

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
імені ІГОРЯ СІКОРСЬКОГО»

**МАТЕМАТИЧНЕ ЗАБЕЗПЕЧЕННЯ
ЦИФРОВИХ СИСТЕМ. ЧАСТИНА 2.
ЦИФРОВА ФІЛЬТРАЦІЯ
ЛАБОРАТОРНИЙ ПРАКТИКУМ**

Навчальний посібник

Рекомендовано Методичною радою КПІ ім. Ігоря Сікорського
як навчальний посібник для здобувачів ступеня бакалавра
за освітньою програмою
«Системи керування літальними апаратами та комплексами»
спеціальності 173 Авіоніка

Укладач: Ю.В. Бобков

Електронне мережне навчальне видання

Київ

КПІ ім. Ігоря Сікорського

2023

Рецензент *Смирнов С. А.*, канд. фіз.-мат. наук, с.н.с., заступник директора ННФТІ з методичних питань, доцент кафедри інформаційної безпеки КПІ ім. Ігоря Сікорського

Відповідальний редактор *Чепілко М. М.*, докт. фіз.-мат. наук, проф.

*Гриф надано Методичною радою КПІ ім. Ігоря Сікорського
(протокол № 6 від 30.03.2023 р.)*

за поданням Вченої ради Навчально-наукового інституту аерокосмічних технологій (протокол № 3 від 27.03.2023 р.)

Навчальний посібник призначений для забезпечення лабораторних занять з кредитного модуля «Математичне забезпечення цифрових систем. Частина 2. Цифрова фільтрація», що належить до нормативних навчальних дисциплін циклу загальної підготовки для здобувачів ступеня бакалавра спеціальності 173 «Авіоніка», за освітньою програмою «Системи керування літальними апаратами та комплексами».

Основними завданнями лабораторного практикуму є закріплення знань, отриманих при вивченні теоретичного курсу, дослідження властивостей дискретного перетворення Фур'є сигналів та структур лінійних дискретних систем, отримання практичних умінь і досвіду з побудови та дослідження цифрових фільтрів з кінцевою та нескінченною імпульсною характеристикою за допомогою програмних середовищ MatLab та Simulink.

Матеріали посібника можуть бути корисними здобувачам інших спеціальностей, а також фахівцям, що займаються проектуванням і дослідження лінійних дискретних систем і цифрових фільтрів частотної селекції із застосуванням програмних середовищ MatLab і Simulink, зокрема, вбудованого пакету Filter Designer.

Реєстр. № НП 22/23-625. Обсяг 3,7 авт. арк.

Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»
проспект Перемоги, 37, м. Київ, 03056
<https://kpi.ua>

Свідоцтво про внесення до Державного реєстру видавців, виготовлювачів і розповсюджувачів видавничої продукції ДК № 5354 від 25.05.2017 р.

© КПІ ім. Ігоря Сікорського, 2023

ЗМІСТ

| | |
|--|----|
| Вступ | 4 |
| Лабораторна робота № 1. Дослідження дискретного перетворення Фур'є сигналів в системах MatLab і Simulink | 6 |
| Лабораторна робота № 2. Дослідження структур лінійних дискретних систем в системах MatLab і Simulink | 17 |
| Лабораторна робота № 3. Дослідження методів синтезу КІХ-фільтрів за допомогою системи MatLab | 47 |
| Лабораторна робота № 4. Дослідження методів синтезу НІХ-фільтрів за допомогою системи MatLab | 59 |
| Список рекомендованої літератури | 75 |

ВСТУП

Навчальна дисципліна «Математичне забезпечення цифрових систем» належить до нормативних навчальних дисциплін циклу загальної підготовки здобувачів першого (бакалаврського) рівня вищої освіти за спеціальністю 173 «Авіоніка» за освітньо-професійною програмою «Системи керування літальними апаратами та комплексами».

На сучасному етапі розвитку систем авіоніки покращення їх характеристики та підвищення ефективності відбувається за рахунок новітніх досягнень в області цифрової техніки. Переважна кількість сучасних систем керування будується з розвиненою цифровою системою, одними з суттєвих елементів якої є різноманітні лінійні дискретні системи та цифрові фільтри. Тому вивчення дисципліни «Математичне забезпечення цифрових систем» є необхідним та обов'язковим для здобувачів спеціальності 173 «Авіоніка».

Кредитний модуль «Математичне забезпечення цифрових систем. Частина 2. Цифрова фільтрація» формує у здобувачів вищої освіти теоретичні знання та практичний досвід у галузі лінійних дискретних систем та цифрової фільтрації.

При виконанні лабораторних робіт здобувачі досліджують властивості дискретного перетворення Фур'є сигналів і базові структури лінійних дискретних систем в програмних середовищах MatLab та Simulink. Вони також отримують необхідні практичні уміння та досвід з побудови та дослідження цифрових фільтрів з кінцевою та нескінченною імпульсною характеристикою. Для розробки цифрових фільтрів частотної селекції використовується розвинений апарат системи MatLab як у вигляді програмних команд, так і у вигляді спеціалізованого додатку Filter Designer для проектування та дослідження цифрових фільтрів. В останньому випадку розглядається застосування двох спеціалізованих програмних пакетів: sptool та fdatool, - кожний з яких має свої особливості.

В теоретичних відомостях до лабораторних робіт наведено достатньо прикладів, що ілюструють застосування технологій та стандартних елементів MatLab та Simulink. Наведено необхідні алгоритми та детально викладені вимоги до порядку виконання робіт, підготовки звітів. Все це сприяє засвоєнню

матеріалу та дає змогу здобувачам самостійно виконувати завдання лабораторних занять.

Матеріали посібника можуть бути корисними здобувачам інших спеціальностей, а також фахівцям, що займаються проектуванням і дослідження лінійних дискретних систем і цифрових фільтрів частотної селекції із застосуванням програмних середовищ MatLab та Simulink, зокрема, вбудованого пакету Filter Designer.

Лабораторна робота № 1

Дослідження дискретного перетворення Фур'є сигналів в системах MatLab і Simulink

Ціль роботи: дослідити дискретне перетворення Фур'є сигналів і методи його реалізації в системах Matlab і Simulink.

1.1 Теоретичні відомості

1.1.1 Дискретне перетворення Фур'є

Дискретним перетворенням Фур'є (ДПФ) називається пара взаємно однозначних перетворень [1,2,4,5]:

- пряме ДПФ (Discrete Fourier Transform — DFT):

$$X(k) = \sum_{n=0}^{N-1} x(n)W_N^{nk}, \quad k = 0, 1, \dots, N-1; \quad (1.1)$$

- зворотне ДПФ (ЗДПФ) (Inverse Discrete Fourier Transform — IDFT):

$$x(n) = \frac{1}{N} \sum_{k=0}^{N-1} X(k)W_N^{-nk}, \quad n = 0, 1, \dots, N-1, \quad (1.2)$$

де n — дискретний нормований час, $n = nT/T$;

k — дискретна нормована частота, $k = k\Delta\omega/\Delta\omega$;

$\Delta\omega = \omega_d/N = 2\pi/NT$ - період дискретизації по частоті (роздільна здатність по частоті);

$x(n)$ — N -точкова послідовність, тобто періодична послідовність в часовій області з періодом N ;

$X(k)$ — N -точкове ДПФ, тобто періодична послідовність в частотній області з періодом N ;

N — період послідовності і ДПФ;

$$W_N^{nk} = e^{-j\frac{2\pi}{N}nk} \quad \text{— множник, що повертає;}$$

$$X(k)W_N^{-nk} = X(k)e^{j\frac{2\pi}{N}nk} \quad \text{— } k\text{-та дискретна гармоніка.}$$

Значення абсолютних частот дискретних гармонік пов'язані зі значеннями дискретних нормованих частот співвідношенням:

$$f = kf_d/N. \quad (1.3)$$

Дискретне перетворення Фур'є (1) трактується по-різному в залежності від виду послідовності $x(n)$: періодична з періодом N або кінцева довжини N .

Для періодичної послідовності $x(n)$ з періодом N ДПФ $X(k)$ (1) являє собою її спектр з точністю до множника $1/N$.

Модуль ДПФ $|X(k)|$ (з точністю до множника $1/N$) називають амплітудним спектром, а аргумент $\arg\{X(k)\}$ — фазовим спектром періодичної послідовності.

Амплітудний спектр дійсної періодичної послідовності дорівнює модулю ДПФ $|X(k)|$ з точністю до множника:

$$\begin{cases} 1/N, & k = 0; \\ 2/N, & k \neq 0. \end{cases} \quad (1.4)$$

При обчисленні ДПФ (1.1) періодичної послідовності вона може бути задана на періоді N або на цілому числі періодів N , що не змінює результату. Для кінцевої послідовності $x(n)$ довжини N ДПФ (1.1) являє собою N дискретних рівновіддалених значень її спектральної густини $X(e^{j\omega T})$ на періоді $\omega_\Delta = 2\pi/T$.

Для дійсних послідовностей, періодичних і скінченних, модуль ДПФ $|X(k)|$ — парна, а аргумент $\arg\{X(k)\}$ — непарна функція частоти k .

Згідно з визначенням, при обчисленні ДПФ передбачається, що послідовність $x(n)$ є періодичною, і кінцева послідовність являє собою один період періодичної послідовності.

При цьому точне виділення гармонік послідовності $x(n)$ з частотами f_i гарантується тільки в тому випадку, якщо вони кратні періоду дискретизації по частоті $\Delta f = f_\Delta/N$:

$$f_i = q \cdot \Delta f, \quad q = 0, 1, \dots, (N-1), \quad (1.5)$$

що, в свою чергу, можливо лише в тому випадку, якщо на інтервалі NT послідовності $x(n)$ розміщується ціле число періодів T_i , тобто співвідношення:

$$P_i = \frac{NT}{T_i} = \frac{Nf_i}{f_\Delta}, \quad (1.6)$$

є цілим числом.

У разі, якщо умова (1.5) не виконується, спостерігається ефект розтікання спектра.

Спектральна густина кінцевої послідовності $x(n)$ довжини N :

$$X(e^{j\omega T}) = \sum_{n=0}^{N-1} x(n)e^{-j\omega Tn} \quad (1.7)$$

на періоді $\omega_\Delta = 2\pi/T$ пов'язана з відліками ДПФ $X(k)$ (1) с співвідношенням:

$$X(k) = X(e^{j\omega T}) \Big|_{\omega=k\frac{2\pi}{NT}}, \quad k = 0, 1, \dots, N-1. \quad (1.8)$$

Значення спектральної густини (1.7) в L рівновіддалених точках на періоді ω_Δ при $L > N$ визначаються за формулою:

$$X\left(e^{j\frac{2\pi}{L}l}\right) = \sum_{n=0}^{N-1} x(n)e^{-j\frac{2\pi}{L}ln}, \quad l = 0, 1, \dots, L-1, \quad (1.9)$$

де l — дискретна нормована частота, а $\Delta\omega$ — період дискретизації по частоті:

$$\Delta\omega = \omega_\Delta/L = 2\pi/LT \quad (1.10)$$

Той же результат буде отримано, якщо кінцеву послідовність $x(n)$ довжини N доповнити нулями до довжини L :

$$\tilde{x}(n) = \begin{cases} x(n), & 0 \leq n \leq (N-1); \\ 0, & N \leq n \leq (L-1), \end{cases} \quad (1.11)$$

і знайти її ДПФ (1.1), замінюючи N на L :

$$\tilde{X}(k) = \sum_{n=0}^{L-1} \tilde{x}(n)W_L^{nk}, \quad k = 0, 1, \dots, L-1. \quad (1.12)$$

З урахуванням (1.11) формула (1.12) набуває вигляду (порівняйте з (1.9)):

$$\tilde{X}(k) = \sum_{n=0}^{N-1} x(n)W_L^{nk} = \sum_{n=0}^{N-1} x(n)e^{-j\frac{2\pi}{L}kn}, \quad k = 0, 1, \dots, L-1.$$

Слід пам'ятати, що роздільна здатність по частоті, під якою розуміють мінімальну відстань між дискретними гармоніками в ДПФ, визначається виключно періодом дискретизації по частоті $\Delta f = f_\delta/N$ і при фіксованій частоті f_δ залежить тільки від довжини (періоду) послідовності, оскільки саме вона і тільки вона визначає спектральний склад (дискретні гармоніки) послідовності.

Тому збільшення довжини кінцевої послідовності за рахунок додавання $(L-N)$ – нулів і, відповідно, зменшення періоду дискретизації по частоті до $\Delta f = f_\delta/L$, не змінює роздільної здатності по частоті, а лише покращує умови розрізнення близько розташованих частот дискретних гармонік.

1.1.2 Відновлення аналогового сигналу

Дискретне перетворення Фур'є $X(k)$ (1.1) може використовуватися для відновлення аналогового періодичного сигналу з фінітним спектром, розташованим в області $(-N/2) \leq k \leq (N/2-1)$ при N парному, за формулою (усічений ряд Фур'є) [1,2,4]:

$$x(t) = \frac{1}{N} \sum_{k=-N/2}^{N/2-1} X_a(k) e^{j\frac{2\pi}{NT}kt}, \quad (1.13)$$

де відліки $X_a(k)$ пов'язані з відліками ДПФ $X(k)$ співвідношенням:

$$X_a(k) = \begin{cases} X(N+k), & -N/2 \leq k \leq -1; \\ X(k), & 0 \leq k \leq (N/2-1). \end{cases} \quad (1.14)$$

При N непарному береться область $(-(N-1)/2 \leq k \leq (N-1)/2)$.

Той же результат буде отримано при відновленні аналогового сигналу безпосередньо за допомогою усіченого ряду Котельникова:

$$x(t) = \sum_{n=0}^{N-1} x(n) \frac{\sin \left[\pi \left(\frac{t}{T} - n \right) \right]}{\pi \left(\frac{t}{T} - n \right)}. \quad (1.15)$$

В MATLAB для цього зручно скористатися функцією: $\text{sinc}(t/T-n)$.

1.1.3 Обчислення ДПФ в Matlab

В MATLAB ДПФ (1.1) і ЗДПФ (1.2) обчислюється з використанням алгоритмів швидкого перетворення Фур'є (ШПФ) і зворотного ШПФ (ЗШПФ) за допомогою функцій [1,2]:

$$X = \text{fft}(x);$$

$$x = \text{ifft}(X);$$

де x и X — N -точкові послідовності $x(n)$ и її ДПФ $X(k)$ — вектори, нижня межа індексів яких дорівнює одиниці, на відміну від ДПФ (1.1) і (1.2), де вона дорівнює нулю.

1.2 Підготовка до лабораторної роботи

1.2.1 Ознайомитися з короткими теоретичними відомостями і порядком роботи з середовищем MatLab і Simulink.

1.2.2 Написати програму у вигляді файлу-сценарію (m-файлу) для обчислення ДПФ періодичної послідовності $x(n)$ з періодом N :

$$x(nT) = A_1 \cos(2\pi f_1 nT + \pi/4) + A_2 \cos(2\pi f_2 nT + \pi/8), \quad (1.16)$$

використовуючи її тотожне представлення у вигляді:

$$\begin{aligned} x(n) &= A_1 \cos\left(\frac{2\pi f_1}{f_d} n + \frac{\pi}{4}\right) + A_2 \cos\left(\frac{2\pi f_2}{f_d} n + \frac{\pi}{8}\right) = \\ &= A_1 \cos(\hat{\omega}_1 n + \pi/4) + A_2 \cos(\hat{\omega}_2 n + \pi/8). \end{aligned} \quad (1.17)$$

Вивести графіки послідовності $x(n)$ (1.17) на періоді N :

- в шкалі дискретного нормованого часу n ;
- в шкалі дискретного часу nT .

Параметри для гармонічних сигналів вибираються з таблиці 1, згідно заданого варіанта¹.

Таблиця 1 – Параметри гармонічних сигналів.

| № Вар. | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|------------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| Парам. | | | | | | | | | | |
| A_1 | 1 | 1 | 3 | 1 | 5 | 2 | 2 | 4 | 3 | 4 |
| A_2 | 3 | 2 | 1 | 4 | 1 | 5 | 3 | 1 | 2 | 5 |
| f_1 , Гц | 100 | 200 | 300 | 100 | 100 | 200 | 300 | 150 | 250 | 350 |
| f_2 , Гц | 200 | 300 | 400 | 300 | 400 | 400 | 200 | 200 | 400 | 250 |

Примітка 1. Номер варіанта визначається номером студента у списку групи (підгрупи, або об'єднаному списку груп згідно їх номерів). Якщо номер за списком 11 (21), то параметри вибираються для першого гармонічного сигналу з варіанта 1 (2), а другого - з варіанта 10. Якщо номер за списком перевищує 11 (21), то параметри вибираються для першого гармонічного сигналу з варіанта 1 (2), а другого - з варіанта, номер якого дорівнює другій цифрі в номері за списком. Наприклад, для номера 12 параметри для першого гармонічного сигналу вибираються з варіанта 1, а другого - з варіанта 2.

1.2.3 Написати програму у вигляді файлу-сценарію (m-файлу) для обчислення амплітудного і фазового спектрів періодичної послідовності $x(n)$ (див. п.1.2.2) з урахуванням (1.4) і виведення їх графіків:

- в шкалі дискретних нормованих частот k ;
- в шкалі абсолютних частот f (Гц).

1.2.4 Написати програму у вигляді файлу-сценарію (m-файлу) для обчислення ЗДПФ від ДПФ послідовності $x(n)$ (див. п.1.2.2) і виведення графіка отриманої послідовності в шкалі дискретного нормованого часу.

1.2.5 Написати програму у вигляді файлу-сценарію (m-файлу) для відновлення періодичного аналогового сигналу $x(t)$ за відліками ДПФ $X(k)$ періодичної послідовності $x(n)$ (1.17). Для обчислення значень сигналу $x(t)$ використовувати формулу (1.13).

У тих же точках обчислити значення початкового аналогового сигналу $x_{\text{поч}}(t)$, на основі якого отримана послідовність $x(nT)$ (1.16):

$$x_{\text{поч}}(t) = A_1 \cos(2\pi f_1 t + \pi/4) + A_2 \cos(2\pi f_2 t + \pi/8). \quad (1.18)$$

Вивести графіки:

- періодичної послідовності $x(n)$ (1.17) і модуля її ДПФ;
- відновленого аналогового сигналу $x(t)$ і його амплітудного спектра;

- початкового аналогового сигналу $x_{\text{поч}}(t)$ (1.18).

1.2.6 Побудувати у середовищі Simulink моделі за п.1.2.2 та 1.2.3.

Моделі повинні працювати незалежно від даних, що можуть знаходитись у Workspace, тобто й при очищеному Workspace.

1.3 Порядок виконання лабораторної роботи

1.3.1 Увімкніть ПЕОМ та запустіть програму MatLab.

1.3.2 Наберіть програму, написану за п. 1.2.2 та запустіть її на виконання. Отримані результати занесіть в протокол лабораторної роботи у вигляді скріншотів.

1.3.3 Наберіть програму, написану за п. 1.2.3 та запустіть її на виконання. Отримані результати занесіть в протокол лабораторної роботи у вигляді скріншотів.

1.3.4 Наберіть програму, написану за п. 1.2.4 та запустіть її на виконання. Отримані результати занесіть в протокол лабораторної роботи у вигляді скріншотів.

1.3.5 Наберіть програму, написану за п. 1.2.5 та запустіть її на виконання. Отримані результати занесіть в протокол лабораторної роботи у вигляді скріншотів.

1.3.6 Запустіть модель, підготовлену в середовищі Simulink за п. 1.2.2. Отримані результати занесіть в протокол лабораторної роботи у вигляді скріншотів.

1.3.7 Запустіть модель, підготовлену в середовищі Simulink за п. 1.2.3. Отримані результати занесіть в протокол лабораторної роботи у вигляді скріншотів.

1.4 Зміст звіту з лабораторної роботи

Звіт з лабораторної роботи повинен містити:

1.4.1 При допуску до лабораторної роботи

1.4.1.1 Ціль лабораторної роботи.

1.4.1.2 Короткі теоретичні відомості.

1.4.1.3 Написані за пунктами 1.2.2 - 1.2.5 програми з коментарями для середовища MatLab.

1.4.1.4 Блок-схему моделі в середовищі Simulink за пунктами 1.2.2 і 1.2.6 із зазначенням значень параметрів, що задаються для всіх блоків.

1.4.2 При здачі лабораторної роботи до матеріалів за п. 1.4.1 повинні бути додані результати виконання роботи.

1.4.2.1 Результати виконання програми за п. 1.3.2.

1.4.2.2 Результати виконання програми за п. 1.3.3.

1.4.2.3 Результати виконання програми за п. 1.3.4.

1.4.2.4 Результати виконання програми за п. 1.3.5.

1.4.2.5 Результати роботи моделі в середовищі Simulink за п. 1.3.6 у вигляді скріншотів.

1.4.2.6 Результати роботи моделі в середовищі Simulink за п. 1.3.7 у вигляді скріншотів.

1.4.2.6 Висновки.

1.5 Контрольні питання

- 1.5.1 Як пов'язаний дискретний нормований час з дискретним часом?
- 1.5.2 Як пов'язана частота f (Гц) з дискретною нормованою частотою?
- 1.5.3 Як пов'язаний модуль ДПФ послідовності зі спектром аналогового сигналу?
- 1.5.4 Як вибирається частота дискретизації?

Лабораторна робота № 2

Дослідження структур лінійних дискретних систем в системах MatLab і Simulink

Ціль заняття: дослідити структури рекурсивних і нерекурсивних лінійних дискретних систем і методи їх реалізації в системах Matlab і Simulink.

2.1 Теоретичні відомості

Структура лінійної дискретної системи (ЛДС) відображає алгоритм обчислення реакції при відомій передавальній функції і нульових початкових умовах (НПУ). [1,2,3,5]

Дробово-раціональній передавальній функції загального вигляду, що описується в Z-області формулою:

$$H(z) = \frac{\sum_{i=0}^{N-1} b_i z^{-i}}{1 + \sum_{k=1}^{M-1} a_k z^{-k}}, \quad (2.1)$$

відповідає алгоритм обчислення реакції, що описується різницеvim рівнянням (далі коротко алгоритм у вигляді різницевого рівняння) при НПУ:

$$y(n) = \sum_{i=0}^{N-1} b_i x(n-i) - \sum_{k=1}^{M-1} a_k y(n-k)$$

Однак передавальна функція (2.1) може мати інші види математичного представлення, кожному з яких буде співставлений свій алгоритм обчислення реакції ЛДС, що відображається відповідною структурою.

Ось чому вид структури визначається видом передавальної функції.

Структуру представляють у вигляді структурної схеми з умовними зображеннями операцій необхідних для обчислення реакції:

- затримки на період дискретизації T (в області дискретного нормованого часу $T = 1$);

- множення на константу;

- алгебраїчного додавання.

У структурі їм ставиться у відповідність три види елементів:

- елемент затримки (рисунок 2.1, а);

- помножувач (рисунок 2.1, б);

- суматор (рисунок 2.1, в).

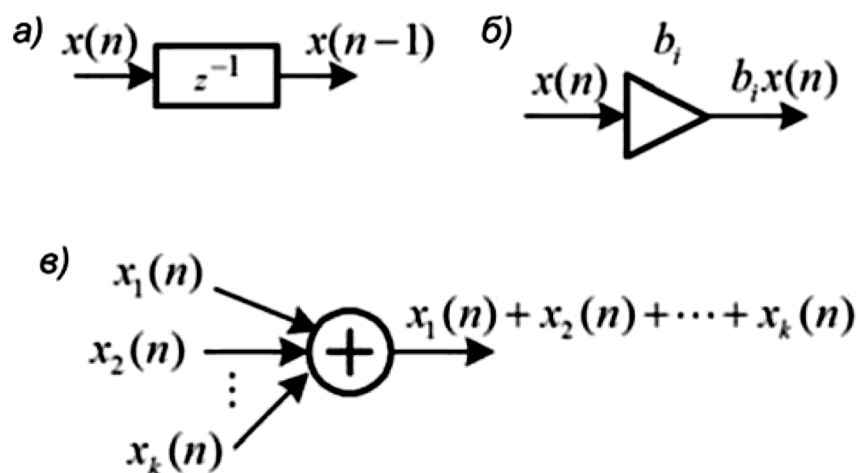


Рисунок 2.1 - Елементи структури ЛДС

2.1.1 Пряма структура та її модифікації

Пряма структура визначається дробово-раціональним видом передавальної функції [1,2]:

$$H(z) = \frac{b_0 + b_1 z^{-1} + b_2 z^{-2}}{1 + a_1 z^{-1} + a_2 z^{-2}}, \quad (2.2)$$

і відображає алгоритм розрахунку реакції у вигляді різницевого рівняння при НПУ (рис. 2.2):

$$y(n) = b_0 x(n) + b_1 x(n-1) + b_2 x(n-2) - a_1 y(n-1) - a_2 y(n-2). \quad (2.3)$$

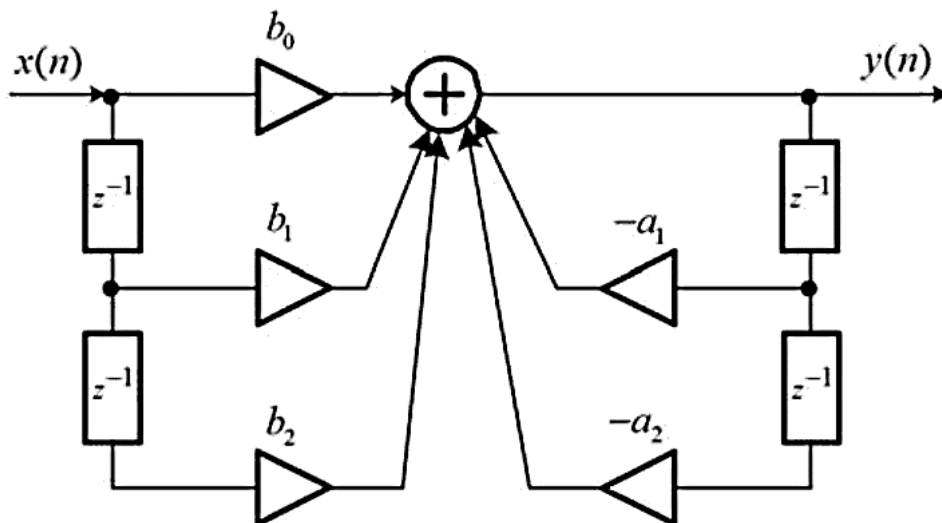


Рисунок 2.2 - Пряма структура ланки 2-го порядку

Пряма канонічна структура визначається еквівалентним представленням передавальної функції (2.2) у вигляді добутку:

$$H(z) = \frac{1}{1 + a_1 z^{-1} + a_2 z^{-2}} (b_0 + b_1 z^{-1} + b_2 z^{-2}) = H_1(z)H_2(z), \quad (2.4)$$

де передавальні функції визначаються наступним чином:

$$\begin{aligned} H(z) &= \frac{Y(z)}{X(z)} = \frac{V(z)}{X(z)} \cdot \frac{Y(z)}{V(z)}; \\ H_1(z) &= \frac{V(z)}{X(z)} = \frac{1}{1 + a_1 z^{-1} + a_2 z^{-2}}; \\ H_2(z) &= \frac{Y(z)}{V(z)} = b_0 + b_1 z^{-1} + b_2 z^{-2}. \end{aligned} \quad (2.5)$$

Записуючи послідовно різниці рівняння, що відповідають передавальним функціям $H_1(z)$ та $H_2(z)$, отримаємо алгоритм обчислення реакції у вигляді системи різницевих рівнянь при НПУ:

$$\begin{cases} v(n) = x(n) - a_1 v(n-1) - a_2 v(n-2); \\ y(n) = b_0 v(n) + b_1 v(n-1) + b_2 v(n-2), \end{cases} \quad (2.6)$$

який відображається структурою у вигляді послідовного (каскадного) з'єднання рекурсивної та нерекурсивної частин прямокутної структури (рисунок 2.3, а).

Об'єднуючи гілки в точці $v(n)$, отримуємо пряму канонічну структуру (рисунок 2.3, б) з вдвічі меншою кількістю елементів затримки у порівнянні з прямою структурою (див. рисунок 2.2).

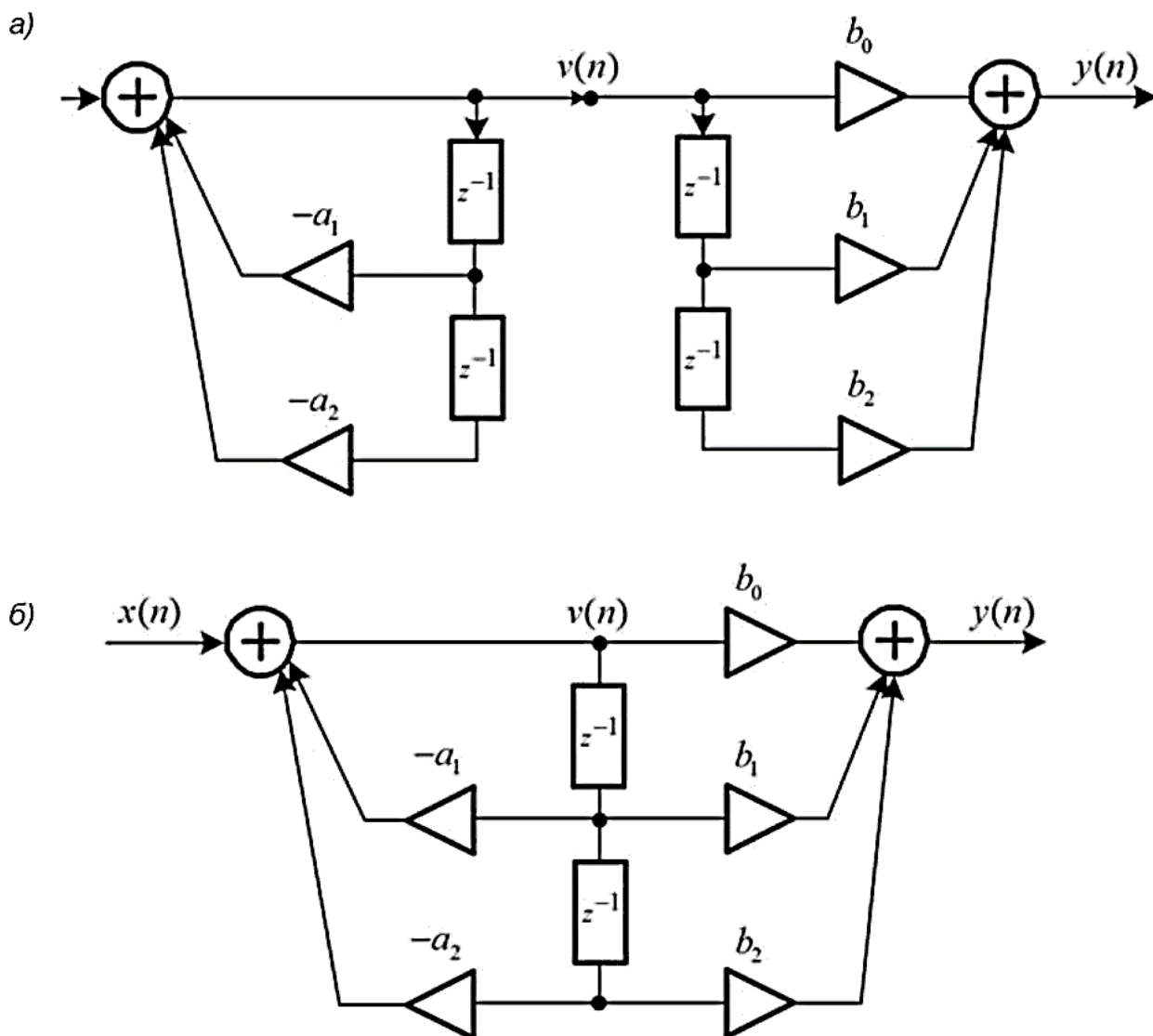


Рисунок 2.3 - Пряма канонічна структура ланки 2-го порядку: послідовне з'єднання прямих структур (а) та їх об'єднання (б)

Пряма транспонована структура ланки 2-го порядку визначається еквівалентним представленням передавальної функції (2.2) у вигляді модифікації добутку (2.5):

$$H(z) = \frac{1}{1 + (a_1 + a_2 z^{-1})z^{-1}} [b_0 + (b_1 + b_2 z^{-1})z^{-1}] = H_1(z)H_2(z), \quad (2.7)$$

де передавальна функція $H(z)$ визначена в (2.5), а передавальні функції $H_1(z)$ та $H_2(z)$ визначаються наступним чином:

$$H_1(z) = \frac{V(z)}{X(z)} = \frac{1}{1 + (a_1 + a_2 z^{-1})z^{-1}} = \frac{1}{1 + W_1(z)z^{-1}};$$

$$H_2(z) = \frac{Y(z)}{V(z)} = b_0 + (b_1 + b_2 z^{-1})z^{-1} = b_0 + W_2(z)z^{-1}.$$

Записуючи послідовно різницеві рівняння, що відповідають передавальним функціям $H_1(z)$ та $H_2(z)$, отримуємо алгоритм обрахування реакції у вигляді системи різницевих рівнянь при НПУ:

$$\begin{cases} v(n) = x(n) - w_1(n-1); \\ w_1(n) = a_1 v(n) + a_2 v(n-1); \\ y(n) = b_0 v(n) + w_2(n-1); \\ w_2(n) = b_1 v(n) + b_2 v(n-1), \end{cases} \quad (2.8)$$

який відображається структурою у вигляді каскадного з'єднання структур з передавальними функціями $H_1(z)$ та $H_2(z)$ (рисунок 2.4, а).

Об'єднавши гілки в точці $v(n)$, отримуємо пряму транспоновану структуру (рисунок 2.4, б).

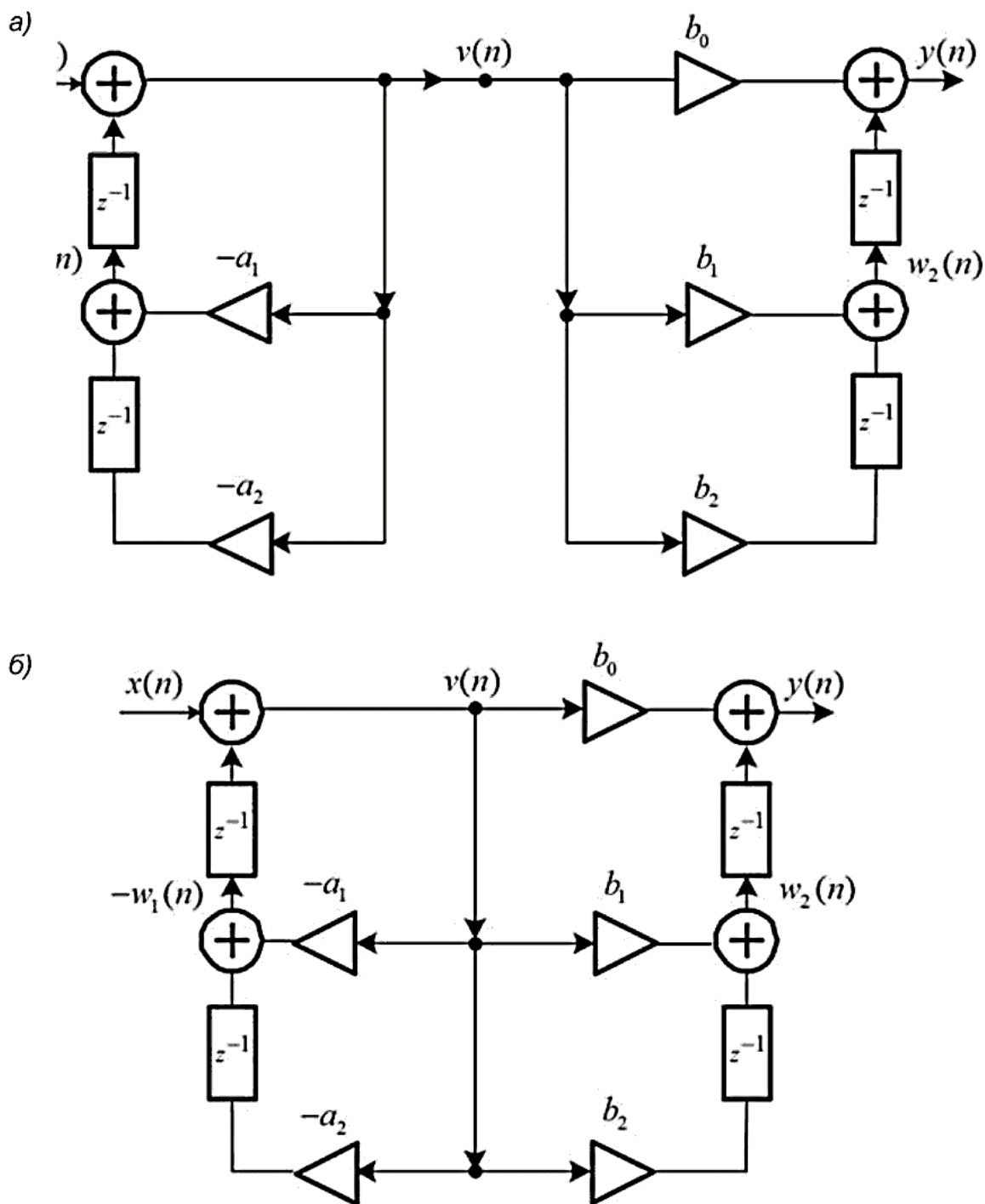


Рисунок 2.4 - Пряма транспонована структура ланки 2-го порядку: послідовне з'єднання структур (а) та їх об'єднання (б)

Пряма канонічна транспонована структура ланки 2-го порядку визначається еквівалентним представленням передавальної функції (2.2) у вигляді модифікації добутку (2.7):

$$H(z) = \left[b_0 + (b_1 + b_2 z^{-1}) z^{-1} \right] \frac{1}{1 + (a_1 + a_2 z^{-1}) z^{-1}} = H_1(z) H_2(z),$$

де передавальна функція $H(z)$ визначена в (2.5), а передавальні функції $H_1(z)$ та $H_2(z)$ визначаються наступним чином:

$$H_1(z) = \frac{V(z)}{X(z)} = b_0 + (b_1 + b_2 z^{-1}) z^{-1} = b_0 + W_2(z) z^{-1};$$

$$H_2(z) = \frac{Y(z)}{V(z)} = \frac{1}{1 + (a_1 + a_2 z^{-1}) z^{-1}} = \frac{1}{1 + W_1(z) z^{-1}}.$$

Записуючи послідовно різницеві рівняння, що відповідають передавальним функціям $H_1(z)$ та $H_2(z)$, отримуємо алгоритм обрахування реакції у вигляді системи різницевих рівнянь при НПУ:

$$\begin{cases} v(n) = b_0 x(n) + w_2(n-1); \\ w_2(n) = b_1 x(n) + b_2 x(n-1); \\ y(n) = v(n) - w_1(n-1); \\ w_1(n) = a_1 y(n) + a_2 y(n-1), \end{cases} \quad (2.9)$$

який відображається структурою у вигляді каскадного з'єднання структур з передавальними функціями $H_1(z)$ та $H_2(z)$ (рисунок 2.5, а).

Об'єднуючи гілки в точці $v(n)$, отримуємо пряму канонічну транспоновану структуру (рисунок 2.5, б) з вдвічі меншою кількістю елементів затримки у порівнянні з прямою (див. рисунок 2.2) та прямою транспонованою (див. рисунок 2.4, б) структурами.

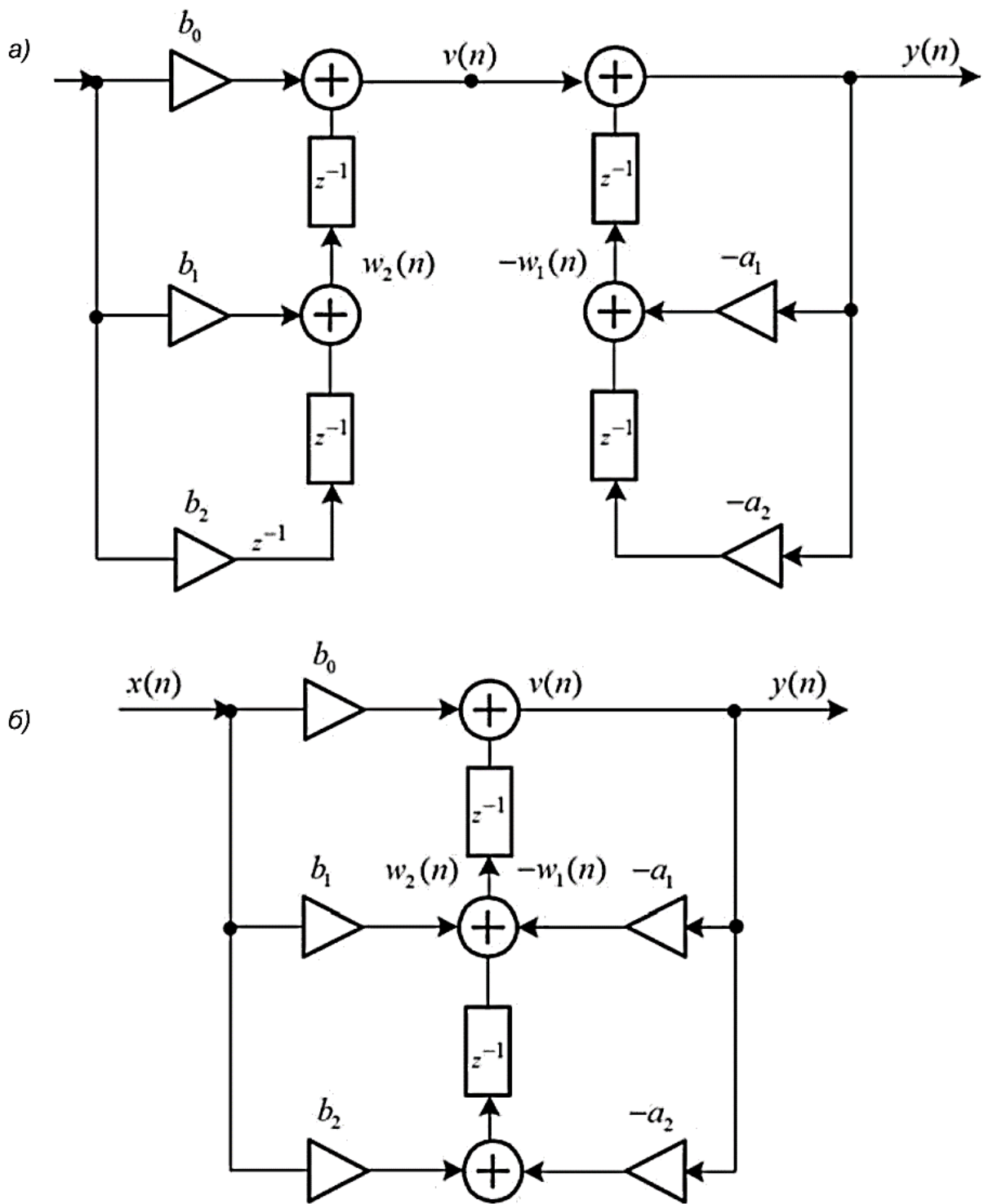


Рисунок 2.5 - Прямая канонична транспонована структура ланки 2-го порядку:
 послідовне з'єднання структур (а) та їх об'єднання (б)

2.1.2 Опис ЛДС в просторі станів

В алгоритмах (2.3), (2.6) та (2.9) для обчислення реакції $y(n)$ використовуються внутрішні сигнали структури на виходах суматорів. Можна, однак, для цієї цілі використовувати інші внутрішні сигнали – на виходах елементів затримки. Такі сигнали називають змінними стану, а відповідний алгоритм обчислення реакції – описом ЛДС у просторі станів (State-space). [1,2]

На практиці, а також при моделюванні в MATLAB, для опису ЛДС в просторі станів використовують пряму канонічну структуру.

Пояснимо на прикладі ланки 2-го порядку. Її пряма канонічна структура зі змінними стану $s_1(n)$ та $s_2(n)$ – сигналами на виходах елементів затримки – представлена на рисунку 2.6.

Запишемо систему рівнянь для обчислення реакції:

$$\begin{cases} s_1(n+1) = -a_1s_1(n) - a_2s_2(n) + x(n); \\ s_2(n+1) = s_1(n); \\ y(n) = b_1s_1(n) + b_2s_2(n) + b_0s_1(n+1). \end{cases}$$

Підставимо в останнє рівняння $s_1(n+1)$:

$$\begin{cases} s_1(n+1) = -a_1s_1(n) - a_2s_2(n) + x(n); \\ s_2(n+1) = s_1(n); \\ y(n) = (b_1 - b_0a_1)s_1(n) + (b_2 - b_0a_2)s_2(n) + b_0x(n), \end{cases} \quad (2.10)$$

де:

$$\begin{cases} s_1(n) = v(n-1); \\ s_2(n) = v(n-2). \end{cases} \quad (2.11)$$

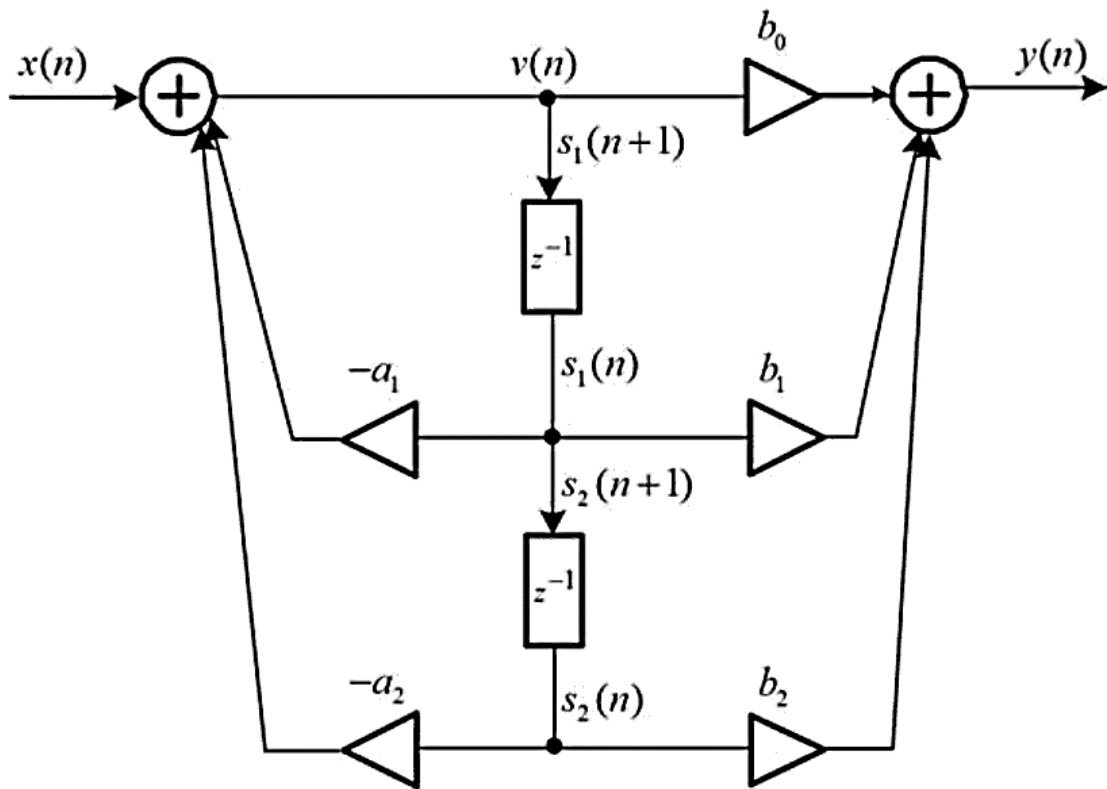


Рисунок 2.6 - Пряма канонічна структура ланки 2-го порядку зі змінними стану

Алгоритм обчислення реакції у вигляді системи рівнянь (2.10) представляє собою її опис у просторі станів, який включає два рівняння стану (по кількості елементів затримки) і одне рівняння спостереження.

Відповідно з даним описом, задаючи в (6.11) НПУ:

$$\begin{cases} s_1(0) = v(-1) = 0; \\ s_2(0) = v(-2) = 0, \end{cases}$$

обчислюється значення реакції $y(0)$:

$$\begin{cases} s_1(1) = -a_1 s_1(0) - a_2 s_2(0) + x(n) = x(0); \\ s_2(1) = s_1(0) = 0; \\ y(0) = b_0 x(0), \end{cases}$$

потім значення $y(1)$:

$$\begin{cases} s_1(2) = -a_1s_1(1) - a_2s_2(1) + x(1) = -a_1x(0) + x(1); \\ s_2(2) = s_1(1) = x(0); \\ y(1) = (b_1 - b_0a_1)s_1(1) + (b_2 - b_0a_2)s_2(1) + b_0x(1) = b_0x(1) + b_1x(0) - a_1b_0x(0) \end{cases}$$

і т. д.

Матричний запис (2.10):

$$\begin{cases} \begin{bmatrix} s_1(n+1) & s_2(n+1) \end{bmatrix} = \begin{bmatrix} -a_1 & -a_2 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} s_1(n) \\ s_2(n) \end{bmatrix} + \begin{bmatrix} 1 \\ 0 \end{bmatrix} x(n); \\ y(n) = \begin{bmatrix} (b_1 - b_0a_1) & (b_2 - b_0a_2) \end{bmatrix} \begin{bmatrix} s_1(n) \\ s_2(n) \end{bmatrix} + b_0x(n) \end{cases} \quad (2.12)$$

легко узагальнити на опис у просторі станів ЛДС будь-якого порядку ($M-1$):

$$\begin{cases} \mathbf{s}(n+1) = \mathbf{A}\mathbf{s}(n) + \mathbf{B}x(n); \\ y(n) = \mathbf{C}\mathbf{s}(n) + Dx(n), \end{cases} \quad (2.13)$$

де по аналогії з (2.12) прийняті наступні позначення:

$$\mathbf{s}(n+1) = \begin{bmatrix} s_1(n+1) & s_2(n+1) & \dots & s_{M-1}(n+1) \end{bmatrix} \quad \text{- вектор-}$$

рядок значень змінних стану (на виходах елементів затримки);

$$\mathbf{s}(n) = \begin{bmatrix} s_1(n) \\ s_2(n) \\ \vdots \\ s_{M-1}(n) \end{bmatrix} \quad \text{- вектор-стовбець значень сигналів на входах}$$

елементів затримки;

\mathbf{A} , \mathbf{B} , \mathbf{C} , D - параметри простору станів;

$$\mathbf{A} = \begin{bmatrix} -a_1 & -a_2 & \dots & -a_{M-2} & -a_{M-1} \\ 1 & 0 & \dots & \dots & 0 \\ 0 & 1 & \dots & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & 0 \\ 0 & 0 & \dots & 1 & 0 \end{bmatrix} - \text{квадратна матриця}$$

порядку $(M-1)$;

$$\mathbf{B} = \begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix} - \text{вектор-стовбець довжини } (M-1);$$

$\mathbf{C} = [(b_1 - b_0 a_1) \quad (b_2 - b_0 a_2) \quad \dots \quad (b_{M-1} - b_0 a_{M-1})]$ - вектор-рядок довжини $(M-1)$;

$D = b_0$ - скаляр.

На основі (2.13), застосувавши Z -перетворення до рівнянь, можна отримати представлення передавальної функції у просторі станів:

$$H(z) = \frac{Y(z)}{X(z)} = \mathbf{C}(z\mathbf{I} - \mathbf{A})^{-1}\mathbf{B} + D,$$

де \mathbf{I} - одинична матриця порядку $(M-1)$.

2.1.3 Каскадна структура

Каскадна структура (рисунок 2.7) визначається еквівалентним представленням дробово-раціональної передавальної функції (2.1) у вигляді добутку[1,2]:

$$H(z) = G \prod_{k=1}^L \left(\frac{1 + b_{1k}z^{-1} + b_{2k}z^{-2}}{1 + a_{1k}z^{-1} + a_{2k}z^{-2}} \right) = G \prod_{k=1}^L H_k(z), \quad (2.14)$$

і відображає алгоритм обчислення реакції $y(n)$ у вигляді системи різницевих рівнянь з послідовним обчисленням реакцій біквдратних ланок $y_k(n)$, $k = 1, 2, \dots, L$, $y_L(n) = y(n)$, з передавальними функціями $H_k(z)$.

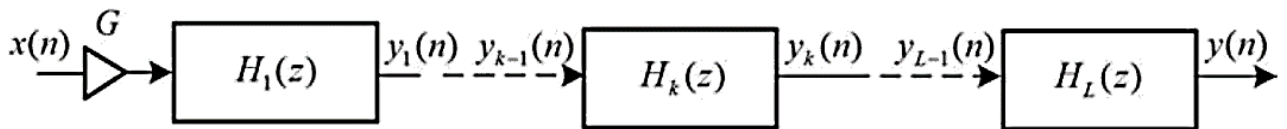


Рисунок 2.7 - Каскадна структура рекурсивної ЛДС

Структура ланок визначає вид каскадної структури:

- з прямою структурою ланок (Direct-form I, SOS);
- з прямою канонічною структурою ланок (Direct-form II, SOS);
- з прямою транспонованою структурою ланок (Direct-form I transposed, SOS);
- з прямою канонічною транспонованою структурою ланок (Direct-form II transposed, SOS).

SOS – скорочення від Second-Order Sections – ланки 2-го порядку (біквдратні).

Структури ланок визначають алгоритм обчислення реакції каскадної структури. Наприклад, при прямій структурі біквдратних ланок цей алгоритм буде описуватися у вигляді наступної системи різницьових рівнянь:

$$\begin{cases} y_1(n) = G[x(n) + b_{11}x(n-1) + b_{21}x(n-2)] - a_{11}y_1(n-1) - a_{21}y_1(n-2); \\ y_2(n) = [y_1(n) + b_{12}y_1(n-1) + b_{22}y_1(n-2)] - a_{12}y_2(n-1) - a_{22}y_2(n-2); \\ \dots \\ y(n) = [y_{L-1}(n) + b_{1L}y_{L-1}(n-1) + b_{2L}y_{L-1}(n-2)] - a_{1L}y(n-1) - a_{2L}y(n-2). \end{cases}$$

2.1.4 Паралельна структура

Паралельна структура (рисунок 2.8) визначається еквівалентним представленням дробово-раціональної передавальної функції (2.1) у вигляді суми

$$H(z) = \sum_{k=1}^L \left(\frac{B_{0k} + B_{1k}z^{-1}}{1 + a_{1k}z^{-1} + a_{2k}z^{-2}} \right) = \sum_{k=1}^L H_k(z), \quad (2.15)$$

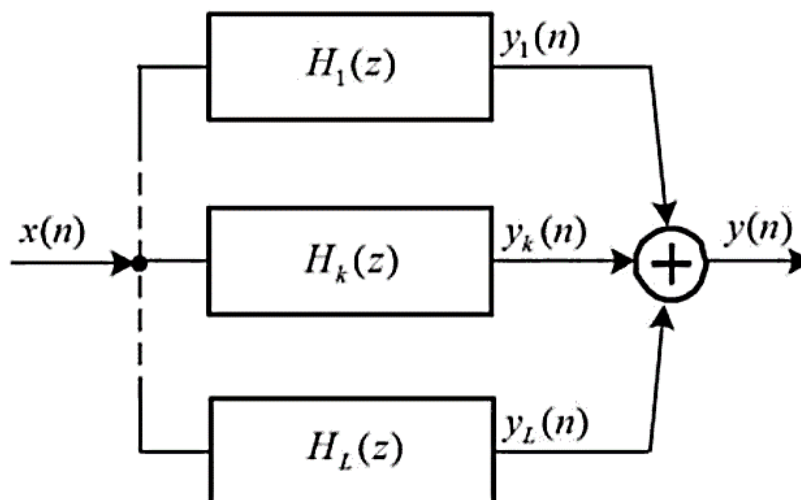


Рисунок 2.8 - Паралельна структура рекурсивної ЛДС

та відображає алгоритм обчислення реакції $y(n)$ у вигляді рівняння з паралельним обчисленням реакцій ланок з передавальними функціями $H_k(z)$ при однаковому впливі $x(n)$ [1]:

$$y(n) = \sum_{k=1}^L y_k(n).$$

Згідно з (2.15), в z -області йому відповідає відношення вхід/вихід:

$$Y(z) = H(z)X(z) = \sum_{k=1}^L H_k(z)X(z) = \sum_{k=1}^L \left(\frac{B_{0k} + B_{1k}z^{-1}}{1 + a_{1k}z^{-1} + a_{2k}z^{-2}} \right) X(z) = \sum_{k=1}^L Y_k(z).$$

Структура ланок визначає алгоритм обчислення їх реакцій $y_k(n)$, $k = 1, 2, \dots, L$, і реакції паралельної структури, що дорівнюють сумі реакцій ланок.

Наприклад, при прямій структурі ланок цей алгоритм буде описуватися у вигляді рівняння:

$$y(n) = \sum_{k=1}^L y_k(n) = \sum_{k=1}^L [B_{0k}x(n) + B_{1k}x(n-1) - a_{1k}v_k(n-1) - a_{2k}v_k(n-2)].$$

2.1.5 Структури нерекурсивних ЛДС

Пряма структура (Direct-form FIR – Finite Impulse Response) нерекурсивної ЛДС (рисунок 2.9) визначається раціональною передавальною функцією [1]:

$$H(z) = \sum_{i=0}^{N-1} b_i z^{-i}, \quad (2.16)$$

і відображає алгоритм обчислення реакції $y(n)$ у вигляді різницевого рівняння при НПУ:

$$y(n) = \sum_{i=0}^{N-1} b_i x(n-i).$$

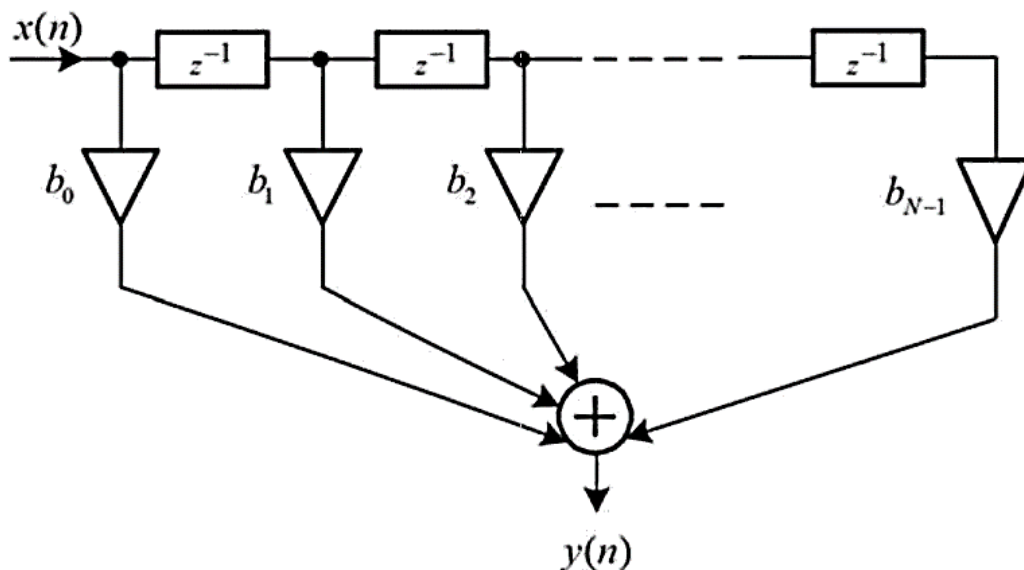


Рисунок 2.9 - Пряма структура нерекурсивної ЛДС

Різновидом прямої структури нерекурсивної ЛДС являється пряма транспонована структура (Direct-form FIR Transposed), еквівалентна прямій канонічній транспонованій структурі на рисунку 2.5 при $a_1 = a_2 = 0$.

Каскадна структура нерекурсивної ЛДС визначається передавальною функцією у вигляді добутку:

$$H(z) = G \prod_{i=1}^L (1 + b_{1i} z^{-1} + b_{2i} z^{-2}) = G \prod_{k=1}^L H_k(z), \quad (2.17)$$

і відображає алгоритм обчислення реакції $y(n)$ у вигляді системи рівнянь з послідовним обчисленням реакції ланок з передавальними функціями $H_k(z)$. Ланки можуть мати пряму або пряму транспоновану структуру.

Дана структура на практиці не знайшла широкого застосування.

2.1.6 Опис структур в MATLAB

В MATLAB структура ЛДС описується у вигляді об'єкту **dfilt** (Discrete-time filter object) [1,2,6]:

Hd = dfilt.structure(input1,...)

де:

Hd – ім'я об'єкту **dfilt**, обране користувачем;

structure – функція, що задає структуру об'єкта **dfilt**;

input1,... – список параметрів функції **structure**.

В таблиці 2.1 приведений список типових структур ЛДС, які підтримує MATLAB, та функцій **structure** з параметрами, що їм відповідають. При описанні структур використовуються аббревіатури НІХ та КІХ.

Перші чотири структури НІХ ЛДС на практиці, як правило, застосовують для рекурсивних ланок 2-го порядку.

Таблиця 2.1 Типові структури ЛДС та їх опис у вигляді об'єктів **dfilt**

| Структура | Функція structure | Параметри функції structure |
|--|--------------------------|--|
| Direct-Form I (пряма НІХ) | df1(a,b) | a,b – вектори коефіцієнтів передавальної функції (2.1) |
| Direct-Form II (пряма канонічна НІХ) | df2(a,b) | |
| Direct-Form I Transposed (пряма транспонована НІХ) | df1t(a,b) | |
| Direct-Form II Transposed (пряма канонічна транспонована НІХ) | df2t(a,b) | |
| Direct-Form I, SOS (каскадна НІХ з біквдратних ланок з прямою структурою) | df1sos(SOS,G) | SOS,G – матриця коефіцієнтів і вектор коефіцієнтів підсилення біквдратних ланок в (2.14) |
| Direct-Form II, SOS (каскадна НІХ з біквдратних ланок з прямою канонічною структурою) | df2sos(SOS,G) | |
| Direct-Form I Transposed, SOS (каскадна НІХ з біквдратних ланок з прямою транспонованою структурою) | df1tsos(SOS,G) | |
| Direct-Form II Transposed, SOS (каскадна НІХ з біквдратних ланок з прямою канонічною транспонованою структурою) | df2tsos(SOS,G) | |
| State-space (пряма канонічна НІХ з описом в просторі станів) | statespace (A,B,C,D,) | A,B,C,D – параметри системи рівнянь (2.13) |
| Direct-Form FIR (пряма КІХ) | dffir(Num) | Num – вектор коефіцієнтів передавальної функції (2.16) |
| Direct-Form FIR Transposed (пряма транспонована КІХ) | dffirt(Num) | |
| Direct-Form Symetric FIR (пряма приведена КІХ з симетричною імпульсною характеристикою) | dfsymfir(Num) | |
| Direct-Form Antisymetric FIR (пряма приведена КІХ з антисиметричною імпульсною характеристикою) | dfasymfir(Num) | |

Вихідними параметрами об'єкту **Hd** являються його властивості, що залежать від конкретної структури та типу арифметики.

Список властивостей виводиться у вікні **Command Window** по імені об'єкту:

```
>> Hd
```

Вивід конкретної властивості об'єкту виконується за допомогою функції:

```
>> get(Hd, 'властивість')
```

Для структур КІХ ЛДС список властивостей об'єкту **dfilt** включає:

FilterStructure – структура ЛДС;

Arithmetic – тип арифметики, під яким приймають формат представлення даних. По замовчуванню обирається **Arithmetic: 'double'** – представлення даних з плаваючою комою з подвійною точністю;

Numerator – вектор коефіцієнтів передавальної функції;

PersistentMemory – початкові умови для обчислення реакції. За замовчуванням **false**, що відповідає НПУ.

Для створеного об'єкта **dfilt** властивість **Numerator** співпадає з параметром **Num** (див. таблицю 2.1).

Приклад 2.1. Для КІХ ЛДС з передавальною функцією (2.16):

$$H(z) = \sum_{i=0}^{N-1} b_i z^{-i},$$

що дорівнює

$$H(z) = 0,05 + 0,1z^{-1} + 0,2z^{-2} + 0,5z^{-3} + 0,2z^{-4} + 0,1z^{-5} + 0,05z^{-6},$$

виберемо пряму структуру (Direct-Form FIR), створимо об'єкт **dfilt** з іменем **H**:

```
>> Num = [0.05 0.1 0.2 0.5 0.2 0.1 0.05];  
>> H = dfilt.dffir(Num);
```

та виведемо властивість **Numerator**:

```
>> get(H, 'Numerator')  
ans =  
    0.0500    0.1000    0.2000    0.5000    0.2000    0.1000    0.0500
```

яке співпадає з параметром **Num**.

Для каскадних структур НІХ ЛДС замість властивості **Numerator** будемо мати дві наступні властивості:

sosMatrix – матриця коефіцієнтів;

ScaleValues – вектор коефіцієнтів підсилення.

Визначимо параметри **SOS** та **G** (див. таблицю 2.1), а потім пояснимо дані властивості.

Передавальній функції у вигляді:

$$H(z) = G \prod_{k=1}^L \left(\frac{1 + b_{1k}z^{-1} + b_{2k}z^{-2}}{1 + a_{1k}z^{-1} + a_{2k}z^{-2}} \right), \quad (2.18)$$

де $G = b_{01} \cdot b_{02} \cdot \dots \cdot b_{0L}$, відповідає матриця коефіцієнтів;

$$\text{SOS} \Rightarrow \begin{bmatrix} 1 & b_{11} & b_{21} & 1 & a_{11} & a_{21} \\ 1 & b_{12} & b_{22} & 1 & a_{12} & a_{22} \\ \dots & \dots & \dots & \dots & \dots & \dots \\ 1 & b_{1L} & b_{2L} & 1 & a_{1L} & a_{2L} \end{bmatrix}, \quad (2.19)$$

та вектор коефіцієнтів підсилення (скаляр):

$$G \Rightarrow G.$$

Еквівалентному представленню передавальної функції (2.18)

$$H(z) = \prod_{k=1}^L b_{0k} \left(\frac{1 + b_{1k}z^{-1} + b_{2k}z^{-2}}{1 + a_{1k}z^{-1} + a_{2k}z^{-2}} \right) \quad (2.20)$$

відповідає матриця коефіцієнтів (2.19) та вектор коефіцієнтів підсилення

$$G \Rightarrow [b_{01} \quad b_{02} \quad \dots \quad b_{0L}].$$

Еквівалентному представленню передавальної функції (2.20)

$$H(z) = \prod_{k=1}^L \left(\frac{b_{0k} + \tilde{b}_{1k}z^{-1} + \tilde{b}_{2k}z^{-2}}{1 + a_{1k}z^{-1} + a_{2k}z^{-2}} \right), \quad (2.21)$$

де $b_{1k} = \tilde{b}_{1k}/b_{0k}$, $b_{2k} = \tilde{b}_{2k}/b_{0k}$, відповідає матриця коефіцієнтів

$$\text{SOS} \Rightarrow \begin{bmatrix} b_{01} & \tilde{b}_{11} & \tilde{b}_{21} & 1 & a_{11} & a_{21} \\ b_{02} & \tilde{b}_{12} & \tilde{b}_{22} & 1 & a_{12} & a_{22} \\ \dots & \dots & \dots & \dots & \dots & \dots \\ b_{0L} & \tilde{b}_{1L} & \tilde{b}_{2L} & 1 & a_{1L} & a_{2L} \end{bmatrix}; \quad (2.22)$$

та вектор коефіцієнтів підсилення (скаляр)

$$G \Rightarrow 1.$$

Для створеного об'єкту **dfilt** властивість **sosMatrix** співпадає з параметром *SOS*.

Властивість **ScaleValues** не співпадає з параметром *G* та має вигляд вектора-стовбчика довжини $(L+1)$, елементи якого дорівнюють:

Перший – коефіцієнт підсилення b_{01} на вході першої ланки;

Другий – коефіцієнт підсилення b_{02} на вході другої ланки і т.д.;

Останній – коефіцієнт підсилення на виході структури.

Цей коефіцієнт розраховується автоматично та повинен враховуватися при обчисленні реакції. В більшості випадків він дорівнює одиниці.

Приведемо приклади створення об'єктів **dfilt** при трьох еквівалентних представленнях передавальної функції.

Приклад 2.2. Для НІХ ЛДС з передавальною функцією (2.18)

$$H(z) = 0,1044 \frac{1 + 1,7021z^{-1} + 1z^{-2}}{1 - 0,6106z^{-1} + 0,3029z^{-2}} \cdot \frac{1 + 0,886z^{-1} + 1z^{-2}}{1 - 0,0013z^{-1} + 0,8093z^{-2}} \quad (2.23)$$

виберемо каскадну структуру з прямою структурою ланок (Direct-Form I, SOS), створимо об'єкт **dfilt** з іменем **H1**:

```
>> SOS = [1 1.7021 1 1 -0.6106 0.3029;1 0.886 1 1 -0.0013 0.8093];  
>> G = 0.1044;  
>> H1 = dfilt.df1sos(SOS,G);
```

та виведемо його властивість **sosMatrix**:

```
>> get(H1,'sosMatrix')  
ans =  
    1.0000    1.7021    1.0000    1.0000   -0.6106    0.3029  
    1.0000    0.8860    1.0000    1.0000   -0.0013    0.8093
```

яке співпадає з параметром *SOS* (6.20)б та властивість **ScaleValues**:

```
>> get(H1,'ScaleValues')  
ans =  
    0.1044  
    1.0000  
    1.0000
```

представлене вектором-стовбцем довжини $L + 1 = 3$.

Приклад 2.3. Для НІХ ЛДС з передавальною функцією (2.20)

$$H(z) = 0,5477 \frac{1 + 1,7021z^{-1} + 1z^{-2}}{1 - 0,6106z^{-1} + 0,3029z^{-2}} \cdot 0,1906 \frac{1 + 0,886z^{-1} + 1z^{-2}}{1 - 0,0013z^{-1} + 0,8093z^{-2}}$$

виберемо ту ж структуру (Direct-Form I, SOS), створимо об'єкт **dfilt** з іменем **H2**:

```
>> SOS = [1 1.7021 1 1 -0.6106 0.3029; 1 0.886 1 1 -0.0013 0.8093];
>> G = [0.5477 0.1906];
>> H2 = dfilt.df1sos(SOS,G);
```

та виведемо його властивість **sosMatrix**:

```
>> get(H2, 'sosMatrix')
ans =
    1.0000    1.7021    1.0000    1.0000   -0.6106    0.3029
    1.0000    0.8860    1.0000    1.0000   -0.0013    0.8093
```

яка співпадає з параметром *SOS* (6.19), та властивість **ScaleValues**:

```
>> get(H3, 'ScaleValues')
ans =
    0.5477
    0.1906
    1.0000
```

представлену вектором-стовбцем довжини $L + 1 = 3$.

Приклад 2.4. Для НІХ ЛДС з передавальною функцією (2.21)

$$H(z) = \frac{0,5477 + 0,9322z^{-1} + 0,5477z^{-2}}{1 - 0,6106z^{-1} + 0,3029z^{-2}} \cdot \frac{0,1906 + 0,1689z^{-1} + 0,1906z^{-2}}{1 - 0,0013z^{-1} + 0,8093z^{-2}},$$

виберемо ту ж структуру (Direct-Form I, SOS), створимо об'єкт **dfilt** з іменем **H3**:

```
>> SOS = [0.5477 0.9322 0.5477 1 -0.6106 0.3029;0.1906 0.1689 0.1906 1 -0.0013
0.8093];
>> G = 1;
>> H3 = dfilt.df1sos(SOS,G);
```

та виведемо його властивість **sosMatrix**:

```
>> get(H3,'sosMatrix')
ans =
    0.5477    0.9322    0.5477    1.0000   -0.6106    0.3029
    0.1906    0.1689    0.1906    1.0000   -0.0013    0.8093
```

яка співпадає з параметром *SOS* (6.22), та властивість **ScaleValues**:

```
>> get(H3,'ScaleValues')
ans =
     1
     1
     1
```

представлену вектором-стовбцем довжини $L + 1 = 3$.

Паралельна структура НІХ ЛДС з передавальною функцією (2.15) не підтримується MATLAB, але при необхідності може бути реалізована за допомогою функції **structure**:

parallel(Hd1,Hd2,...)

де в дужках вказані імена об'єктів **dfilt** для структур рекурсивних ланок 2-го порядку, які попередньо створюються користувачем.

При відомій структурі ЛДС обчислення її реакції виконується за допомогою функції:

y = filter(H,x)

де **H** – ім'я об'єкту **dfilt**; x, y – вектори відліків вхідного та вихідного сигналів.

2.2 Підготовка до лабораторної роботи

2.2.1 Ознайомитись з короткими теоретичними відомостями і порядком роботи з середовищем MatLab.

2.2.2 Для нерекурсивної ЛДС (КІХ-системи) розробити і зобразити в графічному вигляді пряму і пряму транспоновану структури, а також записати різницеве рівняння.

Параметри для реалізації ЛДС вибираються з таблиці 1, згідно заданого варіанта¹.

2.2.3 Написати програму у вигляді файлу-сценарію (m-файлу) для реалізації прямої і прямої транспонованої структур нерекурсивної ЛДС (КІХ-системи), і виведення їх властивостей.

Таблиця 1 – Параметри ЛДС

| № вар. / Парам. | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|-----------------|-----|------|------|------|------|------|------|------|------|------|
| b_0 | 0,1 | 1 | 0,2 | 0 | 1,7 | 0,5 | 2 | 0,4 | 0 | 2,5 |
| b_1 | 0 | 2 | 1 | 4 | 1 | 5 | 0 | 1 | 2 | 5 |
| b_2 | 2 | 0 | 1,2 | 1,8 | 2,1 | 3,4 | 1,1 | 0 | 5,6 | 3,5 |
| b_3 | 1,8 | 2,4 | 0 | 3 | 0,4 | 1,5 | 2,4 | 2,6 | 0 | 2,5 |
| b_4 | 2,2 | 1,5 | 2,5 | 2 | 2,6 | 1,2 | 2 | 2 | 2,4 | 0 |
| b_5 | 0,9 | 1,1 | 1,5 | 5,6 | 0 | 2,7 | 4,6 | 2,8 | 1,5 | 2 |
| b_6 | 1,5 | 1,2 | 1,2 | 2,3 | 3,4 | 0 | 2,2 | 0,7 | 1,1 | 1,8 |
| a_1 | -1 | 2,6 | 0 | 1,3 | -2 | 2 | 2,6 | 0 | 1,2 | 2,2 |
| a_2 | 1,2 | -2 | -2 | 0 | 0,9 | -1,8 | -2 | -2,4 | -2,1 | 0 |
| a_3 | 2,6 | 2,8 | 2,8 | -1,9 | 0 | 2,2 | 2,8 | 1,5 | 0,4 | -2,7 |
| a_4 | -2 | -0,7 | 0,9 | 5 | -2 | 0 | -2,1 | -1,1 | 0 | 2,6 |
| a_5 | 0 | -2,1 | 2,1 | 3,4 | -1,8 | -1,9 | 0 | 1,2 | 2,1 | -2 |
| a_6 | 0,7 | 0 | -1,5 | -2 | 2,2 | 1,5 | 2,5 | 0 | -0,4 | 2,8 |

Примітка 1. Номер варіанта визначається номером студента у списку групи (підгрупи або об'єднаного списку груп згідно їх номерів). Якщо номер за списком 11 (21), то параметри $b_0, b_1, b_2, b_3, a_1, a_2, a_3$ вибираються з варіанта 1, а параметри $b_4, b_5, b_6, a_4, a_5, a_6$ – з варіанта 10. Якщо номер за списком перевищує 11 (21), то параметри $b_0, b_1, b_2, b_3, a_1, a_2, a_3$ вибираються з варіанта 1 (2), а параметри $b_4, b_5, b_6, a_4, a_5, a_6$ - з варіанта, номер якого дорівнює другій цифрі в номері за списком. Наприклад, для номера 12 параметри $b_0, b_1, b_2, b_3, a_1, a_2, a_3$ вибираються з варіанта 1, а $b_4, b_5, b_6, a_4, a_5, a_6$ - з варіанта 2.

2.2.4 Для рекурсивної ЛДС (НІХ-системи) розробити і зобразити в графічному вигляді пряму, пряму канонічну, пряму транспоновану і пряму канонічну транспоновану структури, а також записати їх передавальну функцію і різницеве рівняння.

Параметри для реалізації ЛДС вибираються з таблиці 1, згідно заданому варіанту.

2.2.5 Написати програму у вигляді файлу-сценарію (m-файлу) для реалізації прямої, прямої канонічної, прямої транспонованої і прямої канонічної транспонованої структур рекурсивної ЛДС (НІХ-системи), і виведення їх властивостей.

2.2.6 Побудувати в середовищі Simulink моделі прямої і прямої транспонованої структур нерекурсивної ЛДС (КІХ-системи).

Моделі повинні працювати незалежно від даних, що можуть знаходитись у Workspace, тобто й при очищеному Workspace.

2.3 Порядок виконання лабораторної роботи

2.3.1 Увімкніть ПЕОМ та запустіть програму MatLab.

2.3.2 Наберіть програму, написану за п. 2.2.3 та запустіть її на виконання. Отримані результати занесіть в протокол лабораторної роботи у вигляді скріншотів.

2.3.3 Побудуйте імпульсні характеристики розроблених за п.п. 2.2.2 і 2.2.3 нерекурсивних ЛДС (КІХ-систем).

2.3.4 Змініть коефіцієнти в 10 разів: для непарних номерів коефіцієнтів шляхом зменшення, для парних номерів коефіцієнтів (включаючи нульовий) шляхом збільшення. Для змінених коефіцієнтів повторіть виконання п.п. 2.3.2 и 2.3.3.

2.3.5 Наберіть програму, написану за п. 2.2.5 та запустіть її на виконання. Отримані результати занесіть в протокол лабораторної роботи у вигляді скріншотів.

2.3.6 Побудуйте імпульсні характеристики розроблених за п.п. 2.2.4 і 2.2.5 рекурсивних ЛДС (НІХ-систем).

2.3.7 Змініть коефіцієнти в 10 разів: для непарних номерів коефіцієнтів шляхом зменшення, для парних номерів коефіцієнтів (включаючи нульовий) шляхом збільшення. Для змінених коефіцієнтів повторіть виконання п.п. 2.3.5 и 2.3.6.

2.3.8 Запустіть моделі, підготовлені в середовищі Simulink за п. 2.2.6, на виконання. Отримані результати занесіть в протокол лабораторної роботи у вигляді скріншотів.

2.4 Зміст звіту з лабораторної роботи

Звіт з лабораторної роботи повинен містити:

2.4.1 При допуску до лабораторної роботи

2.4.1.1 Ціль лабораторної роботи.

2.4.1.2 Короткі теоретичні відомості.

2.4.1.3 Розроблені за п.п. 2.2.2 і 2.2.4 структури ЛДС.

2.4.1.4 Написані за по п.п. 2.2.3 і 2.2.5 програми з коментарями для середовища MatLab.

1.4.1.4 Блок-схеми моделей в середовищі Simulink за пунктом 2.2.6.

2.4.2 При здачі лабораторної роботи до матеріалів за п. 2.4.1 повинні бути додані результати виконання роботи:

2.4.2.1 Результати виконання програми за п. 2.2.3 з вихідними (п. 2.2.2) і зміненими (п. 2.3.4) коефіцієнтами.

2.4.2.2 Побудовані за п. 2.3.3 імпульсні характеристики.

2.4.2.3 Результати виконання програми за п. 2.2.5 з вихідними (п. 2.3.5) і зміненими (п. 2.3.7) коефіцієнтами.

2.4.2.4 Побудовані за п. 2.3.6 імпульсні характеристики.

2.4.2.5 Результати роботи моделей в середовищі Simulink за п. 2.3.8 у вигляді скріншотів.

2.4.2.6 Висновки.

2.5 Контрольні питання

2.5.1 Які обчислення і які відповідні їм елементи використовуються в ЛДС?

2.5.2 Які різновиди структур рекурсивних ЛДС (НІХ-систем) використовуються в системі MatLab?

2.5.3 Які різновиди структур нерекурсивних ЛДС (КІХ-систем) використовуються в системі MatLab?

2.5.4 Які особливості каскадної структури рекурсивної ЛДС?

2.5.5 Які особливості паралельної структури рекурсивної ЛДС?

2.5.6 За допомогою яких структур описуються ЛДС в системі MatLab?

Лабораторна робота № 3

Дослідження методів синтезу КІХ-фільтрів за допомогою системи «MatLab»

Ціль роботи: дослідити методи синтезу цифрових фільтрів з кінцевою імпульсною характеристикою в середовищі MATLAB.

3.1 Теоретичні відомості

3.1.1 Цифрові фільтри

Цифрові фільтри призначені для обробки (фільтрації) сигналів, представлених у вигляді часових рядів. По виду імпульсної характеристики цифрові фільтри прийнято ділити на фільтри з кінцевою імпульсною характеристикою (КІХ-фільтри) і з нескінченною імпульсною характеристикою (НІХ-фільтри). [1,2,4,5]

Зазвичай нерекурсивні цифрові фільтри мають КІХ, рекурсивні - НІХ. Перші на відміну від других можуть мати лінійну фазочастотну характеристику (ФЧХ), але вимагають більш значних обчислювальних затрат. Рекурсивні кінцево-імпульсні характеристики фільтрів (РКІХФ) також можуть мати лінійну ФЧХ, але при набагато менших обчислювальних витратах в порівнянні з нерекурсивними фільтрами.

Цифровий фільтр повністю описується його імпульсною характеристикою. Імпульсна характеристика - це реакція фільтра на одиничний імпульс, поданий на його вхід. Інша характеристика, яка використовується при вивченні і

проектуванні цифрових фільтрів ЦФ, називається перехідною. Вона показує, для чого призначений фільтр з різними частотними складовими вхідного сигналу, як він змінює спектр сигналу. Обидві ці характеристики не є незалежними, вони пов'язані між собою перетворенням Фур'є. Знаючи одну з них, за допомогою прямого або зворотного перетворення Фур'є можна отримати іншу. Перехідна характеристика показує, як фільтр змінює амплітуду і фазу частотних складових вхідного сигналу. Так як перехідна характеристика - це комплексна функція частоти, то зобразити її на графіку досить важко. Тому прийнято зображати окремо залежність амплітуди від частоти - амплітудно-частотна характеристика (АЧХ) і залежність зсуву фази від частоти - фазо-частотна характеристика (ФЧХ).

Цифрові фільтри також прийнято ділити по виду АЧХ. Зазвичай виділяють фільтри низьких частот (ФНЧ), фільтри високих частот (ФВЧ), смугові фільтри (СФ), загороджуючі фільтри (ЗФ), диференціюючі фільтри і деякі інші.

Серед безлічі аналітичних/ітераційних методів розрахунку КІХ-фільтрів слід виділити два основні методи:

- метод вагових функцій (вікон);
- метод частотної вибірки.

Крім того, можливе застосування числових методів розрахунку, серед яких виділяються наступні критерії розрахунку:

- критерій мінімального середнього квадрата помилки (мінімального СКВ);
- критерій рівнохвильового наближення.

Більшість методів синтезу КІХ-фільтрів реалізовано в пакеті програм середовища **MATLAB**.

3.1.2 Розрахунок цифрових КІХ-фільтрів методом вікон за допомогою команд MATLAB

Розрахунок коефіцієнтів a_k КІХ-фільтра (нерекурсивного) методом вікон здійснюється за допомогою функції **fir1**. [1,2,6]

Функція **fir1** реалізує обчислення за методом зворотного перетворення Фур'є з використанням вікон. Синтаксис команди:

a=fir1(n,Wn,'ftype',window,'normalization'),

де **n** – порядок фільтра – ціле парне число (кількість коефіцієнтів фільтра дорівнює **n+1**);

Wn – відносна частота зрізу (по відношенню до частоти Найквіста, що дорівнює половині частоти дискретизації **Fd**) - число в діапазоні (0,1); якщо фільтр смуговий або режекторний тоді **Wn** є вектором з двох чисел;

'ftype' – тип фільтра. Для скалярної **Wn**: якщо відсутній або **'low'** – ФНЧ; **'high'** – ФВЧ. Для векторної **Wn**: **'bandpass'** або відсутній – смуговий; **'stop'** – режекторний або загороджуючий;

window – вектор-стовпець з **n+1** елементів (за замовчуванням застосовується вікно Хеммінга **hamming(n+1)**);

'normalization' – нормування АЧХ та імпульсно-смугова характеристика (ІСХ): за замовчуванням значення **'scale'** – одиничне значення АЧХ в центрі смуги пропускання; **'noscale'** – нормування не проводиться.

Примітка: при нормалізації максимальне значення АЧХ в точності дорівнює одиниці. При відсутності нормалізації через ефект Гіббса максимальне значення АЧХ більше одиниці. Цим і пояснюється те, що розраховані коефіцієнти фільтра за відсутності нормалізації більші таких при наявності нормалізації (див. наведений нижче приклад).

Приклад:

```
window=rectwin(7) % синтез прямокутного вікна з 7 відліків;  
a=fir1(6,0.5>window) % розрахунок коефіцієнтів КІХ-фільтра з  
нормалізацією.
```

Результат:

```
a=[ -0.1148 0.0000 0.3443 0.5409 0.3443 0.0000 -0.1148].
```

Порівнюючи ці результати з розрахованими вручну коефіцієнтами, неважко побачити різницю. Наприклад, розрахунки дають $a_0=0.5$, тоді як в **Matlab** ми отримали $a_0=0.5409$. Звичайно можна припустити, що причиною тому проведене за замовчуванням нормування ІСХ. Перевіримо це припущення, задаючи в програмі значення **'noscale'** для параметра нормалізації:

```
window=rectwin(7) % синтез прямокутного вікна з 7 відліків;  
a=fir1(6,0.5>window,'noscale') % розрахунок коефіцієнтів КІХ - фільтра  
без нормалізації.
```

Результат:

```
a=[ -0.1061 0.0000 0.3183 0.5000 0.3183 0.0000 -0.1061].
```

У наведеному вище прикладі прямокутне вікно генерувалося за допомогою функції **rectwin(n+1)**, де **n** – порядок нерекурсивного фільтра. Слід зазначити, що прямокутне вікно можна згенерувати також за допомогою функції **boxcar(n+1)**.

3.1.3 Розрахунок фільтра за допомогою пакета **sptool**

Для активізації пакета потрібно в командному вікні **Matlab** набрати команду **sptool**. Потім у вікні що з'явилося, в колонці кнопок “**Filters**” натиснути кнопку “**New**”. [1,6]

У вікні «**Filter Designer**» що з'явилося:

- задати частоту дискретизації (наприклад, задаємо 100 Гц);
- вибрати в позиції **Algorithm** значення **Kaiser Window FIR** (вибираємо з 3 варіантів: **Equiripple FIR**, **Least Square FIR** и **Kaiser Window FIR**);
- відключити опцію **Minimum Order**;
- задати **Order=6**;
- задати **Type=lowpass**;
- задати **Passband Fp=25**;
- відключити опцію **Autodesigne**;
- закрити вікно **Filter Designer**;
- у вікні **sptool** в стовпці **Filters** натиснути кнопку **View**.

У вікні **Filter Viewer** можна бачити АЧХ, ФЧХ, ІСХ (ІСХ бачимо після активізації відповідної опції).

Примітка: ми вибрали в позиції **Algorithm** значення **Kaiser Window FIR**. Крім даного алгоритму, є ще два алгоритми: **Equiripple FIR**, **Least Square FIR**. З усіх цих трьох алгоритмів тільки алгоритм Кайзера реалізує метод зворотного перетворення Фур'є з ваговим вікном Кайзера. При значенні “0” параметра **Order** вікно Кайзера перетворюється в звичайне прямокутне вікно.

Як показує експеримент, розраховані таким чином коефіцієнти ФНЧ виявляються ненормованими, тобто в точності рівними розрахованим вручну.

Примітка. Щоб дізнатися значення коефіцієнтів, потрібно: активувати графік ІСХ, клікнувши по ньому мишкою; активувати вертикальні маркери (кнопкою, розташованої під меню); помістити один з маркерів (всього є 2 маркера - 1-й зображується суцільною вертикальною лінією, 2-й - пунктирною) навпроти потрібного відліку ІСХ.

На рисунку 3.1 показано вікно **Filter Viewer** (в режимі попереднього перегляду **Print Preview**) для даного конкретного випадку. Пакет **sptool** дозволяє моделювати процес фільтрації за допомогою розрахованого фільтра.

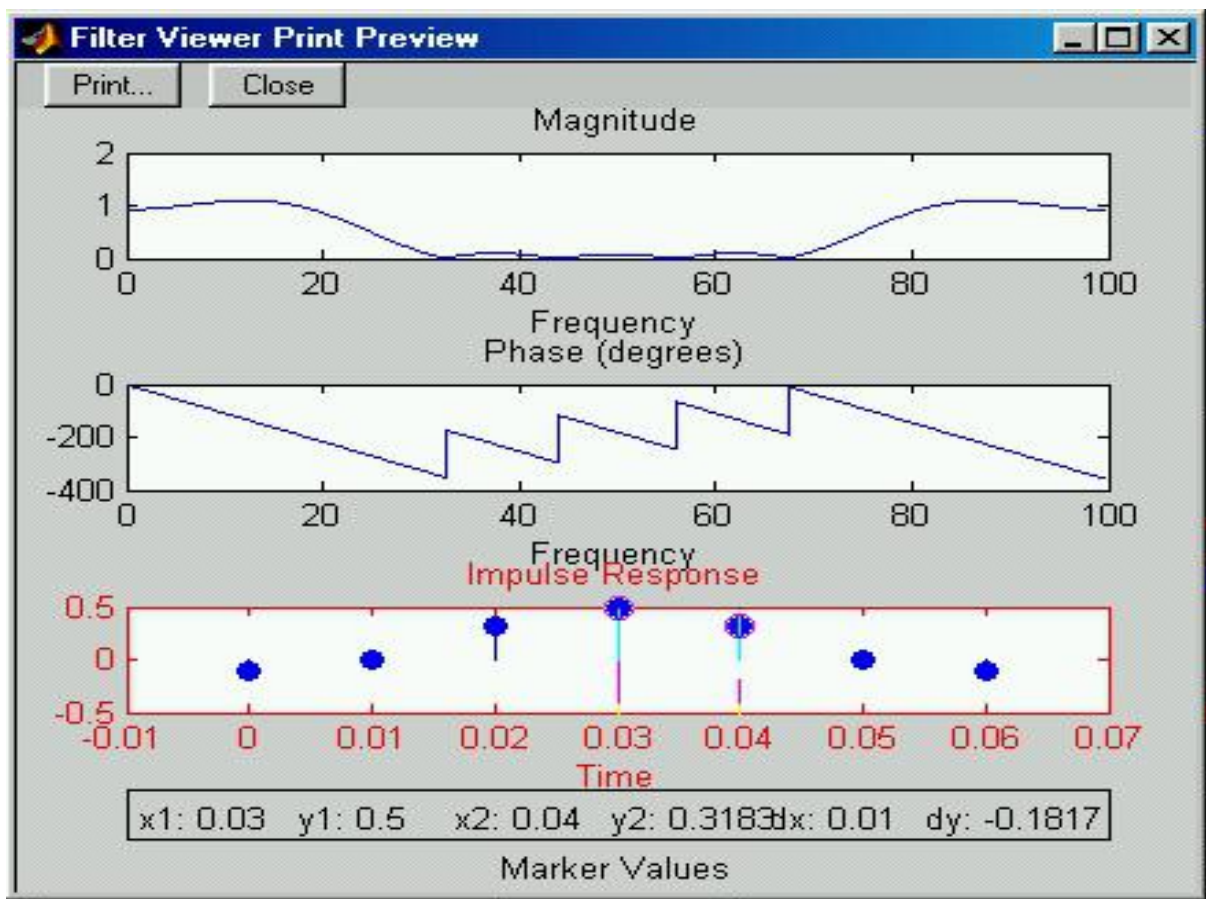


Рисунок 3.1 - Вікно **Filter Viewer** в режимі попереднього перегляду **Print Preview**

3.1.4 Розрахунок КІХ-фільтра за допомогою пакета **fdatool**

Для активізації пакета потрібно в командному вікні набрати команду **fdatool** [1,2,6]. Потім у вікні, що з'явилося, в розділі **Design Filter** задати:

- Designe Method: **FIR=Window**;
- Window Specifications: **Window=Rectangular**;
- Filter order: **Specify order=6**;
- частоту дискретизації ($F_s=100$ Гц); **Filter Type=lowpass**;
- **Passband Fc=25**;
- за допомогою кнопок меню вмикаємо режим перегляду коефіцієнтів фільтра.

В результаті проведених розрахунків переконуємось, що тут за замовчуванням проводиться нормування ІСХ (див. вище **fir1**).

Пакет **fdatool** не дозволяє моделювати процес фільтрації за допомогою розрахованого фільтра. Однак розраховані коефіцієнти фільтра можна імпортувати з середовища **fdatool** в середовище **Matlab**.

3.1.5 Співставлення способів розрахунку

Розрахунки за допомогою пакета **sptool**: перевага - не потрібно пам'ятати синтаксис команд; недолік - обмежений набір вікон для методу зворотного перетворення Фур'є (тільки вікно Кайзера); особливість - обчислюються тільки ненормовані коефіцієнти фільтра.

Розрахунки за допомогою пакета **fdatool**: перевага - не потрібно пам'ятати синтаксис команд; особливість - обчислюються тільки нормовані коефіцієнти фільтра.

3.2 Підготовка до лабораторної роботи

3.2.2 Написати програму у вигляді файлу-сценарію (m-файлу) для розрахунку коефіцієнтів a_k для двох заданих типів КІХ-фільтрів за допомогою функції **fir1** для двох типів вікон - прямокутного і Хеммінга.

Реалізувати обидва задані за варіантом вікна для кожного з заданих типів фільтрів – тобто чотири комбінації.

Побудувати графіки імпульсної реакції, АЧХ і ФЧХ фільтра.

Параметри для фільтрів, що проєктуються, вибираються з таблиці 3.1, згідно заданого варіанта ¹.

3.2.3 Побудувати два заданих типи КІХ-фільтрів за допомогою вбудованої утиліти (пакета) для проєктування цифрових фільтрів **SPTool**.

Параметри фільтрів, що проєктуються, вибираються з таблиці 3.1 згідно заданого варіанта для двох типів вікон: прямокутного і Кайзера.

Побудувати графіки імпульсної реакції, АЧХ і ФЧХ фільтра.

3.2.4 Побудувати два заданих типи КІХ-фільтрів за допомогою вбудованої утиліти (пакета) для проєктування цифрових фільтрів **fdatool**.

Параметри фільтрів, що проєктуються, вибираються з таблиці 3.1 згідно заданого варіанта для двох типів вікон: прямокутного і Блекмана.

Побудувати графіки імпульсної реакції, АЧХ і ФЧХ фільтра.

Таблиця 3.1

| № варіанта | Тип фільтра | Порядок фільтра | Частота дискретизації f_d , Гц | Частота зрізу f_z , Гц |
|------------|-------------|-----------------|----------------------------------|--------------------------|
| 1 | ФНЧ | 3 | 600 | 50 |
| | СФ | 6 | 600 | 80/120 |
| 2 | ФВЧ | 4 | 2100 | 400 |
| | СФ | 7 | 2100 | 200/260 |
| 3 | ФНЧ | 5 | 800 | 80 |
| | ЗФ | 4 | 800 | 130/200 |
| 4 | ФВЧ | 3 | 1500 | 320 |
| | ЗФ | 4 | 1500 | 100/200 |
| 5 | ФНЧ | 7 | 1800 | 140 |
| | ФВЧ | 6 | 1800 | 320 |
| 6 | ФНЧ | 4 | 900 | 175 |
| | СФ | 7 | 900 | 220/300 |
| 7 | ФВЧ | 5 | 2400 | 600 |
| | СФ | 6 | 2400 | 200/260 |
| 8 | ФНЧ | 4 | 1000 | 125 |
| | ЗФ | 5 | 1000 | 160/220 |
| 9 | ФВЧ | 6 | 2000 | 500 |
| | ЗФ | 6 | 2000 | 250/350 |
| 10 | ФНЧ | 3 | 2300 | 260 |
| | ФВЧ | 6 | 2300 | 550 |

Примітка 1. Номер варіанта визначається номером студента у списку групи (підгрупи, або об'єднаного списку груп згідно їх номерів). Якщо номер за списком 11 (21), параметри фільтра 1-го типу вибираються з варіанта 1 (2), а параметри фільтра 2-го типу – з варіанта 10. Якщо номер за списком перевищує 11 (21), то параметри фільтра 1-го типу вибираються з варіанта 1 (2), а параметри фільтра 2-го типу – з варіанта, номер якого дорівнює другій цифрі в номері за списком. Наприклад, для номера 12 параметри фільтра 1-го типу вибираються з варіанта 1, а параметри фільтра 2-го типу – з варіанта 2.

3.3 Порядок виконання роботи

3.3.1 Увімкніть ПЕОМ та запустіть програму MatLab.

3.3.2 Наберіть програму, написану за п. 3.2.2 та запустіть її на виконання. Отримані результати занесіть в протокол лабораторної роботи у вигляді скріншотів.

3.3.3 У програмі, написаній по п. 3.2.2, збільшити частоти дискретизації в два рази, і запустіть її на виконання. Отримані результати занесіть в протокол лабораторної роботи у вигляді скріншотів.

3.3.3 У програмі, написаній по п. 3.2.2, збільшити частоти дискретизації в два рази, і запустіть її на виконання. Отримані результати занесіть в протокол лабораторної роботи у вигляді скріншотів.

3.3.4 У програмі, написаній за п. 3.2.2 змініть на 20 % значення частот зрізу, і запустіть її на виконання. Отримані результати занесіть в протокол лабораторної роботи у вигляді скріншотів.

3.3.5 Запустіть на виконання утиліту для проектування цифрових фільтрів SPTool.

Спроектуйте фільтри відповідно п. 3.2.3 з параметрами з таблиці 3.1 для заданого варіанта для двох типів вікон: прямокутного і Кайзера.

Побудуйте графіки імпульсної реакції, АЧХ і ФЧХ фільтра.

Отримані результати занесіть в протокол лабораторної роботи у вигляді скріншотів.

3.3.6 Повторіть п. 3.3.5 для збільшеної в два рази частоти дискретизації. Отримані результати занесіть в протокол лабораторної роботи у вигляді скріншотів.

3.3.7 Повторіть п. 3.3.5 для змінених на 30 % значень частот зрізу. Отримані результати занесіть в протокол лабораторної роботи у вигляді скріншотів.

3.3.8 Запустіть на виконання утиліту для проектування цифрових фільтрів **fdatool**.

Спроектуйте фільтри відповідно п. 3.2.4 з параметрами з таблиці 3.1 для заданого варіанта для двох типів вікон: прямокутного і Блекмана.

Побудуйте графіки імпульсної реакції, АЧХ і ФЧХ фільтра.

3.3.9 Повторіть п. 3.3.8 для збільшеної в два рази частоти дискретизації. Отримані результати занесіть в протокол лабораторної роботи у вигляді скріншотів.

3.3.10 Повторіть п. 3.3.8 для змінених на 30 % значень частот зрізу. Отримані результати занесіть в протокол лабораторної роботи у вигляді скріншотів.

3.4 Зміст звіту з лабораторної роботи

Звіт з лабораторної роботи повинен містити:

3.4.1 При допуску до лабораторної роботи

3.4.1.1 Ціль лабораторної роботи.

3.4.1.2 Короткі теоретичні відомості.

3.4.1.3 Написану за пунктом 3.2.2 програму з коментарями для середовища MatLab.

3.4.2 При здачі лабораторної роботи до матеріалів по п. 1.4.1 повинні бути додані результати виконання роботи.

3.4.2.1 Результати виконання програми за п. 3.3.2 - 3.3.10 у вигляді скріншотів.

3.4.2.2 Висновки.

3.5 Контрольні питання

3.5.1 Які види фільтрів ви знаєте?

3.5.2 Які способи розрахунку цифрових фільтрів ви знаєте?

3.5.3 Як реалізується метод вікон в MatLab?

3.5.4 Які параметри задаються в команді **fir1** MatLab?

3.5.5 Які особливості пакетів (утиліт) **sptool** і **fdatool** для розрахунку параметрів КІХ-фільтрів?

Лабораторна робота № 4

Дослідження методів синтезу НІХ-фільтрів за допомогою системи «MatLab»

Ціль роботи: дослідити методи синтезу цифрових фільтрів з нескінченною імпульсною характеристикою в середовищі MATLAB.

4.1 Теоретичні відомості

4.1.1 Цифрові НІХ-фільтри

По виду імпульсної характеристики цифрові фільтри прийнято ділити на фільтри з кінцевою імпульсною характеристикою (КІХ-фільтри) і з нескінченною імпульсною характеристикою (НІХ-фільтри). [1,2,5]

Зазвичай нерекурсивні цифрові фільтри мають КІХ, рекурсивні - НІХ. Перші на відміну від других можуть мати лінійну фазочастотну характеристику (ФЧХ), але вимагають більш значних обчислювальних затрат. Рекурсивні кінцево-імпульсні характеристики фільтрів (РКІХФ) також можуть мати лінійну ФЧХ, але при набагато менших обчислювальних витратах в порівнянні з нерекурсивними фільтрами.

Проектування ЦФ в загальному випадку включає:

- проектування передавальної функції і структури фільтра за заданою його частотною або імпульсною характеристикою;

- оцінку необхідної розрядності чисел для коефіцієнтів фільтра і відліків вхідного, вихідного і внутрішніх сигналів.

Для синтезу НІХ-фільтрів широко використовується Z-перетворення.

Передавальну функцію НІХ-фільтра в Z-області можна записати в наступному вигляді:

$$H(z) = \frac{\sum_{k=0}^{N-1} b_k z^{-k}}{1 + \sum_{k=1}^{M-1} a_k z^{-k}}$$

Різницеве рівняння:

$$Y(z) = X(z) \sum_{k=0}^{N-1} b_k z^{-k} - Y(z) \sum_{k=1}^{M-1} a_k z^{-k}.$$

Канонічна структура НІХ-фільтра показана на рисунку 4.1.

Проектування передавальної функції ЦФ $H(z)$ за заданою частотною характеристикою $H_d(j\omega)$ полягає в її апроксимації і визначенні коефіцієнтів передавальної функції.

Методи синтезу поділяються на:

- аналітичні;
- ітераційні;
- числові.

Аналітичні методи забезпечують рішення задачі апроксимації в замкнутій аналітичній формі. До них відносяться методи проектування рекурсивних цифрових фільтрів (РЦФ) за даними аналогового фільтра-прототипу і, в першу чергу, метод білінійного перетворення.

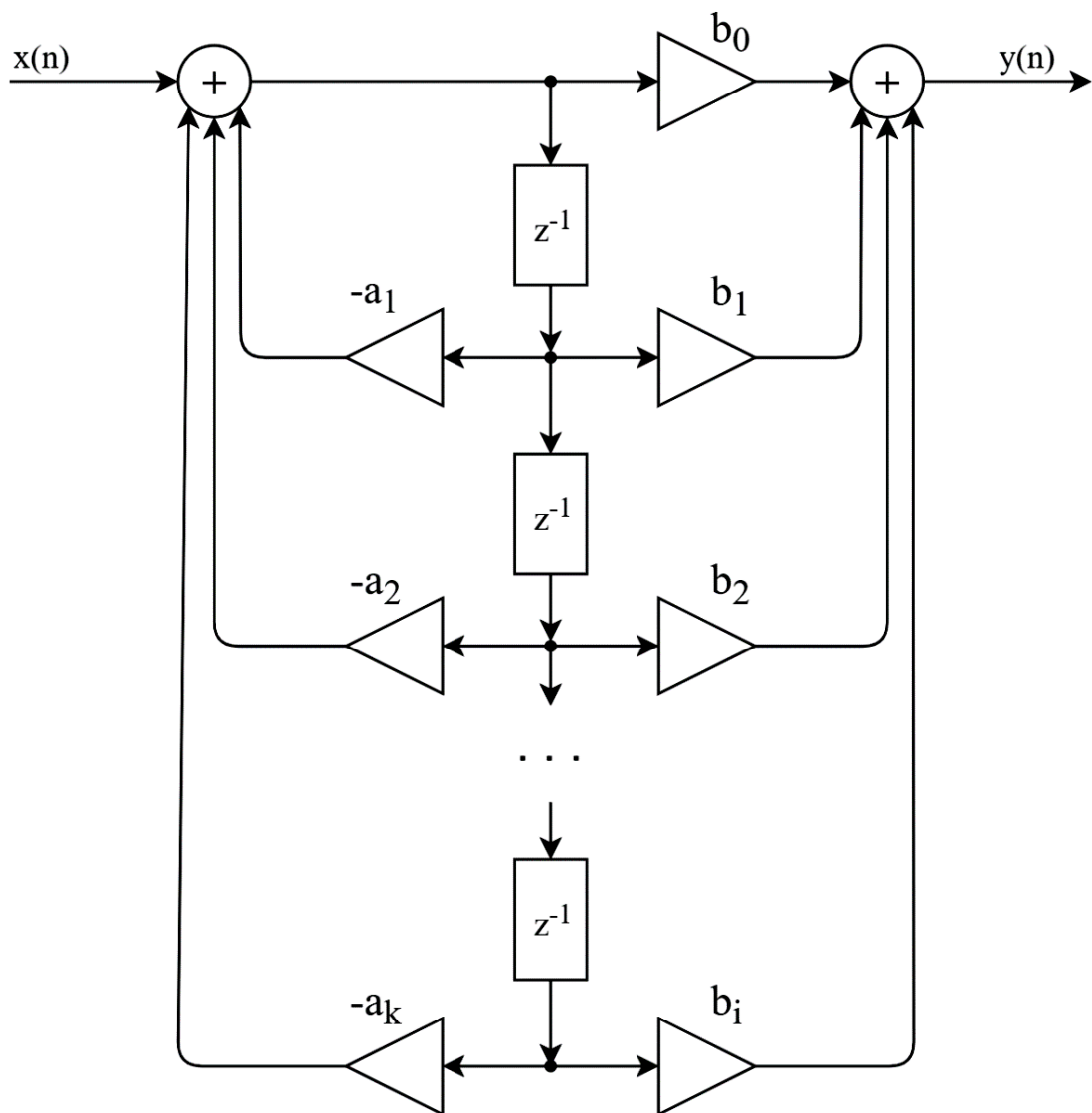


Рисунок 4.1 - Канонічна структура НІХ-фільтра

Ітераційні методи, поєднуючи аналітичний і числовий підходи, не дають однозначної відповідності параметрів розрахункових і заданих характеристик і вимагають, як правило, ряду уточнюючих процедур. Ітераційними є методи вагових функцій і частотної вибірки, які використовуються для проектування нерекурсивних фільтрів.

Числові методи ґрунтуються на безпосередній апроксимації заданої частотної характеристики з мінімальною похибкою відповідно до певного критерію оптимальності.

Вихідними даними для проектування ЦФ за заданою частотною характеристикою є:

1) частоти зрізу ω_c , затримання ω_z , що визначають границі і значення смуг пропускання, затримання та перехідних смуг фільтра;

2) допустима нерівномірність АЧХ фільтра в смузі пропускання (або ослаблення на границях смуги пропускання в разі монотонних АЧХ), дБ;

3) мінімальне затухання АЧХ в смузі затримання, дБ.

Більшість методів синтезу НІХ-фільтрів реалізовано у пакеті програм середовища **MATLAB**.

4.1.2 Розробка НІХ -фільтрів в середовищі MATLAB

В MATLAB НІХ-фільтри можливо проектувати або програмним шляхом у вигляді файлу-сценарію, або за допомогою вбудованої утиліти Filter Designer.

4.1.2.1 Проектування НІХ-фільтра програмним шляхом у вигляді файлу сценарію

Для проектування програмним шляхом у вигляді файлу сценарію використовуються спеціальні функції які дозволяють розрахувати фільтри з різними видами частотних характеристик. Розглянемо основні функції системи MATLAB для проектування НІХ-фільтрів. [1,2,6]

Фільтр Баттерворта.

Функція для знаходження мінімального порядку фільтра Баттерворта при заданих параметрах:

$$[n, W_p] = \text{buttord}(W_p, W_s, r_p, r_s),$$

де n – мінімальний порядок фільтра;

W_p – гранична частота смуги пропускання фільтра (частота зрізу);

W_s – гранична частота смуги затримання фільтра;

r_p – максимально допустиме затухання в смузі пропускання (в дБ);

r_s – мінімально допустиме затухання в смузі затримання (в дБ).

Функція для знаходження коефіцієнтів передавальної функції фільтра Баттерворта:

$$[b, a] = \text{butter}(n, W_p, 'type'),$$

де b – коефіцієнти чисельника;

a – коефіцієнти знаменника;

n – мінімальний порядок фільтра;

W_p – гранична частота смуги пропускання фільтра (частота зрізу);

'type' – тип фільтра.

При заданні типу фільтра ('type') в функції для знаходження передавальної функції фільтра використовуються наступні параметри: **low** – фільтр низьких частот, **high** – фільтр високих частот, **bandpass** – смуговий фільтр, **stop** – загороджуючий фільтр.

Частоти W_p і W_s задаються у нормованому виді. В такому випадку необхідні значення цих параметрів можна обчислити за формулами:

$$Wp = \frac{f_{пп}}{f_d/2};$$

$$Ws = \frac{f_{пз}}{f_d/2};$$

де f_d – частота дискретизації.

Фільтр Чебишева 1-го роду.

Функція для знаходження мінімального порядку фільтра Чебишева 1-го роду при заданих параметрах:

$$[n, Wp] = \text{cheb1ord}(Wp, Ws, rp, rs).$$

Параметри n , Wp , Ws , rp , rs аналогічні попереднім (див. вище).

Функція для знаходження коефіцієнтів передавальної функції фільтра Чебишева 1-го роду:

$$[b, a] = \text{cheby1}(n, rp, Wp, 'type').$$

Параметри b , a , n , rp , Wp , $'type'$ аналогічні попереднім (див. вище).

Фільтр Чебишева 2-го роду.

Функція для знаходження мінімального порядку фільтра Чебишева 2-го роду при заданих параметрах:

$$[n, Wp] = \text{cheb2ord}(Wp, Ws, rp, rs).$$

Параметри n , Wp , Ws , rp , rs аналогічні попереднім (див. вище).

Функція для знаходження коефіцієнтів передавальної функції фільтра Чебишева 2-го роду:

[b,a]= cheby2(n,rs,Wp,'type').

Параметри **b, a, n, rp, Wp, 'type'** аналогічні попереднім (див. вище).

Еліптичний фільтр (Золотарьова-Кауера).

Функція для знаходження мінімального порядку еліптичного фільтра при заданих параметрах:

[n,Wp]=ellipord(Wp,Ws,rp,rs).

Параметри **n, Wp, Ws, rp, rs** аналогічні попереднім (див. вище).

Функція для знаходження коефіцієнтів передавальної функції еліптичного фільтра:

[b,a]=ellip(n,rp,rs,Wp,'type').

Параметри **b, a, n, rp, Wp, 'type'** аналогічні попереднім (див. вище).

4.1.2.2 Проектування НИХ-фільтра за допомогою утиліти Filter Designer

Запуск утиліти **Filter Designer** проводиться набором у командному вікні системи MATLAB команди **filterDesigner**. Вид вікна Filter Designer представлений на рисунку 4.2. [1,2,6]

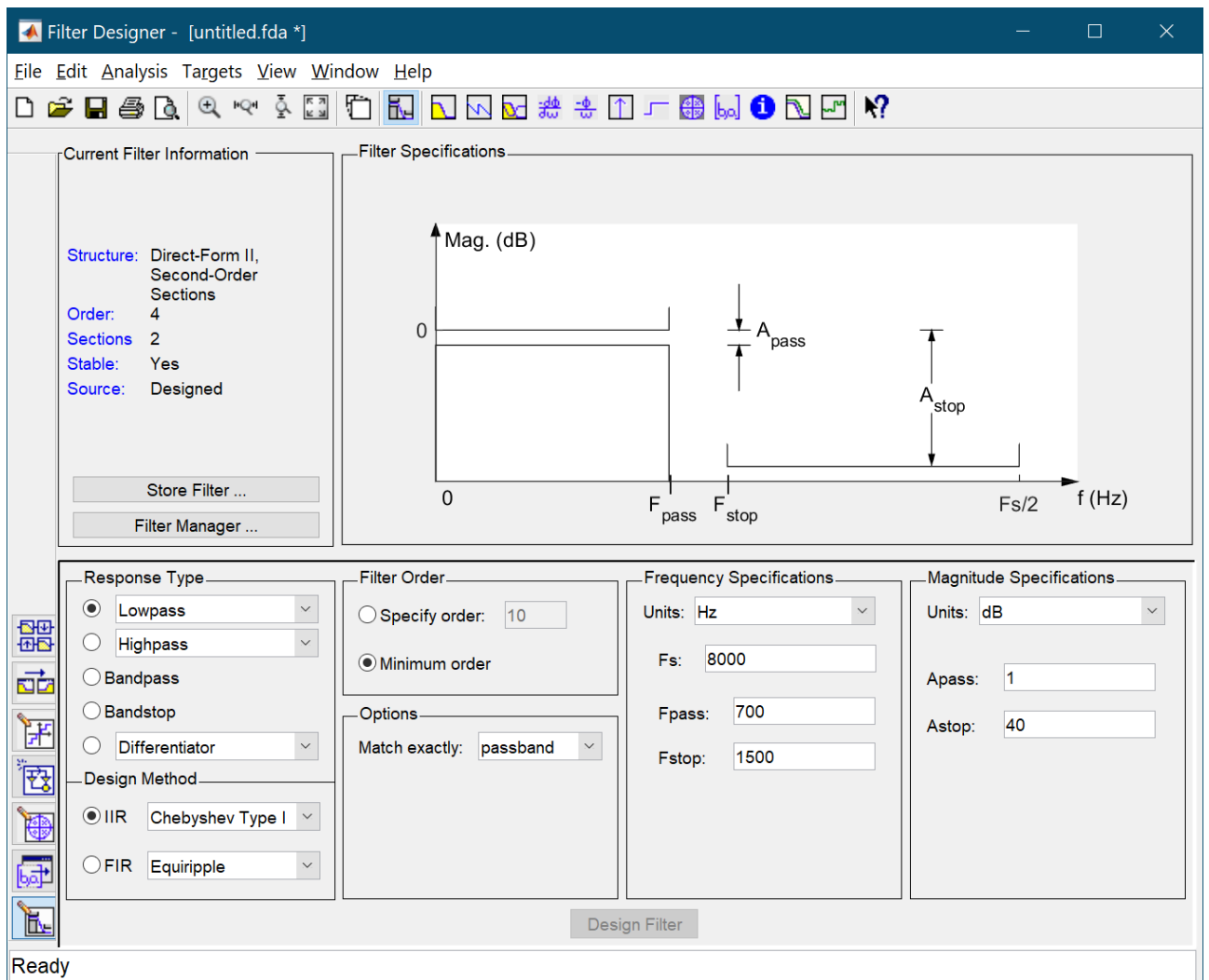


Рисунок 4.2 - Загальний вигляд вікна Filter Designer

При проектуванні фільтра за допомогою утиліти Filter Designer вибір необхідних параметрів фільтра здійснюється на таких вкладках:

Панель **Response Type** (тип характеристики):

- **Lowpass** – фільтр низьких частот;
- **Highpass** – фільтр високих частот;
- **Bandpass** – смуговий фільтр;
- **Bandstop** – загороджуючий фільтр.

Панель **Design Method** (метод синтезу):

- **IIR** – НІХ-фільтр;
- **FIR** – КІХ-фільтр.

Панель **Filter order** (порядок фільтра):

- **Specify order** – можливість вказати порядок фільтра вручну;
- **Minimum order** – буде вибраний мінімальний порядок фільтра.

Панель **Frequency Specifications** (вимоги до частот):

- **Units** – розмірність;
- **Fs** – частота дискретизації;
- **Fpass** - гранична частота смуги пропускання фільтра;
- **Fstop** - гранична частота смуги затримання фільтра.

Панель **Magnitude Specifications** (вимоги до амплітуди):

- **Units** – розмірність;
- **Apass** - максимально допустиме затухання в смузі пропускання;
- **Astop** - мінімально допустиме затухання в смузі затримання.

4.1.2.3 Проектування НІХ-фільтра за допомогою утиліти SPTool

Для запуску утиліти потрібно в командному вікні системи MATLAB набрати команду **sptool** [1,2,6]. Вид вікна SPTool представлений на рисунку 4.3.

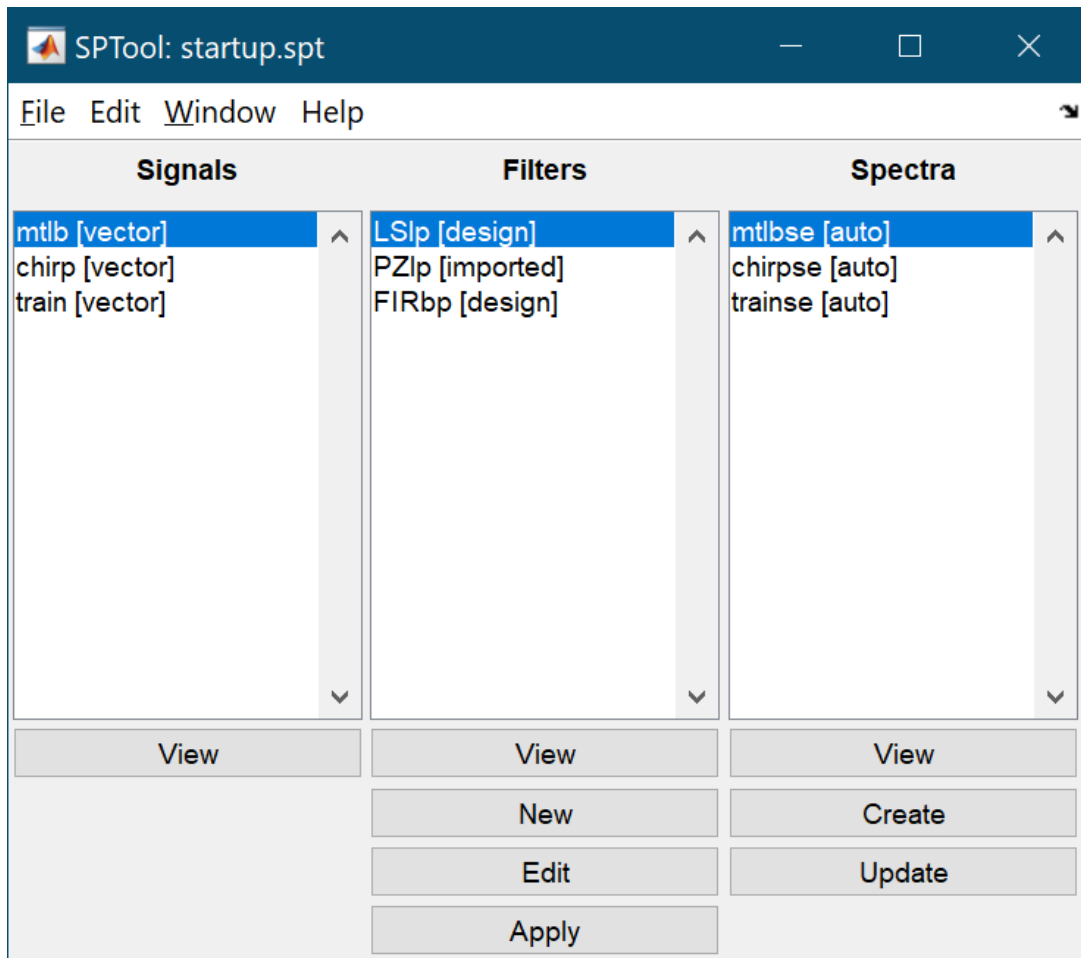


Рисунок 4.3 - Загальний вигляд вікна SPTool

У вікні SPTool в колонці **Filters** необхідно натиснути кнопку **New**. У вікні Filter Designer, що з'явилося, необхідно задати параметри фільтра і натиснути кнопку **Design Filter**. Всі панелі вікна Filter Designer описані в пункті 4.1.2.2.

Потім необхідно закрити вікно Filter Designer і у вікні SPTool в колонці **Filters** вибрати створений фільтр, натиснути на нього, потім натиснути на кнопку **View**. Відкриється вікно Filter Visualization Tool в якому будуть представлені характеристики спроектованого фільтра (рисунок 4.4).

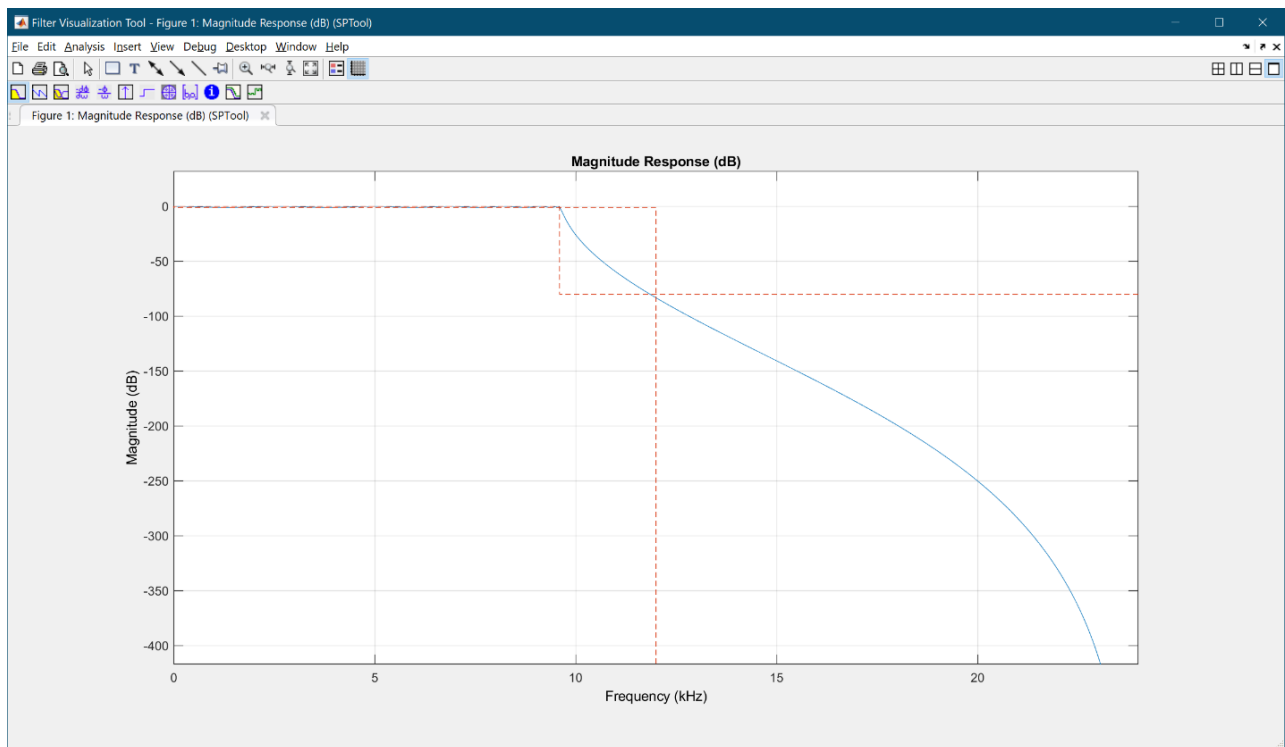


Рисунок 4.4 - Вид вікна Filter Visualization Tool

4.2 Підготовка до лабораторної роботи

4.2.1 Ознайомитися з короткими теоретичними відомостями і порядком роботи з середовищем MatLab.

4.2.2 Написати програму у вигляді файлу-сценарію для проектування НІХ-фільтра.

Параметри для фільтра, що синтезується, вибираються таблиці 4.1, згідно заданого варіанта ¹.

Таблиця 4.1

| № вар. | Тип функції | Тип фільтра | $f_{\text{пн}}^{(2)}$, Гц | $f_{\text{пз}}^{(2)}$, Гц | $A_{\text{пн}}^{(2)}$, дБ | $A_{\text{пз}}^{(2)}$, дБ | $f_{\text{д}}$, Гц |
|--------|--------------------|-------------|----------------------------|----------------------------|----------------------------|----------------------------|---------------------|
| 1 | Чебишева 1-го роду | ФНЧ | 400 | 1000 | 2 | 35 | 8000 |
| | Еліптичний | СФ | 1000/1500 | 500/2500 | | | |
| 2 | Баттерворта | ЗФ | 1200/2500 | 1600/2000 | 4 | 50 | 10000 |
| | Чебишева 1-го роду | ФВЧ | 1500 | 600 | | | |
| 3 | Чебишева 2-го роду | СФ | 1200/1700 | 700/2700 | 6.5 | 48 | 5500 |
| | Баттерворта | ФВЧ | 1700 | 500 | | | |
| 4 | Еліптичний | СФ | 1100/1400 | 600/2800 | 8 | 65 | 8400 |
| | Чебишева 2-го роду | ФНЧ | 300 | 1100 | | | |
| 5 | Чебишева 1-го роду | ФВЧ | 1400 | 500 | 3 | 20 | 7600 |
| | Еліптичний | ЗФ | 900/2200 | 1200/1900 | | | |
| 6 | Еліптичний | ФВЧ | 800 | 2000 | 1 | 15 | 12000 |
| | Баттерворта | СФ | 1500/2000 | 500/2700 | | | |
| 7 | Чебишева 2-го роду | ЗФ | 1000/2900 | 1500/2000 | 2.6 | 60 | 11600 |
| | Чебишева 1-го роду | СФ | 500/900 | 100/1500 | | | |
| 8 | Баттерворта | ФНЧ | 1000 | 2500 | 0.6 | 30 | 7500 |
| | Еліптичний | ЗФ | 800/1600 | 1000/1200 | | | |
| 9 | Чебишева 1-го роду | ЗФ | 300/1500 | 500/800 | 6 | 55 | 6000 |
| | Баттерворта | ФВЧ | 3000 | 1500 | | | |
| 10 | Еліптичний | ЗФ | 400/2000 | 700/1500 | 5 | 70 | 4500 |
| | Чебишева 2-го роду | СФ | 600/800 | 200/1000 | | | |

Примітка 1. Номер варіанта визначається номером студента у списку групи (підгрупи, або об'єднаного списку груп згідно їх номерів). Якщо номер за списком 11 (21), параметри фільтра 1-го типу вибираються з варіанта 1 (2), а параметри фільтра 2-го типу – з варіанта 10. Якщо номер за списком перевищує 11 (21), то параметри фільтра 1-го типу вибираються з варіанта 1 (2), а параметри фільтра 2-го типу – з варіанта, номер якого дорівнює другій цифрі в номері за списком. Наприклад, для номера 12 параметри фільтра 1-го типу вибираються з варіанта 1, а параметри фільтра 2-го типу – з варіанта 2.

Примітка 2. $f_{\text{пп}}$ – гранична частота смуги пропускання фільтра; $f_{\text{пз}}$ – гранична частота смуги затримання фільтра; $A_{\text{пп}}$ – максимально допустиме затухання в смузі пропускання; $A_{\text{пз}}$ – мінімально допустиме затухання в смузі затримання; $f_{\text{д}}$ – частота дискретизації.

4.2.3. Написати програму у вигляді файлу-сценарію для знаходження порядку НІХ-фільтра і коефіцієнтів передавальної функції.

Побудувати АЧХ і ФЧХ НІХ-фільтра, а також вивести коефіцієнти передавальної функції НІХ-фільтра. АЧХ і ФЧХ побудувати на проміжку $[0, f_{\text{д}}/2]$.

Реалізувати обидві задані за варіантом функції для кожного з заданих типів фільтрів – тобто чотири комбінації.

4.2.4. Побудувати НІХ-фільтр за допомогою вбудованої утиліти для проектування цифрових фільтрів Filter Designer. Побудувати АЧХ і ФЧХ фільтра.

Параметри фільтра, що проектується, вибираються з таблиці 4.1, згідно заданого варіанта.

4.2.5. Побудувати НІХ-фільтр за допомогою вбудованої утиліти для проектування цифрових фільтрів SPTool. Побудувати АЧХ і ФЧХ фільтра.

Параметри фільтра, що проектується, вибираються з таблиці 4.1, згідно заданого варіанта.

4.3 Порядок виконання роботи

4.3.1 Увімкніть ПЕОМ та запустіть програму MatLab.

4.3.2 Наберіть програму, написану за п.п. 4.2.2 і 4.2.3 і запустіть її на виконання. Отримані результати занесіть в протокол лабораторної роботи у вигляді скріншотів.

4.3.3 У програмі, написаній за п.п. 4.2.2 і 4.2.3 збільшить частоти дискретизації в два рази, і запустіть її на виконання. Отримані результати занесіть в протокол лабораторної роботи у вигляді скріншотів.

4.3.4 У програмі, написаній за п.п. 4.2.2 і 4.2.3 змініть на 20 % значення частот зрізу, і запустіть її на виконання. Отримані результати занесіть в протокол лабораторної роботи у вигляді скріншотів.

4.3.5 Запустіть на виконання утиліту для проектування цифрових фільтрів Filter Designer.

Спроектуйте фільтри відповідно п. 4.2.4 з параметрами з таблиці 4.1 для заданого варіанту.

Побудуйте графіки АЧХ і ФЧХ фільтра.

Отримані результати занесіть в протокол лабораторної роботи у вигляді скріншотів.

4.3.6 Повторіть п. 4.3.5 для збільшеної в два рази частоти дискретизації. Отримані результати занесіть в протокол лабораторної роботи у вигляді скріншотів.

4.3.7 Повторіть п. 4.3.5 для змінених на 30 % значень частот зрізу. Отримані результати занесіть в протокол лабораторної роботи у вигляді скріншотів.

4.3.8 Запустіть на виконання утиліту для проєктування цифрових фільтрів SPTool.

Спроєктуйте фільтри відповідно п. 4.2.5 з параметрами з таблиці 4.1 для заданого варіанту.

Побудуйте графіки АЧХ і ФЧХ фільтра.

Отримані результати занесіть в протокол лабораторної роботи у вигляді скріншотів.

4.3.9 Повторіть п. 4.3.8 для збільшеної в два рази частоти дискретизації. Отримані результати занесіть в протокол лабораторної роботи у вигляді скріншотів.

4.3.10 Повторіть п. 4.3.8 для змінених на 30 % значень частот зрізу. Отримані результати занесіть в протокол лабораторної роботи у вигляді скріншотів.

4.4 Зміст звіту з лабораторної роботи

Звіт з лабораторної роботи повинен містити:

4.4.1 При допуску до лабораторної роботи:

4.4.1.1 Ціль лабораторної роботи.

4.4.1.2 Короткі теоретичні відомості.

4.4.1.3 Написані за п.п. 4.2.2 - 4.2.3 програми з коментарями для середовища MatLab.

4.4.2 При здачі лабораторної роботи до матеріалів по п. 4.4.1 повинні бути додані результати виконання роботи.

4.4.2.1 Результати виконання програми за п. 4.3.2 - 4.3.10.

4.4.2.2 Висновки.

4.5 Контрольні питання

4.5.1. Що таке НІХ-фільтр?

4.5.2. Запишіть передавальну функцію і різницеve рівняння НІХ-фільтра в загальному вигляді?

4.5.3. Чим визначається порядок фільтра?

4.5.4. Які функції використовуються для реалізації НІХ -фільтрів?

4.5.5. Які існують методи реалізації НІХ-фільтрів?

СПИСОК РЕКОМЕНДОВАНОЇ ЛІТЕРАТУРИ

Основна

1 Ушенко Ю.О., Гавриляк М.С. та ін, Основи та методи цифрової обробки сигналів: від теорії до практики: навч. посібник / уклад. : Ю.О. Ушенко, М.С. Гавриляк, М.В. Талах, В.В. Дворжак. – Чернівці : Чернівецький нац. ун-т ім. Ю. Федьковича, 2021. - 308 с.

2 Заболотній С. В. Цифрове оброблення сигналів: Посібник для студентів напряму підготовки 6.050901 "Радіотехніка" усіх форм навчання [Електронний ресурс] / Авт.-укл. С. В.Заболотній; За ред. проф. Ю. Г. Леги ; М-во освіти і науки України, Черкас. держ. технол. ун-т. – Черкаси: ЧДТУ, 2010. – 119 с.

3 Тотосько О.В., Стухляк П.Д. Цифрова обробка сигналів та зображень / Навчальний посібник. – Тернопіль: ТНТУ, 2016 – 132с.

4 Бабак В.П., Хандецький В.С., Шрюфер Е. Обробка сигналів. / Підручник для студентів технічних спеціальностей вузів. — К.: «Либідь», 1999 – 496 с.

Допоміжна

5 Richard G. Lyons. Understanding Digital Signal Processing, 3rd Edition. - Prentice Hall, 2011. – 954 p.

6 Колєнов С. О. Цифрова обробка інформації: Методичний посібник до лабораторного практикуму для студентів радіофізичного факультету / Колєнов С. О./ – Київ: Радіофізичний факультет Київського національного університету імені Тараса Шевченка, 2008. – 56 с.

7 Метрологія та вимірювальна техніка : Підручник — 2-е вид., доп. та переробл. /Є.С.Поліщук, М.М. Дорожовець, В.О. Яцук, В.М. Ванько, Т.Г. Бойко; За ред. Є.С. Поліщука; — Львів: Вид-во Львівської політехніки, 2012. — 544 с.