

SPRAWOZDANIE 10

Zajęcia nr 11 - Parametry sygnałów cyfrowych

Michał Midor, gr. 5 - 9.12.2025, godz: 15:00

WSTĘP

Filtracja cyfrowa to proces przetwarzania sygnału w dziedzinie czasu, którego celem jest modyfikacja jego widma częstotliwościowego. W praktyce inżynierskiej proces ten służy do ekstrakcji użytecznych informacji poprzez redukcję niepożądanych składowych (szumu) lub separację określonych pasm częstotliwości. Z matematycznego punktu widzenia, filtracja cyfrowa realizowana jest poprzez operację splotu sygnału wejściowego z odpowiedzią impulsową filtru. Dzięki temu możliwe jest nie tylko poprawienie jakości sygnału, ale również rekonstrukcja jego uszkodzonych fragmentów.

Filtry o Skończonej Odpowiedzi Impulsowej (FIR) są klasą układów nierekursywnych, co oznacza, że wyjście zależy wyłącznie od próbek wejściowych (bieżących i przeszłych), co gwarantuje ich bezwarunkową stabilność. Kluczową cechą filtrów FIR jest możliwość uzyskania liniowej charakterystyki fazowej, co zapobiega zniekształceniom fazowym sygnału i umożliwia jednakowe opóźnienie wszystkich składowych. Projektowanie tych filtrów polega na doborze odpowiednich współczynników (wag), które determinują charakterystykę częstotliwościową. Aby zminimalizować zafalowania w paśmie przepustowym i zaporowym (Listki boczne widoczne na wykresach widma), stosuje się technikę okienkowania (np. okna Hamminga, Hann, Czebyszewa czy Tukeya). Wyższy rząd filtra pozwala uzyskać bardziej strome zbocza charakterystyki, jednak odbywa się to kosztem zwiększonej złożoności obliczeniowej.

WYKONANIE ĆWICZEŃ

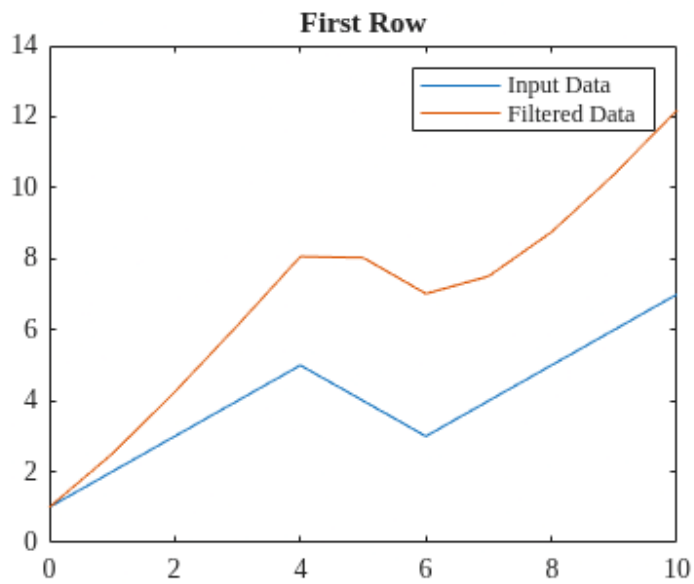
1)

```
display("Zadanie 1")
```

"Zadanie 1"

```
x=[1, 2, 3, 4, 5, 4, 3, 4, 5, 6, 7];
b = [1];
a = [1, -0.5];
y = filter(b, a, x, [], 2); % 2wymiarowy
t = 0:length(x)-1;

figure();
plot(t,x(1,:))
hold on
plot(t,y(1,:))
legend('Input Data','Filtered Data')
title('First Row')
hold off;
```



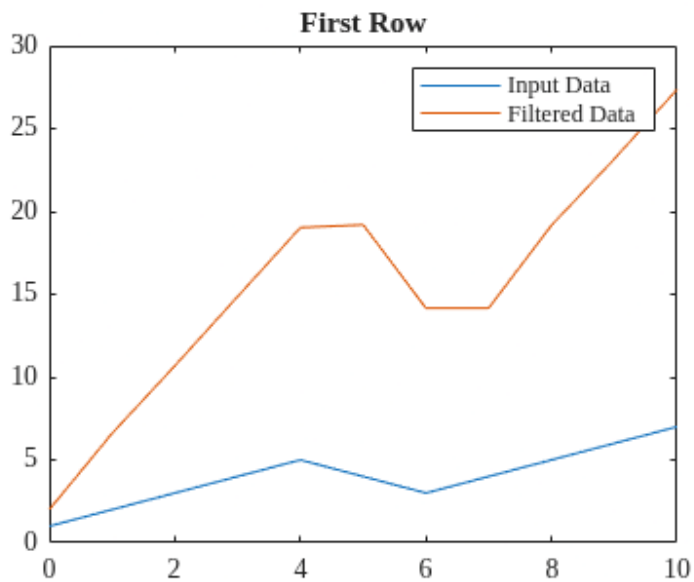
2)

```
display("Zadanie 2")
```

"Zadanie 2"

```
b = [2, 3];
a = [1, 0.2];
y = filter(b, a, x, [], 2); % 2 wymiary

figure();
plot(t,x(1,:))
hold on
plot(t,y(1,:))
legend('Input Data','Filtered Data')
title('First Row')
hold off;
```



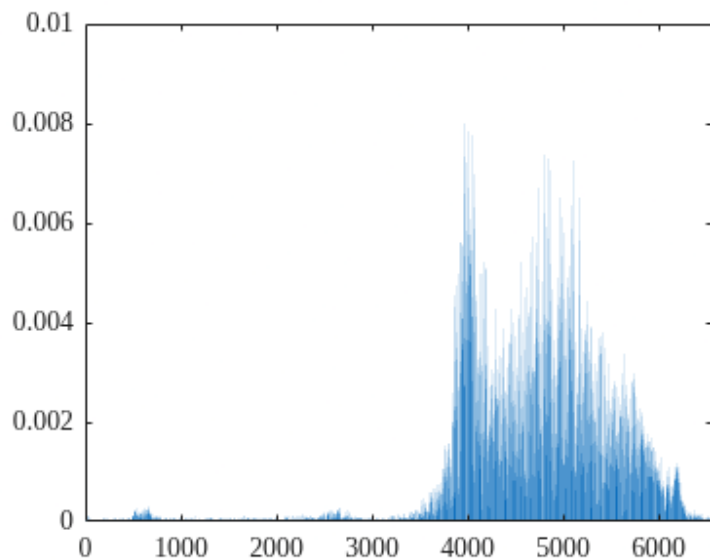
3)

```
display("Zadanie 3")
```

"Zadanie 3"

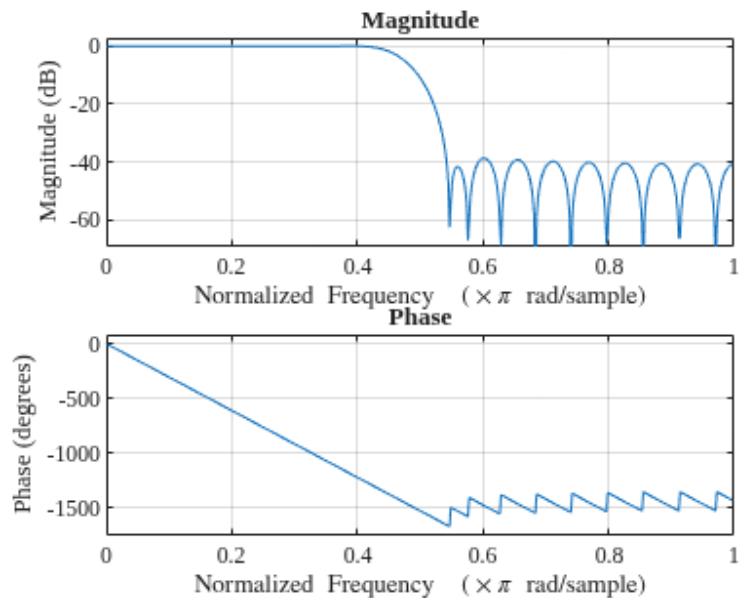
```
load chirp % daje y oraz Fs
t = (0:length(y)-1)/Fs;

xfft=abs(fft(y));
xfft=xfft/13129; % normalizacja - amplituda nie zależy od długości sygnału
x1=1:1:6564; % tylko pierwsza połowa widma, druga to lustrzane odbicie
bar(x1(1:6564), xfft(1:6564)); % wykres słupkowy
axis([0,6564, 0,0.01]) ;
```



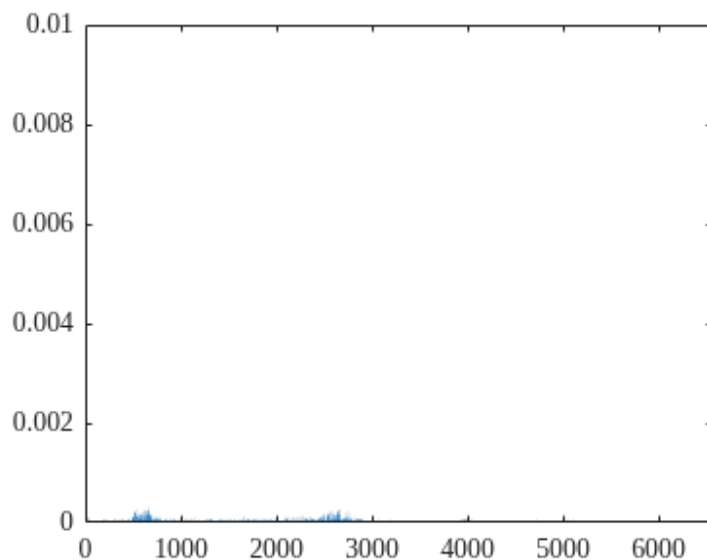
```
% projektowanie filtra
```

```
bhi = fir1(34,0.48,'low',chebwin(35,30)); % filtr dolnoprzepustowy  
freqz(bhi,1) % charakterystyka częstotliwościowa - amplituda i faza
```



```
%fvtool(bhi) % interaktywne Filter Vizualization Tool  
outhi = filter(bhi,1,y); % sygnał po filtracji
```

```
xfft=abs(fft(outhi));  
xfft=xfft/13129;  
x1=1:1:6564;  
bar(x1(1:6564), xfft(1:6564));  
axis([0,6564, 0,0.01]);
```



```

subplot(2,1,1)
plot(t,y)
title('Original Signal')
ys = ylim;

subplot(2,1,2)
plot(t,outhi)
title('Highpass Filtered Signal')
xlabel('Time (s)')
ylim(ys)

```

4)

```
display("Zadanie 4")
```

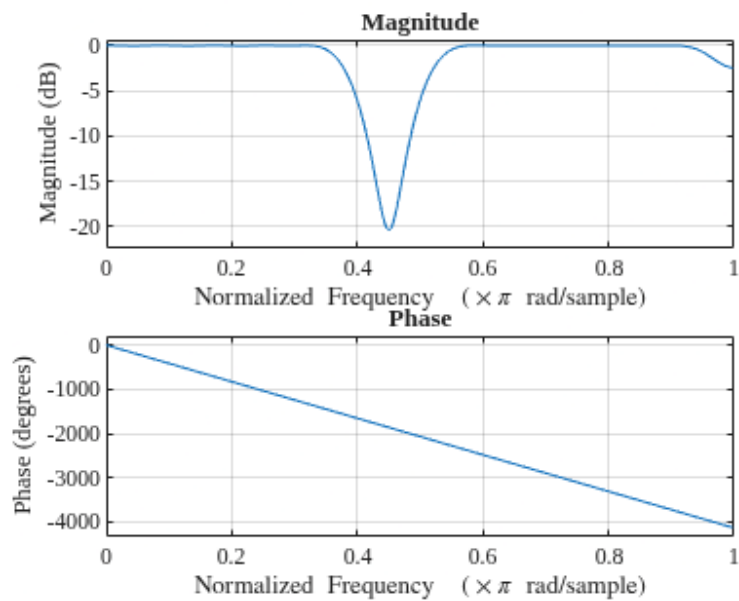
"Zadanie 4"

```

ord = 46;
low = 0.4;
bnd = [0.5 0.99];
bM = fir1(ord,[low bnd], "DC-1");

freqz(bM,1)

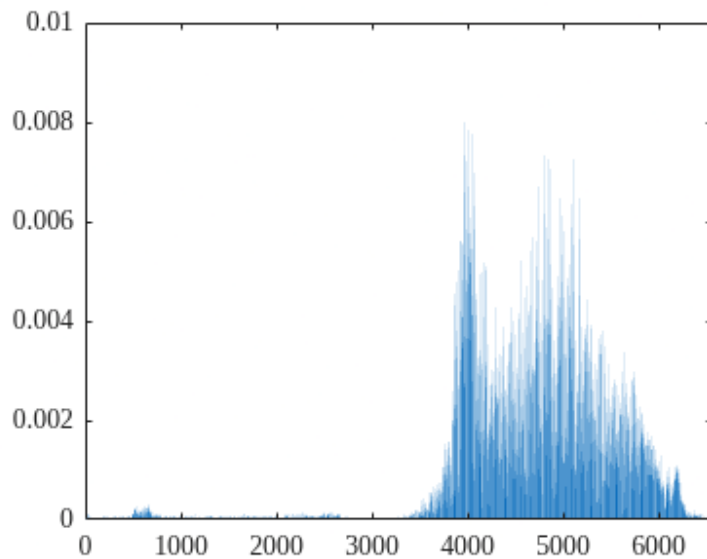
```



```

outF = filter(bM,1,y);
xfft=abs(fft(outF));
xfft=xfft/13129;
x1=1:1:6564;
bar(x1(1:6564), xfft(1:6564));
axis([0,6564, 0,0.01]) ;

```



5)

```
display("Zadanie 5")
```

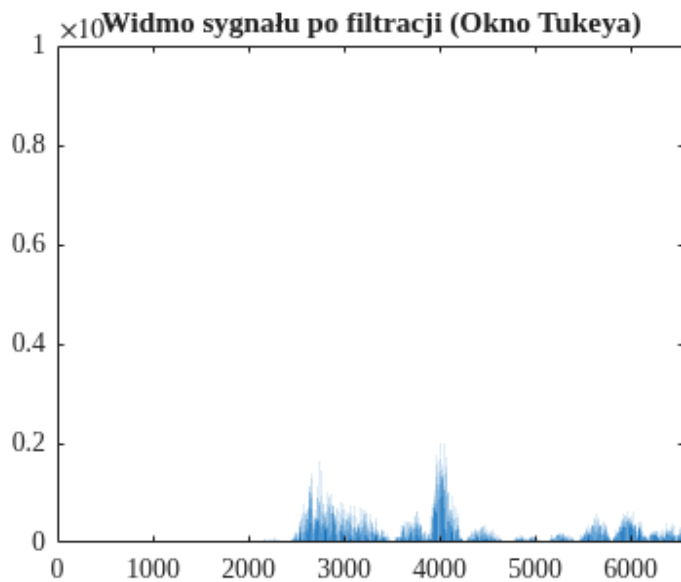
"Zadanie 5"

```
bHamming = fir1(ord, [low bnd], 'DC-0');
hTukey = fir1(ord, [low bnd], 'DC-0', tukeywin(ord+1));
hfvt = fvtool(bHamming, 1, hTukey, 1);
legend(hfvt, 'Hamming', 'Tukey');
```

```
outTukey = filter(hTukey, 1, y);
xfft=abs(fft(outTukey));
xfft=xfft/13129;display("Zadanie 6");
```

"Zadanie 6"

```
x1=1:1:6564;
bar(x1(1:6564), xfft(1:6564));
axis([0,6564, 0,0.001]) ;
title('Widmo sygnału po filtracji (Okno Tukeya)');
```



6)

```
display("Zadanie 6");
```

"Zadanie 6"

```
file_list = {
    'wiatrak_20.wav',      'FFT_filtracja_wiatrak20.txt';
    'wiatrak_21.wav',      'FFT_filtracja_wiatrak21.txt';
    'przekladnia20.wav',   'FFT_filtracja_przekladnia20.txt';
    'przekladnia21.wav',   'FFT_filtracja_przekladnia21.txt';
    'wiatrak_23.wav',      'FFT_filtracja_wiatrak23.txt';
    'przekladnia23.wav',   'FFT_filtracja_przekladnia23.txt';
    'wiatrak_24.wav',      'FFT_filtracja_wiatrak24.txt';
    'przekladnia24.wav',   'FFT_filtracja_przekladnia24.txt'
};

N_len = 44100;
F_max = N_len/2;
IND_START = 501;
IND_STOP = 1001;

for i = 1:size(file_list, 1)
    input_wav = file_list{i,1};
    output_txt_base = file_list{i, 2};

    % 1. Wczytanie pliku
    try
        [x2, Fs] = audioread(input_wav);
    catch
        warning('Nie znaleziono pliku: %s. Sprawdź ścieżkę!', full_path);
        continue;
    end
end
```

```

end

% 2. Normalizacja [-1, 1]
max_val = max(abs(x2));
if max_val > 0
    x2 = x2 / max_val;
end

% 3. FFT, wspólne dla zad 6 i zad 7
raw_fft = abs(fft(x2, N_len)) / Fs; % Surowe widmo

% Zad 6 (500Hz - 1000Hz)
xfft_6 = raw_fft;
xfft_6(1:IND_START-1) = 0;
xfft_6(IND_STOP + 1 : N_len) = 0;

% Zapis do pliku
out_name_6 = output_txt_base;
fid = fopen(out_name_6, 'w+t', 'n');
fprintf(fid, '%f\n', xfft_6(1:22050));
fclose(fid);
end

% Próbkki uczące
load FFT_filtracja_wiatrak20.txt
load FFT_filtracja_wiatrak21.txt
load FFT_filtracja_przekladnia20.txt
load FFT_filtracja_przekladnia21.txt

% Próbkki testowe
load FFT_filtracja_wiatrak23.txt
load FFT_filtracja_wiatrak24.txt
load FFT_filtracja_przekladnia23.txt
load FFT_filtracja_przekladnia24.txt

% Obliczenie odległości

% Testowa probka wiatrak_23
D1 = sum(abs(FFT_filtracja_wiatrak23 - FFT_filtracja_wiatrak20));
D2 = sum(abs(FFT_filtracja_wiatrak23 - FFT_filtracja_wiatrak21));
D3 = sum(abs(FFT_filtracja_wiatrak23 - FFT_filtracja_przekladnia20));
D4 = sum(abs(FFT_filtracja_wiatrak23 - FFT_filtracja_przekladnia21));

display("Odległości dla FFT_filtracja_wiatrak23 (500-1000Hz):");

```

```
"Odległości dla FFT_filtracja_wiatrak23 (500-1000Hz):"
```

```
display(['D (wiatrak20): ', num2str(D1)]);
```

```
D (wiatrak20): 0.60459
```



```
display(['D (wiatrak21): ', num2str(D2)]);
```

```
D (wiatrak21): 0.59571
```

```
display(['D (przekladnia20): ', num2str(D3)]);
```

```
D (przekladnia20): 0.63487
```

```
display(['D (przekladnia21): ', num2str(D4)]);
```

```
D (przekladnia21): 0.61346
```

```
% Testowa probka wiatrak_24
```

```
D5 = sum(abs(FFT_filtracja_wiatrak24 - FFT_filtracja_wiatrak20));
```

```
D6 = sum(abs(FFT_filtracja_wiatrak24 - FFT_filtracja_wiatrak21));
```

```
D7 = sum(abs(FFT_filtracja_wiatrak24 - FFT_filtracja_przekladnia20));
```

```
D8 = sum(abs(FFT_filtracja_wiatrak24 - FFT_filtracja_przekladnia21));
```

```
display("Odległości dla FFT_filtracja_wiatrak24 (500-1000Hz):");
```

```
"Odległości dla FFT_filtracja_wiatrak24 (500-1000Hz):"
```

```
display(['D (wiatrak20): ', num2str(D5)]);
```

```
D (wiatrak20): 0.69764
```

```
display(['D (wiatrak21): ', num2str(D6)]);
```

```
D (wiatrak21): 0.57607
```

```
display(['D (przekladnia20): ', num2str(D7)]);
```

```
D (przekladnia20): 0.65681
```

```
display(['D (przekladnia21): ', num2str(D8)]);
```

```
D (przekladnia21): 0.65608
```

```
% Testowa probka przekladnia_23
```

```
D9 = sum(abs(FFT_filtracja_przekladnia23 - FFT_filtracja_wiatrak20));
```

```
D10 = sum(abs(FFT_filtracja_przekladnia23 - FFT_filtracja_wiatrak21));
```

```
D11 = sum(abs(FFT_filtracja_przekladnia23 - FFT_filtracja_przekladnia20));
```

```
D12 = sum(abs(FFT_filtracja_przekladnia23 - FFT_filtracja_przekladnia21));
```

```
display("Odległości dla FFT_filtracja_przekladnia23 (500-1000Hz):");
```

```
"Odległości dla FFT_filtracja_przekladnia23 (500-1000Hz):"
```

```
display(['D (wiatrak20): ', num2str(D9)]);
```

```
D (wiatrak20): 0.63089
```

```
display(['D (wiatrak21): ', num2str(D10)]);
```

```
D (wiatrak21): 0.65249
```

```
display(['D (przekladnia20): ', num2str(D11)]);
```

```
D (przekladnia20): 0.47557
```

```
display(['D (przekladnia21): ', num2str(D12)]);
```

```
D (przekladnia21): 0.44115
```

```
% Testowa probka przekladnia_24
```

```
D13 = sum(abs(FFT_filtracja_przekladnia24 - FFT_filtracja_wiatrak20));
```

```
D14 = sum(abs(FFT_filtracja_przekladnia24 - FFT_filtracja_wiatrak21));
```

```
D15 = sum(abs(FFT_filtracja_przekladnia24 - FFT_filtracja_przekladnia20));
```

```
D16 = sum(abs(FFT_filtracja_przekladnia24 - FFT_filtracja_przekladnia21));
```

```
display("Odległości dla FFT_filtracja_przekladnia24 (500-1000Hz):");
```

```
"Odległości dla FFT_filtracja_przekladnia24 (500-1000Hz):"
```

```
display(['D (wiatrak20): ', num2str(D13)]);
```

```
D (wiatrak20): 0.66112
```

```
display(['D (wiatrak21): ', num2str(D14)]);
```

```
D (wiatrak21): 0.61636
```

```
display(['D (przekladnia20): ', num2str(D15)]);
```

```
D (przekladnia20): 0.44108
```

```
display(['D (przekladnia21): ', num2str(D16)]);
```

```
D (przekladnia21): 0.48382
```

7)

```
display("Zadanie 7");
```

```
"Zadanie 7"
```

```
IND_START = 101;
```

```
IND_STOP = 1501;
```

```
for i = 1:size(file_list, 1)
```

```
    input_wav = file_list{i,1};
```

```
    output_txt_base = file_list{i, 2};
```

```
    % 1. Wczytanie pliku
```

```
    try
```

```
        [x2, Fs] = audioread(input_wav);
```

```
    catch
```

```

        warning('Nie znaleziono pliku: %s. Sprawdź ścieżkę!', full_path);
        continue;
    end

    % 2. Normalizacja [-1, 1]
    max_val = max(abs(x2));
    if max_val > 0
        x2 = x2 / max_val;
    end

    % 3. FFT, wspólne dla zad 6 i zad 7
    raw_fft = abs(fft(x2, N_len)) / Fs; % Surowe widmo

    % Zad 7 (100Hz - 1500Hz)
    xfft_6 = raw_fft;
    xfft_6(1:IND_START-1) = 0;
    xfft_6(IND_STOP + 1 : N_len) = 0;

    % Zapis do pliku
    out_name_7 = output_txt_base;
    fid = fopen(out_name_7, 'w+t', 'n');
    fprintf(fid, '%f\n', xfft_6(1:22050));
    fclose(fid);

end

% Próbkki uczące
load FFT_filtracja_wiatrak20.txt
load FFT_filtracja_wiatrak21.txt
load FFT_filtracja_przekladnia20.txt
load FFT_filtracja_przekladnia21.txt

% Próbkki testowe
load FFT_filtracja_wiatrak23.txt
load FFT_filtracja_wiatrak24.txt
load FFT_filtracja_przekladnia23.txt
load FFT_filtracja_przekladnia24.txt

% Obliczenie odległości

% Testowa probka wiatrak_23
D1 = sum(abs(FFT_filtracja_wiatrak23 - FFT_filtracja_wiatrak20));
D2 = sum(abs(FFT_filtracja_wiatrak23 - FFT_filtracja_wiatrak21));
D3 = sum(abs(FFT_filtracja_wiatrak23 - FFT_filtracja_przekladnia20));
D4 = sum(abs(FFT_filtracja_wiatrak23 - FFT_filtracja_przekladnia21));

display("Odległości dla FFT_filtracja_wiatrak23 (500-1000Hz):");

```

"Odległości dla FFT_filtracja_wiatrak23 (500-1000Hz):"

```
display(['D (wiatrak20): ', num2str(D1)]);
```

```
D (wiatrak20): 1.6371
```

```
display(['D (wiatrak21): ', num2str(D2)]);
```

```
D (wiatrak21): 1.5565
```

```
display(['D (przekladnia20): ', num2str(D3)]);
```

```
D (przekladnia20): 1.8099
```

```
display(['D (przekladnia21): ', num2str(D4)]);
```

```
D (przekladnia21): 1.7837
```

```
% Testowa probka wiatrak_24
```

```
D5 = sum(abs(FFT_filtracja_wiatrak24 - FFT_filtracja_wiatrak20));
```

```
D6 = sum(abs(FFT_filtracja_wiatrak24 - FFT_filtracja_wiatrak21));
```

```
D7 = sum(abs(FFT_filtracja_wiatrak24 - FFT_filtracja_przekladnia20));
```

```
D8 = sum(abs(FFT_filtracja_wiatrak24 - FFT_filtracja_przekladnia21));
```

```
display("Odległości dla FFT_filtracja_wiatrak24 (500-1000Hz):");
```

```
"Odległości dla FFT_filtracja_wiatrak24 (500-1000Hz):"
```

```
display(['D (wiatrak20): ', num2str(D5)]);
```

```
D (wiatrak20): 1.8764
```

```
display(['D (wiatrak21): ', num2str(D6)]);
```

```
D (wiatrak21): 1.4932
```

```
display(['D (przekladnia20): ', num2str(D7)]);
```

```
D (przekladnia20): 1.8075
```

```
display(['D (przekladnia21): ', num2str(D8)]);
```

```
D (przekladnia21): 1.8049
```

```
% Testowa probka przekladnia_23
```

```
D9 = sum(abs(FFT_filtracja_przekladnia23 - FFT_filtracja_wiatrak20));
```

```
D10 = sum(abs(FFT_filtracja_przekladnia23 - FFT_filtracja_wiatrak21));
```

```
D11 = sum(abs(FFT_filtracja_przekladnia23 - FFT_filtracja_przekladnia20));
```

```
D12 = sum(abs(FFT_filtracja_przekladnia23 - FFT_filtracja_przekladnia21));
```

```
display("Odległości dla FFT_filtracja_przekladnia23 (500-1000Hz):");
```

```
"Odległości dla FFT_filtracja_przekladnia23 (500-1000Hz):"
```

```
display(['D (wiatrak20): ', num2str(D9)]);
```

D (wiatrak20): 1.8032

```
display(['D (wiatrak21): ', num2str(D10)]);
```

D (wiatrak21): 1.687

```
display(['D (przekladnia20): ', num2str(D11)]);
```

D (przekladnia20): 1.1435

```
display(['D (przekladnia21): ', num2str(D12)]);
```

D (przekladnia21): 1.0544

```
% Testowa probka przekladnia_24
```

```
D13 = sum(abs(FFT_filtracja_przekladnia24 - FFT_filtracja_wiatrak20));
```

```
D14 = sum(abs(FFT_filtracja_przekladnia24 - FFT_filtracja_wiatrak21));
```

```
D15 = sum(abs(FFT_filtracja_przekladnia24 - FFT_filtracja_przekladnia20));
```

```
D16 = sum(abs(FFT_filtracja_przekladnia24 - FFT_filtracja_przekladnia21));
```

```
display("Odległości dla FFT_filtracja_przekladnia24 (500-1000Hz):");
```

```
"Odległości dla FFT_filtracja_przekladnia24 (500-1000Hz):"
```

```
display(['D (wiatrak20): ', num2str(D13)]);
```

D (wiatrak20): 1.8608

```
display(['D (wiatrak21): ', num2str(D14)]);
```

D (wiatrak21): 1.6608

```
display(['D (przekladnia20): ', num2str(D15)]);
```

D (przekladnia20): 1.0781

```
display(['D (przekladnia21): ', num2str(D16)]);
```

D (przekladnia21): 1.1397

8)

```
display("Zadanie 8");
```

```
"Zadanie 8"
```

```
file_wiatrak = 'wiatrak_20.wav';
```

```
file_przekladnia = 'przekladnia20.wav';
```

```
try
```

```
[y_wiatrak, Fs_w] = audioread(file_wiatrak);
```

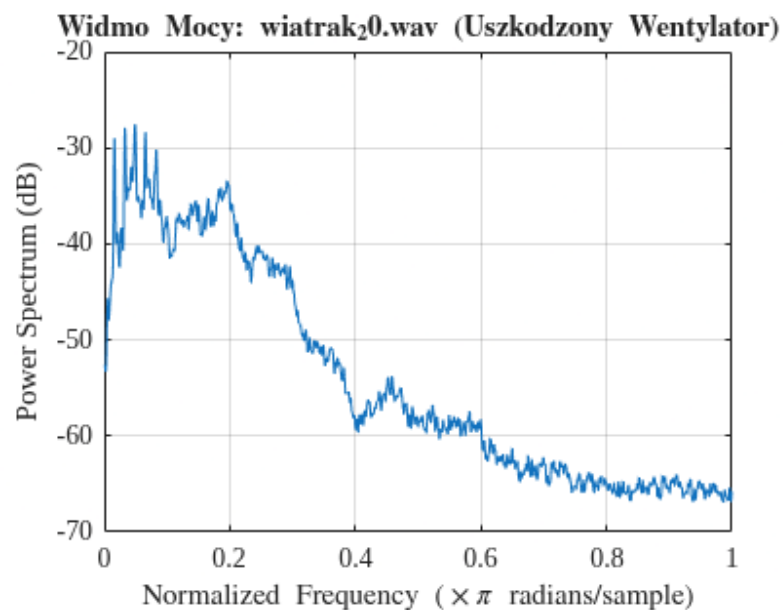
```
[y_przekladnia, Fs_p] = audioread(file_przekladnia);
```

```
catch
```

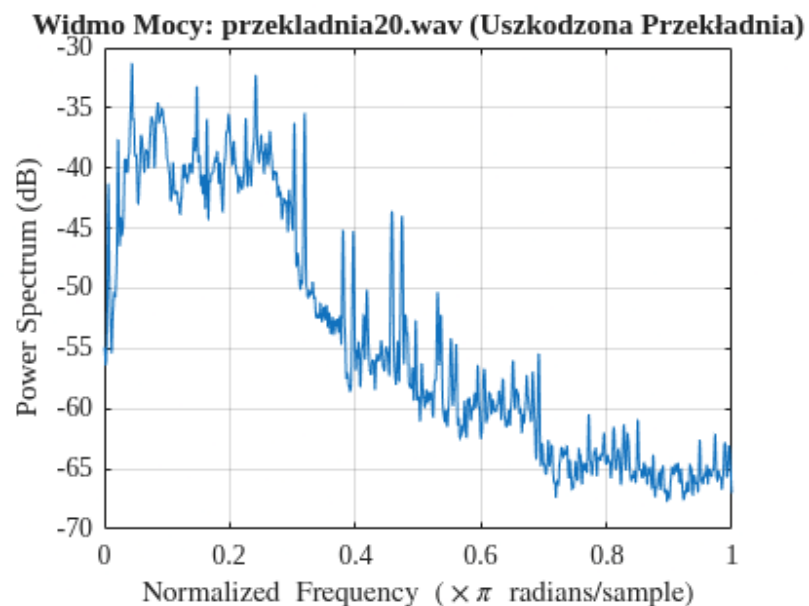
```
error('Nie znaleziono plików WAV.');
```

```
end
```

```
figure;  
pspectrum(y_wiatrak);  
title(['Widmo Mocy: ', file_wiatrak, ' (Uszkodzony Wentylator)']);
```



```
figure;  
pspectrum(y_przekladnia);  
title(['Widmo Mocy: ', file_przekladnia, ' (Uszkodzona Przekładnia)']);
```



WNIOSKI

Zastosowanie okien (np. Hamminga, Hann, Tukeya) w projektowaniu filtrów FIR pozwala na redukcję zjawiska przecieku widma i zmniejszenie listków bocznych w charakterystyce częstotliwościowej, odbywa się to jednak kosztem poszerzenia pasma przejściowego. Różne okna oferują różny kompromis na tym obszarze.

Analiza widmowa (FFT) połączona z odpowiednią filtracją pasmową stanowi skuteczne narzędzie diagnostyczne. W badaniu uszkodzeń wiertarki, zawężenie analizowanego pasma do zakresu 500–1000 Hz (Zadanie 6) pozwoliło na uzyskanie mniejszych odległości w klasyfikatorze K-NN dla próbek tej samej klasy w porównaniu do szerszego pasma 100–1500 Hz (Zadanie 7). Oznacza to, że kluczowe cechy (informacja diagnostyczna) odróżniające uszkodzenie wentylatora od przekładni znajdują się w wyselekcjonowanym, wąskim paśmie częstotliwości.

Funkcja `pspectrum` jest bardziej użyteczna w analizie sygnałów stochastycznych i zaszumionych (jak wibracje maszyn) niż proste FFT, ponieważ dzięki uśrednianiu i segmentacji pozwala lepiej zobrazować ogólny rozkład energii sygnału.