

УДК 621.317.08

А.В. РУДИК

Національний авіаційний університет, м. Київ

**СИНТЕЗ ТА МОДЕЛЮВАННЯ ЦИФРОВИХ ФІЛЬТРІВ
ПРОГРАМНИМИ ЗАСОБАМИ MATLAB**

Цифровий фільтр є лінійною дискретною системою, що виконує перетворення вхідної послідовності відліків у вихідну за алгоритмом, який описується різницевою рівнянням і реалізується апаратно або програмно. Залежно від того, чи є параметри цифрового фільтра незмінними або змінюються з часом, фільтр є стаціонарним або адаптивним.

Структура цифрового фільтра (алгоритм обчислення реакції або вихідного сигналу) в більшості практичних випадків реалізується програмно на базі цифрових сигнальних процесорів або програмованих логічних інтегральних схем, тому необхідно проводити відладку програмної частини цифрового фільтра апаратними засобами.

В статті за допомогою графічного інтерфейсу користувача GUI SPTool програмних засобів в MATLAB проведено моделювання цифрової фільтрації, а саме:

- синтез оптимального фільтра нижніх частот з лінійною фазочастотною характеристикою порядку 31 на базі фільтра з кінцевою імпульсною характеристикою другого типу з частотою дискретизації 8000 Гц, граничною частотою смуги пропускання 1000 Гц, граничною частотою смуги затримування 1500 Гц, максимально допустимими відхиленнями амплітудно-частотної характеристики в смугах пропускання 0.05 дБ та затримування 0.01 дБ, максимально допустимими затуханнями в смугах пропускання 0.4455 дБ та затримування 40 дБ;

- аналіз синтезованого цифрового фільтра, в результаті якого отримано амплітудно-частотну характеристику фільтра відповідно до заданих при синтезі вимог;

- імпорт вхідного сигналу, в результаті чого створено вхідний сигнал-вектор, який є сумішшю двох гармонік з нормальним білим шумом і використовується для подальшого аналізу;

- моделювання цифрової фільтрації за результатами синтезу цифрового фільтра та імпорту вхідного сигналу-вектора;

- аналіз сигналів в часовій області, в результаті якого отримано графіки вхідного та вихідного сигналів цифрового фільтра на заданому інтервалі часу ($0 \div 0.128$) с, який визначається довжиною сигналу та частотою дискретизації;

- аналіз сигналів в частотній області методом швидкого перетворення Фур'є, в результаті якого отримано графіки спектральної густини потужності (PSD) вхідного та вихідного сигналів синтезованого цифрового фільтра в основній смузі частот ($0 \div 8000$) Гц, яка визначається значенням частоти дискретизації;

- експорт даних з GUI SPTool, який виконується при необхідності подальшого використання результатів моделювання в MATLAB.

Ключові слова: цифровий фільтр, моделювання, синтез цифрового фільтра, аналіз цифрового фільтра, MATLAB, GUI SPTool, амплітудно-частотна характеристика, імпульсна характеристика, порядок фільтра.

A.V. RUDYK

National Aviation University, Kyiv, Ukraine

SYNTHESIS AND MODELING OF DIGITAL FILTERS BY MATLAB SOFTWARE

The digital filter is a linear discrete system that performs the transformation of the input sequence of readings into output by an algorithm, which is described by differential equations and is implemented hardware or software. Depending on whether the parameters of the digital filter are unchanged or change over time, the filter is stationary or adaptive.

The structure of the digital filter (algorithm reactions or output) in most practical cases implemented software-based digital signal processors or programmable logic integrated circuits, as necessary to carry out debugging of software digital filter hardware.

In the article, using graphical user interface GUI SPTool software in MATLAB carried out the simulation of digital filtering, namely:

- synthesis of an optimal low pass filter with linear phase- frequency characteristic of order 31 on the basis of a filter with a finite pulse characteristic of the second type with a sampling rate of 8000 Hz, a limiting frequency of the bandwidth of 1000 Hz, a limiting frequency of the delay band of 1500 Hz, the maximum permissible deviations of the amplitude-frequency characteristic in the bands A 0.05 dB transmittance and a delay of 0.01 dB, the maximum permissible damping in the bandwidth of 0.4455 dB and a delay of 40 dB;

- analysis of the synthesized digital filter, which resulted in the amplitude-frequency response of the filter according to the requirements specified in the synthesis;

- import input, resulting in a set up input vector, which is a mixture of two harmonics of the normal white noise and is used for further analysis;

- digital filtering simulation based on the results of the synthesis of the digital filter and import of the input signal-vector;

- analysis of signals in the time domain, which resulted in the graphs of the input and output signals of the digital filter at a given time interval ($0 \div 0.128$) s, which is determined by the signal length and sampling frequency;

- analysis of signals in the frequency domain by fast Fourier transform (FFT), in which the received graphics power spectral density (PSD) input and output signals synthesized digital filter in the main frequency band ($0 \div 8000$) Hz, which is determined by the sampling rate;

- export data from a GUI SPTool, which is executed if necessary to further use the simulation results in MATLAB.

Вступ

Цифрові фільтри (еквалайзери) дуже поширені завдяки інтенсивному розвитку ЕОМ, а їх можливості обмежені в основному складністю програми. При обробці сигналу цифровим фільтром є можливість встановити добротність до 10000, коефіцієнт підсилення на певній частоті може досягати 50 дБ, а послаблення – до повного придушення складової з даною частотою, чого не вдасться отримати на аналогових фільтрах. Цифрові еквалайзери не призводять до фазових зсувів (не вносять фазових спотворень), хоча за потребою можуть їх зімітувати. Цифрові фільтри (ЦФ) не додають шуму в сигнал, тому що обробляється оцифрований сигнал, а якість цієї обробки залежить від складності алгоритму, частоти дискретизації та розрядності перетворювачів [1].

ЦФ є лінійною дискретною системою, що виконує перетворення вхідної послідовності відліків у вихідну за алгоритмом, який описується різними рівняннями і реалізується апаратно або програмно. Залежно від того, чи є параметри ЦФ незмінними або змінюються з часом, фільтр називають стаціонарним або адаптивним. Якщо це спеціально не оговорюється, фільтр вважається стаціонарним.

Проектування ЦФ проводиться за чотири етапи.

1. Синтез ЦФ на базі лінійних динамічних схем (ЛДС), що реалізується у такій послідовності:

- вибір типу ЦФ, при цьому нерекурсивним та рекурсивним фільтрам відповідають фільтри з кінцевою (КІХ-фільтри, *FIR filters*) та безкінечною (БІХ-фільтри, *IIR filters*) імпульсною характеристикою;
- формування вимог до характеристик ЦФ (залежать від його типу та призначення (частотно-вибірковий, перетворювач Гільберта або ін.), за замовчуванням ЦФ вважається частотно-вибірковим);
- вибір метода синтезу та розрахунок коефіцієнтів передавальної функції ЦФ.

2. Вибір структури ЦФ.

3. Моделювання структури ЦФ з врахуванням ефектів квантування.

4. Реалізація структури ЦФ.

Структура ЦФ (алгоритм обчислення реакції, тобто вихідного сигналу) в більшості практичних випадків реалізується програмно на базі цифрових сигнальних процесорів (ЦСП) або програмованих логічних інтегральних схем (ПЛІС), тому необхідно проводити відладку програмної частини ЦФ апаратними засобами.

В *MATLAB* є багато функцій, призначених для проектування ЦФ, більшість з яких знаходиться в пакетах розширення *Signal Processing Toolbox*, *Filter Design Toolbox* та *Fixed Point Toolbox* [2]. На базі даних функцій розроблено стандартні програми *GUI FDATool* та *SPTool*. Однак кількість задач, що розв'язуються за допомогою стандартних програм, обмежена, тому для моделювання складних систем цифрової обробки сигналів необхідним є володіння програмними засобами проектування ЦФ. Крім того, знання основних функцій *MATLAB* забезпечує більш глибоке розуміння організації стандартних програм типу *GUI FDATool* та *SPTool*.

Постановка задачі

Метою даної роботи є моделювання цифрової фільтрації засобами програм *GUI SPTool*, а саме аналіз цифрового фільтра та його вхідного і вихідного сигналів в часовій та частотній областях.

Таким чином, в статті необхідно:

- 1) синтезувати оптимальний ФНЧ з лінійною ФЧХ на базі КІХ-фільтра 2-го типу;
- 2) проаналізувати синтезований ЦФ та провести моделювання цифрової фільтрації за результатами синтезу ЦФ та імпорту вхідного сигналу;
- 3) провести аналіз вхідного і вихідного сигналів ЦФ в часовій та частотній областях.

Розв'язання поставленої задачі

На основі програмних засобів в *MATLAB* розроблені програми *GUI* (*Graphic User Interface* – графічний інтерфейс користувача), що є засобами, призначеними для моделювання шляхом інтерактивного спілкування без прямого доступу до програмних засобів з графічним виведенням результатів. Зокрема, для моделювання ЦФ розроблено дві програми *GUI* – *FDATool* та *SPTool*. В даній роботі проведемо моделювання ЦФ засобами *GUI SPTool* (*Signal Processing Toolbox* – засоби обробки сигналу).

1. Моделювання цифрових фільтрів засобами *GUI SPTool*

Програма *GUI SPTool* розроблена на основі пакету розширення *Signal Processing Toolbox* та призначена для розв'язування таких задач, пов'язаних з моделюванням цифрової фільтрації [3]: синтез ЦФ; аналіз ЦФ; імпорт вхідного сигналу; моделювання цифрової фільтрації; аналіз сигналів в часовій області; аналіз сигналів в частотній області; експорт даних з *GUI SPTool*. Можливості синтезу ЦФ в програмах *GUI SPTool* та *GUI FDATool* схожі, однак *GUI FDATool* має можливість моделювання структур ЦФ з фіксованою комою. Звернення до *GUI SPTool* відбувається за командою *sptool*.

2. Синтез цифрових фільтрів

Синтезуємо оптимальний КІХ-фільтр ФНЧ за заданими вимогами до АЧХ методом чебишевської апроксимації засобами *GUI SPTool* за такими вихідними даними: частота дискретизації f_D (Fs) – 8000 Гц; гранична частота смуги пропускання f_x (f_l) – 1000 Гц; гранична частота смуги затримання f_k (f_h) – 1500

Гц; максимально допустиме відхилення АЧХ у смузі пропускання δ_1 (d1) – 0.05; максимально допустиме відхилення АЧХ у смузі затримування δ_2 (d2) – 0.01; максимально допустиме затухання в смузі пропускання $a_{max} = 0.4455$ (дБ); мінімально допустиме затухання в смузі затримування $a_{min} = 40$ (дБ).

Основні етапи синтезу є такими.

1. Вибір типу та методу синтезу ЦФ. В списку *Algorithm* («Алгоритм») вибираємо значення *Equiripple FIR* («Метод найкращої рівномірної (чебишевської) апроксимації»).

2. Задавання вхідних параметрів. Прапорець *Minimum Order* («Мінімальний порядок») встановлено, вхідні параметри відсутні.

3. Задавання вимог до АЧХ:

- тип вибірконості задається в групі *Specifications* («Вимоги») в списку *Type* («Тип») – *lowpass*;
- частота дискретизації та граничні частоти смуг пропускання і затримування в Гц вказуємо відповідно в полі вводу *Sampling Frequency* («Частота дискретизації») – 8000 і в групі *Specifications* у підгрупах *Passband* («Смуга пропускання») в полі вводу *Fp* – 1000 та *Stopband* («Смуга затримування») в полі вводу *Fs* – 1500;

- максимально допустимі відхилення АЧХ (дБ) вказуємо в групі *Specifications* в підгрупах *Passband* в полі вводу *Rp* – 0.4455 та *Stopband* в полі вводу *Rs* – 40.

4. Синтез фільтра відбувається після натискання кнопки *Apply* (при скинутому прапорці *Auto Design*), а після завершення автоматично видаються (рис. 1, а):

- в полі *Frequency Response* («Частотна характеристика») – графік АЧХ (дБ) (характеристика затухання);

- в групі *Measurements* («Значення») – *Order* – 26;

- в групі *Passband* дійсне відхилення АЧХ (дБ) *Actual Rp* – 0.5697 і вага в смузі пропускання *Weight* – 1;

- в групі *Stopband* дійсне відхилення АЧХ (дБ) *Actual Rs* – 37.82 і вага в смузі затримування 2.564;

- в списку *Filter* – ім'я синтезованого фільтра *fil2*.

Порівнюючи дійсні значення АЧХ (дБ) *Actual Rp* та *Actual Rs* в групі *Measurements* із заданими користувачем значеннями *Rp* та *Rs* в групі *Specifications*, бачимо, що вимоги до АЧХ (дБ) в смузі пропускання не виконуються. Для зміни порядку КІХ-фільтра скинемо прапорець *Minimum Order*, а в полі вводу *Order* («Порядок») встановлюємо більший порядок фільтра і задаємо вхідні параметри за замовчуванням: в підгрупі *Passband* в полі вводу *Weight* – 1, а в підгрупі *Stopband* в полі вводу *Weight* – 5.

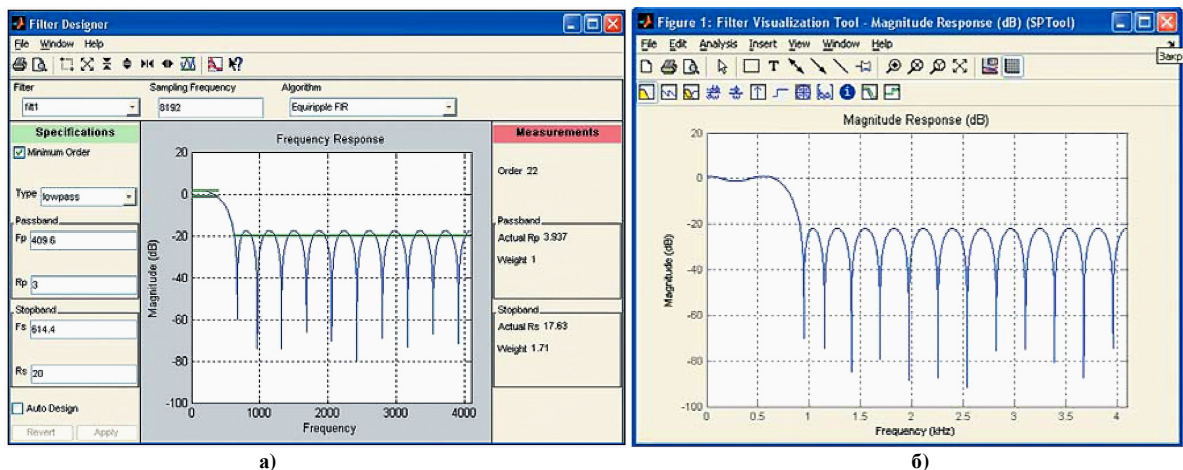


Рис. 1. Вікна Filter Designer (а) та Figure...: Filter Visualization Tool (б)

Вага *Weight* визначається так: вага, що дорівнює 1, присвоюється смузі пропускання з найбільшим максимально допустимим відхиленням 0.05, а вага в смузі затримування розраховується як відношення $0.05/0.01=5$.

При повторному синтезі після натискання кнопки *Apply*, яка активізується після клацання лівою кнопкою миші на полі графіку, в групі *Measurements* автоматично виводяться вихідні параметри в підгрупах *Passband* та *Stopband*. Порівнюючи значення АЧХ (дБ) *Actual Rp* та *Actual Rs* із заданими користувачем в групі *Specifications*, бачимо, що вимоги до АЧХ (дБ) будуть виконуватися при порядку фільтра $R=31$. Таким чином, синтезовано оптимальний ФНЧ з лінійною ФЧХ порядку $R_{opt} = 31$ на базі КІХ-фільтра 2-го типу.

3. Аналіз цифрових фільтрів

Для аналізу характеристик ЦФ, синтезованого або імпортованого в *GUI SPTool*, необхідно в списку *Filters* вікна *SPTool*: *startup.spt* виділити ім'я ЦФ і натиснути кнопку *View*. Далі відбувається автоматичне звертання до *GUI FVTool* і відкривається вікно *Figure...: Filter Visualization Tool*. Список команд пункту меню *Analysis* даного вікна не відрізняється від пункту меню *Analysis* вікна *Filter Design & Analysis Tool GUI*

FDATool [4].

4. Імпорт вхідного сигналу

Джерелом сигналу для *GUI SPTool* можуть бути робочий простір пам'яті *Workspace* при створенні сигналу в командному вікні *MATLAB* або диск, якщо сигнал збережений як *mat*-файл даних в папці *work*.

Для подальшого аналізу створюємо вхідний сигнал – 1024-точкову послідовність, що є сумішшю двох дискретних гармонік $x(n) = \sin(2\pi f_1 nT) + 0.5 \sin(2\pi f_2 nT)$ зі значеннями частот $f_d = 8 \text{ кГц}$, $f_1 = 500 \text{ Гц}$, $f_2 = 2000 \text{ Гц}$ з нормальним білим шумом та збережемо його на диску.

Сформуємо вхідний сигнал – вектор x – підставивши в попереднє співвідношення значення частот:

$$x(n) = \sin(0.125\pi n) + 0.5 \sin(0.5\pi n). \quad (1)$$

Підсумковий сигнал позначимо як вектор s і збережемо на диску:

```
>> N=1024; n=0:(N-1);
>> x=sin(0.125*pi.*n)+0.5*sin(0.5*pi.*n);
>> s=x+randn(1,length(x));
>> save s
```

Для імпорту вхідного сигналу в *GUI SPTool* необхідно у вікні *SPTool: startup.spt* в пункті меню *File* вибрати команду *Import*, після чого відкривається вікно *Import to SPTool*, в якому в групі *Source* («Джерело») вибирають перемикач *From Workspace* (з робочого простору пам'яті *Workspace*) або *From Disk* («З диску»).

Для видалення імпортованих сигналів потрібно у вікні *SPTool: startup.spt* в групі *Signals* курсором виділити ім'я сигналу, що видаляється, та в пункті меню *Edit* вибрати команду *Clear*.

5. Моделювання цифрової фільтрації

Після синтезу ЦФ та імпорту вхідного сигналу можна проводити моделювання цифрової фільтрації. Для цього у вікні *SPTool: startup.spt* проводимо такі дії [5]:

1. В групі *Signals* виділити ім'я сигналу, наприклад, *sig1* (вектор s).
2. В групі *Filters* виділити ім'я фільтра, наприклад, КІХ-фільтра ФНЧ *filt2*.
3. Натиснути кнопку *Apply*, після чого відкривається вікно *Apply Filter* («Використати до фільтра») з іменами вхідного сигналу *sig1*, фільтра *filt2* та вихідного сигналу *sig3*.
4. Натиснути кнопку *OK* і впевнитися, що сигнал *sig3* відобразився в групі *Signals*.

6. Аналіз сигналів в часовій області

Для аналізу сигналів в часовій області потрібно у вікні *SPTool: startup.spt* в групі *Signals* виділити ім'я сигналу (одне або декілька) і натиснути кнопку *View*, після чого відкривається вікно *Signal Browser* («Перегляд сигналу»). На рис. 2 наведені графіки вхідного та вихідного сигналів – *sig1* та *sig3* відповідно.

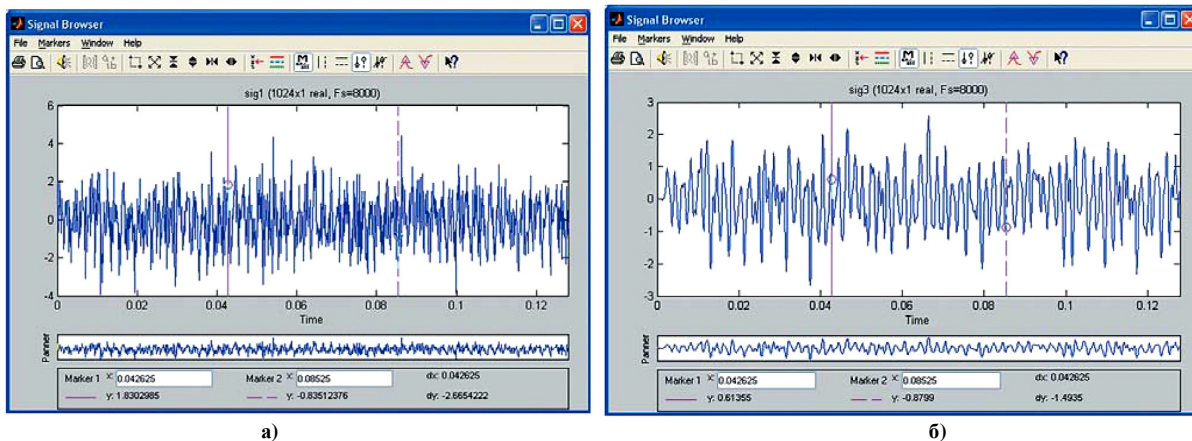


Рис. 2. Вікна *Signal Browser* – вхідний сигнал *sig1* (а) та вихідний сигнал *sig3* (б)

У вікні *Signal Browser* (рис. 2) автоматично виводяться два графіки дискретного сигналу (або декількох сигналів), зображені неперервними лініями: нижній *Panner* («Панорама»), тобто загальний вигляд (панорама) сигналу на заданому інтервалі, та верхній, тобто загальний вигляд сигналу на заданому інтервалі з активізованим набором засобів для аналізу сигналу.

На верхньому графіку вхідний дискретний сигнал *sig1* представлений на інтервалі часу $[0; NT] = [0; N/f_d]$, де N – довжина сигналу (при $N=1024$ та $f_d = 8000 \text{ Гц}$ маємо інтервал $[0; 0.128] \text{ с}$).

7. Аналіз сигналів в частотній області

Для аналізу сигналів в частотній області необхідно:

1. У вікні *SPTool: startup.spt* в групі *Signals* виділити ім'я сигналу.
2. В групі *Spectra* натиснути кнопку *Create*, після чого відкривається вікно *Spectrum Viewer* («Перегляд спектру»), в якому в групі *Parameters* в списку *Method* знаходяться методи спектрального аналізу сигналів [6].

3. Натискаємо кнопку *Apply*, після чого у вікні *Spectrum Viewer* виводиться графік спектральної

густини потужності (PSD) в основній смузі частот $[0; f_d]$.

В групі *Signals* послідовно виділяємо імена вхідного і вихідного сигналів – $sig1$ та $sig3$. Для кожного з них вибираємо метод спектрального аналізу FFT (швидке перетворення Фур'є, ШПФ), вхідним параметром якого є задане число точок сигналу – 1024. Графіки PSD вхідного та вихідного сигналів $sig1$ та $sig3$ представлені на рис. 3, де маркери *Marker 1* та *Marker 2* підведені до частот сигналу 500 Гц та 2000 Гц (1), а змінні $y1$ та $y2$ фіксують значення PSD (дБ/Гц). Обчисленим PSD автоматично присвоюються імена $spect1$, $spect2$ та ін. В *GUI SPTool* при використанні методу FFT спектральна густина потужності $PSD P(k)$ (дБ/Гц) розраховується як

$$P(k) = 10 \lg \left(\frac{|X(k)|^2}{N f_d} \right), \quad k = 0, 1, K, (N-1), \quad (2)$$

де N – розмірність ШПФ; $X(k)$ – N -точкове ШПФ послідовності $x(n)$; k – дискретна нормована частота.

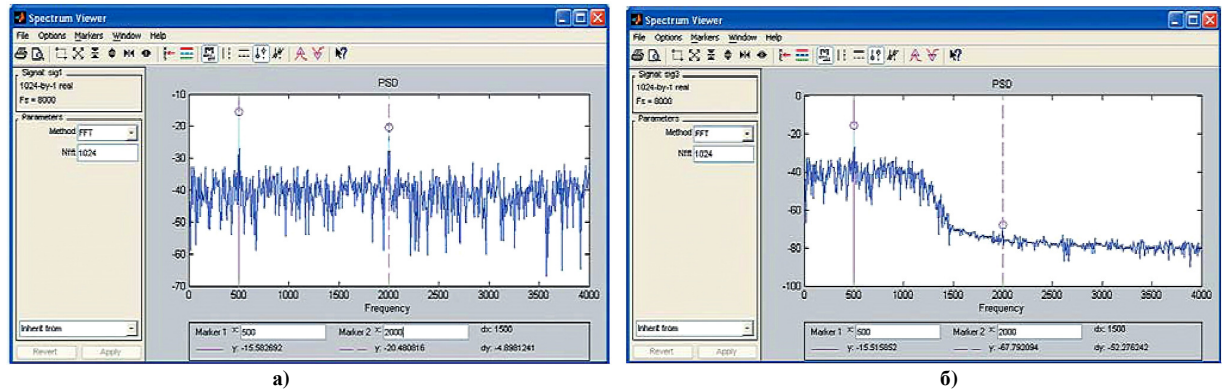


Рис. 3. Вікна *Spectrum Viewer* – $PSD spect1$ вхідного сигналу $sig1$ (а) та $PSD spect2$ вихідного сигналу $sig3$ (б)

Визначимо значення спектральної густини потужності (вектора PDB) на частотах 500 Гц та 2000 Гц. Для цього знайдемо відповідні дискретні нормовані частоти k_1 та k_2 за допомогою зовнішньої функції fft_e1 аналогічно обчисленням ШПФ в *MATLAB* [6].

```
>> [max(P) min(P)]
ans=0.0277          2.0302e-007
>> e1=0.05;
>> [MODm,m]=fft_e1(N,P,e1)
>> MODm=0.0277    0.0090    0.0090    0.0277
m=      65      257      769      961
```

З врахуванням того, що нижня границя індексів в *MATLAB* дорівнює одиниці, в основній смузі дискретних нормованих частот $0 \leq k \leq 511$ маємо $k_1 = 64$ та $k_2 = 256$, а в *MATLAB* – $k_1 = 65$ та $k_2 = 257$.

Виведемо значення спектральної густини потужності $P(k_1)$ та $P(k_2)$ (дБ/Гц):

```
>> [PDB(65) PDB(257)]
ans=      -15.5827          -20.4808
```

Отримані значення майже збігаються з виведеними у вікні *Spectrum Viewer* (рис. 3, а). На рис. 3, б наведена спектральна густина $spect2$ вихідного сигналу $sig3$ КІХ-фільтра ФНЧ $filt2$, при цьому одна складова вхідного сигналу $sig1$ потрапляє до смуги пропускання ФНЧ, а друга з частотою 2 кГц – до смуги затримання.

8. Експорт даних з *GUI SPTool*

Якщо результати моделювання в *GUI SPTool* потрібно далі використовувати в *MATLAB*, необхідно провести їх експорт. Для цього у вікні *SPTool: startup.spt* в пункті меню *File* потрібно вибрати команду *Export* («Експорт»), після чого відкривається вікно *Export from SPTool* («Експорт з *SPTool*»), в якому необхідно:

1. В групі *Export List* («Список імен експортованих даних») виділяємо імена експортованих даних.
2. За допомогою кнопок вказуємо, куди експортуються дані (*Export to Disk* або *Export to Workspace*).

При експорті даних у *Workspace* вони автоматично зберігаються з тими самими іменами, що і в *GUI SPTool*. Експорт даних на диск відбувається послідовно, при цьому кожний раз у вікні *Export to Disk*, що відкривається автоматично, необхідно вказати ім'я *mat*-файлу (без розширення), яке може збігатися або не збігатися з іменами в *GUI SPTool*, після чого натиснути кнопку *Save*, і дані зберігаються в папці *work* або в попередньо відкритій папці користувача. Завантаження *mat*-файлів з диску у *Workspace* проводиться за командою *load*.

Для проведення подальшого моделювання експортуємо сигнал $sig3$, фільтр $filt2$ та $PSD spect2$ у

Workspace з тими самими іменами. Дані, що зберігаються, представляються у вигляді масивів записів *struct array* [6].

Поле *data* зберігає значення сигналу та може використовуватися для побудови його графіку (рис. 4, а):

```
>> s=sig3.data;
>> n=0:1023;
>> subplot(2,1,1), stem(n,s,'fill','MarkerSize',3),...
grid, xlabel('n'), title('sig3')
```

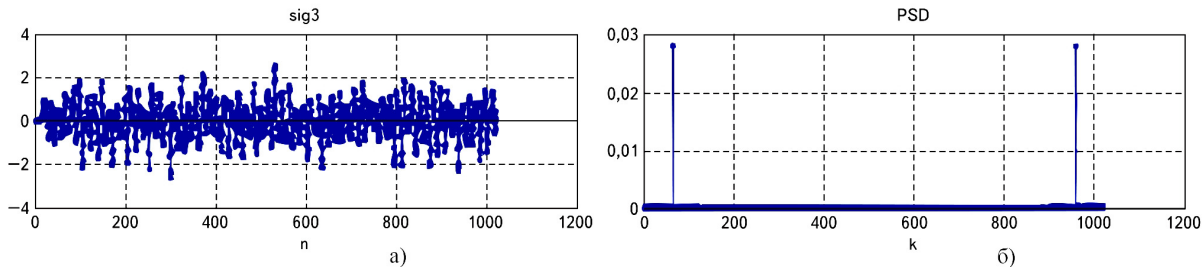


Рис. 4. Вихідний сигнал *sig3* (а) та *PSD* (б)

В масиві записів *filt2* зарезервовано ряд пустих полів [] з фіксованими іменами для зберігання інших представлень передавальної функції, імпульсної характеристики, перехідної характеристики та ін. Після проведення необхідних розрахунків цим полям присвоюються відповідні значення. Так, за коефіцієнтами передавальної функції можна обчислити імпульсну характеристику КІХ-фільтра за допомогою функції *impz* та зберегти її значення в полі *impz*:

```
>> N=32;
>> filt2.imp=impz(filt2.tf.num,filt2.tf.den,N);
```

Інтервал між сусідніми значеннями частот (Δf) дорівнює $\Delta f = f_d / N = 8000 / 1024 = 7.8125$ Гц, а діапазон частот $-\left[f_d / 2; f_d - \Delta f\right] = [4000; 7992.2]$ Гц.

При побудові графіка залежності (2) за шкалою дискретних нормованих частот k використовується 1024 значення, які відповідають частотному діапазону $[0; f_d - \Delta f] = [0; 7992.2]$ Гц, де частоти 0 Гц

відповідає дискретна нормована частота $K = \frac{f_d}{2\Delta f} + 1$ (рис. 4, б):

```
>> K=4000/delta_f=1
K=513
>> spect2.f(513)
ans=0
>> k=K:(K+1023);
kk=0:1023;
>> subplot(2,1,2), stem(k,spect2.P(k),'fill','MarkerSize',3),...
grid, xlabel('k'), title('PSD')
```

В полі *label* зберігається ім'я *PSD* в *GUI SPTool*, в полі *signal* – інформація про довжину вихідного сигналу *sig3* (перший елемент вектора), а в полях *confid*, *specs*, *lineinfo* та *SPTIdentifier* – допоміжна інформація.

При виході з програми *GUI SPTool* автоматично з'являється запит, чи потрібно зберігати дані (сигнали, фільтри, спектральні густини потужності). У випадку позитивної відповіді натискається кнопка *Save*, дані зберігаються у файлі *startup.spt* в папці *work* або в папці користувача і можуть використовуватися в наступних сеансах роботи в *GUI SPTool*.

Висновки

В статті за допомогою графічного інтерфейсу користувача *GUI SPTool* програмних засобів в *MATLAB* проведено моделювання цифрової фільтрації, а саме:

- синтез оптимального ФНЧ з лінійною ФЧХ порядку $R_{opt} = 31$ на базі КІХ-фільтра 2-го типу;
- аналіз синтезованого ЦФ, в результаті якого отримано АЧХ фільтра;
- імпорту вхідного сигналу, в результаті чого створено вхідний сигнал-вектор, який є сумішшю двох гармонік і використовується для подальшого аналізу;
- моделювання цифрової фільтрації за результатами синтезу ЦФ та імпорту вхідного сигналу;
- аналіз сигналів в часовій області, в результаті якого отримано графіки вхідного та вихідного сигналів ЦФ на заданому інтервалі часу $[0; NT] = [0; N/f_d] = [0; 1024/8000] = [0; 0.128]$ с);

- аналіз сигналів в частотній області, в результаті якого отримано графіки спектральної густини потужності (*PSD*) вхідного та вихідного сигналів синтезованого ЦФ в основній смузі частот

$$[0; f_D] = [0; 8000] \text{ Гц};$$

- експорт даних з *GUI SPTool*, який виконується при необхідності подальшого використання результатів моделювання в *MATLAB*.

Література

1. Лэм, Г. Аналоговые и цифровые фильтры. Расчёт и реализация / Г. Лэм. – М. : Мир, 1992. – 592 с.
2. Ingle, V. K. Digital Signal Processing Using MATLAB. V.4. / V. K. Ingle, J. G. Proakis. – Boston : PWS Publishing Company, 2009. – 420 p. – ISBN: 0-534-93805-1.
3. Солонина, А. И. Цифровая обработка сигналов. Моделирование в MATLAB / А. И. Солонина, С. М. Арбузов. – СПб. : БХВ-Петербург, 2008. – 816 с. – ISBN 978-5-9775-0259-7.
4. Солонина, А. И. Основы цифровой обработки сигналов. 2-е изд. / А. И. Солонина, Д. А. Улахович, С. М. Арбузов, Е. Б. Соловьёва. – СПб. : БХВ-Петербург, 2005. – 334 с.
5. Signal Processing Toolbox: [Электронный ресурс]. – Режим доступа: <https://matlab.ru/products/signal-processing-toolbox>. (Дата звернення: 16.07.2017).
6. Сергиенко, А. Б. Цифровая обработка сигналов / А. Б. Сергиенко. – СПб. : БХВ-Петербург, 2011. – 758 с. – ISBN: 978-5-9775-0606-9.

References

1. Lem, G. Analogovyie i tsifrovyye filtryi. Raschyot i realizatsiya / G. Lem. – M. : Mir, 1992. – 592 s.
2. Ingle, V. K. Digital Signal Processing Using MATLAB. V.4. / V. K. Ingle, J. G. Proakis. – Boston : PWS Publishing Company, 2009. – 420 p. – ISBN: 0-534-93805-1.
3. Solonina, A. I. Tsifrovaya obrabotka signalov. Modelirovanie v MATLAB / A. I. Solonina, S. M. Arbutov. – SPb. : BHV-Peterburg, 2008. – 816 s. – ISBN 978-5-9775-0259-7.
4. Solonina, A. I. Osnovy tsifrovoy obrabotki signalov. 2-e izd. / A. I. Solonina, D. A. Ulahovich, S. M. Arbutov, E. B. Solovyova. – SPb. : BHV-Peterburg, 2005. – 334 s.
5. Signal Processing Toolbox: [Elektronniy resurs]. – Rezhim dostupu: <https://matlab.ru/products/signal-processing-toolbox>. (Data zvernennya: 16.07.2017).
6. Sergienko, A. B. Tsifrovaya obrabotka signalov / A. B. Sergienko. – SPb. : BHV-Peterburg, 2011. – 758 s. – ISBN: 978-5-9775-0606-9.

Рецензія/Peer review : 18.06.2017 р.

Надрукована/Printed : 03.10.2017 р.

Стаття рецензована редакційною колегією