|  |  |  |  |
| --- | --- | --- | --- |
| LABORATORIUM  Cyfrowe Przetwarzanie Sygnałów | | | |
| Data  27.11.2024 | Skład podgrupy  1. Karol Cegliński  2.Przemysław Cichoń  3. | | Ćwiczenie prowadził  dr inż. Bartosz Chaber |
| Grupa | | Ocena | |

ĆWICZENIE NR 2

Badanie właściwości dyskretnej transformacji Fouriera

1. Opracowanie procedury sygnału testującego

Procedury do wykorzystania:

**linspace(x1, x2, N)** - generacja N wartości równomiernie rozłożonych pomiędzy x1 i x2.

**cos(x)** - wartość funkcji kosinus dla skalara x podanego w radianach. Gdy x jest wektorem funkcja zwraca ciąg wartości funkcji kosinus przypisanych poszczególnym jego elementom.

Polecenia:

1. Korzystając z zestawu procedur podanych powyżej napisać funkcję MATLAB-a umożliwiającą utworzenie dyskretnego przebiegu kosinusoidalnego o zadawanej pulsacji unormowanej ω0, amplitudzie a, fazie początkowej ϕ i długości N. Jako pulsację unormowaną przyjąć: ω0=2πf0Tp, gdzie f0 jest rzeczywistą częstotliwością sygnału a Tp okresem próbkowania. Funkcji nadać nazwę **sygnal** i zapamiętać ją (!).
2. Sprawdzić działanie napisanej funkcji przez zobrazowanie pojedynczego przebiegu, sumy, różnicy itp.

Przykład:

function sig=sygnal(w0,a,fi,N);

n=linspace(0,N-1,N);

sig=a\*cos(w0\*n+fi)';

plot(sig)

2. Badanie postaci prostej i odwrotnej DFT sygnału okresowego

Procedury do wykorzystania

**odwrotna(sig,ω0,a,ϕ)** - funkcja wyznaczająca sygnał sig2 na podstawie odwrotnego dyskretnego przekształcenia Fouriera widma uzyskanego na drodze DFT sygnału kosinusoidalnego sig1 o zadanej pulsacji unormowanej ω0, amplitudzie a i fazie początkowej ϕ. Wywołanie procedury z zadanymi parametrami skutkuje

* wyrysowaniem sygnału analizowanego sig1
* wyświetleniem liczby próbek przypadającej na okres sygnału (L)
* zapytaniem o długość rekordu danych (L'<L)
* zapytaniem o numer próbki początku pobierania danych (k)
* wyrysowaniem modułu i fazy DFT wyznaczonych dla wybranego rekordu
* wyrysowaniem sygnału odtworzonego na podstawie odwrotnej DFT.

Polecenia:

1. Utworzyć sygnał kosinusoidalny za pomocą procedury sygnal z następującymi parametrami wejściowymi:

N=256;

w0=(2\*pi/256)\*8;

a=1;

fi=pi/2;

sig=sygnal(w0,a,fi,N);

1. Korzystając z procedury odwrotna

**odwrotna(sig,w0,a,fi)**

zbadać wpływ długości rekordu wyciętych do analizy danych na postać prostej DFT oraz odwrotnej DFT. Należy zaobserwować przebiegi dla L' będącego całkowitą wielokrotnością okresu L=256/8, np.

L'=32, 64, 128, ...

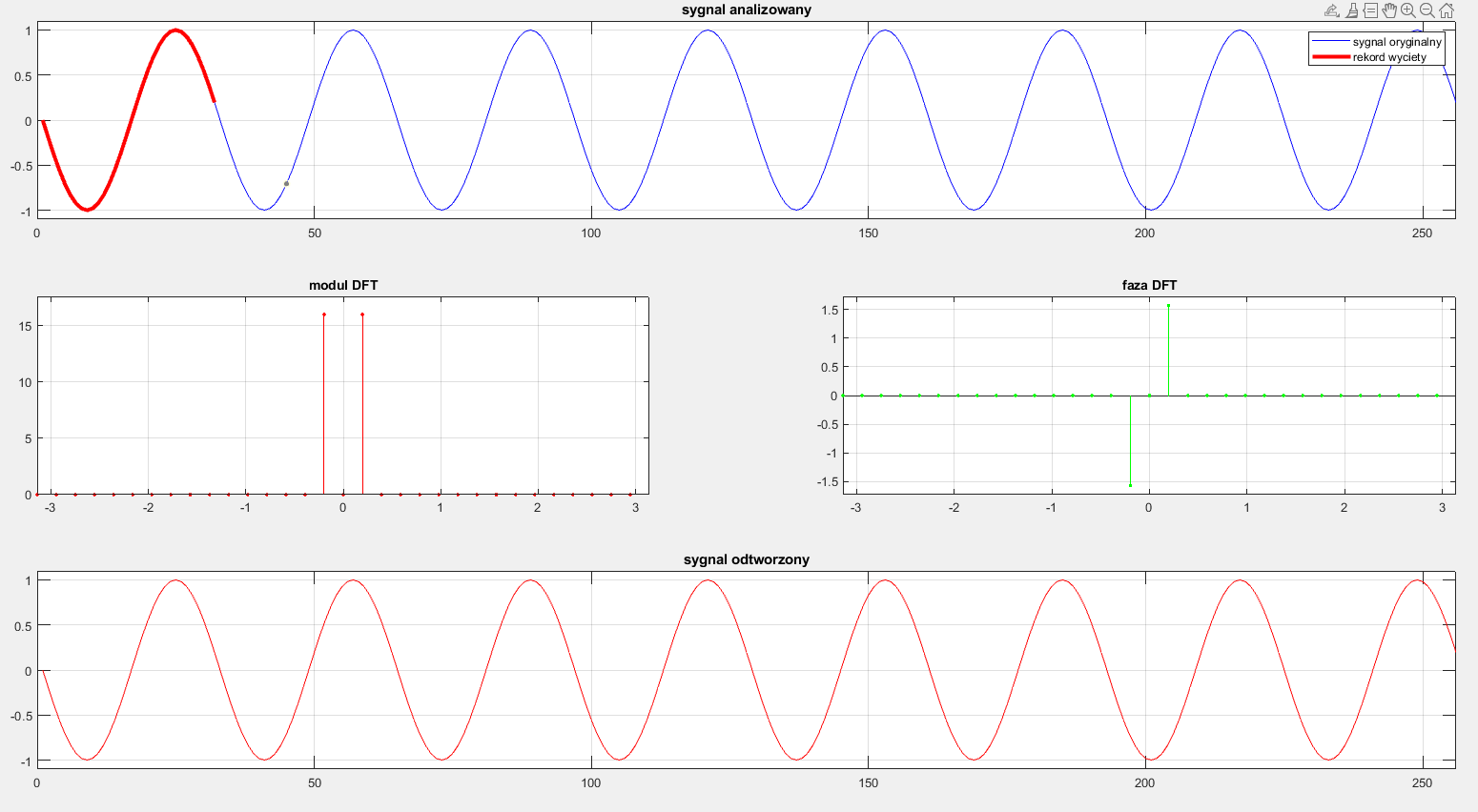
oraz dla L' nie będącego całkowitą wielokrotnością L, np.

L'=20, 56, 100, 140, ...

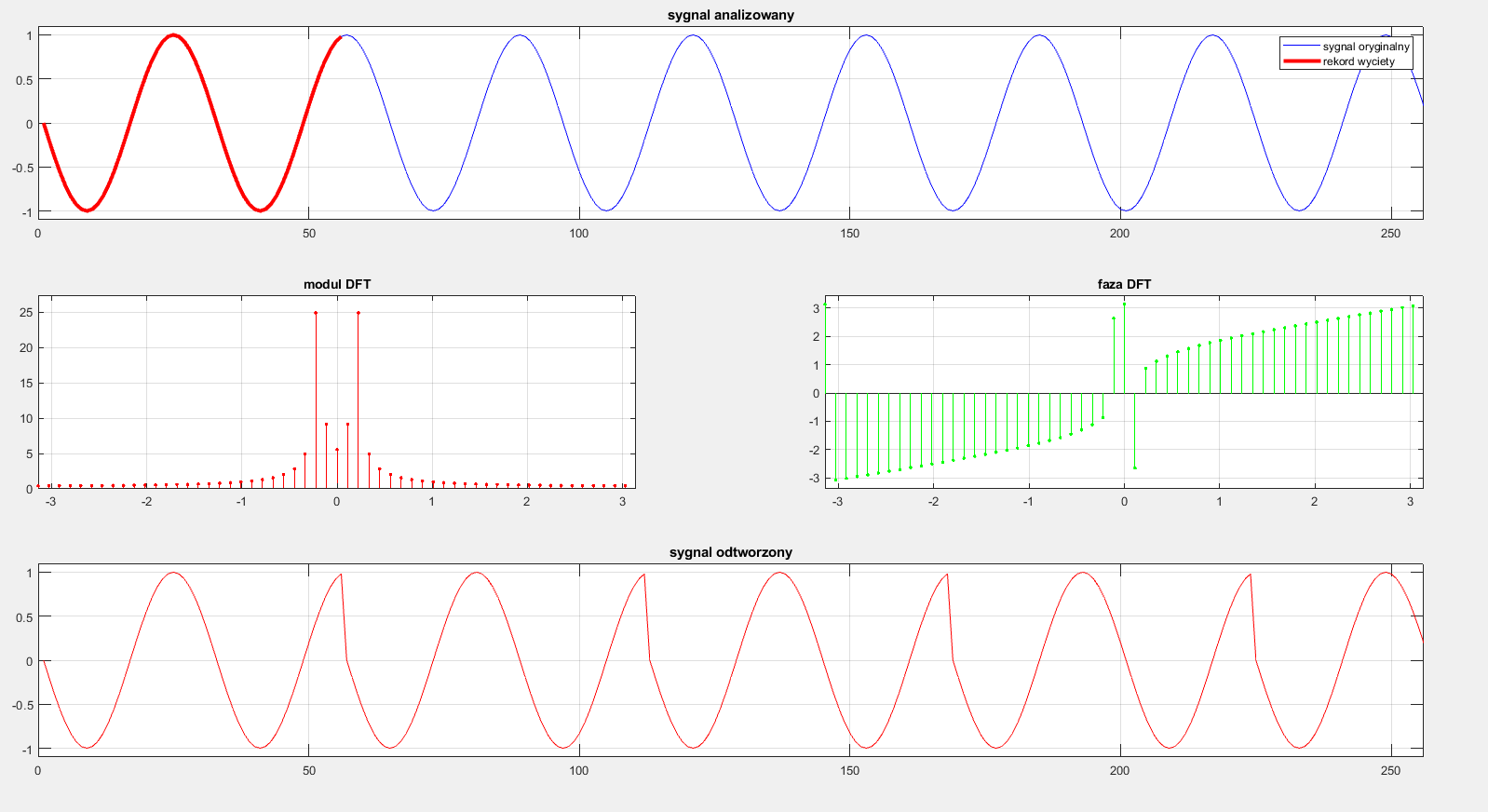
1. Narysować na rys. 1 przebiegi odwrotnej DFT dla L'=32 oraz dla L'=56.

odwrotna

Rys. 1

****

dla L'=32 k =1

****

dla L'=56 k =1

**We wnioskach zapisać kiedy sygnał okresowy może być odtworzony bez zniekształceń na podstawie znajomości DFT skończonego rekordu danych. Jakiego sygnału widmo uzyskuje się w rzeczywistości wyznaczając je na podstawie skończonego rekordu?**

3. Badanie zjawiska aliasingu

Procedury do wykorzystania

**alias(sig)** - wyznacza DFT sygnału sig wykreślając dodatkowo jego postać czasową oraz moduł DFT w zakresie pulsacji unormowanej ±3π.

Polecenia:

1. Wykorzystując zaprezentowaną powyżej procedurę

**alias(sig)**

zaobserwować zmiany zachodzące w module DFT dyskretnego sygnału kosinusoidalnego o wymiarze N=8 przy zmianie jego unormowanej pulsacji a) (2π/N)⋅2, b) (2π/N)⋅3, c) (2π/N)⋅4, d) (2π/N)⋅5 i przy **N=8**. Przy obserwacji można wykorzystać program:

N=8;

w0=(2\*pi/N)\*i; % dla i=2,3,4,5

sig=sygnal(w0,1,0,N);

alias(sig);

Narysować uzyskane postacie widma amplitudowego dla każdego przypadku.

alias

alias

alias

alias

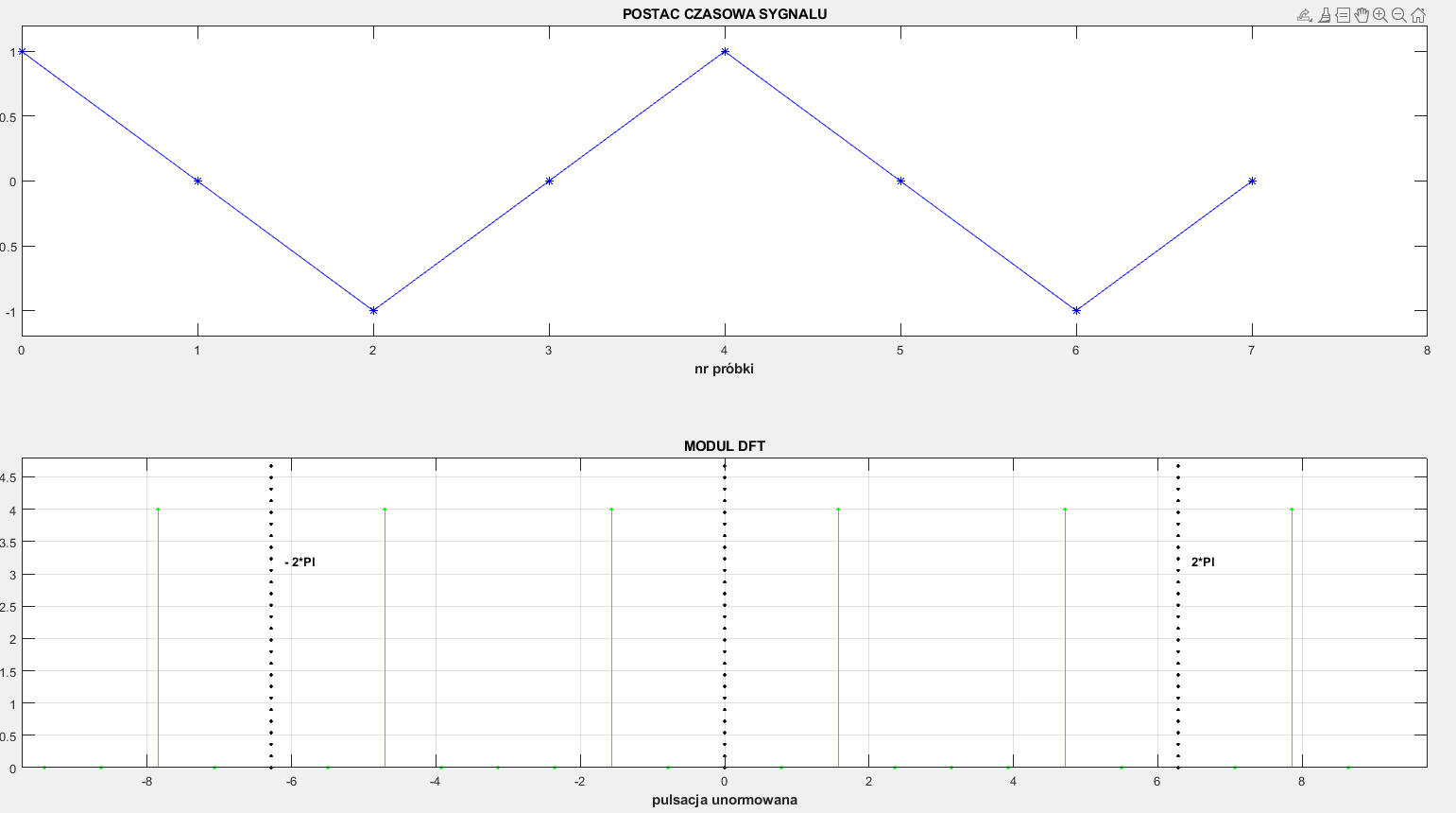
a)

b)

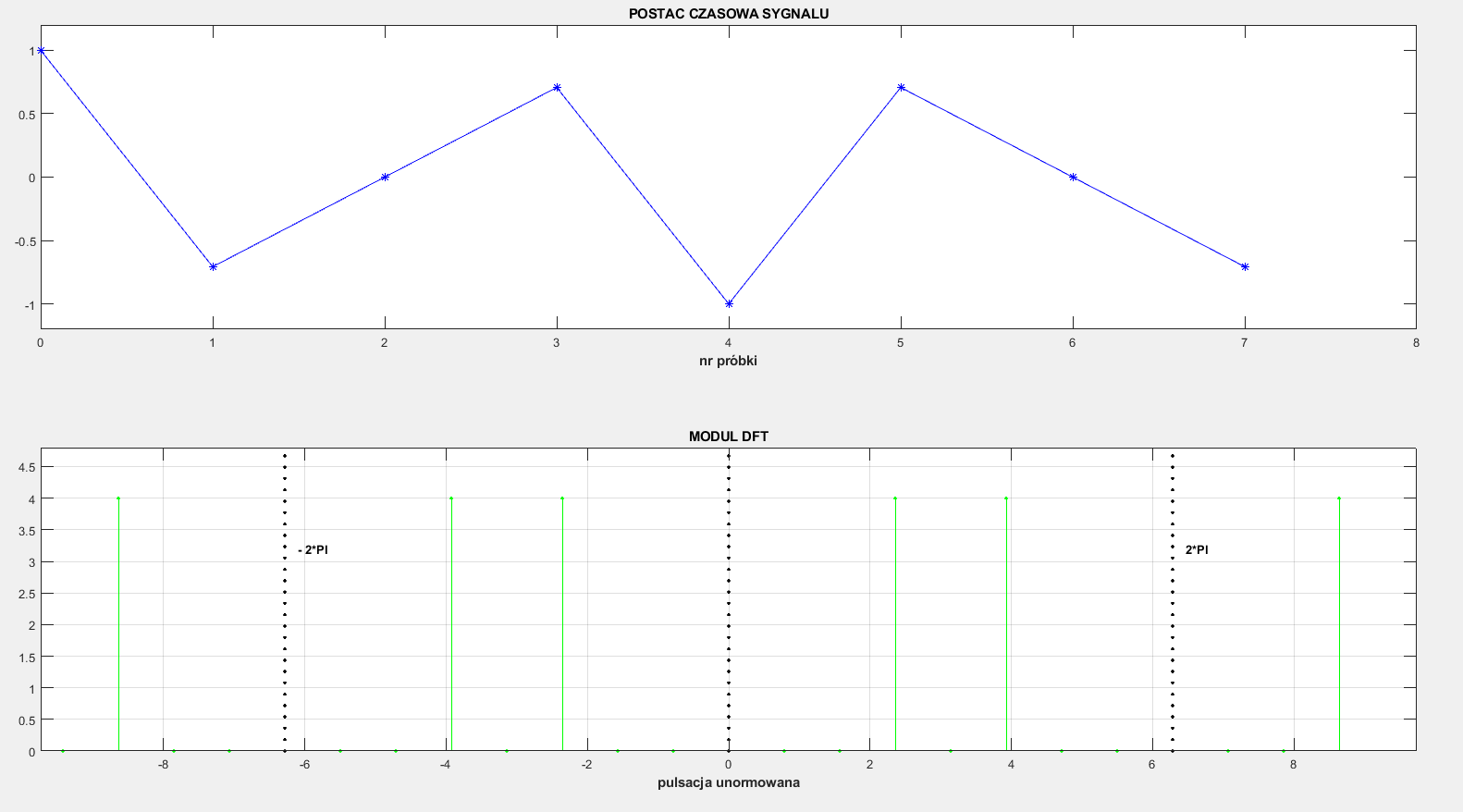
c)

d)

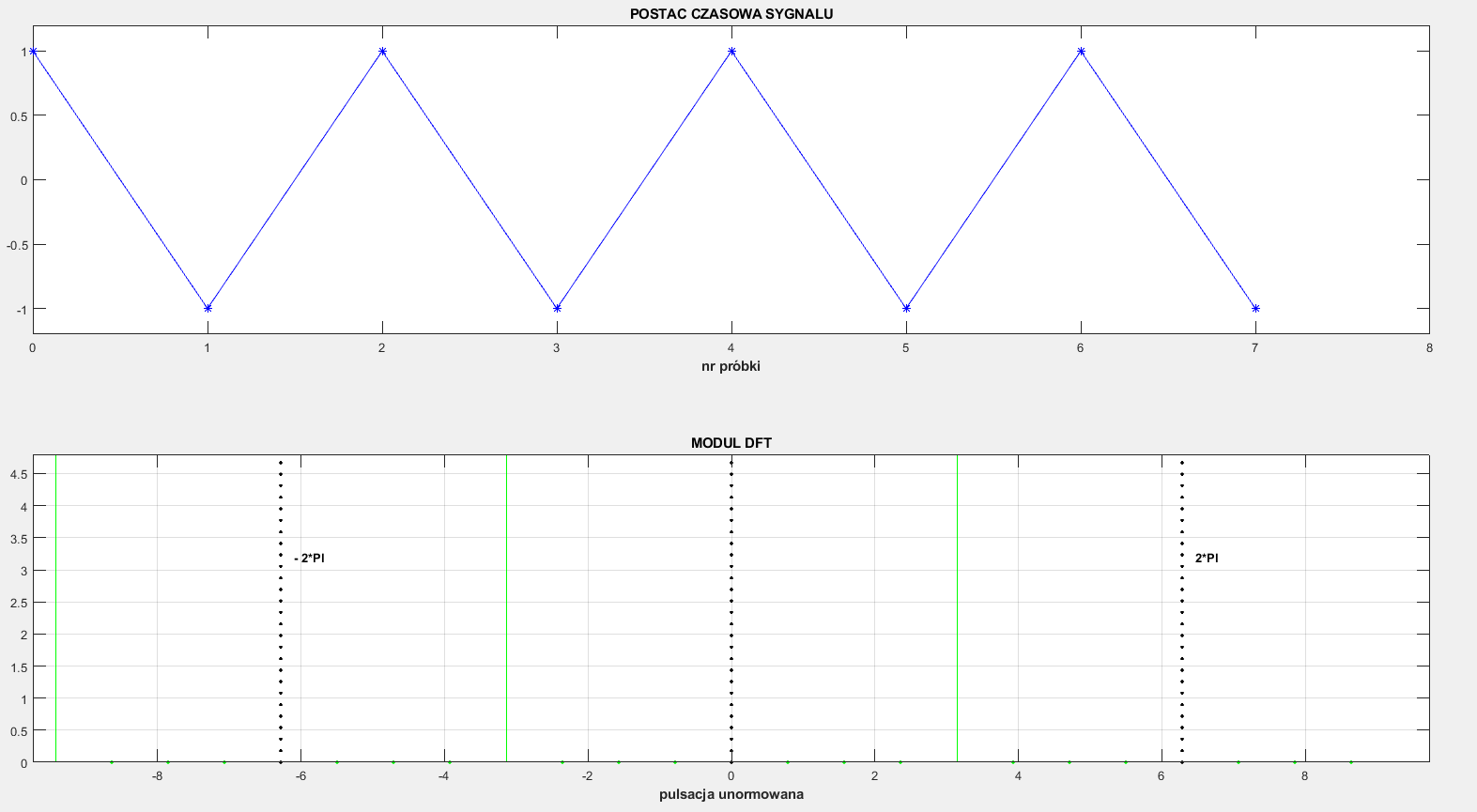
Rys. 2

****

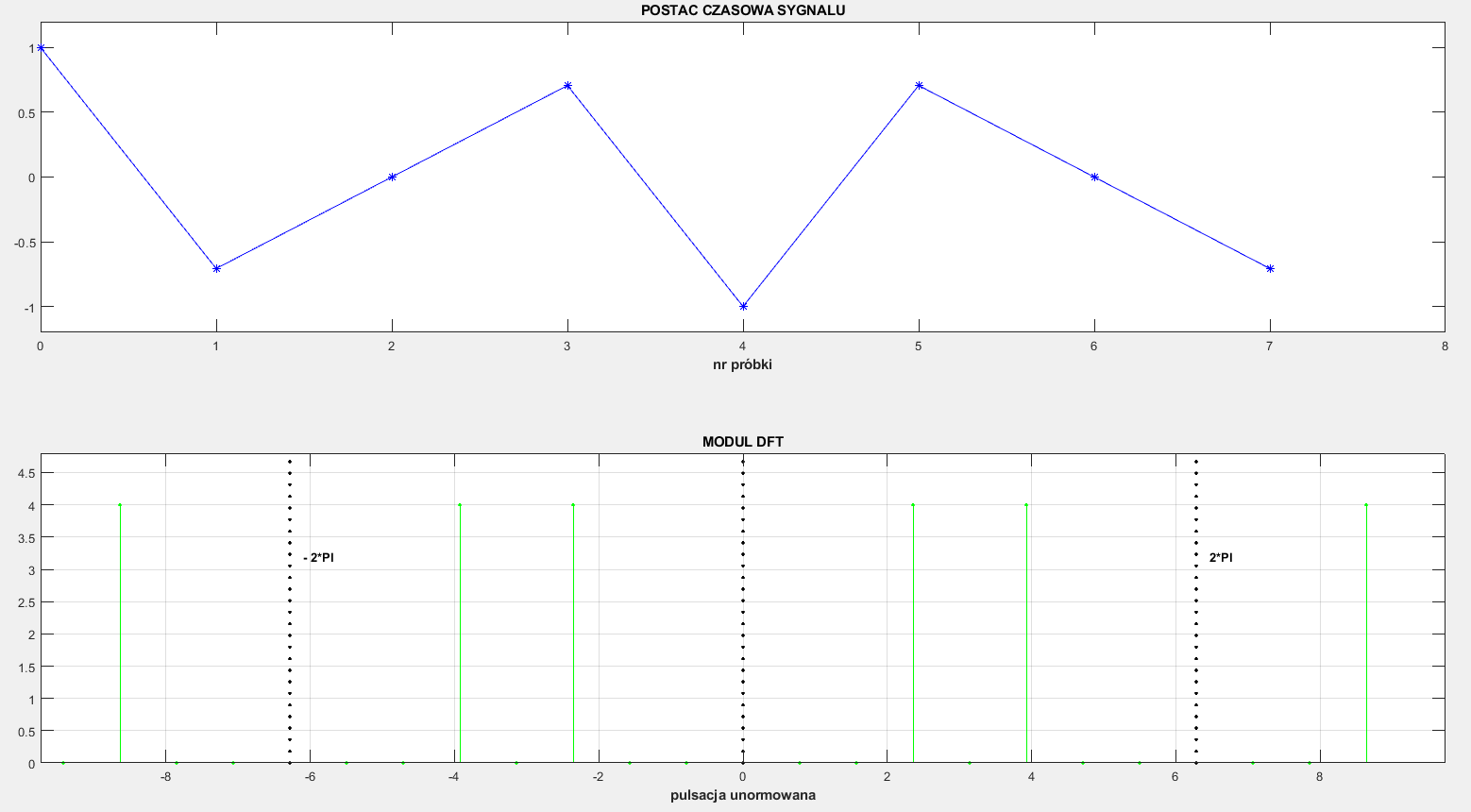
I = 2

****

I = 3

****

I=4

****

I = 5

Dla i = 5 jest chyba aliasing i chyb a i 4

**We wnioskach zapisać przyczyny obserwowanego przekłamania w widmie. Jaka musi być wartość pulsacji unormowanej a jaka częstotliwości rzeczywistej sygnału aby uniknąć zjawiska aliasingu?**

4. Badanie własności okien czasowych

Procedury do wykorzystania

**okna(N,typ)** - procedura wyznaczająca widmo amplitudowe okna czasowego o długości N określonego przez string typ:

typ='p' - okno prostokątne,

typ='h' - okno Hammninga,

typ='t' - okno trójkątne,

typ='b' - okno Blackmana.

typ='f' – okno flattopwin

Wykres widma sporządzany jest samoczynnie w skali liniowej i w unormowanej skali decybelowej. Wielokrotne wywoływanie procedury sprawia, że kolejne widma pojawiają się na tym samym wykresie. Wykorzystanie polecenia legend('p','h','t','b', 'f'), którego parametry wejściowe muszą odpowiadać kolejności badanych okien zapewnia uzyskanie czytelnego opisu uzyskanego rysunku.

Polecenia:

1. Wykorzystując procedurę przy zastosowaniu różnych okien

**okna(N,** 'p'**); okna(N,** 'h'**);**  **okna(N,** 't'**);**  **okna(N,** 'b'**); );**  **okna(N,** 'f'**);**

dla **N=30** zobrazować widma amplitudowe następujących okien czasowych: prostokątnego, Hamminga, trójkątnego, Blackmana i flattopwin.

1. Korzystając z przycisku lupy czasowej określić dla każdego z nich:

* wysokość listka głównego ze skali liniowej,
* poziom pierwszego listka bocznego ze skali decybelowej,
* szerokość listka głównego określoną przez pierwsze miejsce zerowe.

Wyniki zanotować w tabeli 1.

Tabela 1. Wyniki badania różnych typów okien funkcyjnych

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Parametr Rodzaj okna | prostokątne | Hamminga | trójkątne | Blackmana | flattopwin |
| Wysokość listka głównego | 1 | 0,5167 | 0,5 | 0,406 | 0,208379 |
| Poziom pierwszego listka bocznego [dB] | -13,2299 | -31,4689 | -26,658 | -58,14,34 | -83,199 |
| Szerokość listka głównego | 2\* 0,066406 | 2\* 0,128906 | 2\* 0,132812 | 2\* 0,205078 | 2\* 0,339844 |
| Poziom tłumienia listków bocznych | 5 | 3 | 4 | 2 | 1 |

1. Uszeregować okna za pomocą liczb naturalnych od 1 do 5 ze względu na poziom tłumienia listków bocznych (przyjąć poziom pierwszego listka boczneg0). Zastosować oznaczenia: 1 - największe tłumienie, 5 - najmniejsze.

**We wnioskach zapisać czym różnią się poszczególne typy okien i na czym polega kompromis w ich doborze.**

5. Badanie złożoności numerycznej algorytmów wyznaczania dyskretnej transformaty Fouriera

Procedury do wykorzystania

**[czas\_oper,widmo]=wydajnosc(sig,metoda)** - procedura wyznaczająca złożoność obliczeniową algorytmów DFT sygnału sig. Czas\_oper jest czasem wykonywanych przez algorytm operacji zmiennoprzecinkowych (dodawanie, odejmowanie, mnożenie, dzielenie). widmo jest modułem DFT sygnału. Pod string ‘metoda’ należy wstawić jednoliterową nazwę jednej z metod, tzn. 'A', 'B' lub 'C'.

Polecenia:

1. Korzystając z zaprezentowanej powyżej procedury wydajnosc wyznaczyć czas operacji zmiennoprzecinkowych wymaganą do wyznaczenia DFT przez następujące metody:
2. obliczanie DFT z definicji przy założeniu, że wartości współczynników WNkn są wyznaczane na bieżąco,
3. obliczanie DFT z definicji przy założeniu, że współczynniki WNkn nie są obliczane, ale pobierane z gotowej tablicy wyznaczonej wcześniej,
4. obliczanie DFT za pomocą FFT.
5. Sygnał poddawany transformacji może mieć dowolną postać - można np. wykorzystać sygnał uzyskiwany za pomocą procedury sygnal. Badania przeprowadzić dla długości sygnałów podanych w tabeli 2, do której każdorazowo należy wpisywać czas operacji oraz wartość odchylenia standardowego różnicy pomiędzy widmami uzyskanymi w metodach A i B oraz A i C (wartości odchyleń standardowych zaokrąglić do jednej cyfry znaczącej).

Tabela 2. Porównanie różnych rozwiązań transformacji DFT

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| długość sygnału N | | 256 | 512 | 1024 | 2048 | 4096 | 8192 |
| czas.operacji | metoda A |  |  |  |  |  |  |
| metoda B |  |  |  |  |  |  |
| metoda C (FFT) |  |  |  |  |  |  |
| std | widmoA-widmoB |  |  |  |  |  |  |
| widmoA-widmoC |  |  |  |  |  |  |

1. Narysować w skali logarytmicznej we wspólnym układzie współrzędnych zależność liczby operacji od wymiaru sygnału dla metody A, B i C. Wyznaczyć zyski czasowe (stosunki liczby operacji) dla N=4096 i N=8192 i wyciągnąć wnioski.

Sprawne wyznaczenie pożądanych wartości można uzyskać przez wywołanie skryptu:

N=256 %podlega zmianie

sig=sygnal(2,1,0,N);

[A1,widmoA]=wydajnosc(sig,'A');

[B1,widmoB]=wydajnosc(sig,'B');

[C1,widmoC]=wydajnosc(sig,'C');

roznica1=widmoA-widmoB;

roznica2=widmoA-widmoC;

disp('Czas wykonywania operacji')

[A1; B1;C1]

disp('odchylenia standardowe różnic')

[std(roznica1);std(roznica2)]

**We wnioskach skomentować uzyskane wyniki czasów obliczeń dla różnych wersji implementacji DFT.**

**5. format long w mat labie ?????**