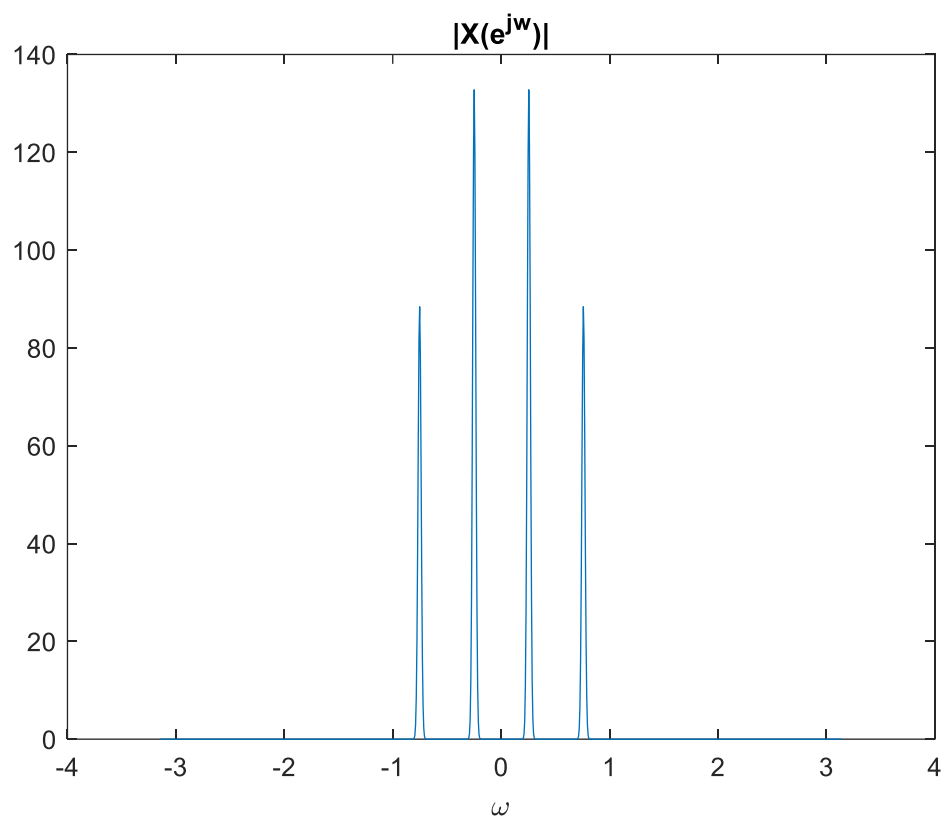
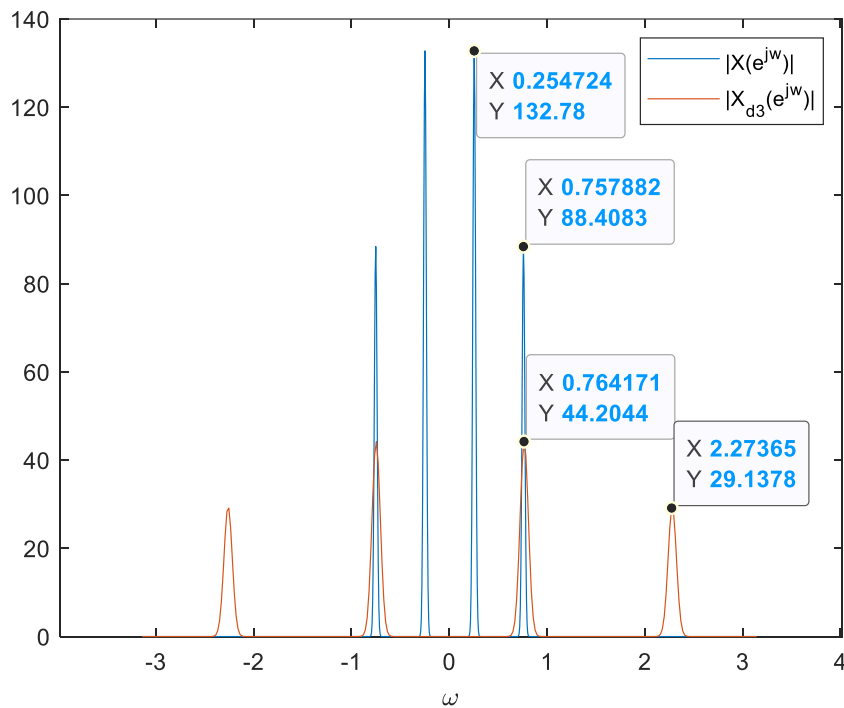
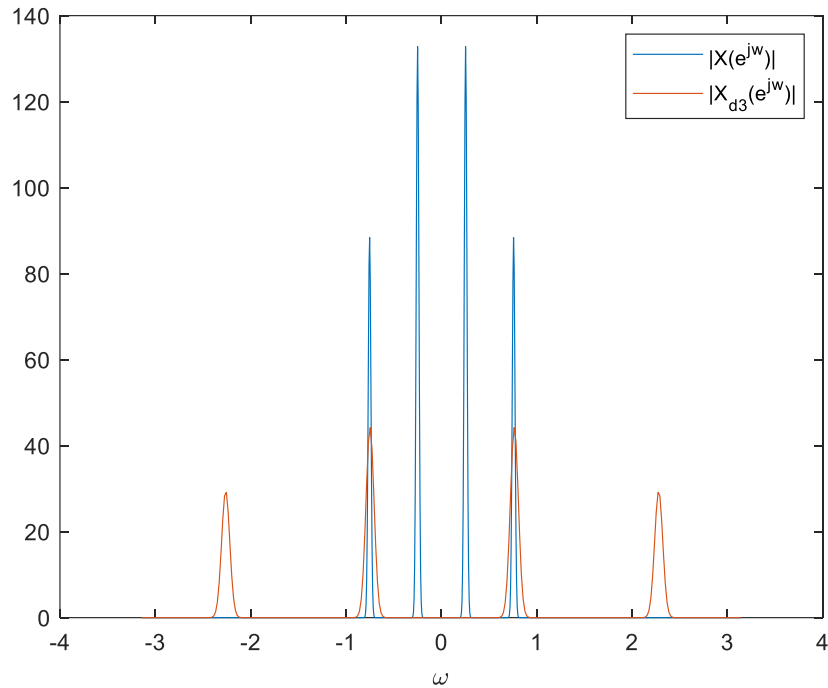


תרגיל מטלב #2

(א)



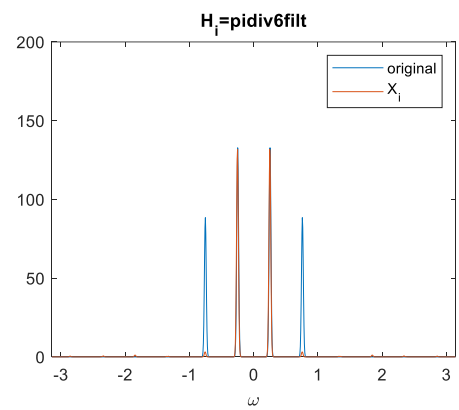
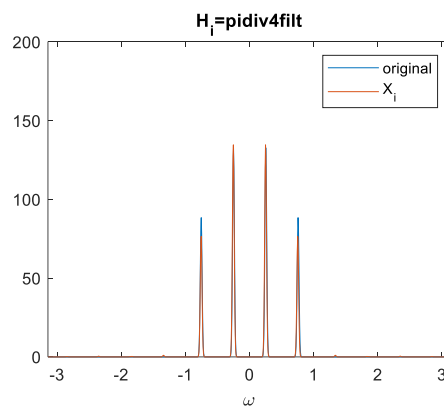
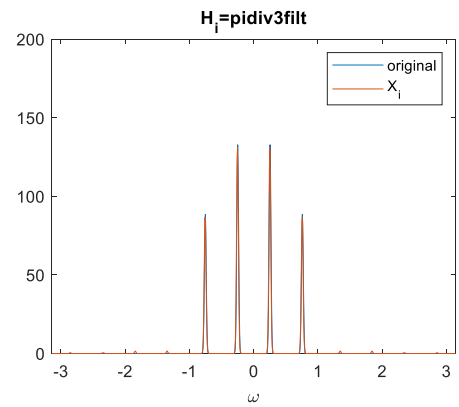
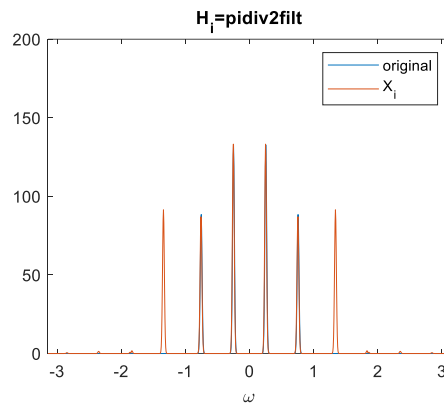
(ב)



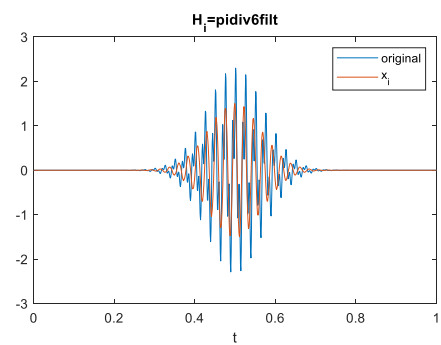
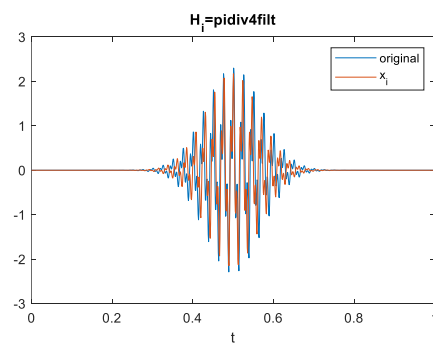
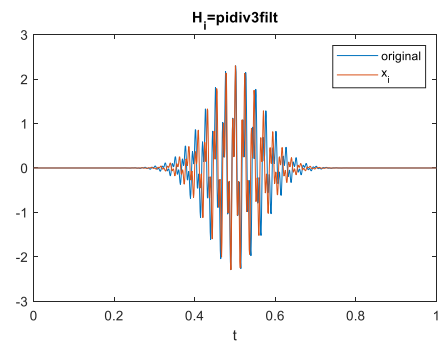
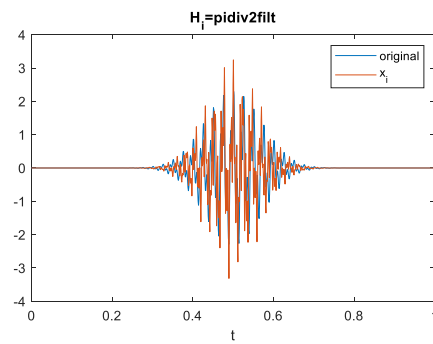
מהשוואה בין התמרת האות המקורי להתמרת האות שעבר הורדת קצב פי 3, אפשר לראות כי התמרת האות שעבר הורדה עברה מתיחה של ציר התדר פי 3 וגם האמפליטודה של האות קטנה פי 3, כמצופה שהרי כפי שלמדנו, מתקיים הקשר:

$$X_{d3}(e^{j\omega}) = \frac{1}{3} * \sum_{k=0}^{M-1} X\left(e^{j(\omega - 2\pi k/3)}\right)$$

**לאחר ביצוע אינטרפולציה פי 3:
במישור התדר:**

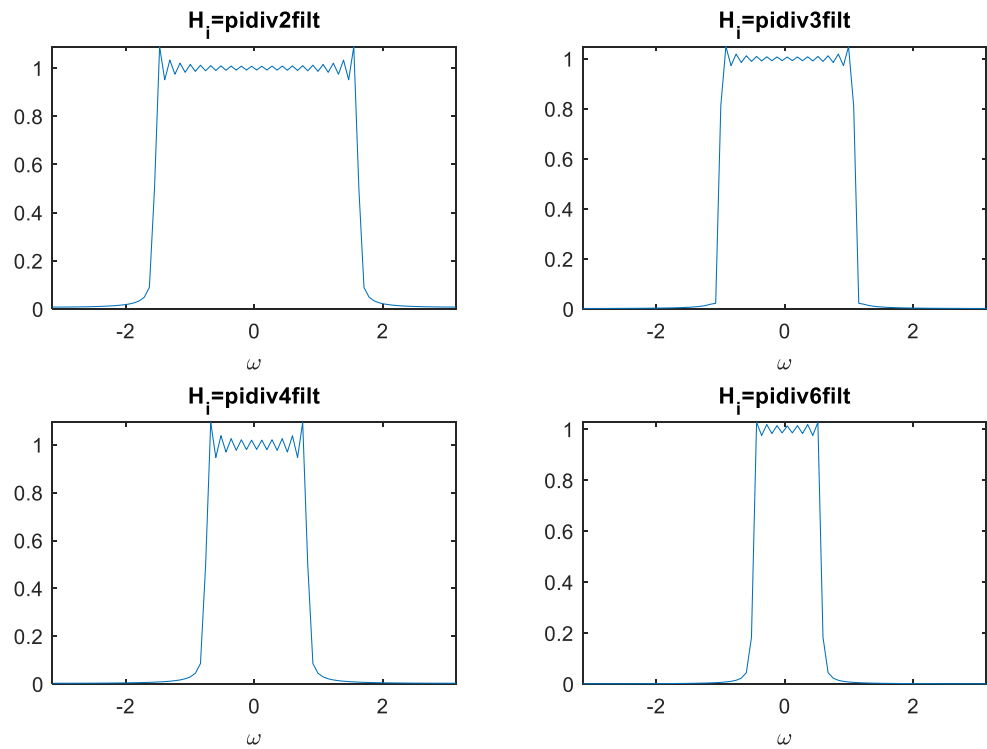


במישור הזמן:

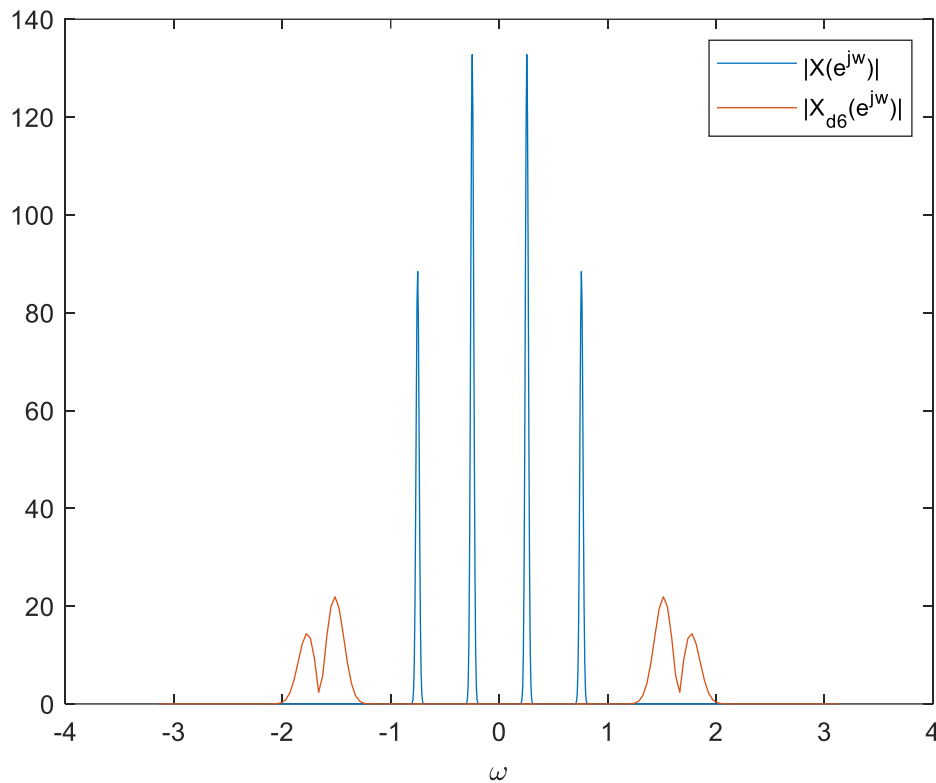


לפי התוצאות אפשר לראות כי שימוש במסנן pdiv3filt (המוכפל ב 3) מניב את השחזור המיטבי, אך לפי ההצגה בזמן אפשר לראות שהשחזור אינו מדויק, והסיבה לכך היא שמדובר במסנן לא אידיאלי, כפי שניתן לראות בגרף הבא:

$$|H_i(e^{j\omega})|$$



(ג)



אפשר לראות כי כתוצאת מהורדת הקצב פי 6, נוצר aliasing.
 כפי שלמדנו בהרצאה, כדי לבצע הורדת קצב ללא aliasing, צריך לעמוד בדרישה: $\omega_b < \frac{\pi}{M}$
 במקרה שלנו:

$$\begin{aligned}\omega_b &= 0.76 \text{ rad} \\ \frac{\pi}{M} &= \frac{\pi}{6} = 0.52 \\ \rightarrow \omega_b &> \frac{\pi}{M} \rightarrow \text{aliasing occurs.}\end{aligned}$$

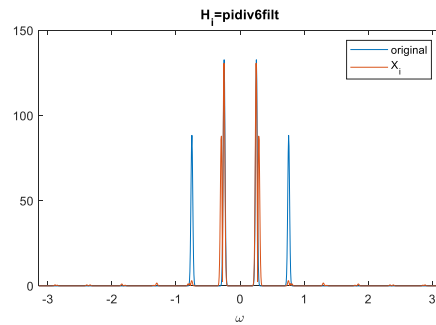
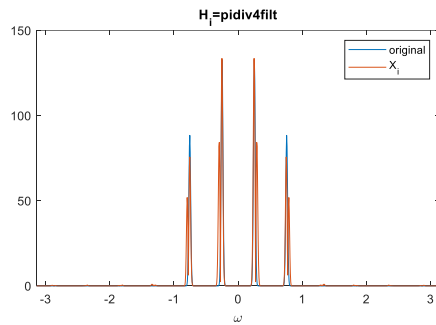
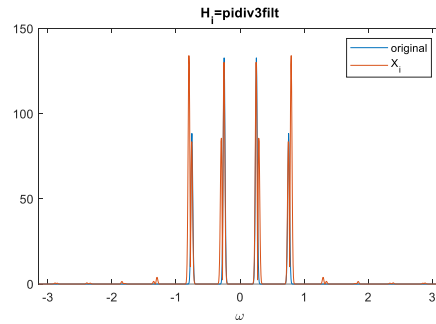
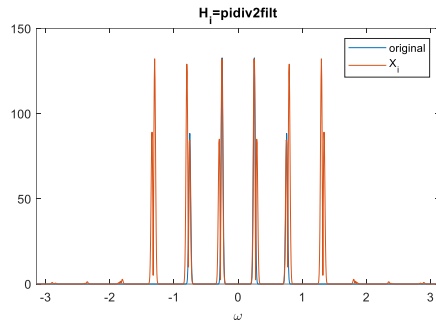
בסעיף (ב), ביצענו הורדת קצב פי 3, לכן במקרה זה:

$$\frac{\pi}{M} = \frac{\pi}{3} = 1.05 \rightarrow \omega_b < \frac{\pi}{M} \rightarrow \text{no aliasing occurs.}$$

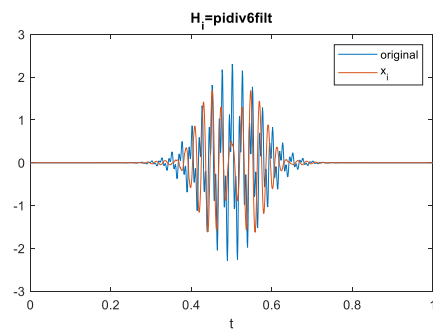
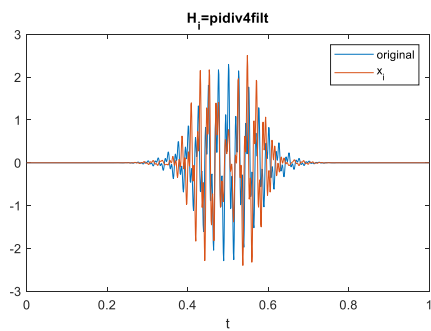
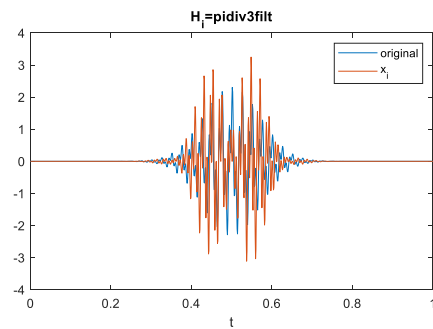
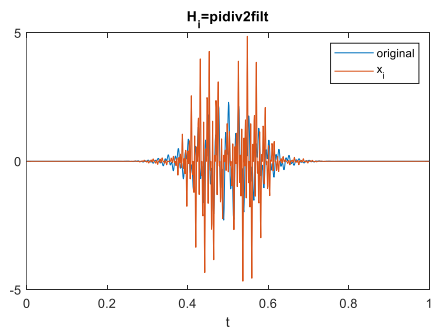
לאחר ביצוע אינטרפולציה פי 6:

במישור התדר:

במישור הזמן:

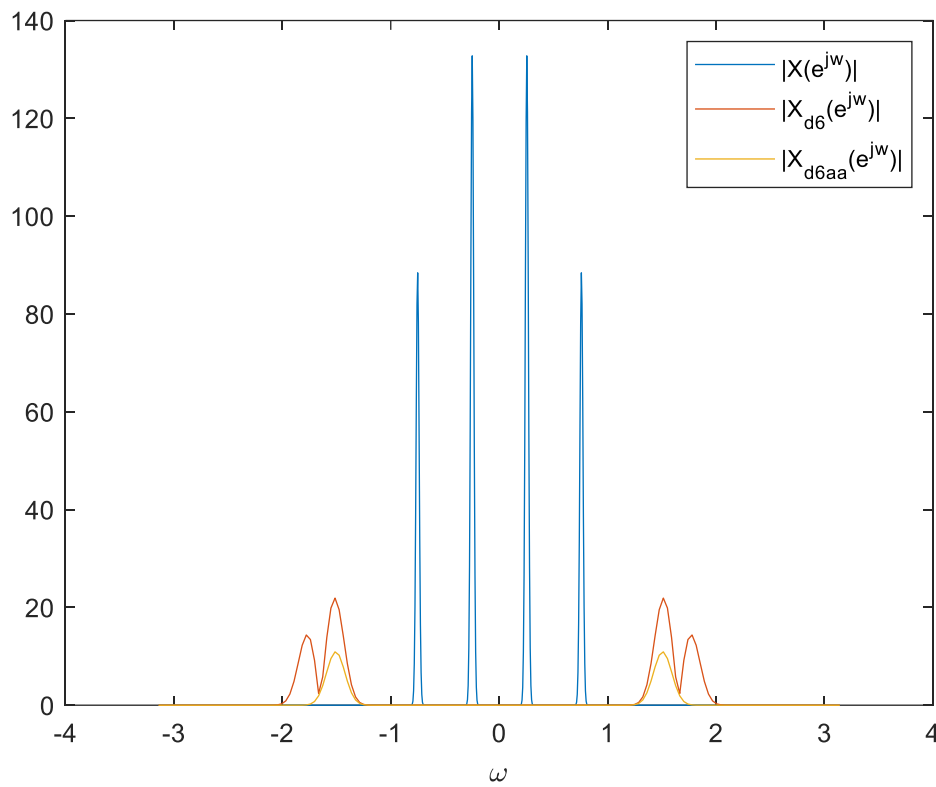


במישור הזמן:



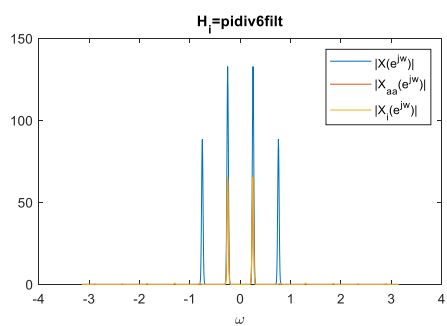
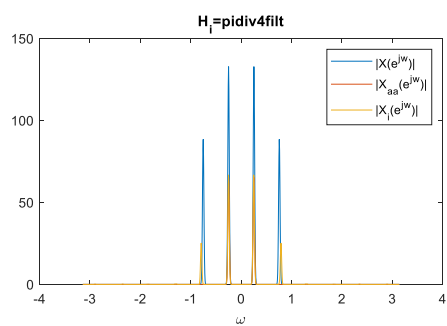
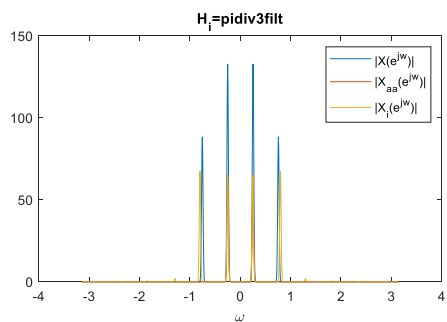
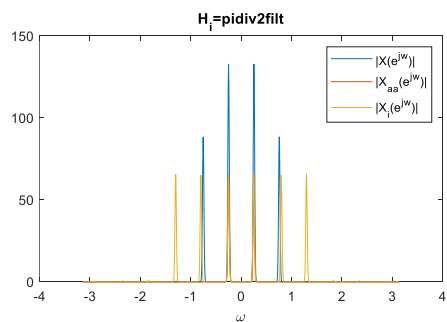
ניתן לראות שהאות המשוחזר שונה משמעותית מהאות המקורי, וזאת כתוצאה מהעיוות שנגרם מה *aliasing*

(ד)
לאחר הורדת הקצב:

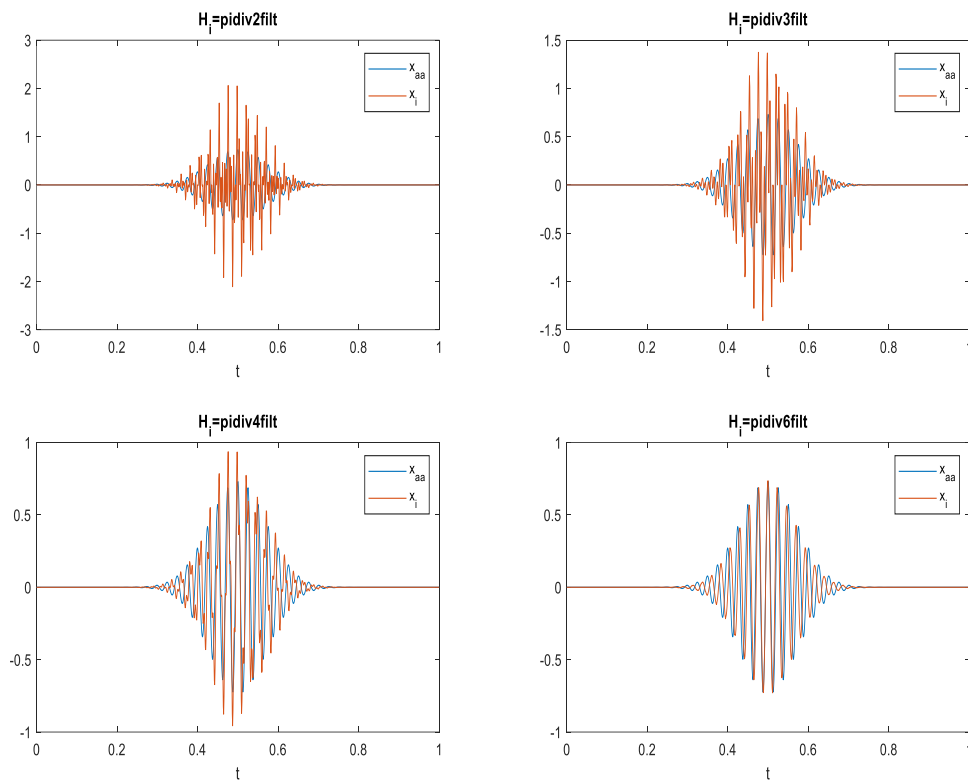


ניתן לראות שהעברה דרך מסנן ה *anti aliasing* מנעה *aliasing* כתוצאה מהורדת הקצב (בשונה מסעיף c)

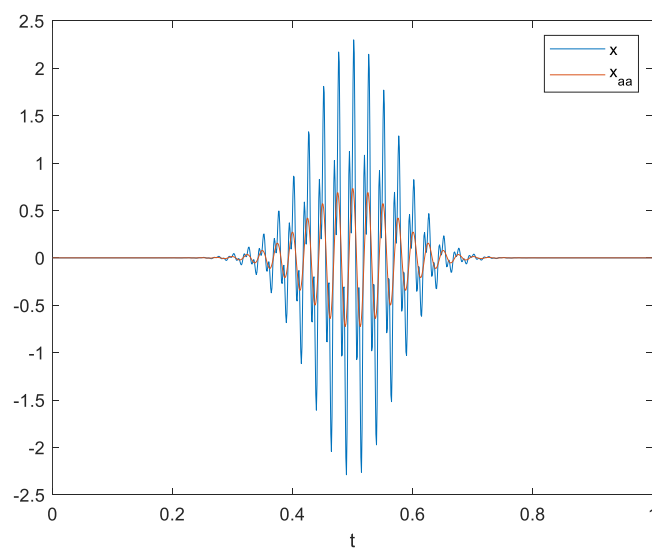
לאחר ביצוע אינטרפולציה פי 6:
במישור התדר:



במישור הזמן:



אפשר לראות שכתוצאה מהעברת האות דרך מסנן האנטי אליאסינג, האות המשוחזר (x_i) כמעט מתלכד עם x_{aa} (כשאר משתמשים במסנן $pidiv6filt$) כלומר בשונה מהסעיף הקודם, האות המשוחזר כמעט מלכד עם האות ההתחלתי (שהוא x_{aa}) אבל עדיין שונה מהאות המקורי (x) מכיוון שהאות במוצא המסנן אנטי אליאסינג אינו מכיל מידע שקיים באות המקורי. השוני בין האות המשוחזר לאות x_{aa} נובע מהעובדה שהמסנן אינו אידיאלי כפי שהדגמתי בסעיף ב.



קוד:

```

% written by Michal Keren 7.6.21
close all;
clc;
%% ----A-----%
fs= 1000;
Ts=1/fs;
t= linspace(0,1,1000); % the spacing is exactly 1/fs.
x=(1.5*cos(2*pi*40.*t) + sin(2*pi*120.*t)).*exp(-100.*(t-0.5).^2);
%FFT:
X= fftshift(fft(x));
plotFFT(X, '|X(e^jw)|')
%% ----B-----%
%decimation by 3:
L=3;
x_d3= x(1:L:end);
X_d3= fftshift(fft(x_d3));
figure(2);
plotFFT(X, '|X(e^jw)|')
hold on;
plotFFT(X_d3, '')
legend('|X(e^jw)|', '|X_d_3(e^jw)|')
%interpolation by 3:
x_p3= zeros(1000,1);
x_p3(1:3:end)= x_d3; %zero padding
h= L.*[pidiv2filt; pidiv3filt ;pidiv4filt ;pidiv6filt];
h_names= ["pidiv2filt"; "pidiv3filt" ;"pidiv4filt" ;"pidiv6filt"];
figure(3);
for i=1:4
    x_int= conv(h(i,:),x_p3);
    X_int= fftshift(fft(x_int));
    subplot(2,2,i)
    plotFFT(X, '|X(e^jw)|')
    hold on;
    plotFFT(X_int, 'H_i='+h_names(i))
    legend('original', 'X_i')
    hold off;
    ylim([0 200])
    xlim([-pi pi])
end
%plots in time domain
figure(4);
for i=1:4
    x_int= conv(h(i,:),x_p3);
    subplot(2,2,i)
    plot(t,x);
    hold on;
    ti= linspace(0,1,length(x_int));
    plot(ti,x_int);
    title('H_i='+h_names(i));
    legend('original', 'x_i')
    xlabel('t')
    hold off;
end

%% ----C-----%
%decimation by 6:
L=6;
x_d6= x(1:6:end);
X_d6= fftshift(fft(x_d6));
figure(5);
plotFFT(X, '|X(e^jw)|')
hold on;
plotFFT(X_d6, '')
legend('|X(e^jw)|', '|X_d_6(e^jw)|')
%interpolation by 6:
x_p6= zeros(1000,1);
x_p6(1:6:end)= x_d6; %zero padding
h= L.*[pidiv2filt; pidiv3filt ;pidiv4filt ;pidiv6filt];
h_names= ["pidiv2filt"; "pidiv3filt" ;"pidiv4filt" ;"pidiv6filt"];
figure(6);
for i=1:4
    x_int= conv(h(i,:),x_p6);

```

```

X_int= fftshift(fft(x_int));
subplot(2,2,i)
plotFFT(X, '|X(e^jw)|')
hold on;
plotFFT(X_int, 'H_i='+h_names(i))
legend('original', 'X_i')
ylim([0 150])
xlim([-pi pi])
hold off;
end
%plots in time domain
figure(7);
for i=1:4
    x_int= conv(h(i,:),x_p6);
    subplot(2,2,i)
    plot(t,x);
    hold on;
    ti= linspace(0,1,length(x_int));
    plot(ti,x_int);
    legend('original', 'x_i')
    title('H_i='+h_names(i));
    xlabel('t')
    hold off;
end
%---plot the filters---%
figure(8);
for i=1:4
    H= fftshift(fft(h(i,:)));
    subplot(2,2,i)
    plotFFT(H, 'H_i='+h_names(i))
    % hold on;
    % plotFFT(X_int, 'H_i='+h_names(i))
    % legend('original', 'X_i')
    % ylim([0 300])
    xlim([-pi pi])
    hold off;
end
sgtitle('|H_i(e^jw)|')
%% ----D-----%
%Anti Aliasing filter:
wc =pi/6;
x_aa= lowpass(x,wc,fs);
figure(9);
plot(t,x)
hold on;
plot(t,x_aa);
legend('x', 'x_a_a');
xlabel('t')
hold off
X_aa = fftshift(fft(x_aa));
%decimation by 6:
x_d6_aa= x_aa(1:6:end);
X_d6_aa= fftshift(fft(x_d6_aa));
figure(10);
plotFFT(X, '|X(e^jw)|')
hold on;
plotFFT(X_d6, '')
hold on;
plotFFT(X_d6_aa, '')
legend('|X(e^jw)|', '|X_d_6(e^jw)|', '|X_d_6_a_a(e^jw)|')

%interpolation by 6:
x_p6= zeros(1000,1);
x_p6(1:6:end)= x_d6; %zero padding
h= 6.*[pidiv2filt; pidiv3filt ;pidiv4filt ;pidiv6filt];
h_names= ["pidiv2filt"; "pidiv3filt" ;"pidiv4filt" ;"pidiv6filt"];
figure(11);
for i=1:4
    x_int= conv(h(i,:),x_p6);
    X_int= fftshift(fft(x_int));
    subplot(2,2,i)
    plotFFT(X, '|X(e^jw)|')
    hold on;
    plotFFT(X_aa, '|X_a_a(e^jw)|')
    hold on;

```

```

plotFFT(X_int,'H_i='+h_names(i))
legend('|X(e^jw)|','|X_a_a(e^jw)|','|X_i(e^jw)|')
hold off;
ylim([0 150])
end
figure(12);
for i=1:4
    x_int= conv(h(i,:),x_p6);
    subplot(2,2,i)
    % plot(t,x);
    % hold on;
    plot(t,x_aa);
    hold on;
    ti= linspace(0,1,length(x_int));
    plot(ti,x_int);
    legend('x_a_a','x_i')
    title('H_i='+h_names(i));
    xlabel('t')
    hold off;
end

```

```

function [] = plotFFT(X,name)
    w= linspace(-pi,pi,length(X));
    plot(w,abs(X))
    title(name);
    xlabel('\omega')
end

```