\frac{\Omega_p}{2F_s}\right)}{2\pi}$$

$$\Omega_s' = \frac{2F_s \tan\left(\frac{\Omega_s}{2F_s}\right)}{2\pi}$$

2. 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.

ג. תכנן/י מסנן אנלוגי מסוג Butterworth כתוב/י ביטוי כללי לאפסים של המסנן Butterworth ושרטט במחשב את מגניטודת תגובת התדר $\tilde{H}(j\Omega)$ במחשב.

פתרון:

1. סדר המסנן:

סדר n של מסנן Butterworth ניתן לחשב באמצעות הנוסחה הבאה:

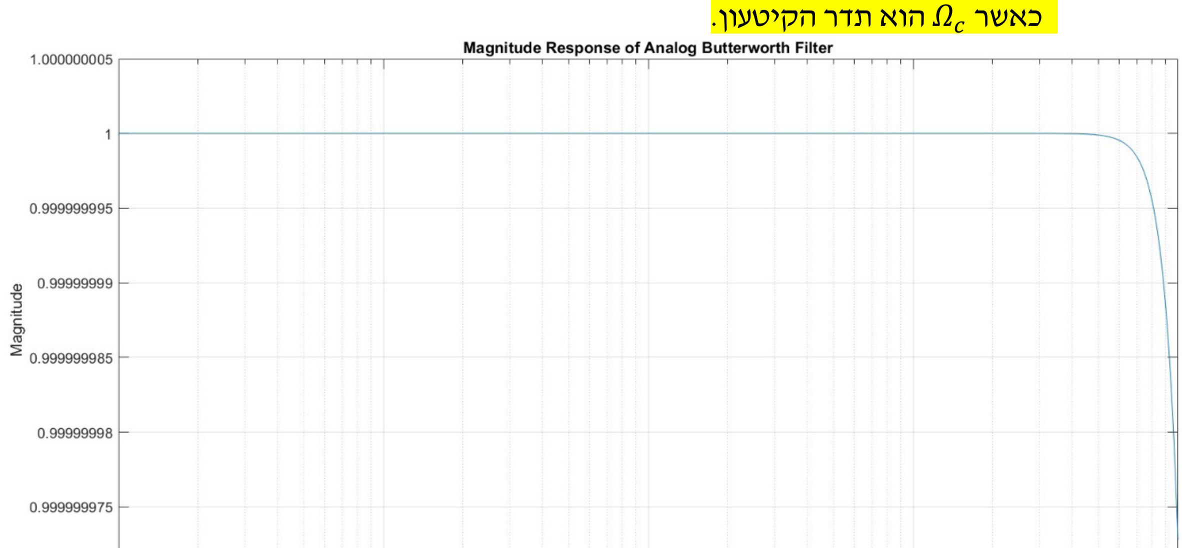
$$n = \left\lceil \frac{\log_{10}\left(\frac{\sqrt{10^{0.1A_s} - 1}}{\sqrt{10^{0.1A_p} - 1}}\right)}{2\log_{10}\left(\frac{\Omega_s}{\Omega_p}\right)} \right\rceil$$

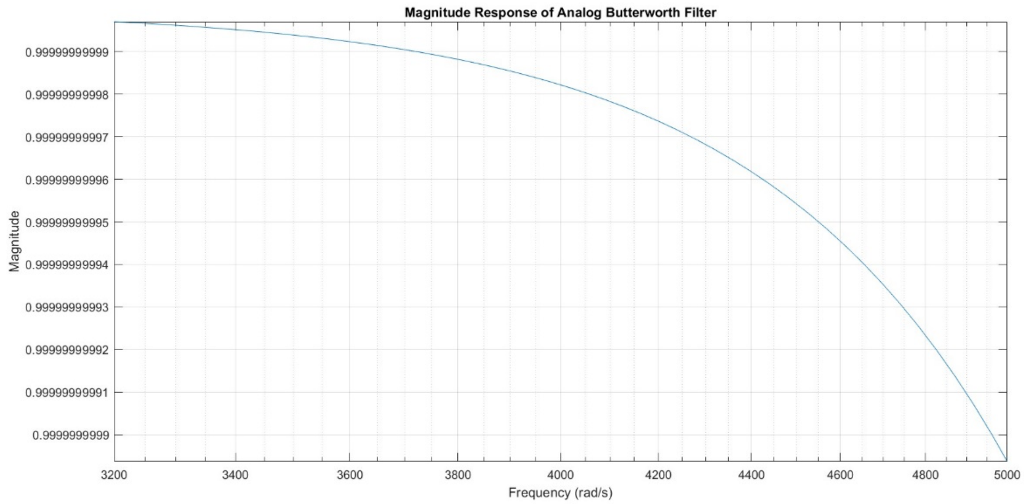
2. פונקציית התמסורת:

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

$$s_k = \Omega_c e^{j\frac{(2k+1+n)\pi}{2n}}$$

כאשר Ω_c הוא תדר הקיטעון.





ד. שרטט/י את מגניטודת תגובת התדר של המסנן הספרתי $H\left(e^{j\omega}\right)$.

1. התמרה בי-לינארית

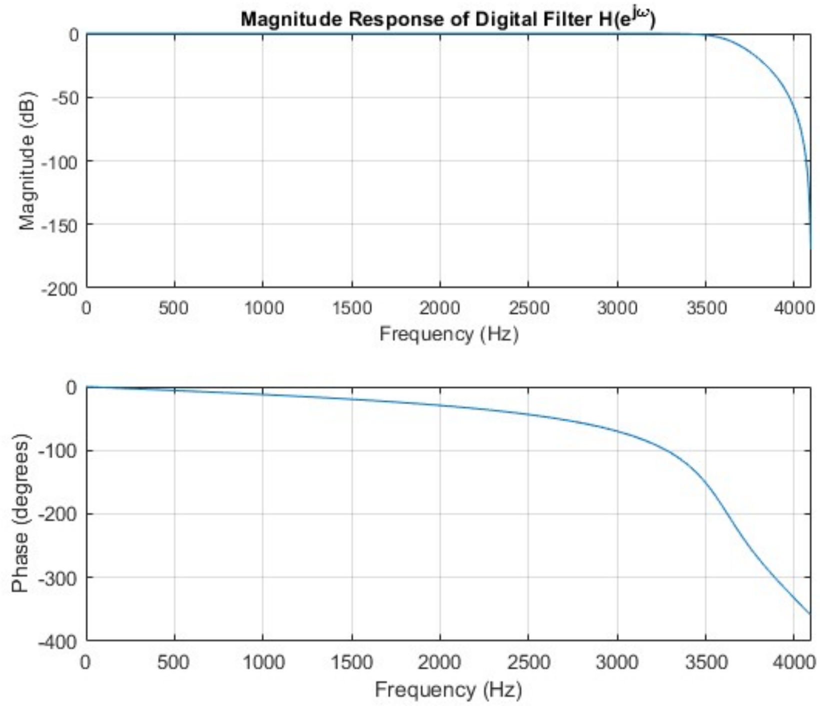
נפעיל את ההתמרה ההבי-לינארית כדי להמיר את מסנן Butterworth האנלוגי $H(s)$ לתחום הדיגיטלי $H(z)$:

$$H(z) = H(5)\Big|_{s=\frac{2}{T}\cdot\frac{1-z^{-1}}{1+z^{-1}}}$$

2. 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})');
```



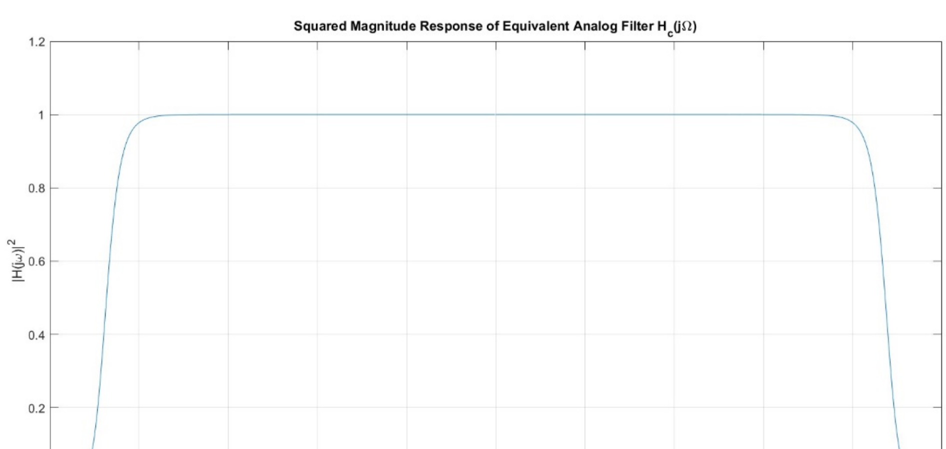
ה. שרטט/י את תגובת התדר של המסנן הספרתי השקול $H_c(j\Omega)$.

פתרון:

1. כדי לשרטט את תגובת התדר של המסנן האנלוגי השקול $H_c(j\Omega)$, אנו צריכים להמיר את תגובת התדר הדיגיטלית $H(e^{j\omega})$ חזרה לתחום האנלוגי. נעשה זאת באמצעות נוסחת ההמרה:

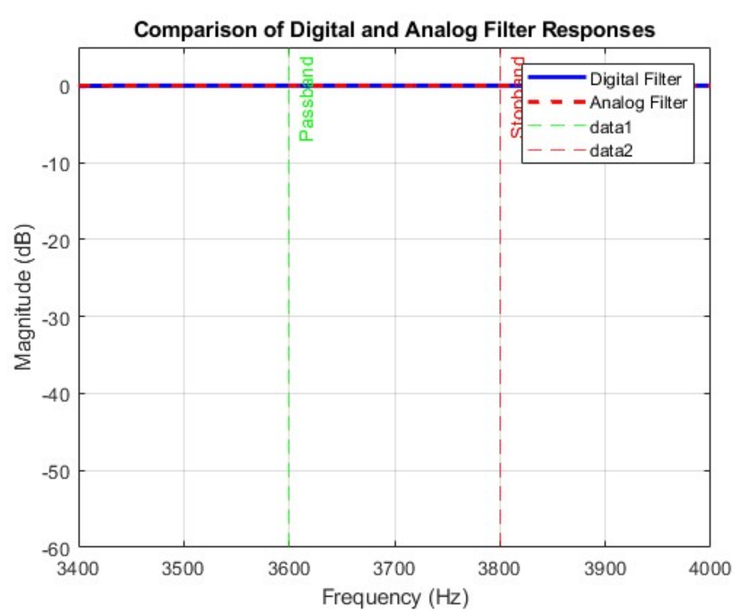
$$\Omega = \frac{2}{T} tan\left(\frac{\omega}{2}\right)$$

נשרטט את $H_c(j\Omega)$ באותו האופן בו שרטטנו את $H(e^{j\omega})$:



```
% 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');
```

ו. סנן את אחד מהאותות ($y(t)$ או $z(t)$) 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

.2

2. בתרגיל זה נתבונן ב- $x(t) = x_0(t) + x_1(t)$ כאשר:

$x_0(t) = A_0 \sin(\Omega_0 t)$

$x_1(t) = A_1 \sin(\Omega_1 t)$

כאשר Ω_0 ו- Ω_1 הינם תדרים לא ידועים למעט $\Omega_0, \Omega_1 < 3200 \times 2\pi$. האות x(t) נדגם בתדר $\Omega_s = 6720 \times 2$ וממור לסידרה $x[n], n = 0, \dots, N-1$.

נתון לנו כי:

$$x(t) = x_0(t) + x_1(t), x_0(t) = A_0 \sin(\omega_0 t), x_1(t) = A_1 \sin(\omega_1 t)$$

$$\omega_s = 6720$$

$$A_0 = A_1$$

$$\Delta\omega = |\omega_1 - \omega_0| min = ?$$

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: $x[n] = A_0 \sin[w_0 n] + A_1 \sin[w_1 n]$

והתמרת הפורייה תהיה

$$X(w) = \frac{\pi}{j}(A_0\delta(w - w_0) - A_0\delta(w + w_0) + \frac{\pi}{j}(A_1\delta(w - w_1) - A_1\delta(w + w_1)$$

ומכיוון שנכפיל בחלון- $w[n] = 1\ 0 \le n \le N-1, = 0 else$

$$Y(W) = X(W) * SINC =$$

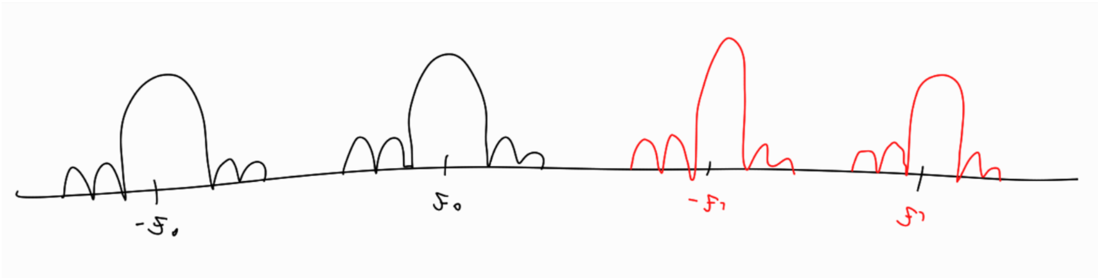
$$= \frac{\pi}{j}(A_0 W(w - 2\pi f_0) - A_0 W(w + 2\pi f_0) + \frac{\pi}{j}(A_1 W(w - 2\pi f_1) - A_1 W(w + 2\pi f_1)$$

כאשר $W = D(w,N)e^{-\frac{jw(N-1)}{2}}$ כאשר $d(w,N) = \frac{\sin\left(\frac{N}{2}w\right)}{\sin\left(\frac{w}{2}\right)}$

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:

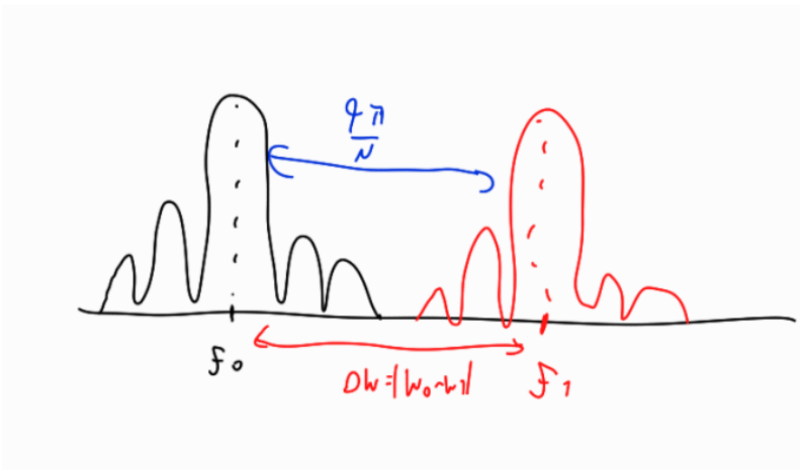
כאשר מכיוון שנקח את העצמה-בעצם כשנבצע את ה DFT נקבל את הערך המוחלט כלומר:

$$\pi(A_0 W(w - 2\pi f_0) - A_0 W(w + 2\pi f_0) + \pi(A_1 W(w - 2\pi f_1) - A_1 W(w + 2\pi f_1)$$



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

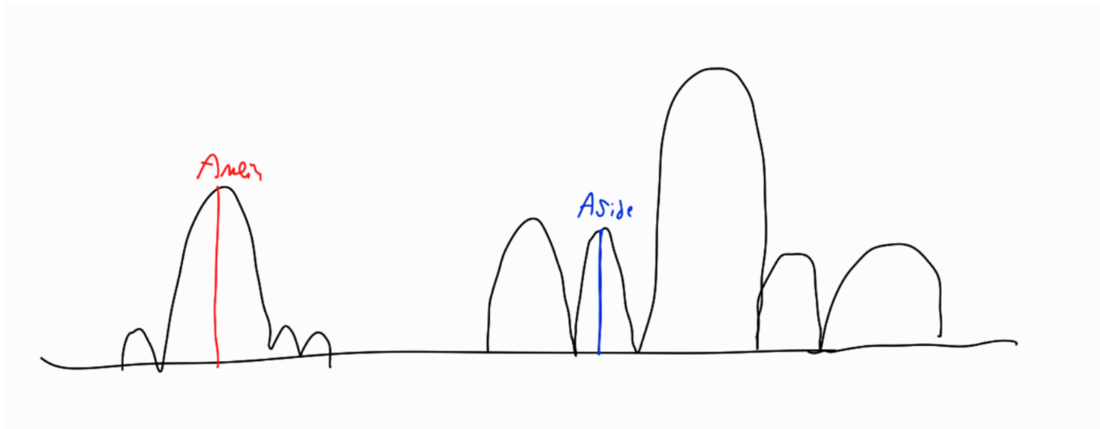
$\Delta\omega = |\omega_0 - \omega_1| > \frac{4\pi}{N}$ כאשר $\frac{4\pi}{N}$ הוא הרוחב של האונה הראשית



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

אם $A_0 \neq A_1$ נרצה שהאונה הראשית של אחד לא תחתוך את האונה הראשית של אחר כלומר-אם נניח ש$A_1 > A_0$ שהאונה המשנית של $A1$ לא תהיה גדולה יותר מהראשית של $A0$ כי לא נוכל להבדיל:

כלומר נדרוש: $\frac{A_{side}}{A_{main}} < \frac{A_0}{A_1}[dB]$



א. עבור $A_1 = A_0$ מהו הפרש התדרים $\Delta\Omega = |\Omega_1 - \Omega_0|$ המינימאלי המאפשר להבחין בין התדרים Ω_0 ו- Ω_1 עבור $\mathrm{N} = 16, 32, 64, 128, 256$. יש להציג את הספקטרום לכל N ולהדגים את ההפרדה.

ב. עבור $A_1 = 1$, $A_0 = 0.05$, משתמשים בחלון Hann. מהו $\Delta\Omega = |\Omega_1 - \Omega_0|$ המינימאלי המאפשר להבחין בין התדרים Ω_0 ו- Ω_1 עבור $\mathrm{N} = 16, 32, 64, 128, 256$. יש להציג את הספקטרום לכל N ולהדגים את ההפרדה.

בסעיף א-כש $A_0 = A_1$ נוכל להשתמש בחלון רגיל ולקיים את הדרישה בקלות-בסעיף ב-הדרישה חמורה יותר:

שוב- נדרוש $[dB] \frac{A_0}{A_1} > \frac{A_{side}}{A_{main}}$ ואצלנו $\frac{A_0}{A_1} = -26.02[dB]$ 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: $\Delta\omega < \frac{4\pi}{N}$

נציב כל פעם מחדש עבור כל N ונקבל את הפרשי התדירויות: $\Delta f = |f_0 - f_1| < \frac{2}{N}$:

ג. $A_1 = 1$, $A_0 = 0.001$. האם ניתן להבחין בין התדרים השונים? אם כן, איזה חלון דרוש ומהו $\Delta\Omega = |\Omega_1 - \Omega_0|$ המינימאלי המאפשר להבחין בין התדרים Ω_0 ו- Ω_1 עבור $\mathrm{N} = 16, 32, 64, 128, 256$. יש להציג את הספקטרום לכל N ולהדגים את ההפרדה.

כעת $\frac{A_0}{A_1} = -60[dB]$ 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 שההנחתה היא *60* כלומר $A=60$:

$$\alpha = \begin{cases} 0.1102(A-8.7) & A > 50 \\ 0.5842(A-21)^{0.4} + 0.07886(A-21) & 21 < A \leq 50 \\ 0 & A \leq 21 \end{cases}$$

ולכן a=0.1102(A-8.7)=5.65

נמצא כעת את התנאי על $\Delta\omega$: **כפי שלמדנו** $N = \frac{60-7.95}{14.36\Delta f} = 2\pi\frac{a-7.95}{14.36\Delta\omega} \rightarrow \Delta\omega = N\frac{14.36}{2\pi(60-7.95)}$

וכפי שלמדנו:

פונקציית החלון מוגדרת:

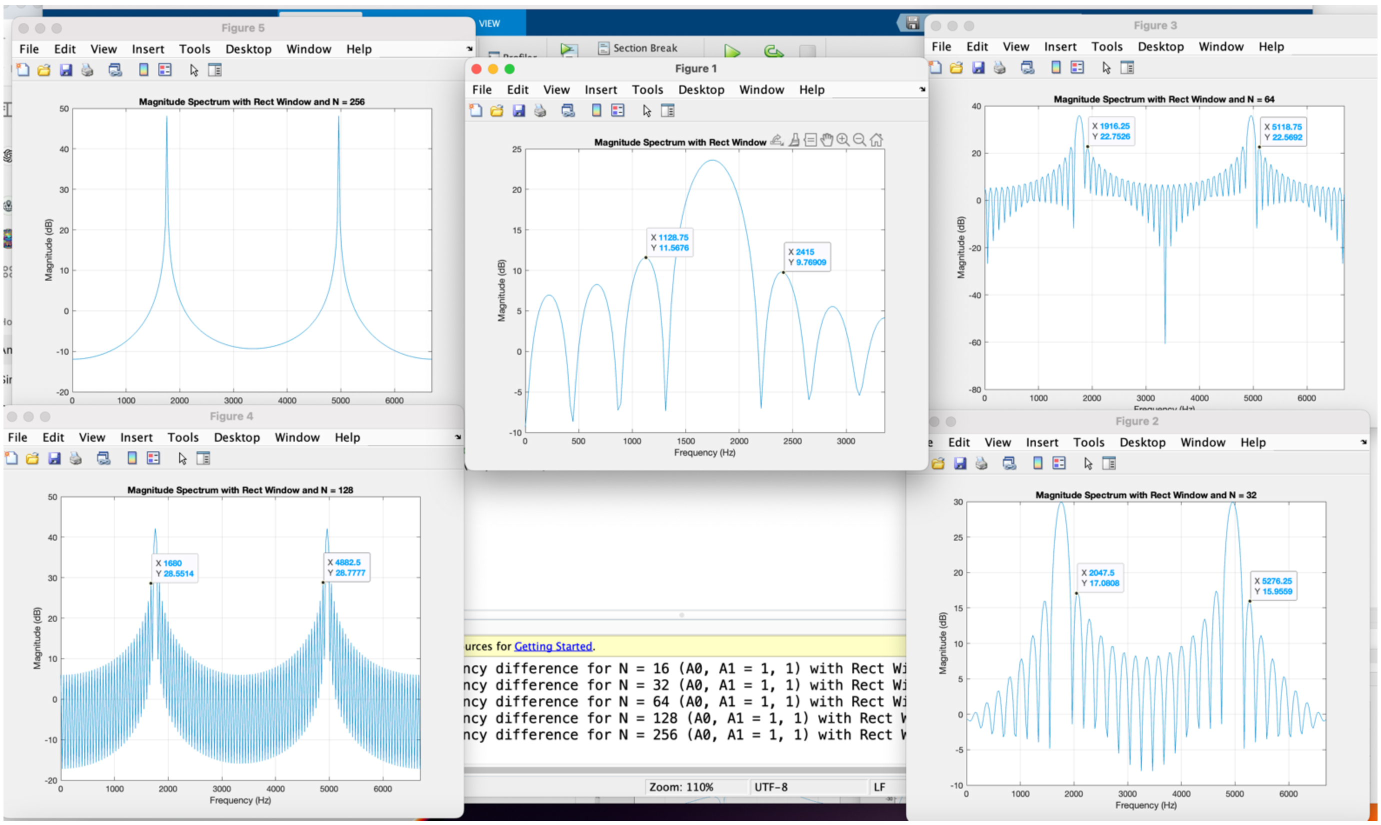
$$w[n] = \frac{I_0\left(\beta\sqrt{1-\left(\frac{n-\gamma}{\gamma}\right)^2}\right)}{I_0(\beta)} \quad , \quad 0 \leq n \leq N$$

כאשר $\gamma = N/2$, ו $I_0(\cdot)$ פונקציית Bessel מסדר 0, הנתונה על ידי:

$$I_0(x) = 1 + \sum_{k=1}^{\infty}\left(x/2\right)^{2k}(k!)^{-2}$$

ונציב...

פתרון סעיף א :



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

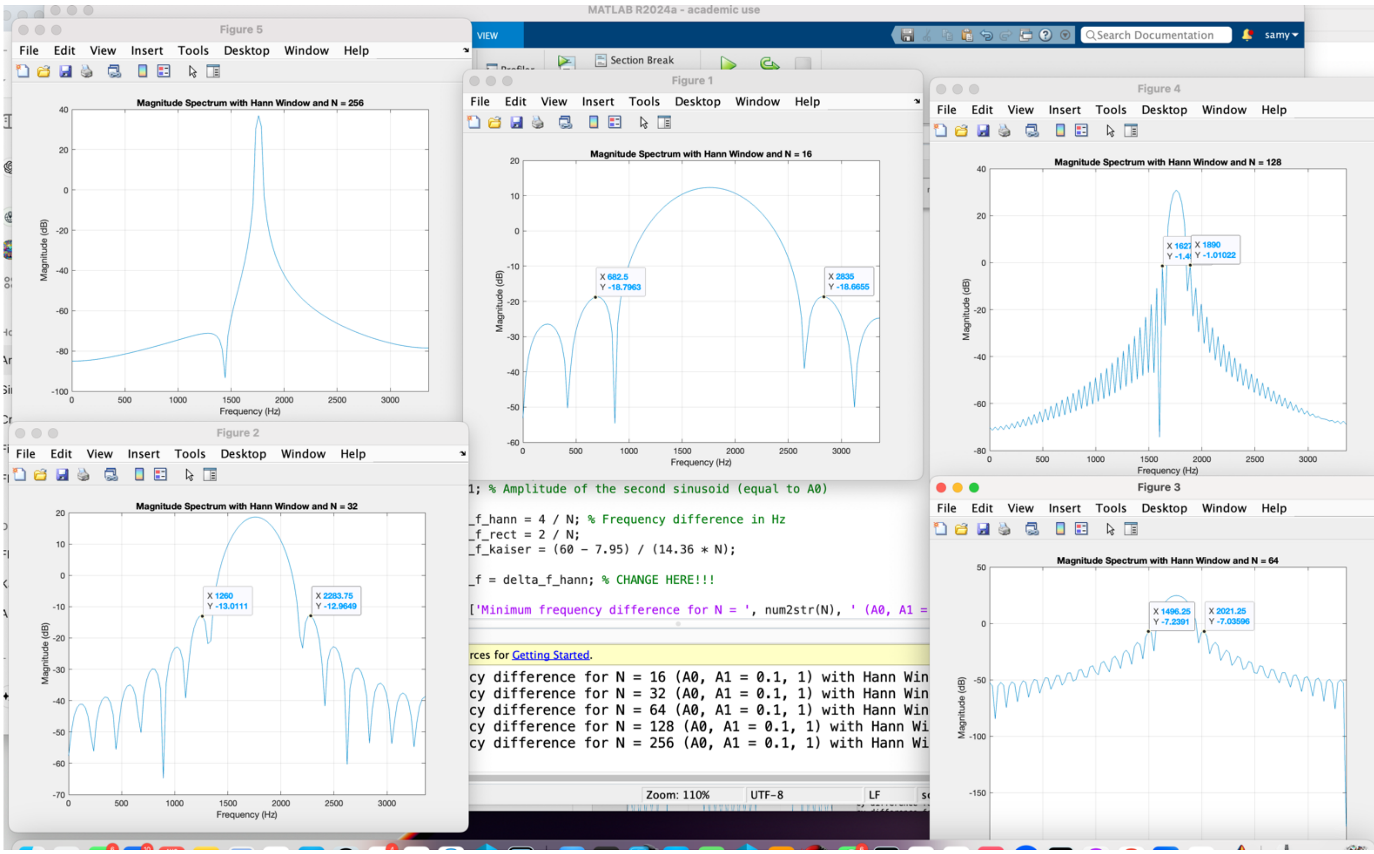
Minimum frequency difference for N = 32 (A0, A1 = 1, 1) with Rect Window is 0.0625 Hz

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

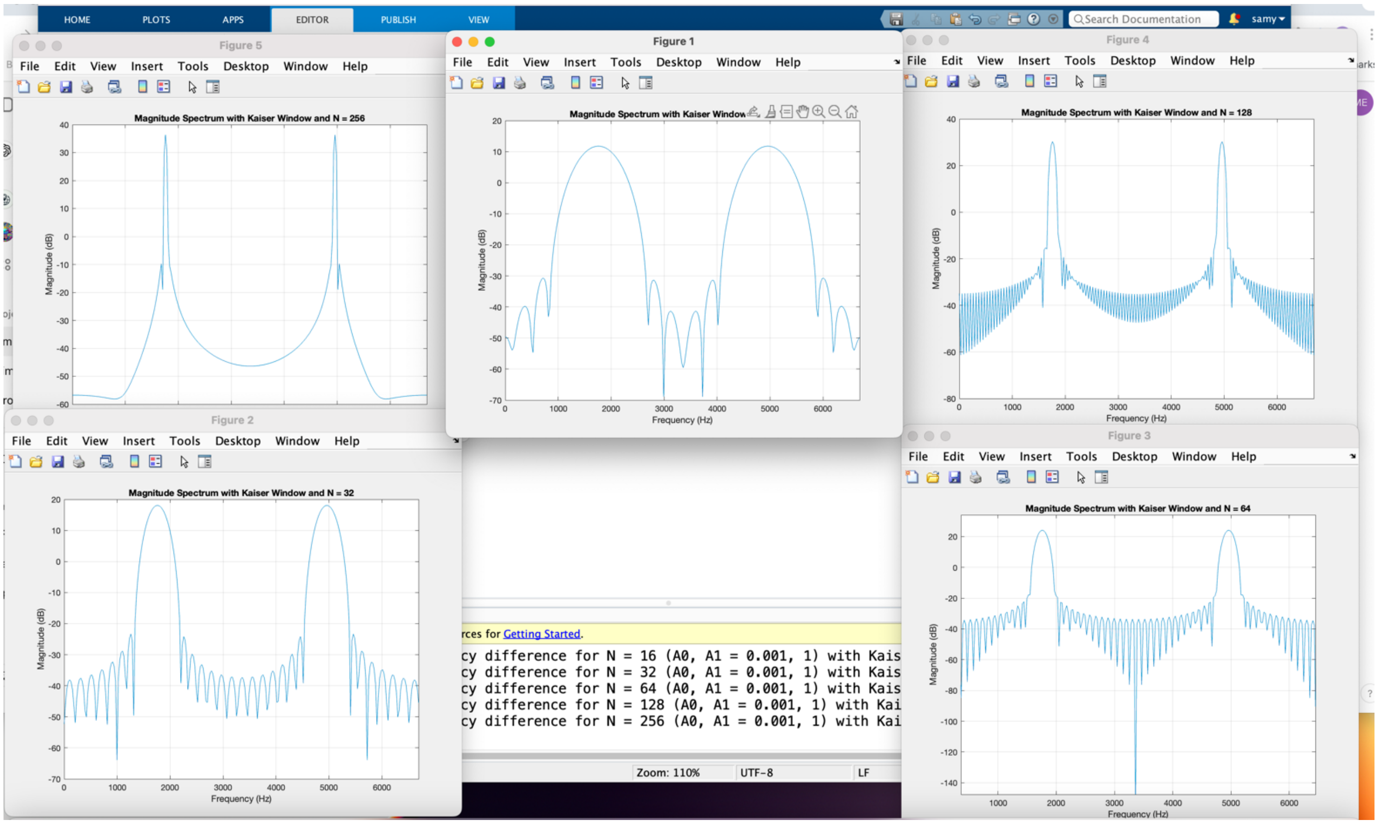
Minimum frequency difference for N = 32 (A0, A1 = 0.05, 1) with Hann Window is 0.125 Hz

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;
```