# Лабораторна робота №4

***Тема:*** ***«Лінійне перетворення сигналів»***

***Мета роботи:*** **навчитися створювати сигнали складної форми, використовуючи властивості лінійності.**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Посібник з лабораторної роботи

Для виконання лабораторної роботи вам необхідно повторити відповідні лекції.

# ТЕОРЕТИЧНІ ВІДОМОСТІ

Формування складних послідовностей сигналів можна здійснювати за допомогою лінійних математичних операцій - множення на константу, складання і тимчасового зсуву. Як приклад розглянемо формування послідовності прямокутних імпульсів з відповідних гармонійних складових.

Формула, що описує розкладання імпульсного періодичного сигналу на гармонійні складові виглядає наступним чином:



Програма, що реалізує цю функцію для восьми гармонійних складових, виглядає наступним чином (приклад № 1)

(**meandr.m**)  
N=8; %число ненульових гармонік

t = -1:0.01:1; %вектор моментів часу

A = 1; %амплітуда

T = 1; %період

nh = (1:N)\*2 - 1; %номери ненульових гармонік

% обчислення рядків-гармонік

harmonics = cos(2\*pi\*nh'\*t/T);

Am = 2/pi./nh; %амплітуди гармонік

Am(2:2:end)=-Am(2:2:end); %чергування знаків

s1 = harmonics.\*repmat(Am', 1, length(t));

%формування рядків – часткових сум гармонік

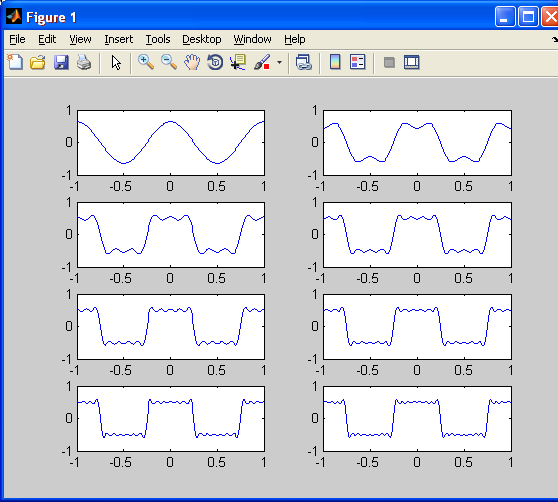
s2 = cumsum(s1);

for k=1:N,

subplot(4,2,k); plot(t,s2(k,:))

end

Результат роботи даної програми зображений на малюнку 1.



Малюнок 1. Формування імпульсного сигналу з восьми гармонійних складових.

Для повного розуміння принципу роботи програми - подивіться документацію для наступних функцій - **repmat** і **cumsum**.

Генерувати періодичні сигнали різної форми можна за допомогою вбудованих функцій середовища MatLab:

**square** - послідовність прямокутних імпульсів;

**sawtooth** - послідовність трикутних імпульсів;

**diric** - функція Діріхле (періодична sinc-функція);

**chirp** - генерація коливань з мінливою частотою.

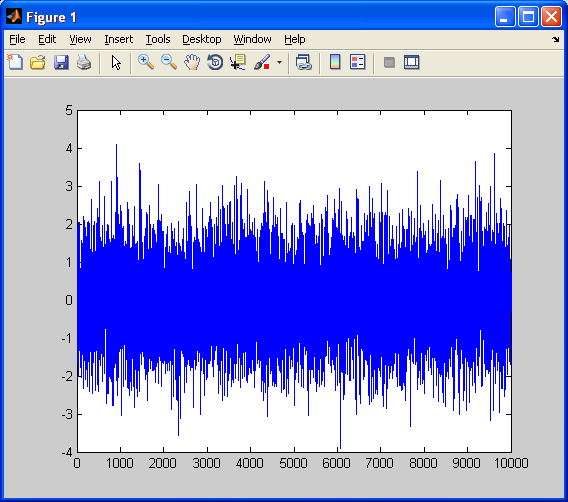
Для генерації дискретного білого шуму з нормальним розподілом можна використовувати функцію **randn (m, n)**. В результаті виклику цієї функції генерується масив, що містить **m** рядків і **n** стовпців псевдовипадкових чисел, що мають нормальний розподіл з нульовим математичним очікуванням і одиничною дисперсією. Приклад роботи цієї функції показаний нижче:

**Приклад № 2**

x=randn(1, 10000);

plot(x);

Результат генерації випадкового сигналу зображений на малюнку 2.



Малюнок 2. Генерація випадкового сигналу.

У пакеті MatLab є ще одна корисна функція - awgn. Вона дозволяє додавати до сигналу білий шум із заданим рівнем. Формат виклику даної функції наступний:

**y** = **awgn (x, snr, sigpower, state, 'powertype'**), де

**x** - вектор відліку сигналу, **snr** - скаляр, що задає співвідношення сигнал / шум в одиницях, визначених параметром **'powertype'** (за замовчуванням - в децибелах),

**sigpower** - потужність сигналу, **state** - примусова установка генератора випадкових чисел (останні три параметра не є обов'язковими).

# ЗАВДАННЯ

**1.** **Опрацюйте основні приклади, викладені вище, в системі MATLAB.**

**2. Створіть М-файл, який реалізує наступні сигнали з відповідних гармонійних складових:**

• Напишіть програму, що синтезує пилкоподібний сигнал з 10 гармонійних складових. Функція для синтезу описується наступним виразом:

****

• Збільшіть кількість гармонійних складових. Як це вплинуло на синтезується сигнал?

• Напишіть програму, що синтезує трикутний сигнал з 8 гармонійних складових. Функція для синтезу описується наступним виразом:

****

• Збільште кількість гармонійних складових. Як це вплинуло на синтезується сигнал?

**3.** **Дослідіть самостійно функції square, sawtooth, diric і chirp. Для візуалізації роботи функції chirp зручно використовувати функцію побудови спектрограми (specgram (s, [], Fs), де s - досліджуваний сигнал, [] - діапазон його значень, Fs - частота дискретизації).**

**4.** **Для випадкового сигналу, що генерується за допомогою функції randn (m, n) створіть М-файл, в якому: генерувався б сигнал, для нього розраховувалося середнє значення, СКО і будувалася гістограма.**

**5.** **Створіть М-файл, в якому до згенерованого періодичного сигналу додавався випадковий шум. Обчисліть статистичні характеристики такого сигналу.**

**6.** **Створіть М-файл, в якому до гармонійного сигналу додається білий шум в заданому співвідношенні (використовуйте для цього функцію awgn).**

**7.** Відкрийте програму, що реалізовує імпульсне розкладання сигналу (**decomp.m**). Вихідний сигнал - один період синусоїдального сигналу, представлений шістнадцятьма відліками. Програма виконує розкладання вихідного сигналу на шістнадцять імпульсних послідовностей (амплітуда і положення імпульсу в кожній послідовності залежить від значення відповідного відліку в вихідному сигналі).

t=0:pi/8:2\*pi; % Period - 16 points

y=sin(t); %Ouer signal

N=length(y); %Namber of pulse

for IC=1:N %Pulse decomposition

decY{IC}=y;

decY{IC}(1:IC-1)=0;

decY{IC}(IC+1:N)=0;

end

stem(t,y);

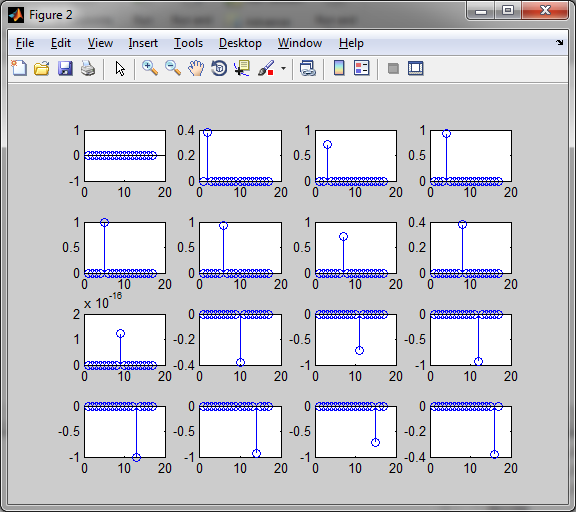
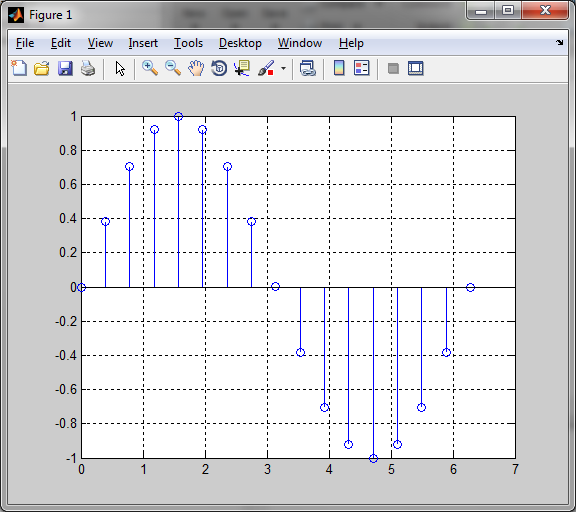
grid;

figure

for k=1:N-1

subplot(4,4,k); stem(decY{1,k})

end

Результати роботи даної програми зображені на малюнках нижче.  
  
 

На першому малюнку - вихідний сигнал. На другому - результат розкладання сигналу на імпульсну послідовність.

**ЗАВДАННЯ**

1) Розробіть програму, що виконує для вихідного сигналу розкладання на ступінчасті функції (сигнал типу «скачок»). Перевірте коректність роботи програми.

2) Розробіть програму, що виконує для вихідного сигналу розкладання на дві послідовності - з парною і непарною симетрією. Перевірте коректність роботи програми.

3) Розробіть програму, що виконує для вихідного сигналу розкладання на дві послідовності, що чергуються - з парними і непарними відліками. Перевірте коректність роботи програми.