

Sprawozdanie z laboratoriów „Próbkowanie sygnałów ciągłych”

Autor: Filip Pasternak

Grupa lab. 7 – 18:30 piątek.

Zadanie 1.

W środowisku programistycznym Matlab przeprowadzić za pomocą udostępnionego w konspekcie m-kodu rekonstrukcję sygnału ciągłego $\sin(\omega t)$ próbkowanego z częstotliwością 200Hz.

```
clear all;
close all;
syms t x w K

fp = 200;
fg = fp/2; %Hz
wp = 2*pi*fp;
wg = 2*pi*fg;
|
s = 4/5;
ws = s*wg;

x_sin = sin(ws*t);
X_FT_sin_org = fourier(x_sin);

X_FT_sin = X_FT_sin_org + ... % oryginal widma
    symsum((subs(X_FT_sin_org, w, w - K*wp ) + ...% 3 aliasy lewe
    subs(X_FT_sin_org, w, w + K*wp)), K , 1, 3); % 3 aliasy prawe

FILT_FT = rectangularPulse(-wg,wg,w); % filtr rekonstruujący
x_sin_rek = ifourier(X_FT_sin*FILT_FT); % odwr. tarnsf. Fouriera

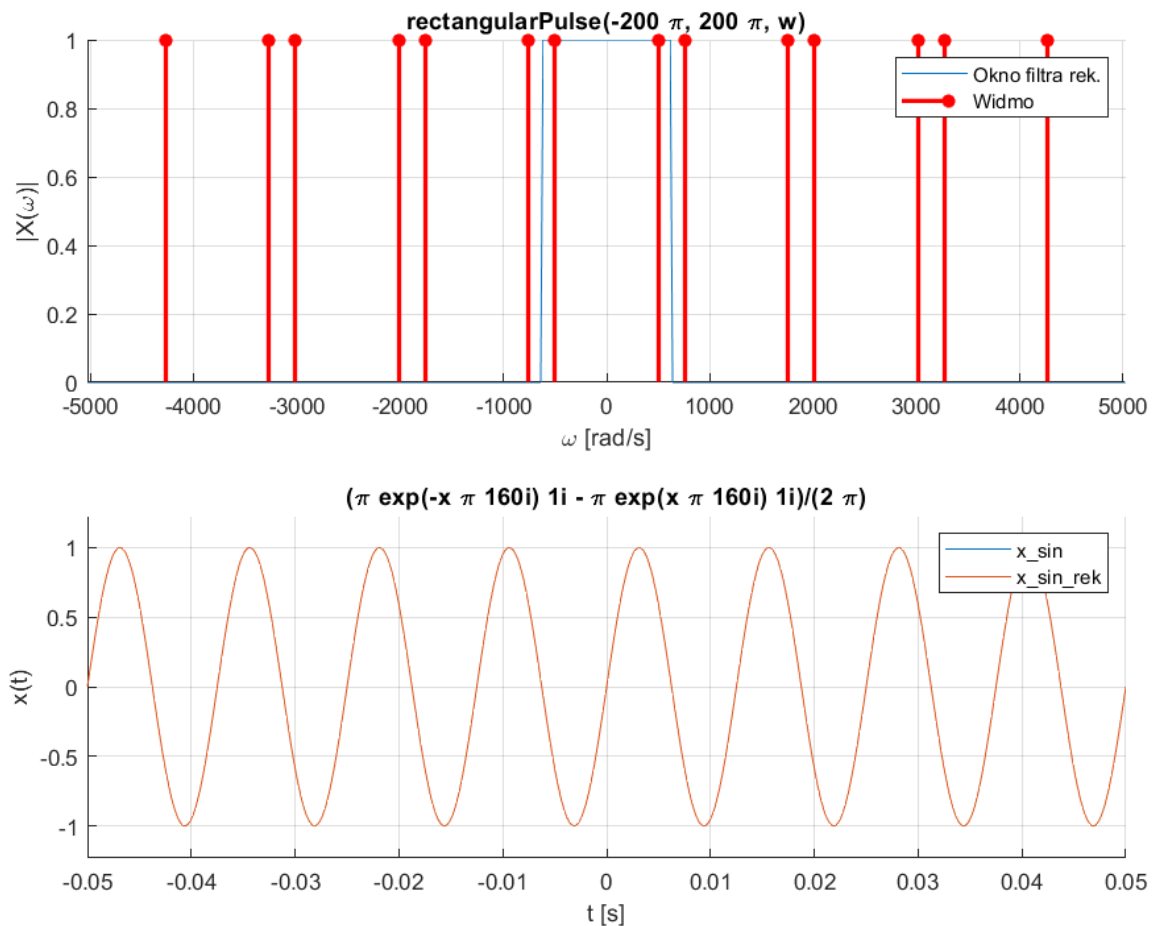
BND_t = [-10/fp;10/fp];
%t_SMP = [BND_t(1):1/(10*fp):BND_t(2) ];
BND_w = [-4*wp;4*wp];
w_SMP = [BND_w(1):wp/10:BND_w(2)];

figure; subplot(2,1,1); hold on; grid on;
ezplot(FILT_FT,BND_w); %okno filtru rek.
%ezplot(X_FT_sin,BND_w)
```

```
v_num = abs(double(subs(X_FT_sin, w, w_SMP)));
n = find(abs(v_num) == Inf);
stem(w_SMP(n),sign(v_num(n)), 'r*', 'Linewidth', 2);
xlabel('\omega [rad/s]'); ylabel('|X(\omega)|')
legend('Okno filtra rek.', 'Widmo');

subplot(2,1,2); hold on; grid on;
ezplot(x_sin, BND_t); % syg. próbkowany
ezplot(x_sin_rek, BND_t) % syg. odtworzony
xlabel('t [s]'); ylabel('x(t)')
legend('x_sin', 'x_sin_rek');
```

Kod został wzbogacony przeze mnie o dodatkowe przerwy, bądź spacje aby poprawić czytelność i ułatwić wprowadzanie zmian w kodzie. Reszta zadań opiera się tak naprawdę na tym samym kodzie, różniąc się często jedną liczbą bądź funkcją więc w celu poprawienia czytelności sprawozdania, m-kod zostanie umieszczony tylko raz.



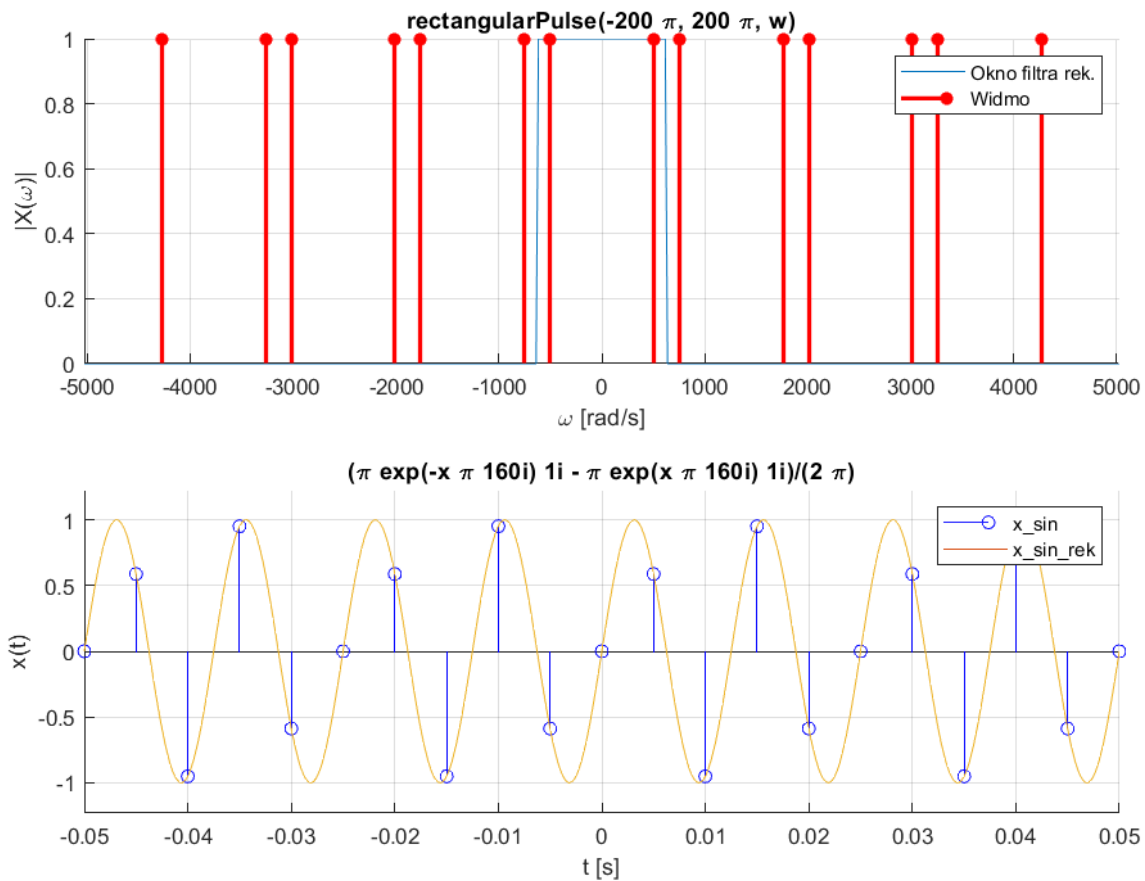
Zadanie 2.

Oznaczyć na wykresie czasowym węzły próbkowania wyznaczone przez okres próbkującej funkcji grzebieniowej.

```
wezly = BND_t(1) : 1/fp : BND_t(2);
wezly_val = subs(x_sin_rek, x, wezly);

subplot(2,1,2); hold on; grid on;
stem(wezly, wezly_val, 'b');
ezplot(x_sin, BND_t); % syg. próbkowany
ezplot(x_sin_rek, BND_t) % syg. odtworzony
xlabel('t [s]'); ylabel('x(t)')
legend('x_sin', 'x_sin_rek');
```

Jak widać zostało dodanych tylko kilka linijek odpowiedzialnych za zaznaczenie węzłów na wykresie.



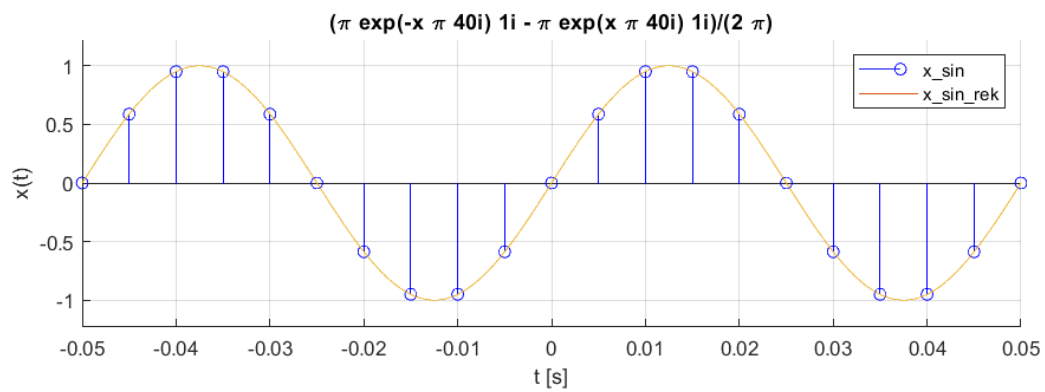
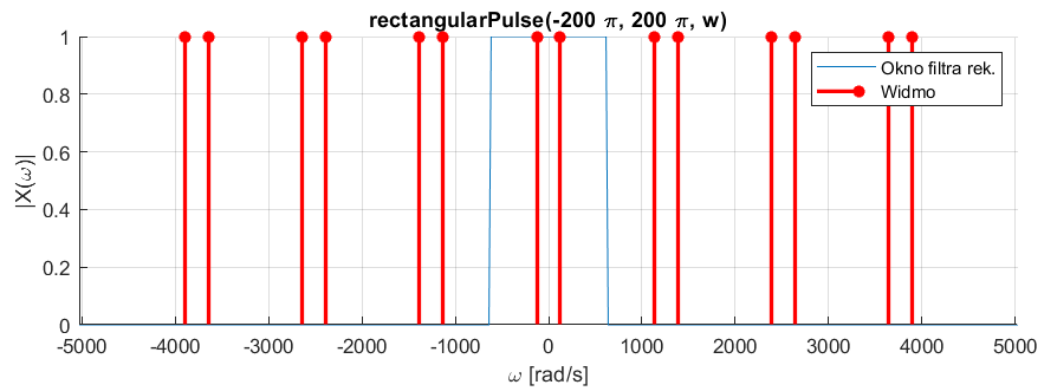
Zadanie 3.

Należy wykonać rekonstrukcję sygnałów sinusoidalnych o odpowiednich częstotliwościach:

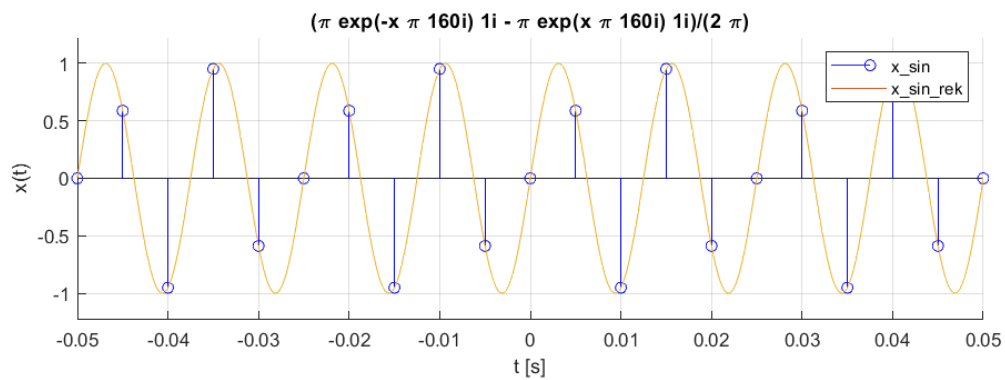
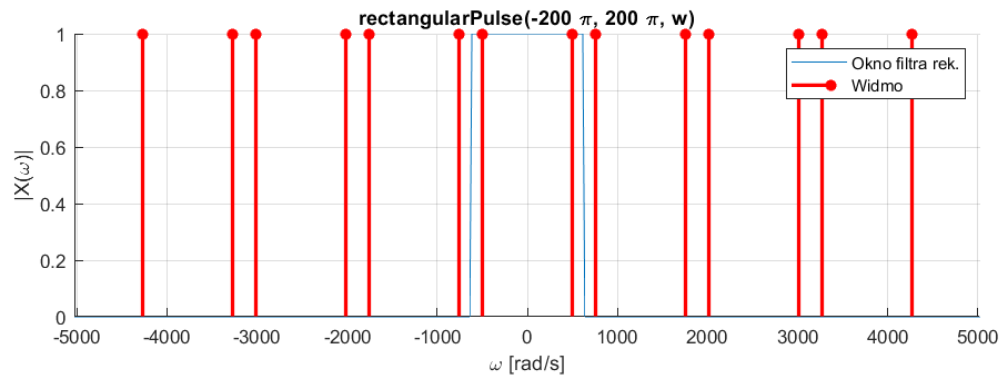
- a) $\frac{1}{5}f_g$, b) $\frac{6}{5}f_g$, c) $\frac{11}{5}f_g$, d) $\frac{16}{5}f_g$,
- e) $\frac{4}{5}f_g$, f) $\frac{9}{5}f_g$, g) $\frac{14}{5}f_g$.

W celu poprawy czytelności przeprowadzonego ćwiczenia oraz ułatwieniu dalszej analizy, częstotliwości zostaną posortowane od najmniejszej do największej.

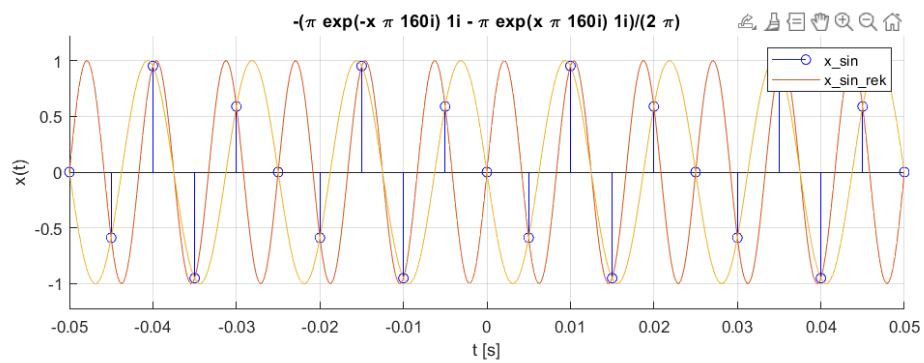
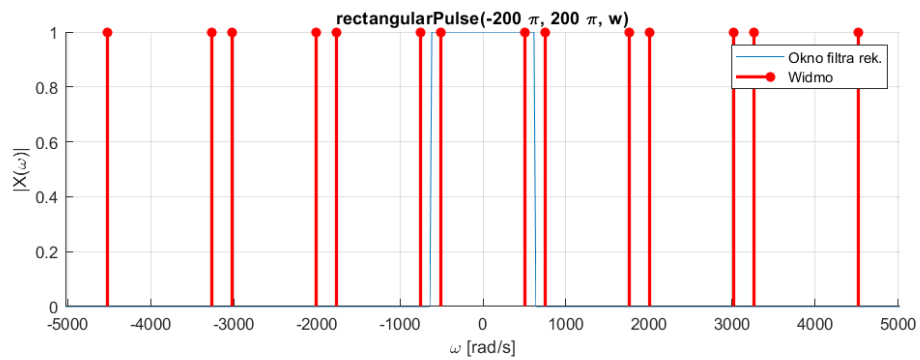
$$\frac{1}{5}f_g$$



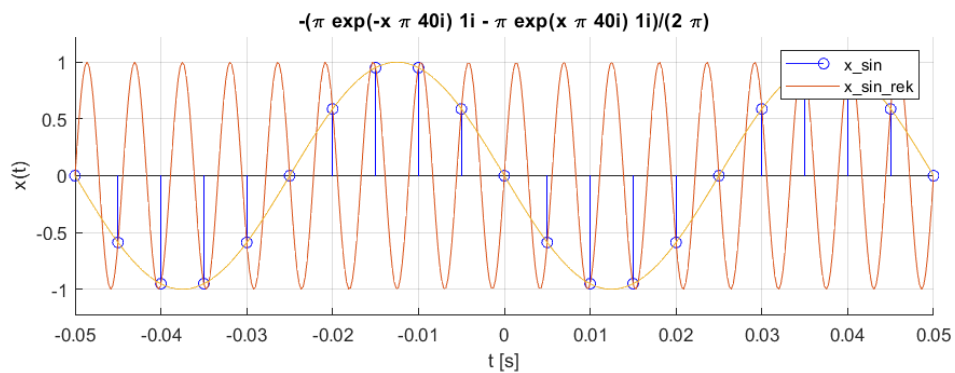
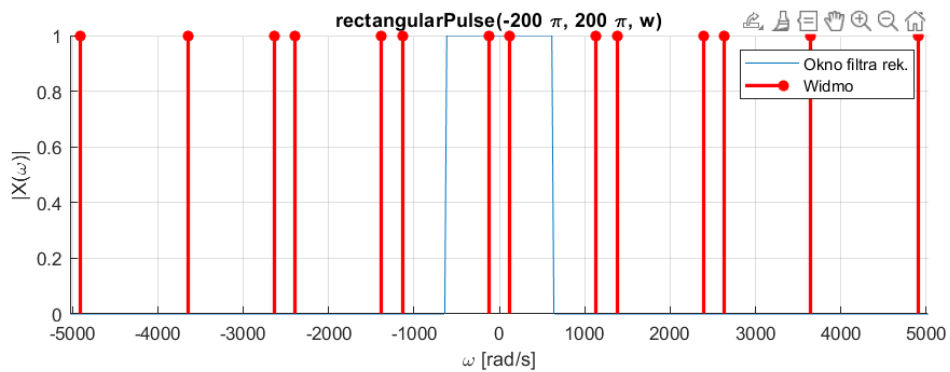
$$\frac{4}{5}f_g$$



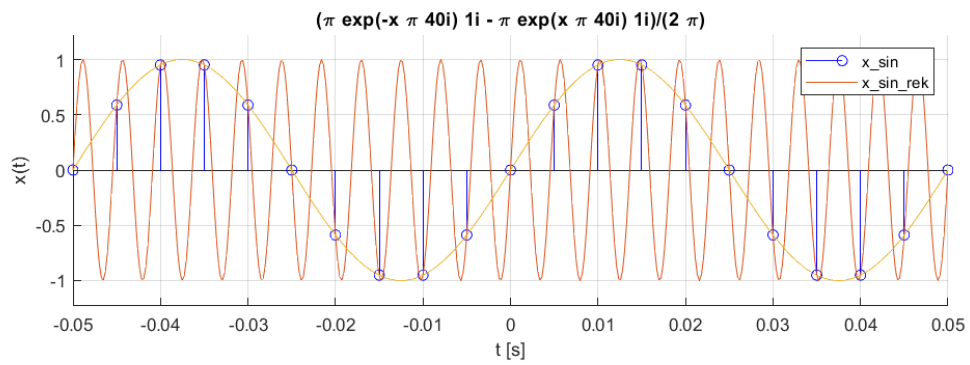
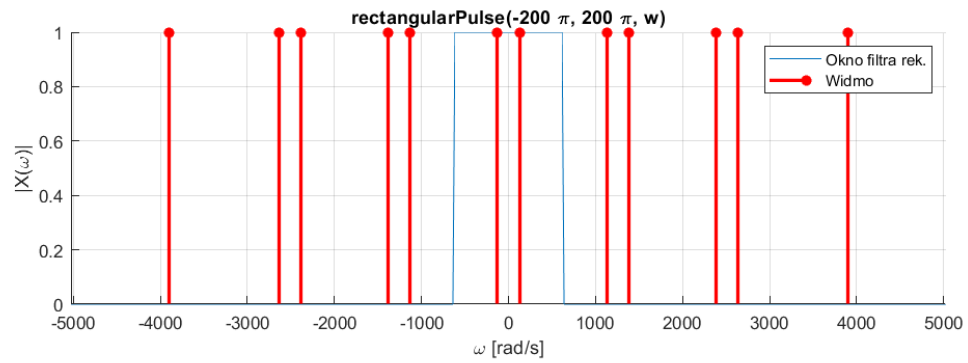
$$\frac{6}{5}f_g$$



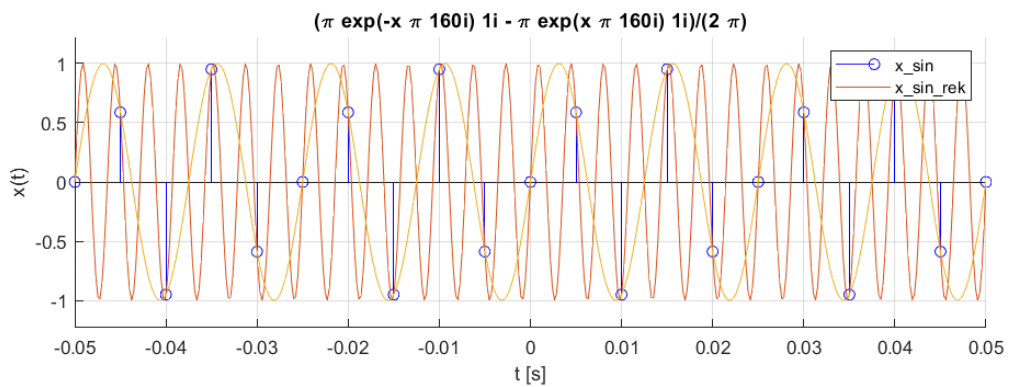
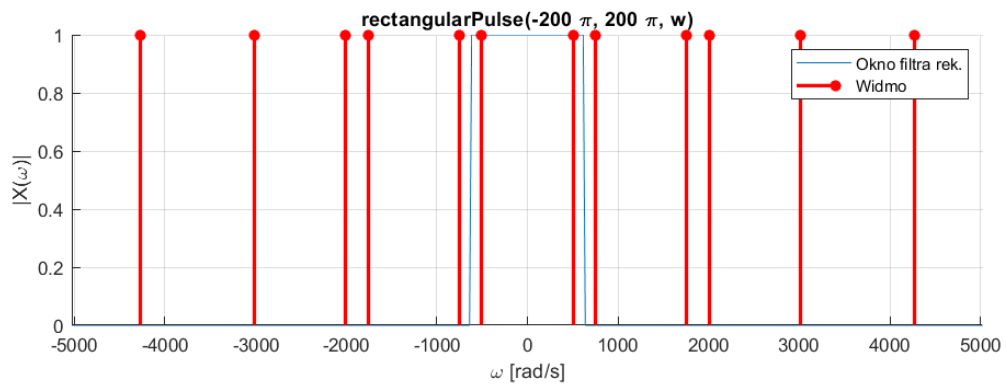
$$\frac{9}{5}f_g$$



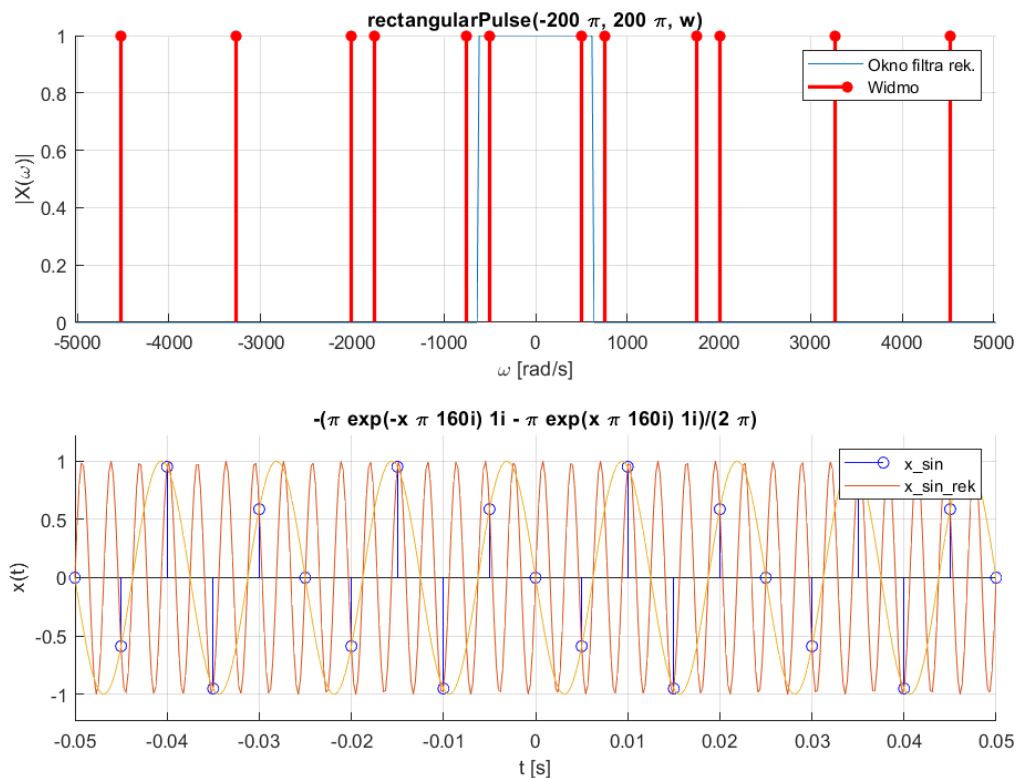
$$\frac{11}{5} f_g$$



$$\frac{14}{5} f_g$$



$$\frac{16}{5}f_g$$



Dla częstotliwości próbkowania mniejszej od dwukrotności częstotliwości sygnału próbkowanego, zrekonstruowany sygnał jest o częstotliwości zmniejszonej względem próbkowanego. Pomimo tego sygnały przechodzą przez wszystkie węzły próbkowania.

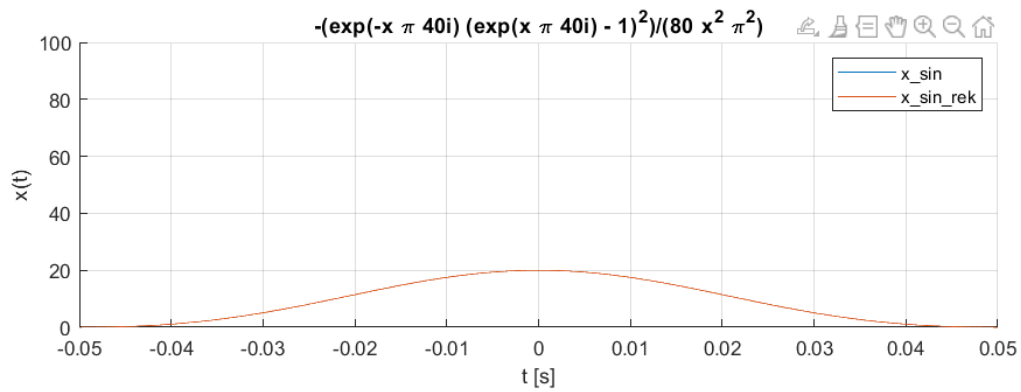
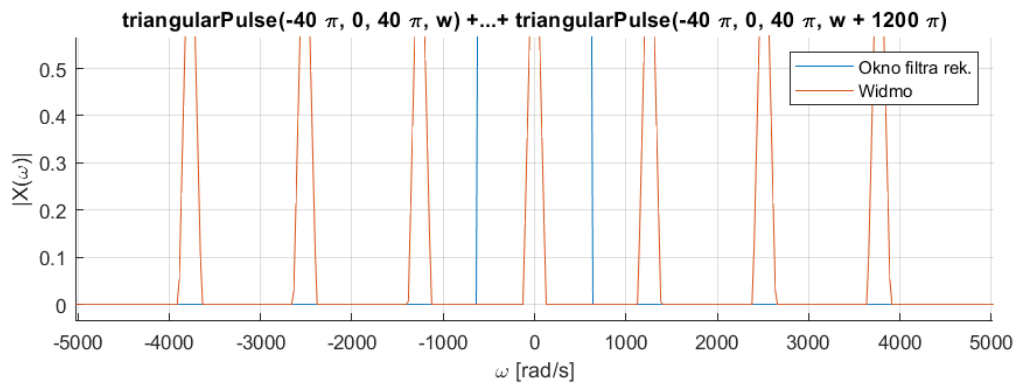
Zadanie 4.

Należy zastąpić widmo sygnału sinusoidalnego symetrycznym widmem o kształcie trójkątnym oraz przeprowadzić analizę jak w zadaniu 3.

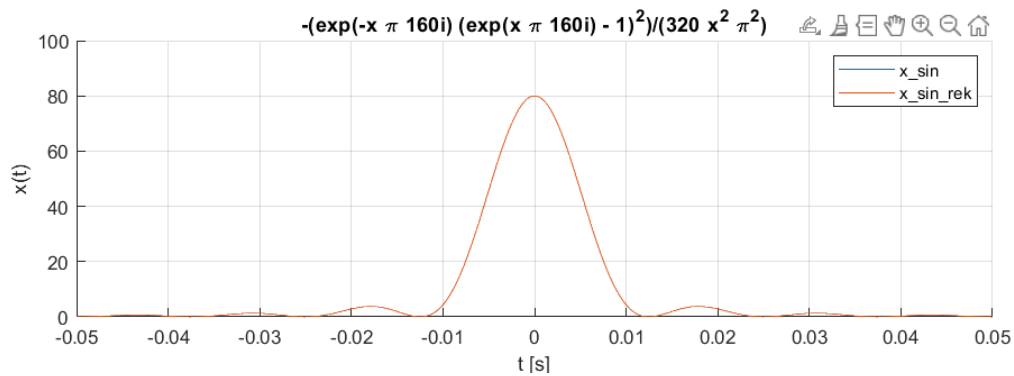
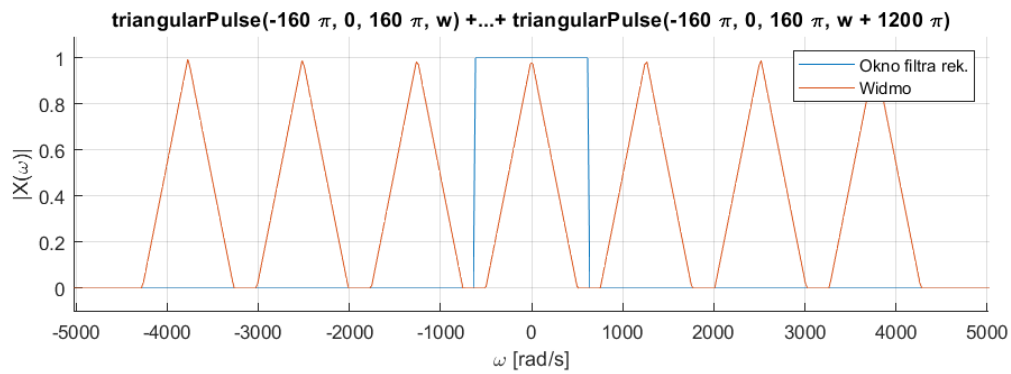
Kod został zmieniony aby być przystosowanym na wyświetlanie widma sygnału trójkątnego. Główną zmianą była podmiana funkcji sin na odwrotną transformatę Fouriera sygnału trójkątnego. Z powodu użycia do tego zadania kodu do wyświetlania widma funkcji sinusoidalnej, zostało odpuszczone zmienianie nazewnictwa zmiennych.

```
x_sin = ifourier(triangularPulse(-ws, ws, w));
X_FT_sin_org = triangularPulse(-ws, ws, w);
```

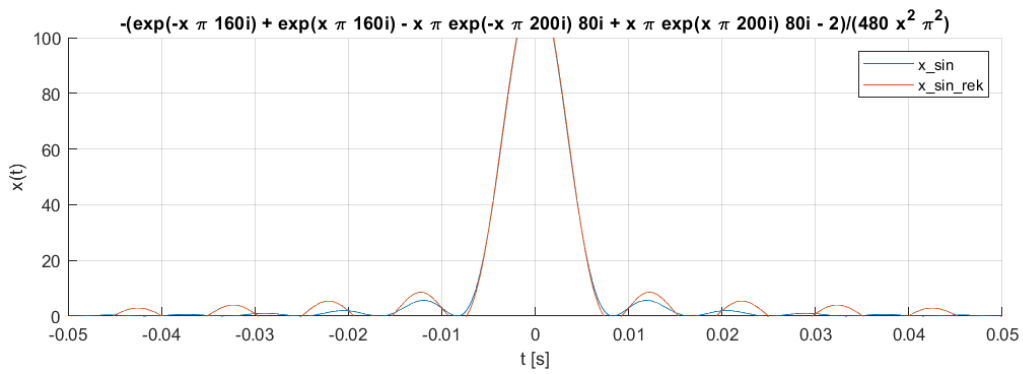
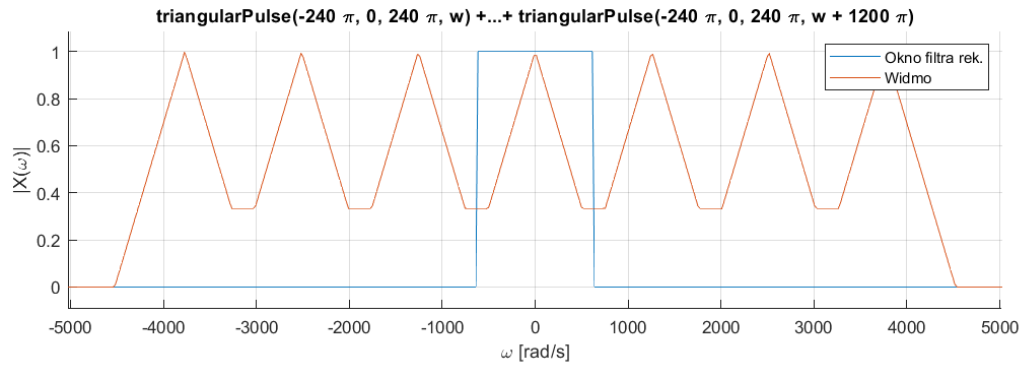
$$\frac{1}{5} f_g$$



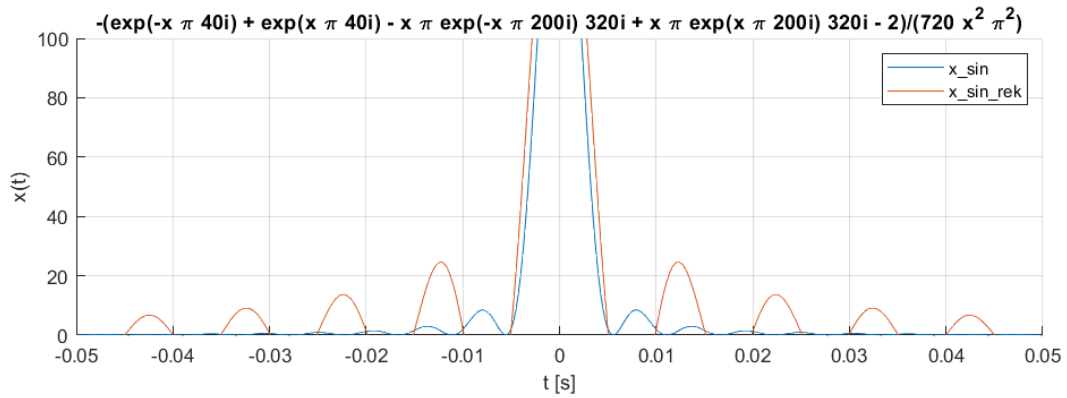
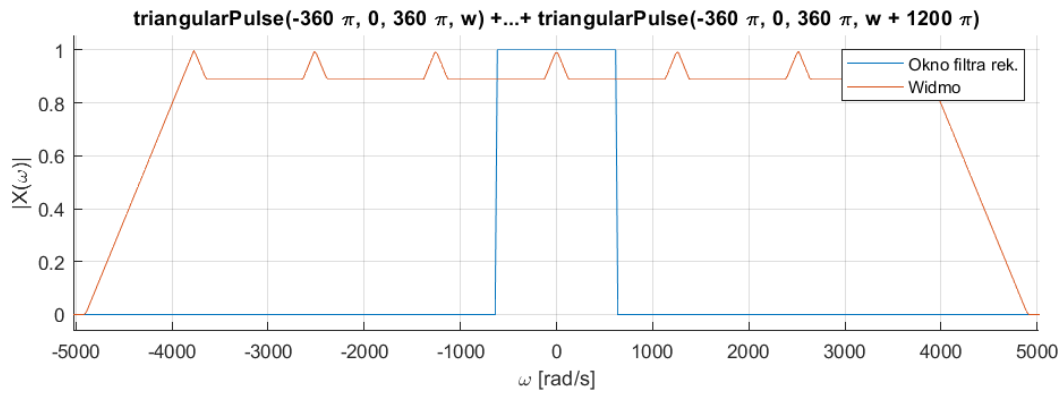
$$\frac{4}{5} f_g$$



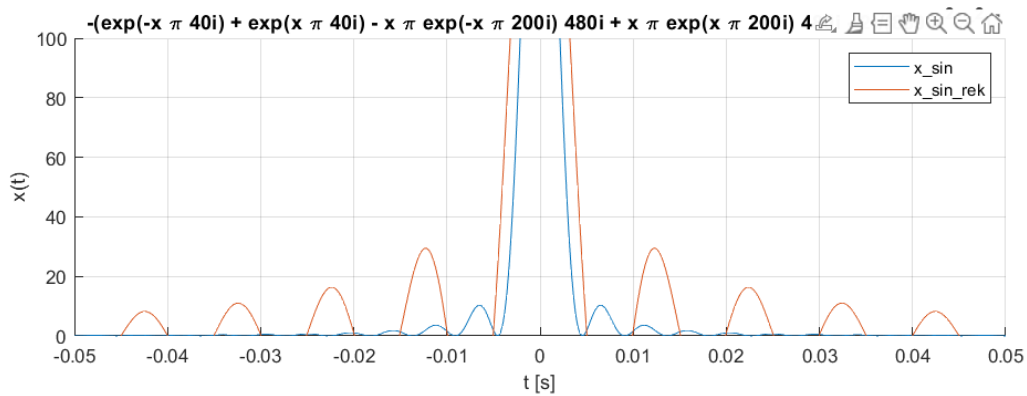
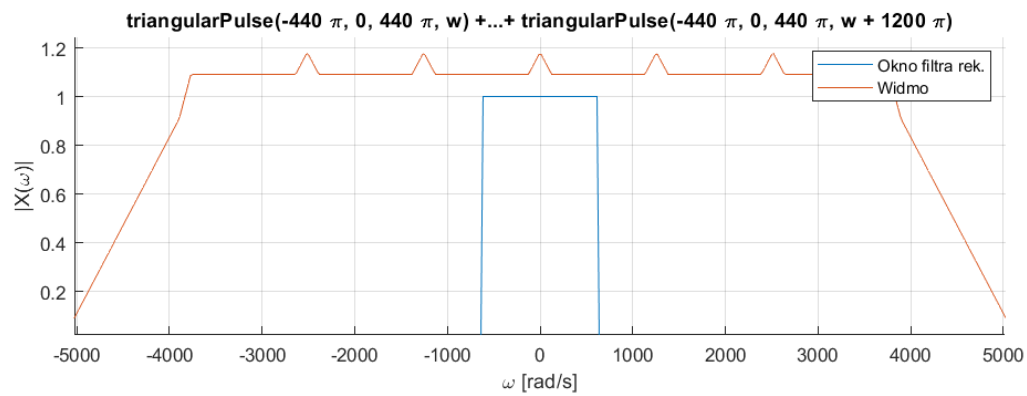
$$\frac{6}{5}f_g$$



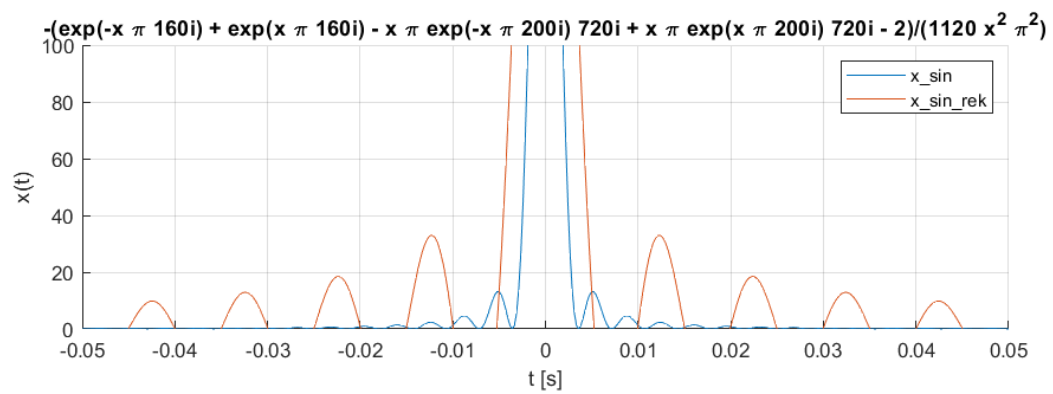
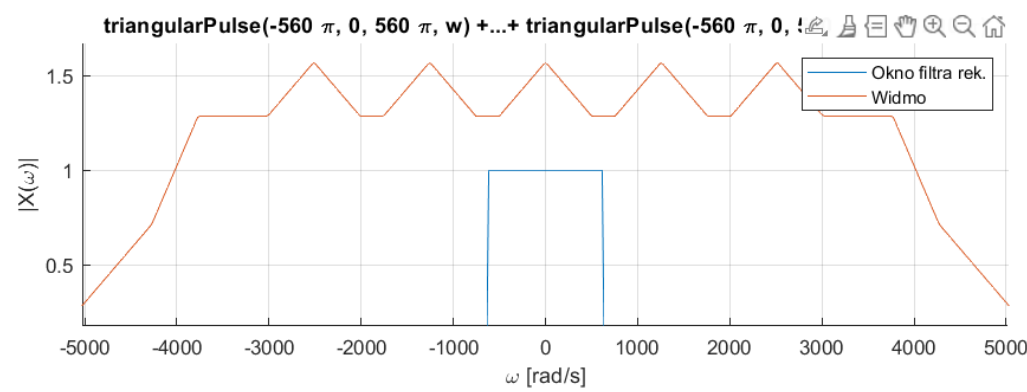
$$\frac{1}{5}f_g$$



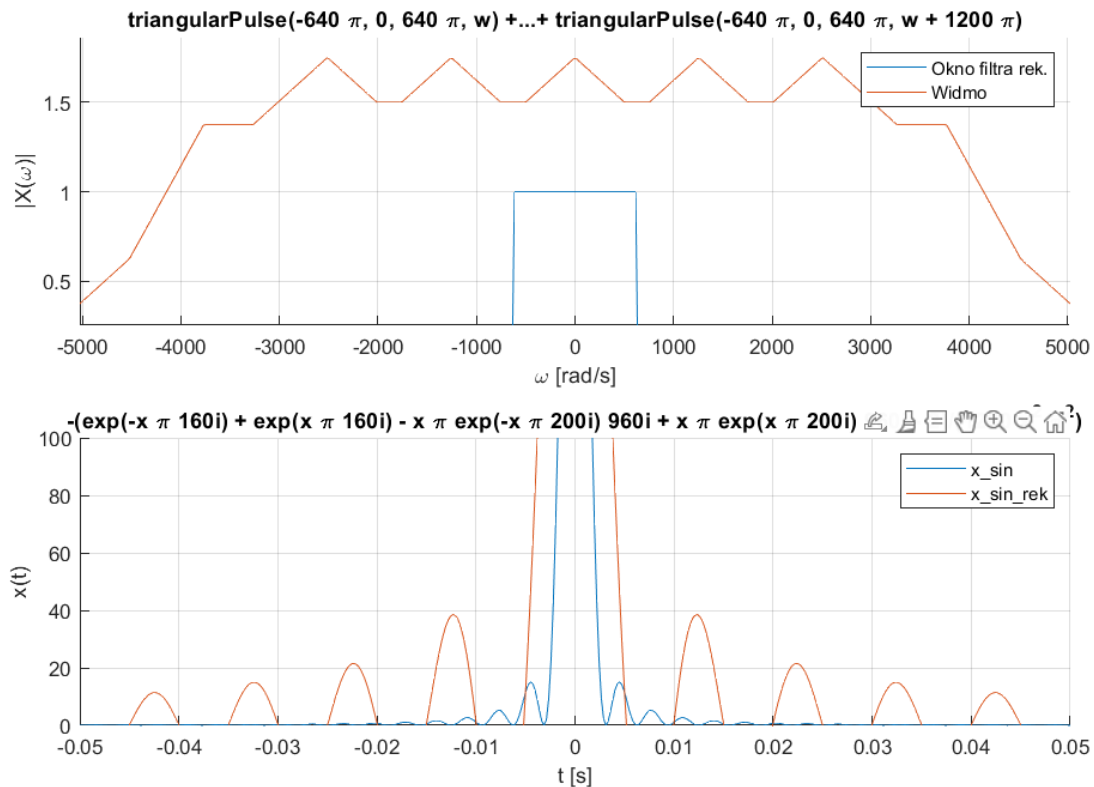
$$\frac{11}{5}f_g$$



$$\frac{14}{5}f_g$$



$$\frac{16}{5}f_g$$



Zadanie 5.

Pomimo wszelkich prób i starań nie udało się napisać mi działającego m-kodu do zadania 5.

Problem pojawia się w następującej liniice:

```
figure; subplot(2,1,1); hold on; grid on;
ezplot(FILT_FT,BND_w); %okno filtru rek.
ezplot(X_FT_org,BND_w)
```



I jest co najmniej nieczytelny:

```
Error using inlineeval (line 13)
Error in inline expression ==> 2.*pi.*(20.*dirac(w -
40.*pi).*int(exp(-t.*pi.*40i).*rectangularPulse(0, t./2, t), t,
-1./40, 1./40) + 20.*dirac(w +
40.*pi).*int(exp(t.*pi.*40i).*rectangularPulse(0, t./2, t), t,
-1./40, 1./40) + 20.*dirac(w - 80.*pi).*int(exp(-
t.*pi.*80i).*rectangularPulse(0, t./2, t), t, -1./40, 1./40) +
20.*dirac(w + 80.*pi).*int(exp(t.*pi.*80i).*rectangularPulse(0,
t./2, t), t, -1./40, 1./40) + 20.*dirac(w - 120.*pi).*int(exp(-
t.*pi.*120i).*rectangularPulse(0, t./2, t), t, -1./40, 1./40) +
20.*dirac(w +
120.*pi).*int(exp(t.*pi.*120i).*rectangularPulse(0, t./2, t), t,
-1./40, 1./40) + 20.*dirac(w - 160.*pi).*int(exp(-
t.*pi.*160i).*rectangularPulse(0, t./2, t), t, -1./40, 1./40) +
20.*dirac(w +
160.*pi).*int(exp(t.*pi.*160i).*rectangularPulse(0, t./2, t), t,
-1./40, 1./40) + 20.*dirac(w - 200.*pi).*int(exp(-
t.*pi.*200i).*rectangularPulse(0, t./2, t), t, -1./40, 1./40) +
```