



МІНІСТЕРСТВО ОСВІТИ ТА НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ ІМЕНІ ІГОРЯ
СІКОРСЬКОГО»

Методичні вказівки до виконання лабораторних робіт

по курсу

Технологія проектування комп'ютерних систем — 2

(назва дисципліни)

для напрямку підготовки (спеціальностей)

123 Комп'ютерна інженерія .

(шифр та назва напрямку, спеціальностей)

Уклав

професор Анатолій Михайлович

Сергієнко, д.т.н., с.н.с. кафедри ОТ

Рекомендовано

Вченою радою факультету

інформатики та обчислювальної техніки

НТУУ «КПІ ім. Ігоря Сікорського»

Протокол № 11_ від _7_.__05____.2019 р.

Київ - 2020

Лабораторна робота 1

Проектування рекурсивного цифрового фільтра

Мета роботи: одержати знання та навички в розробці і тестуванні швидкодіючих цифрових фільтрів на ПЛІС.

Теоретичні відомості

Передаточна функція

У цифровій обробці сигналів часто використовуються рекурсивні цифрові фільтри (РЦФ). РЦФ на базі програмованої логічної інтегральної схеми (ПЛІС) має високу пропускну здатність. РЦФ описується *різницеvim рівнянням* N -го порядку з постійними коефіцієнтами:

$$y(n) = -\sum_{k=1}^N a_k y(n-k) + \sum_{r=0}^M b_r x(n-r). \quad (1)$$

Отже, n -е значення виходу можна обчислити на основі n -го значення входу $x(n)$ та відповідно N та M минулих значень виходу $y(n-k)$ та входу $x(n-r)$. Тоді *імпульсну реакцію* такої системи можна визначити як:

$$h(n) = \frac{y(n)}{x(n)}. \quad (2)$$

Фільтрацію сигнала x фільтром з імпульсною реакцією h називають *згорткою*. Імпульсна реакція (2) у загальному випадку є нескінченною і тоді це – система з нескінченною імпульсною характеристикою (НІХ).

Операції згортки $h(n)*x(n)$ відповідає множення: $H(z)X(z)$ у z -просторі. Слід наголосити, що ці властивості згортки справедливі лише для області поширення змінної z , де розглянуті функції не розбігаються.

Імпульсна реакція $h(n)$ системи (2) має відображення у z -просторі як *передаточна функція*:

$$H(z) = \frac{Y(z)}{X(z)} = \frac{\sum_{r=0}^M b_r z^{-r}}{1 + \sum_{k=1}^N a_k z^{-k}}, \quad (3)$$

де a_k, b_r – дійсні числа, функції z^{-m} відповідає затримка сигналу на m тактів дискретизації. Тут число N є **порядком** системи.

Передаточну функцію (3) можна розкласти на суму елементарних дробів. Частіше, наприклад, якщо $N \geq M$; $Q = N/2$, вона факторизується на добуток дробів:

$$H(z) = \prod_{k=0}^Q \frac{b_{0,k} + b_{1,k}z^{-1} + b_{2,k}z^{-2}}{1 + a_{1,k}z^{-1} + a_{2,k}z^{-2}} \quad . \quad (4)$$

Передаточна функція свідчить про спектральні властивості лінійної системи, що розглядається. На її основі знаходять **амплітудо-частотну** $|H(e^{j\omega t})|$ та **фазо-частотну** $\arg(H(e^{j\omega t}))$ характеристики системи.

Графове представлення алгоритму РЦФ

У переважній більшості алгоритмів ЦОС сигнальні потоки є синхронними. Тому такі алгоритми можна представити графом синхронних потоків даних (ГСПД). Якщо у результуючому потоці наявність даних умовно залежить від вхідного потоку, то такі потоки можуть бути несинхронними. Це, наприклад, потоки в компресорі сигналів, який замінює ланцюжки нульових відліків кодами довжини цих ланцюжків.

Сигнальний граф, який зазвичай використовується в ЦОС та однорідний ГСПД є еквівалентними моделями. В табл. 1 показані графічні позначення елементів сигнального графа, ГСПД та їх відповідність частинам алгоритма ЦОС. Розглянемо приклад задавання алгоритму рекурсивного фільтру високих частот другого порядку за допомогою сигнального графа і ГСПД. Передаточна характеристика такого фільтра дорівнює

$$H(z) = \frac{1-z^{-2}}{1+a_1z^{-1}+a_2z^{-2}} = (1-z^{-2}) \cdot \frac{1}{1+a_1z^{-1}+a_2z^{-2}} \quad , \quad (5)$$

(порівняйте з (4)). Множникам передаточної функції фільтру відповідають різницеві рівняння (порівняйте з (1)):

Таблиця 1. позначення елементів сигнального графа і ГСПД

Елемент алгоритму	Сигнальний граф	Однорідний ГСПД
Сигнал $x(n)$	$\xrightarrow{x(n)}$	$\xrightarrow{x(n)}$
Вхідний та вихідний порти, джерело та приймач сигналів $x(n), y(n)$	$x(n) \begin{array}{c} \text{D} \\ \text{O} \end{array} \xrightarrow{\quad} \begin{array}{c} \text{D} \\ \text{O} \end{array} y(n)$ або $\circ \rightarrow \circ$	$\bigcirc(x) \rightarrow \rightarrow \bigcirc(y)$
Затримка на k циклів	$\xrightarrow{x(n)} \boxed{z^{-k}} \xrightarrow{x(n-k)}$	$\xrightarrow{x(n)} \parallel \dots \parallel \xrightarrow{x(n-k)}$ k
Додавання сигналів, вершина суматора $y(n) = a(n) + b(n)$	$\begin{array}{c} a(n) \\ b(n) \end{array} \rightarrow \bigoplus \rightarrow y(n)$	$\begin{array}{c} a(n) \\ b(n) \end{array} \rightarrow \bigoplus \rightarrow y(n)$
Множення сигналу на константу $y(n) = a \cdot x(n)$, вершина блока множення	$\xrightarrow{x(n)} \triangle a \rightarrow$	$\xrightarrow{x(n)} \bigcirc * a \rightarrow$

$$u(n) = x(n) - x(n-2);$$

$$y(n) = u(n) - a_1 y(n-1) - a_2 y(n-2).$$

Рівняння реалізуються у сигнальному графі (рис.1). Вважається, що перед початком виконання усі елементи затримки зберігають нульові значення. Як тільки надходить чергове вхідне дане $x(n)$, яке дискретизоване з частотою дискретизації f_s , воно зразу іде на елемент затримки на два цикли z^{-2} та на суматор “+”, де воно додається до затриманого даного $x(n-2)$. Решта елементів моделі функціонують так само: як тільки є в наявності вхідні дані – одразу вони спрацьовують і видають результат на свій вихід.

Граф на рис.1 можна розглядати як структурну схему певного

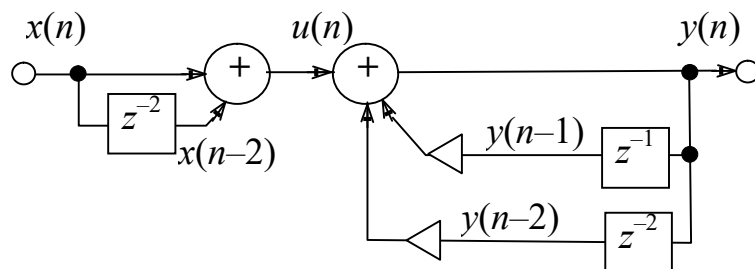


Рис. 1. Сигнальний граф алгоритму фільтра

верхніх частот

спеціалізованого обчислювача, у якого суматори – це відображення вершин додавання, блоки множення – вершин множення, а регістри проміжних операндів – елементів затримки. З цієї точки зору цей варіант алгоритму не є раціональним через надмірну кількість затримок. Для його оптимізації можна переставити місцями множники в формулі (5) та представити затримку z^{-2} як дві послідовні затримки z^{-1} :

$$u(n) = x(n) - a_1 u(n-1) - a_2 v(n-1);$$

$$v(n) = u(n-1);$$

$$y(n) = u(n) - v(n-1);$$

Результуючий сигнальний граф показано на рис. 2,а. Йому відповідає ГСПД на рис. 2,б. Цей граф прийнято називати канонічною формою рекурсивного фільтра, так як він вміщує мінімальне число елементів затримки для зберігання проміжних результатів – затриманих сигналів $u(n)$ та $v(n)$.

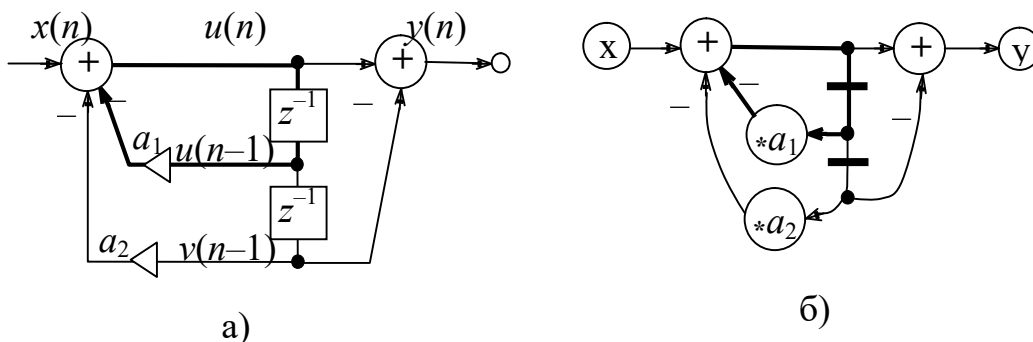


Рис. 2. Сигнальний граф алгоритму фільтра верхніх частот, а) та відповідний ГСПД, б)

Сигнальний граф і ГСПД можуть мати замкнені цикли. На рис. 2 такий цикл виділено товстою лінією. Якщо у замкненому циклі немає жодного елементу затримки, то сигнали в ньому будуть нескінченно переприсвоюватись протягом одного такту, тобто виникає **блокування** алгоритму. Тому обов'язковою умовою відсутності блокувань у сигнальному графі чи ГСПД є наявність принаймні по одному елементу затримки у всіх замкнених циклах. Іншою умовою відсутності блокувань є

наявність початкових, наприклад, нульових даних у всіх елементах затримки, які входять у такі цикли.

РЦФ на базі фазових фільтрів

Фазовий фільтр має модуль передавальної функції $H(z)$, що дорівнює $|H(z)| = 1$. Його фазо-частотна характеристика (ФЧХ) на частоті f_R має фазовий зсув, що дорівнює 180° . Якщо додаються сигнали від двох паралельних фазових фільтрів, вихідний сигнал придушується на частотах, для яких різниця фаз дорівнює 180° . Результуюча передавальна функція:

$$H_S = (H_1(z) \pm H_2(z))/2, \quad (6)$$

відповідає різним фільтрам нижніх (ФНЧ), високих частот (ФВЧ), смуговому або режекторному фільтру в залежності знака додавання (ФНЧ або ФВЧ) і від порядку фазових фільтрів. Наприклад, якщо функція $H_1(z)$ — другого порядку, як у даній лабораторній роботі, то при $H_2(z) = 1$ одержуємо режекторний фільтр, при $H_2(z) = -1$ — смуговий фільтр, при $H_2(z) = \pm z^{-1}$ — ФНЧ (+) та ФВЧ (–).

РЦФ на базі фазових фільтрів відрізняються стійкістю при малій розрядності коефіцієнтів, високою лінійністю АЧХ і ФЧХ в смузі пропускання, а також високою швидкодією. Параметри їх АЧХ, такі як положення частоти зрізу, крутизна перехідної смуги, прямо залежать від коефіцієнтів фільтра.

Передавальна функція ФНЧ, параметри якого можна перенастроювати, дорівнює:

$$H_S = (H_1(z) + z^{-1})/2, \quad (7)$$

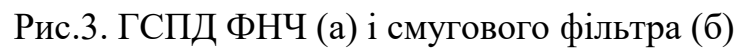
$$H_1(z) = \frac{b + a(b + 1)z^{-1} + z^{-2}}{1 + a(b + 1)z^{-1} + bz^{-2}}$$

$$a = \cos(2\pi f_R),$$

$$b = (1-t)/(1+t), \quad (8)$$

$$t = \operatorname{tg}(\pi \Delta f),$$

ГСПД ФНЧ та смугового фільтрів, які побудовані згідно з (6) показані на рис.3. Отже, ці ГСПД відрізняються лише знаком додавання і наявністю чи відсутністю затримки у другій гілці поширення вхідного сигналу.



Розглянемо детально кілька прикладів ГСПД ланок фазового фільтра другого порядку. При безпосередній реалізації формули (7) у ГСПД необхідно принаймні 6 вершин множення і 6 вершин суматорів. На щастя, є більш ефективний ГСПД для цієї формули, який називають графом хвильового фільтра за те, що він є моделлю хвильоводу (рис.4).



Формулу (7) можна скоротити за рахунок відмови від можливості прямого регулювання характеристик фільтра:

$$H_{12}(z) = \frac{b + cz^{-1} + z^{-2}}{1 + cz^{-1} + bz^{-2}}, \quad (9)$$

де $c = a(b + 1)$. Відповідний ГСПД, який відображає канонічну структуру фільтра, показаний на рис. 5. **Канонічна** структура фільтра має мінімальну кількість затримок, яка дорівнює порядку фільтра.

Формулі (9) відповідає різницеве рівняння (10).

$$y_i = b \cdot x_i + c \cdot x_{i-1} + x_{i-2} - c \cdot y_{i-1} - b \cdot y_{i-2}. \quad (10)$$

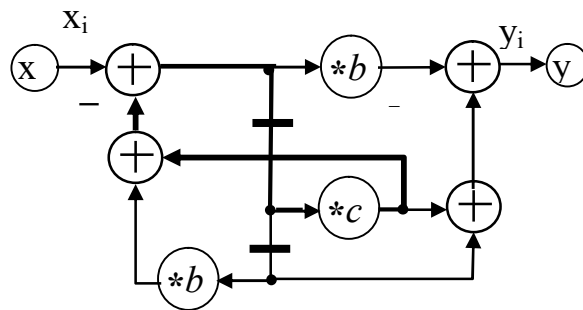


Рис.5. ГСПД для передаточної функції (9) у канонічній формі

Рівняння (10) можна переписати як

$$q_i = x_i - c \cdot q_{i-1} - b \cdot q_{i-2}; \quad (11)$$

$$y_i = b \cdot q_i + c \cdot q_{i-1} + q_{i-2}.$$

Якщо не економити затримки в дугах, то рівняння (11) виконуються у ГСПД, такому як на рис.6. Зате одержимо лише дві операції множення.

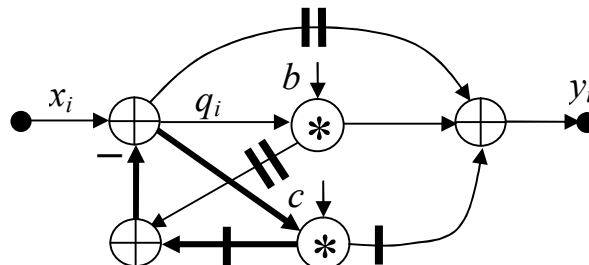


Рис.6. ГСПД для передаточної функції (9) з надлишковою кількістю затримок

При винесені у (10) загальних коефіцієнтів за дужки одержимо рівняння:

$$y_i = b \cdot (x_i - y_{i-2}) + c \cdot (x_{i-1} - y_{i-1}) + x_{i-2}. \quad (12)$$

Безпосередньо за рівнянням (12) побудований ГСПД, показаний на рис. 7. За рахунок повторного використання затримок авторам Mitra і Нірано вдалось побудувати ГСПД з назвою МН2В, яка показана на рис. 8.

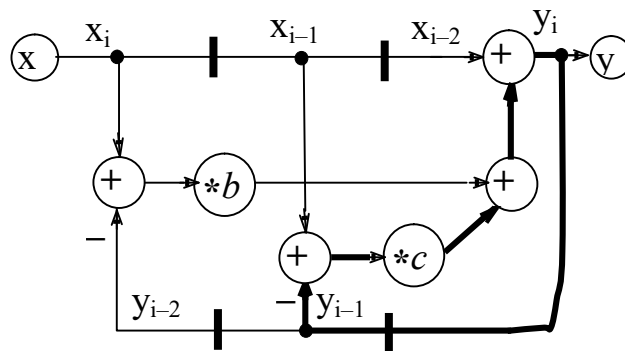


Рис.7. ГСПД для рівняння 12

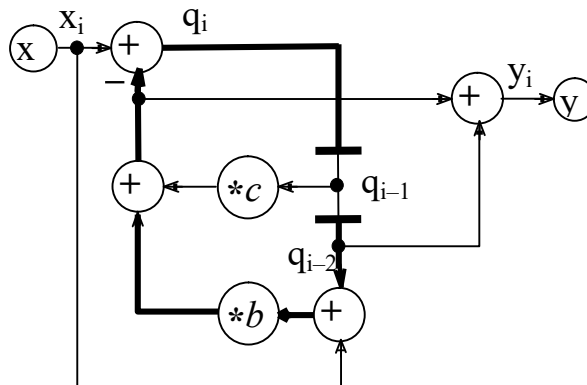


Рис.8. ГСПД МН2В

Товстою лінією у ГСПД на рис. 4 – 8 показаний критичний шлях. Довжина цього шляху є максимальною — $4t_s + 2t_m$ у ГСПД на рис. 4 і мінімальною — у ГСПД на рис. 5 та 6 — $2t_s + t_m$ за рахунок більшого числа вершин множення.

Тестування моделі цифрового фільтра

Визначення АЧХ є типовою процедурою його діагностики та тестування цифрових фільтрів. Для аналізу чи вимірювання АЧХ та ФЧХ на вхід РЦФ, що перевіряється, слід подавати комплексний сигнал $e^{-j\omega n}$, який називають *аналітичним*. Тут $\omega = 2\pi f$ — це частота сигналу, що розглядається. Часто використовують простий спосіб перевірки, коли замість аналітичного сигналу подають лише його складову — $\cos(\omega n)$, а АЧХ вимірюють на виході системи, як максимум результуючого сигналу $Re(H(\omega))$, тобто в моменти, коли друга складова — $Im(H(\omega)) = \sin(\omega n) = 0$. Недолік цього способу – у неточності вимірювання максимуму сигналу.

Більш точний спосіб полягає у одержанні уявної складової $Im(H(\omega))$ після пропускання результату $Re(H(\omega))$ через фільтр Гільберта. Але такий фільтр вносить суттєві перекручення в частотну характеристику. Тому для аналізу РЦФ слід застосувати сигнальний граф на рис. 9. В ньому використовуються два ідентичних екземпляри РЦФ з функцією $H(z)$, на які подаються синусна та косинусна складова аналітичного сигналу.

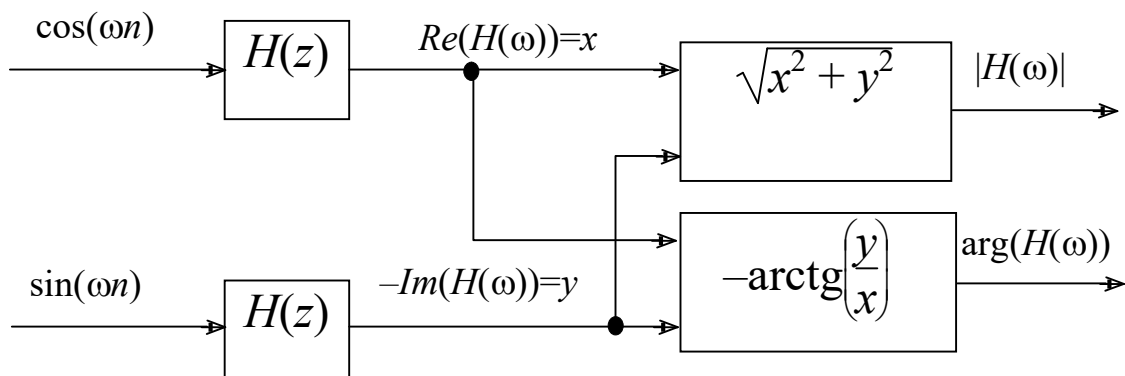


Рис. 9. Вимірювання АЧХ, ФЧХ реальної системи $H(z)$

Відповідно, з виходів РЦФ знімаються складові аналітичного сигналу відгуку РЦФ.

У лабораторній роботі пропонується стенд для іспитів, що має структуру як на рис. 10, який береться за адресою:

<http://kanyevsky.kpi.ua/en/useful-ip-cores/testbench-for-the-filter-testing/>

Порти та настроювальні константи цього стенду представлені у табл.2.

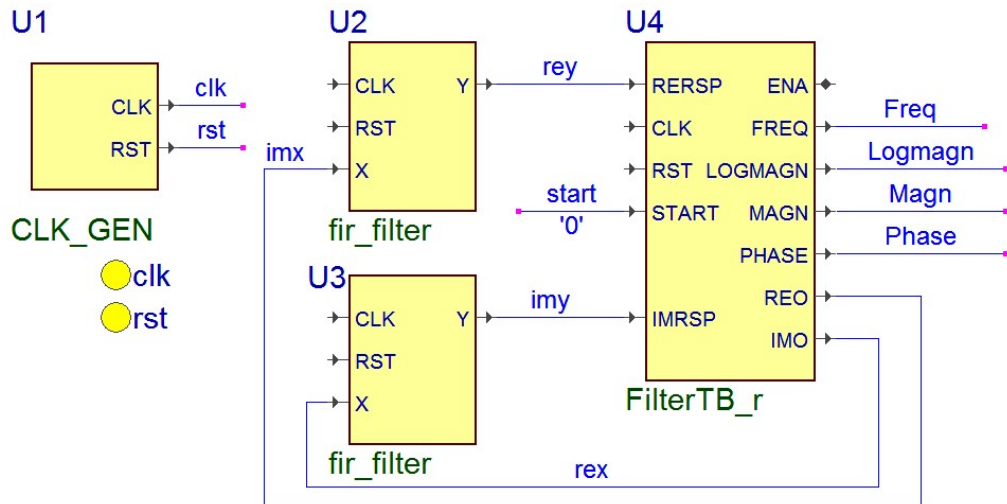


Рис.10. Стенд для іспитів фільтрів

Вибір розрядності фільтра

Переважаюча більшість ЦФ обчислюється в комп'ютерах чи ПЛІС у арифметиці цілих чисел або чисел з фіксованою комою. При програмуванні такого фільтра синтезують набір коефіцієнтів, що задовольняють задані вимоги і представлені з плаваючою комою. Потім вибирають кількість бітів квантування коефіцієнтів n_k , вхідних даних n_x та результатів n_y .

Як правило, $n_x, n_y \geq \log_2 D$, де D – динамічний діапазон зміни сигналу. Тобто на кожні 6 децибел динамічного діапазону припадає не менше одного розряду. Коефіцієнти фільтра масштабують та округлюють, так що при $|b_i| < 1$ цілий коефіцієнт дорівнює

$$b'_i = \lfloor 2^{n_x} \cdot b_i + 0.5 \rfloor. \quad (11)$$

Таблиця 2. Порти та настроювальні константи віртуального модуля стенду для випробувань цифрових фільтрів

Ім'я порта	Призначення порта
fsampl	Частота дискретизації f_o , наприклад 1000 кГц
fstrt	Початкова частота діапазону аналізу
deltaf	Приріст частоти d , так що через k кроків генератор видаватиме частоту $f_o + k \cdot d$
maxdelay	Затримка у кількості відліків від початку генерації частоти, після якої виконується оцінка результатів фільтрації. Має бути більшою за подвоєну максимальну групову затримку фільтра, який тестується.
slowdown	Коефіцієнт зменшення швидкості фільтра відносно тактової частоти. Якщо входні відліки поступають у кожному такті, то slowdown=1, якщо вони приходять у кожному парному такті, то slowdown=2 і т.д.
magnitude	Амплітуда синусного і косинусного сигналів, що генеруються.
REO,IMO	Синусний та косинусний сигнали, що генеруються
RERSP, IMRSP	Сигнали з виходів фільтрів, які є відгуками на синусний та косинусний сигнали, відповідно
FREQ	Код частоти сигналів, що генеруються, який дорівнює $f_o + k \cdot d$
MAGN	Обчислена амплітуда комплексного сигналу RERSP, IMRSP на частоті FREQ
LOGMAGN	Обчислена амплітуда комплексного сигналу RERSP, IMRSP на частоті FREQ у логарифмічному масштабі, тобто у децибелах, причому сигнал з амплітудою magnitude приймається за 0 дБ.
PHASE	Обчислена фаза комплексного сигналу RERSP, IMRSP на частоті FREQ, представлена у діапазоні $\pm\pi$
RST,START	Сигнали початкового встановлення
ENA	Сигнал дозволу прийому RERSP, IMRSP при slowdown >1

Результати фільтра обчислюють за формулою (1). При цьому накопичувальний суматор повинен мати таку розрядність, щоб не виникало переповнення. Причому розрядність добутку дорівнює $n_d = n_k + n_x$, а розрядність суматора повинна бути не меншою за величину $n_c = \log_2 S + n_k + n_x$, де S – теоретично можливий максимальний результат формули (1). Для правої половини формули, яка відповідає обчисленням у СІХ-фільтрі, максимум суми дорівнює сумі модулів усіх коефіцієнтів фільтра, що стоять у чисельнику формули передаточної функції, тобто

$$n_c = \log_2 \sum_{i=0}^M |b_i| + n_k + n_x. \quad (12)$$

Для лівої половини формули (1) максимум суми може бути значно більшим через підсилення сигналу завдяки зворотньому зв'язку. Таке підсилення пропорційне **добротності фільтра** — властивості підсилення сигналу на резонансній частоті. Для фазового фільтра (7), (8) добротність фільтра різко зростає при $b \rightarrow 1$.

На практиці, для фазових фільтрів та фільтрів на їх основі, дякуючи їх малій чутливості до похибок округлення, розрядність проміжних результатів вибирають як

$$n_c = n_d + n_x + 3, \quad (13)$$

де $n_d = 1, \dots, 6$ вибирається в залежності від добротності фільтра і уточнюється під час його моделювання. При цьому при недостатній величині n_d у деяких внутрішніх сигналів на частотах близьких до резонансної відбуватиметься переповнення розрядної сітки. Тобто, це значення зменшується до такої величини, поки не станеться подія переповнення. Наприклад, при $b < 0.7$ в (7) достатньо $n_d = 1$.

Розрядність коефіцієнтів фазових фільтрів може бути меншою за розрядність коефіцієнтів у інших моделях рекурсивних фільтрів. Достатньо розрядності $n_k = n_x$. Ця розрядність може бути зменшена за результатами тестування фільтра при умові задоволення вимог до АЧХ.

Результат фільтра $y(n)$ береться як старші n_y розрядів суми (1) з відкиданням молодших розрядів, тобто з усіканням або за іншим алгоритмом округлення.

Оскільки n_c може бути доволі великим числом, а ймовірність досягнення результатом максимального значення може бути мала (лише на резонансних частотах), то на практиці n_c вибирають дещо меншим, а додавання в (1) виконують за алгоритмом накопичення з **насиченням**. За

цим алгоритмом, якщо виникає переповнення суми, то результат замінюється максимальним числом розрядності n_c з відповідним знаком.

Насичення сигналу моделює аналогічний процес в аналогових схемах і дає значно менше перекручування сигналу ніж переповнення розрядної сітки.

Наступний приклад показує реалізацію насичення на VHDL. Нехай $n_c = 14$, $n_d = 3$ і $n_y = 8$, причому вважається, що результат за амплітудою менший 1, а фіксована кома стоїть перед 8-м розрядом. Тоді операція насичення накопиченого результату S програмується як

```
Y <= x"7F" when S(13 downto 10) > signed("0001"),  
    X"80" when S(13 downto 10) < signed("1111"), else  
    S(11 downto 3);
```

Тут константи $x"7F"$ і $x"80"$ представляють максимальне число 0,99 і мінімальне число $-0,99$.

Завдання для роботи

Розробити VHDL-проект цифрового фільтра з передаточною функцією (6).

Вид складових $H_1(z)$ і $H_2(z)$ задається за варіантом з таблиці 3. Номер завдання цієї та інших лабораторних робіт співпадає з номером студента у списку групи. Причому в таблиці 3 задано $H_2(z)$, частота зрізу (резонансу) f_R , ширина перехідної смуги Δf та вид ГСПД. Коефіцієнти a , b для функції (7) розраховуються за формулами (8), а коефіцієнт c для функції (9) обчислюється як $c = a(b + 1)$.

Розрядність вхідних і вихідних даних n_x , n_y :

у першій групі потоку розрядність – 14,

у другій — розрядність – 16,

у третій — розрядність – 24,

у четвертій — розрядність – 18.

Розрядність коефіцієнтів дорівнює $n_k = n_x - 2$. Розрядність внутрішніх проміжних результатів визначається за відношенням (13).

Таблиця 3. Параметри та функції до лабораторної роботи

№ вар.	f_R	Δf	Рис. ГСПД $H_1(z)$	$H_2(z)$
1	0,1	0,05	4	z^{-1}
2	0,125	0,05	5	z^{-1}
3	0,15	0,05	6	z^{-1}
4	0,175	0,05	7	z^{-1}
5	0,20	0,05	8	z^{-1}
6	0,225	0,05	4	z^{-1}
7	0,25	0,05	5	z^{-1}
8	0,275	0,05	6	z^{-1}
9	0,1	0,05	7	$-z^{-1}$
10	0,125	0,05	8	$-z^{-1}$
11	0,15	0,05	4	$-z^{-1}$
12	0,175	0,05	5	$-z^{-1}$
13	0,20	0,05	6	$-z^{-1}$
14	0,225	0,05	7	$-z^{-1}$
15	0,25	0,05	8	$-z^{-1}$
16	0,275	0,05	4	$-z^{-1}$
17	0,1	0,1	6	1
18	0,125	0,1	7	1
19	0,15	0,1	8	1
20	0,175	0,1	4	1
21	0,20	0,1	5	1
22	0,225	0,1	6	1
23	0,25	0,1	7	1
24	0,275	0,1	8	1
25	0,125	0,1	4	-1
26	0,15	0,1	5	-1
27	0,175	0,1	6	-1
28	0,20	0,1	7	-1
29	0,225	0,1	8	-1
30	0,25	0,1	4	-1

Модель фільтра є описом заданого ГСПД мовою VHDL. ГСПД оптимізується через ресинхронізацію чи конвеєризацію. Наприклад, доцільно додати затримки на вході та виході ГСПД, які відображаються у відповідні регістри вхідного та вихідного сигналів.

VHDL-опис фільтра повинна мати відповідні коментарі, які вказують автора і пояснюють виконання алгоритму.

Розроблений фільтр протестувати у стенді для іспитів, такому, як на рис. 8.

Також фільтр просинтезувати в САПР ПЛІС (Xilinx або Intel) для ПЛІС, вибраної довільно, з розміщенням і трасуванням.

Оформити протокол лабораторної роботи, у якому привести

- алгоритм фільтра, оптимізований ГСПД фільтра;
- VHDL-текст опису фільтра;
- графіки АЧХ випробування фільтра у лінійному і логарифмічному масштабах;
- результати синтезу – апаратні витрати і мінімальний період тактової частоти вибраної ПЛІС як скріншот роботи САПР ПЛІС.

Приклад виконання

Нехай маємо ГСПД як на рис. 8 і треба розробити ФНЧ з частотою зрізу $f_R=0,25$ (це, так званий, напівсмуговий фільтр) та перехідною смугою $\Delta f=0,1$. Розрядність вхідних даних – 12.

Тоді, згідно з (8) і (9), $a = \cos(2\pi \cdot 0,25) = 0$; $t = \text{tg}(\pi \cdot 0,1) = 0,325$; $b = (1-0,325)/(1+0,325) = 0,509$; $c = a(1 + b) = 0$. Отже, множення на c видаляється. Результуючий ГСПД фільтра показаний на рис. 11.

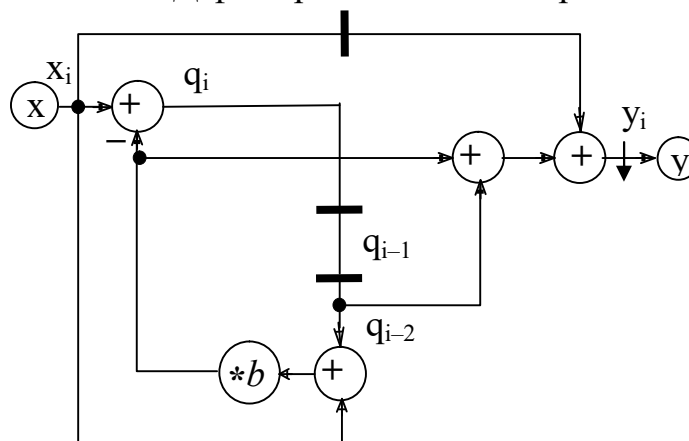


Рис. 11. ГСПД напівсмугового ФНЧ

На ньому стрілочкою поперек дуги представлено зсув вправо на один розряд, тобто, ділення навпіл.

Далі цей ГСПД підлягає ресинхронізації через конвеєризацію. При цьому дуги графа навантажуються затримками таким чином, щоб алгоритм залишався незмінним за виключенням латентної затримки і щоб критичний шлях був мінімізований. Результуючий ГСПД фільтра показаний на рис. 12. На ньому показані критичний шлях та усі сигнали, які беруть участь у обчисленнях.

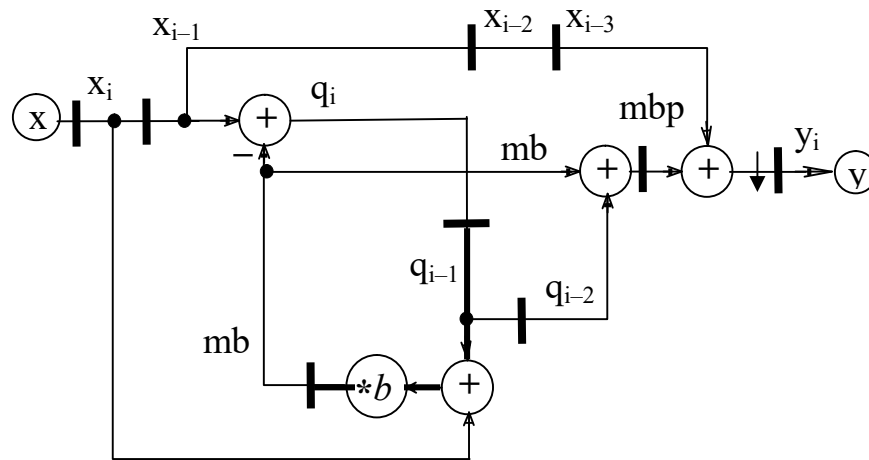


Рис. 12. Конвеєризований ГСПД напівсмугового ФНЧ

Вибирається розрядність проміжних даних $n_c = n_d + n_x + 3 = 2 + 12 + 3 = 17$. Розрядність коефіцієнтів $n_k = 12$. Розрядність добутку $n_d = n_k + n_c = 17 + 12 = 29$.

Результуючий VHDL-опис фільтра представлено нижче.

```
library IEEE;
use IEEE.STD_LOGIC_1164.all;
use IEEE.NUMERIC_STD.all;
entity LPF_HB_LAB is
    port(
        CLK : in STD_LOGIC;
        RST : in STD_LOGIC;
        X : in STD_LOGIC_VECTOR(11 downto 0);
        Y : out STD_LOGIC_VECTOR(11 downto 0)
    );
end LPF_HB_LAB;

architecture synt of LPF_HB_LAB is
    constant b:signed(11 downto 0):=
        to_signed(integer(0.509*2.0**11),12);
```

```

constant max:signed(3 downto 0):="0111";
constant min:signed(3 downto 0):="1100";
signal xi,xi_1,xi_2,xi_3:signed(16 downto 0);
signal qi,qi_1,qi_2,mbp:signed(16 downto 0);
signal mb:signed(28 downto 0);
signal yi:signed(11 downto 0);
begin
  LPF:process(CLK,RST)
    variable yt:signed(16 downto 0);
  begin
    if RST = '1' then
      xi<=(others=>'0');
      xi_1<=(others=>'0');
      xi_2<=(others=>'0');
      xi_3<=(others=>'0');
      yi<=(others=>'0');
      qi_1<=(others=>'0');
      qi_2<=(others=>'0');
      mb<=(others=>'0');
      mbp<=(others=>'0');
    elsif CLK='1' and CLK'event then
      xi<= RESIZE(signed(x&"000"),17);
      xi_1<= xi;
      xi_2<= xi_1;
      xi_3<= xi_2;
      mb <= b*(xi + qi_1);
      qi_1<= xi_1 - mb(27 downto 11);
      qi_2<= qi_1;
      mbp <= qi_2 + mb(27 downto 11);

      yt:= mbp + xi_3;
      if yt(16 downto 13) > max then
        yi<= x"7ff";
      elsif yt(16 downto 13) < min then
        yi<= x"800";
      else
        yi<=yt(14 downto 3);
      end if;
    end if;
  end process;
  Y<= std_logic_vector(yi);
end synt;

```

Результати випробувань моделі фільтра у вигляді АЧХ, логарифмічної АЧХ та ФЧХ показані на рис. 13. При цьому вхідний сигнал має амплітуду $2000 < 2^{11}$, частота дискретизації дорівнює 1000.

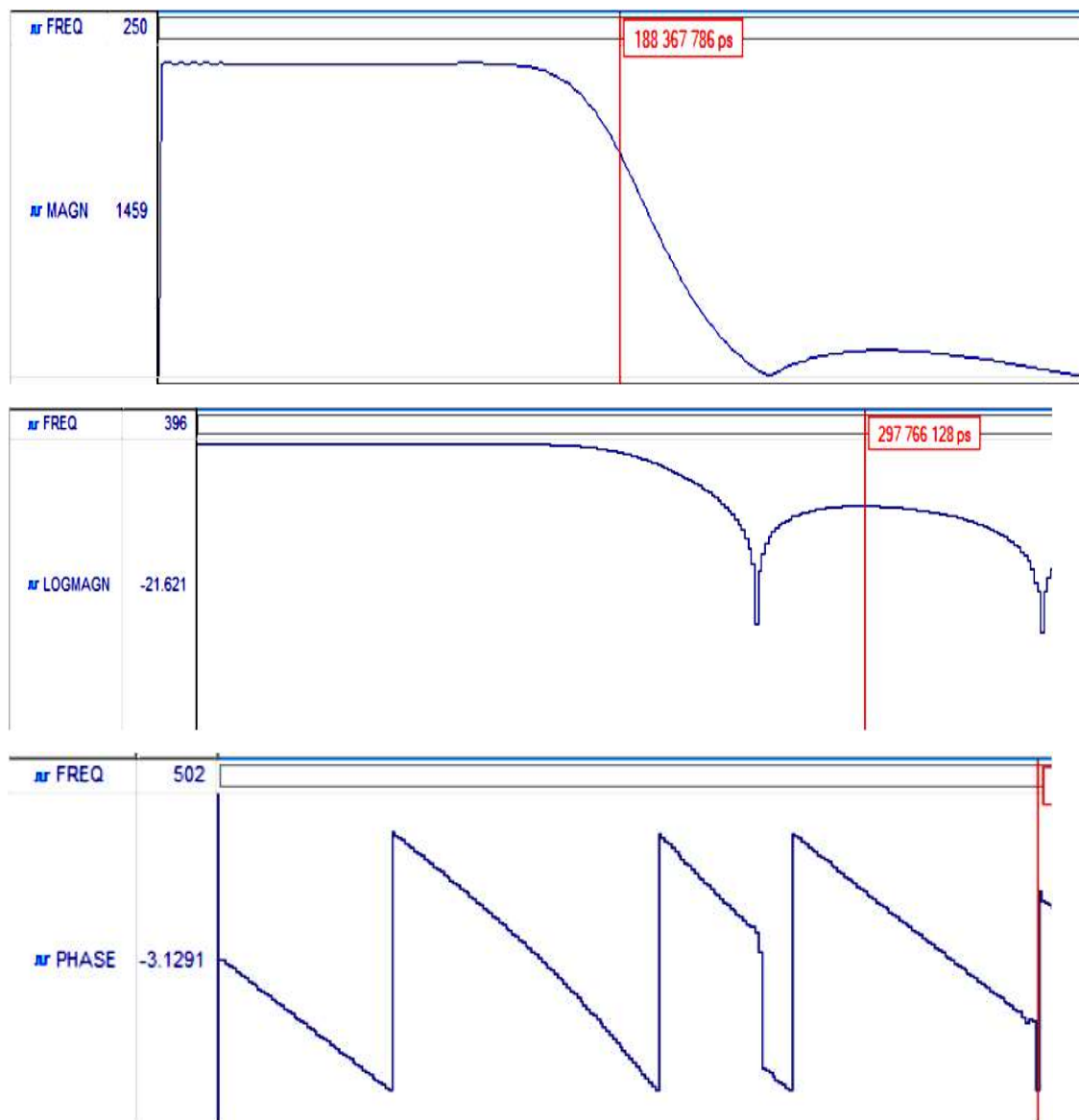


Рис.13. АЧХ, логарифмічна АЧХ і ФЧХ побудованого фільтра

З одержаних графіків видно, що фільтр має рівень придушення 21 дБ, є справді напівхвильовим (на частоті 250 передаточна функція дорівнює $1459/2000 \approx \sqrt{0,5}$), його фазова характеристика більш-менш лінійна, але на частоті 332 фаза різко змінюється на кут π , причому на цій частоті якраз відбувається провал у АЧХ.

При конфігуруванні фільтра в ПЛІС Xilinx Spartan-6 одержано наступні результати.

Device Utilization Summary			
Slice Logic Utilization	Used	Available	Utilization
Number of Slice Registers	125	18,224	1%
Number used as Flip Flops	125		
Number used as Latches	0		
Number used as Latch-thrus	0		
Number used as AND/OR logics	0		
Number of Slice LUTs	91	9,112	1%
Number used as logic	68	9,112	1%
Number using O6 output only	59		
Number using O5 output only	0		
Number using O5 and O6	9		
Number used as ROM	0		
Number used as Memory	0	2,176	0%
Number used exclusively as route-thrus	23		
Number with same-slice register load	23		
Number with same-slice carry load	0		
Number with other load	0		
Number of occupied Slices	31	2,278	1%
Number of MUXCYs used	72	4,556	1%
Number of LUT Flip Flop pairs used	107		
Number of DSP48A1s	1	32	3%

Constraint	Check	Worst Case Slack	Best Case Achievable	Timing Errors	Timing Score
ts_clk = PERIOD TIMEGRP "CLK" 7.93 ns HIG H 50%	SETUP HOLD	0.000ns 0.382ns	7.930ns	0 0	0 0

Отже, апаратні витрати синтезованого фільтра складають 31 CLB slices і 1 блок DSP48. Тактова частота фільтра досягає $f_C = 1/7.93 = 126 \text{ МГц}$.

Лабораторна робота 2

Проектування цифрового фільтра без блоків множення

Мета роботи: одержати знання та навички в розробці і тестуванні швидкодіючих цифрових фільтрів на ПЛІС з використанням технік зменшення апаратних витрат.

Теоретичні відомості

Маскуючі фільтри

При послідовному з'єднанні ступенів цифрових фільтрів результуюча АЧХ є перетином АЧХ цих ступенів. При цьому говорять, що АЧХ ступені маскує АЧХ інших ступенів, тобто така ступінь є маскуючим фільтром (рис. 14). Завдяки маскуванню, результуючий фільтр, що складається з простих ступенів фільтра, має високу якість АЧХ.

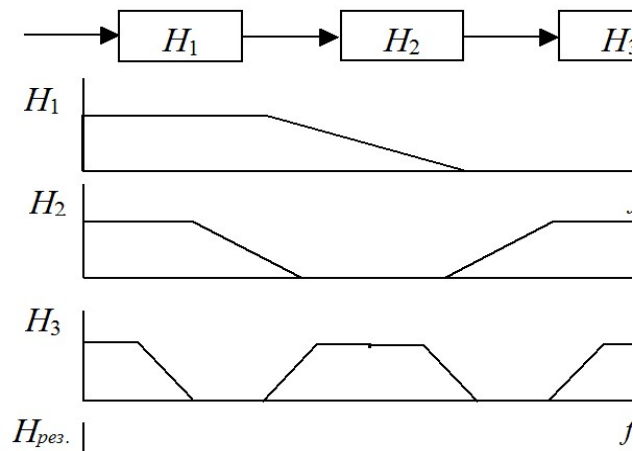


Рис.14. Приклад трьохступеневого фільтра з маскуючими фільтрами

Фільтри з кратними затримками

Кожному члену z^{-k} в передавальній функції $H(z)$ в сигнальному графі фільтра відповідає затримка на k циклів або ланцюжок k регістрів затримки в структурі ЦФ. Якщо в фільтрі кількість регістрів затримки збільшити в n разів, то виходить фільтр з АЧХ $H_n(z) = H(zn)$. АЧХ цього фільтра має форму таку саму, як у фільтра-прототипу $H(z)$, але в діапазоні $0 \rightarrow f_s$ вона

повторюється n разів, де f_s — частота дискретизації. На рис. 14 $H_1 = H(z)$, $H_2 = H(2z)$, $H_3 = H(4z)$.

Заміна блоку множення на константу

Більшість блоків множення у цифровій обробці даних — це блоки множення на константу. Якщо константа може бути представлена у канонічній системі числення з невеликою кількістю ненульвих розрядів, то варто блок множення замінити на спеціалізований помножувач у вигляді дерева суматорів часткових добутоків.

Наприклад, нехай є константа $y = 93_{10} = 1011101_2$. Тоді добуток буде

$$y \cdot x = 93x = (2^6 + 2^4 + 2^3 + 2^2 + 1)x,$$

тобто, при представленні множника y у двійковій системі маємо 5 ненульових розрядів, множення на які полягає у зсуві множеного x на відповідну кількість розрядів та додаванні на дереві з 4 суматорів. Схема такого дерева показана на рис. 15,а, на якому горизонтальними стрілочками показано зсув вправо на відповідну кількість розрядів.

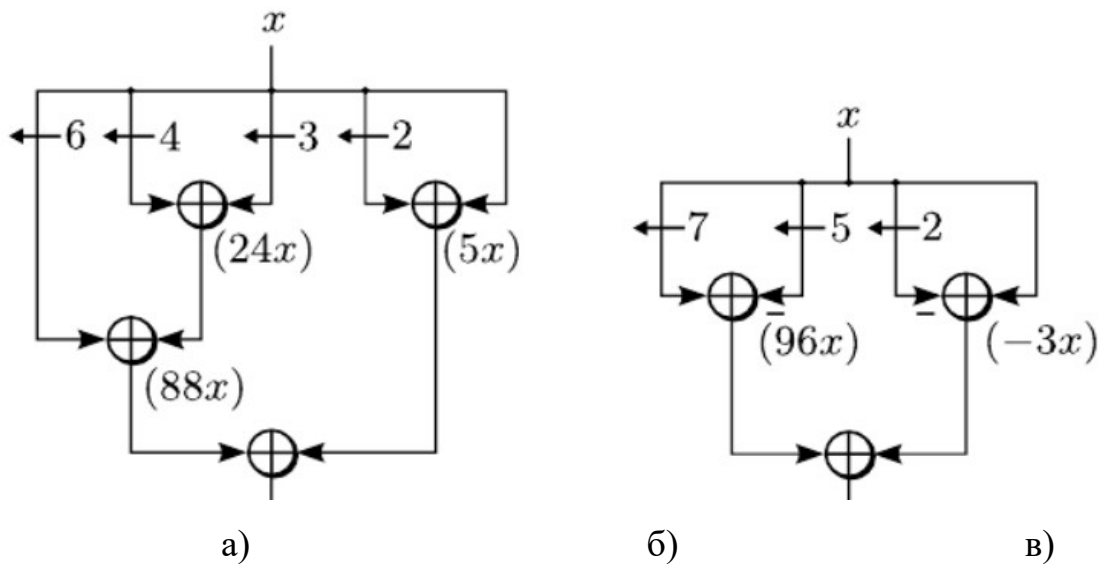


Рис.15. Приклади реалізації блоку множення на коефіцієнт

Якщо константу представити у знаковій канонічній системі числення, то кількість ненульових розрядів зменшується. При цьому слід зауважити, що складності суматора і віднімача є однаковими. Наприклад,

$$y = 93_{10} = 1011101_2 = 1100\bar{1}01_2 = 10\bar{1}00\bar{1}01_2 = (2^7 - 2^5 - 2^2 + 1).$$

Тоді кількість суматорів зменшується до трьох, як на рис. 15,б. Ще покращити схему можна після факторизації множника. Наприклад,

$$10\bar{1}00\bar{1}01_2 = (2^1 + 1) \cdot (2^5 - 1).$$

Тоді кількість суматорів зменшується до двох, як на рис. 15,в.

Завдання для роботи

Розробити VHDL-проект цифрового фільтра без використання блоків множення з передаточною функцією

$$H(z) = H_4(z) \cdot H_M(z),$$

$$\text{де } H_4(z) = H_3(zk),$$

$$H_3(z) = (H_1(z) + H_2(z))/2,$$

$H_1(z)$ є такою самою, як у лабораторній роботі 1 і реалізується у відповідному ГСПД, $H_M(z)$, $H_2(z)$, a , b , k вибираються з таблиці 4.

Модель фільтра є описом заданого ГСПД мовою VHDL. ГСПД обов'язково оптимізується через ресинхронізацію чи конвеєризацію, тому що цьому сприяє використання затримок кратності k .

Розроблений фільтр протестувати у стенді для іспитів, такому, як на рис. 8.

Також фільтр просинтезувати в САПР ПЛІС (Xilinx або Intel) для ПЛІС, вибраної довільно.

Оформити протокол лабораторної роботи, у якому привести

- алгоритм фільтра, ГСПД фільтра;
- VHDL-текст опису фільтра, який повинен мати відповідні коментарі, що вказують автора і пояснюють виконання алгоритму.

Таблиця 4. Параметри та функції до лабораторної роботи

№ вар.	a	b	$H_2(z)$	k	$H_M(z)$
1	-0,3125	0,75	-1	2	$(1+z^{-1}+z^{-2}+z^{-3}+z^{-4})/8$
2	-0,125	0,75	-1	2	$(1-z^{-1}+z^{-2}-z^{-3}+z^{-4})/8$
3	-0,625	0,75	-1	2	$(1+4z^{-1}+6z^{-2}+4z^{-3}+z^{-4})/16$
4	-0,875	0,75	-1	2	$(1-4z^{-1}+6z^{-2}-4z^{-3}+z^{-4})/16$
5	-0,3125	0,5	$-z^{-1}$	2	$(1+4z^{-1}+6z^{-2}+4z^{-3}+z^{-4})/16$
6	-0,75	0,5	$-z^{-1}$	2	$(1+z^{-1}+z^{-2}+z^{-3}+z^{-4})/8$
7	-0,5	0,5	$-z^{-1}$	2	$(-1+3z^{-1}+5z^{-2}+3z^{-3}-z^{-4})/8$
8	-0,25	0,5	$-z^{-1}$	2	$(1+5z^{-1}+10z^{-2}+10z^{-3}+5z^{-4}+z^{-5})/32$
9	-0,875	0,5	z^{-1}	2	$(1+z^{-1}+z^{-2}+z^{-3}+z^{-4}+z^{-5}+z^{-6}+z^{-7})/8$
10	0,5	0,5	z^{-1}	2	$(1+z^{-1})(1+z^{-1}+z^{-2}+z^{-3})/8$
11	-0,25	0,25	z^{-1}	3	$(2+5z^{-1}+7z^{-2}+5z^{-3}+2z^{-4})/32$
12	-0,5	0,25	z^{-1}	3	$(1+z^{-1}+z^{-2}+z^{-3}+z^{-4}+z^{-5})/8$
13	-0,125	0,25	z^{-1}	3	$(1+z^{-1}+2z^{-2}+z^{-3}+z^{-4})/8$
14	-0,75	0,25	z^{-1}	3	$(1-z^{-1}+z^{-3}-z^{-4})/4$
15	-0,75	0,25	z^{-1}	3	$(1+0.7z^{-1}-0.7z^{-3}-z^{-4})/4$
16	-0,625	0,25	z^{-1}	3	$(1-z^{-1}+z^{-2}-z^{-3}+z^{-4}-z^{-5})/8$
17	-0,75	0,25	z^{-1}	3	$(1-0.7z^{-1}+0.7z^{-3}-z^{-4})/4$
18	-0,5	0,25	z^{-1}	2	$(1+z^{-1}+z^{-3}+z^{-4})/4$
19	0,25	0,25	z^{-1}	2	$(-1+z^{-1}+z^{-3}-z^{-4})/4$
20	-0,625	0,25	z^{-1}	2	$(1-z^{-1}+z^{-3}-z^{-4})/4$
21	-0,5	0,25	z^{-1}	2	$(-1+2z^{-2}-z^{-4})/4$
22	-0,75	0,25	z^{-1}	2	$(1+2z^{-1}+2z^{-2}+2z^{-3}+z^{-4})/8$
23	-0,6	0,25	z^{-1}	2	$(1-2z^{-1}+2z^{-2}-2z^{-3}+z^{-4})/8$
24	0	0,25	z^{-1}	2	$(1+z^{-1}+z^{-2}+z^{-3}+z^{-4}+z^{-5}+z^{-6})/8$
25	-0,1	0,25	z^{-1}	2	$(1-z^{-2}+z^{-4}-z^{-6})/4$
26	-0,15	0,25	z^{-1}	3	$(1+1.4z^{-1}+z^{-2}-z^{-4}-1.4z^{-5}-z^{-6})/8$
27	-0,5	0,25	z^{-1}	3	$(1+z^{-1}+z^{-2}+z^{-3}+z^{-4})/8$
28	-0,625	0,25	z^{-1}	3	$(-3-2z^{-1}+5z^{-2}+5z^{-3}-2z^{-4}-3z^{-5})/32$
29	-0,5	0,25	z^{-1}	3	$(1+5z^{-1}+10z^{-2}+10z^{-3}+5z^{-4}+z^{-5})/32$
30	-0,75	0,25	z^{-1}	3	$(3-2z^{-1}+5z^{-2}+5z^{-3}-2z^{-4}+3z^{-5})/32$

– графіки АЧХ випробування фільтра у лінійному і логарифмічному масштабах;

– результати синтезу – апаратні витрати і мінімальний період тактової частоти вибраної ПЛІС як скріншот роботи САПР ПЛІС.

Номер завдання цієї та інших лабораторних робіт співпадає з номером студента у списку групи.

Розрядність вхідних, вихідних та проміжних даних така сама, як у лабораторній роботі 1.

Приклад виконання

Нехай маємо ГСПД частини НІХ-фільтра таку, як на рис. 11, і треба розробити ФНЧ з $a = 0$, $b = 0,5625$, $k = 2$; $H_M(z) = (1 + 4z^{-1} + 6z^{-2} + 4z^{-3} + z^{-4})/16$. Розрядність вхідних даних – 12.

ГСПД фільтра показаний на рис. 16. Він складається з НІХ-частини (зліва) і СІХ-частини (справа).

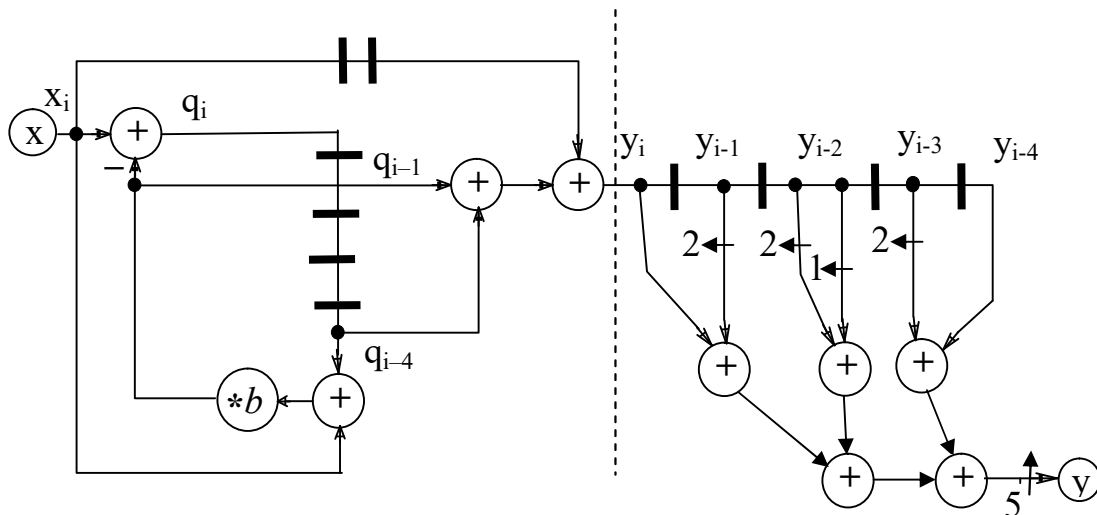


Рис. 16. ГСПД ФНЧ

Далі цей ГСПД підлягає ресинхронізації через конвеєризацію. Також блок множення на b замінюється на спеціалізований блок множення на основі суматора. При цьому $b = 0,4375 = 0,100\bar{1}$ означає, що множення виконується як віднімання операнду, який зсунуто на 4 розряди від операнду, який зсунуто на 1 розряд. Аналогічно реалізоване множення на коефіцієнти в СІХ-частині. Результуючий ГСПД фільтра показаний на рис. 17. На ньому показані усі сигнали, які беруть участь у обчисленнях.



Вибирається розрядність проміжних даних в НІХ-частині

$n_c = n_d + n_x + 3 = 2+12+3 = 17$. Розрядність коефіцієнтів $n_k = 12$. З урахуванням формули (12) розрядність проміжних даних в СІХ-частині $n_v = n_c + 4 = 17+4=21$.

Результуючий VHDL-опис фільтра представлено нижче.

```

library IEEE;
use IEEE.STD_LOGIC_1164.all;
use IEEE.NUMERIC_STD.all;
entity LPF_HB_LAB2 is
    port(
        CLK : in STD_LOGIC;
        RST : in STD_LOGIC;
        X : in STD_LOGIC_VECTOR(11 downto 0);
        Y : out STD_LOGIC_VECTOR(11 downto 0)
    );
end LPF_HB_LAB2;

architecture synt of LPF_HB_LAB2 is
    constant max:signed(3 downto 0):="0111";
    constant min:signed(3 downto 0):="1100";
    signal xi,xi_1,xi_2,xi_3,xi_4,xi_5:signed(16 downto 0);
    signal qi,qi_1,qi_2,qi_3,mb,qpx,mbp:signed(16 downto 0);
    signal yi,yi_1,yi_2,yi_3,yi_4:signed(16 downto 0);

```

```

signal y01,y22,y34,y012,y34_1:signed(20 downto 0);
signal ys:signed(11 downto 0);
begin
  IIR:process(CLK,RST)
  begin
    if RST = '1' then
      xi<=(others=>'0');          xi_1<=(others=>'0');
      xi_2<=(others=>'0');        xi_3<=(others=>'0');
      xi_4<=(others=>'0');        xi_5<=(others=>'0');
      yi<=(others=>'0');          qi<=(others=>'0');
      qi_1<=(others=>'0');        qi_2<=(others=>'0');
      qi_3<=(others=>'0');        qpx<=(others=>'0');
      mb<=(others=>'0');          mbp<=(others=>'0');
    elsif CLK='1' and CLK'event then
      xi<= RESIZE(signed(x&"000"),17);
      xi_1<= xi;
      xi_2<= xi_1;
      xi_3<= xi_2;
      xi_4<= xi_3;
      xi_5<= xi_4;
      qpx <= xi + qi_1;
      mb  <= shift_right(qpx,1) - shift_right(qpx,4);
      qi  <= xi_2 - mb;
      qi_1<= qi;
      qi_2<= qi_1;
      qi_3<= qi_2;
      mbp <= qi_3 + mb;
      yi<= mbp + xi_5;
    end if;
  end process;

  FIR:process(CLK,RST)
  variable yt:signed(20 downto 0);
  begin
    if RST = '1' then
      yi_1<=(others=>'0');        yi_2<=(others=>'0');
      yi_3<=(others=>'0');        yi_4<=(others=>'0');
      y01 <=(others=>'0');        y012 <=(others=>'0');
      y22 <=(others=>'0');        y34  <=(others=>'0');
      y34_1<=(others=>'0');        ys  <=(others=>'0');
    elsif CLK='1' and CLK'event then
      yi_1<= yi;
      yi_2<= yi_1;
      yi_3<= yi_2;
      yi_4<= yi_3;
      y01 <= yi + resize((yi_1 &"00"),21) ;
      y22 <= resize((yi_2 &"00"),21) + (yi_2 & "0");
      y34 <= yi_4 + resize((yi_3 & "00"),21) ;
      y34_1<= y34;
      y012 <= y01 + y22;
      yt:= y012 + y34_1;
      ys<=yt(19 downto 8);
    end if;
  end process;
  Y<= std_logic_vector(ys);
end synt;

```

Результати випробування на стенді для іспитів, такому як на рис.10, показані на рис. 18.



Рис. 18. АЧХ і логарифмічна АЧХ фільтра на рис. 17

З одержаних графіків видно, що фільтр має рівень придушення 35,6 дБ, частоту зрізу 0,112, частоту початку смуги придушення 0,175.

При конфігуруванні фільтра в ПЛІС Xilinx Spartan-6 одержано наступні результати.

Отже, апаратні витрати синтезованого фільтра складають 77 CLB slices, в тому числі 382 тригери і 263 логічні таблиці.

Тактова частота фільтра досягає $f_c = 1/2.666 = 375$ МГц. Це майже втричі більше, ніж для фільтра, який використовує блок множення (див. лабораторну роботу 1). Таким чином, методика використання спеціалізованих блоків множення не тільки зменшує апаратні витрати (блоки множення DSP48, кожен з яких вартий 20 суматорів), але й суттєво збільшує швидкодію фільтра.

Slice Logic Utilization	Used	Available	Utilization
Number of Slice Registers	382	18,224	2%
Number used as Flip Flops	382		
Number used as Latches	0		
Number used as Latch-thrus	0		
Number used as AND/OR logics	0		
Number of Slice LUTs	263	9,112	2%
Number used as logic	173	9,112	1%
Number using O6 output only	169		
Number using O5 output only	0		
Number using O5 and O6	4		
Number used as ROM	0		
Number used as Memory	1	2,176	1%
Number used as Dual Port RAM	0		
Number used as Single Port RAM	0		
Number used as Shift Register	1		
Number using O6 output only	1		
Number using O5 output only	0		
Number using O5 and O6	0		
Number used exclusively as route-thrus	89		
Number with same-slice register load	88		
Number with same-slice carry load	1		
Number with other load	0		
Number of occupied Slices	77	2,278	3%
Number of MUXCYs used	192	4,556	4%
Number of LUT Flip Flop pairs used	286		
Number with an unused Flip Flop	9	286	3%
Number with an unused LUT	23	286	8%
Number of fully used LUT-FF pairs	254	286	88%

constraint	Check	worst Case Slack	Best Case Achievable	Timing Errors	Timing Score
ts_clk = PERIOD TIMEGRP "CLK" 2.8 ns HIGH 50%	SETUP	0.248ns	2.552ns	0	0
	HOLD	0.388ns		0	0
	MINPERIOD	0.134ns	2.666ns	0	0

Література

1. Сергиенко А.М. VHDL для проектирования вычислительных устройств. Киев: ЧП "Корнейчук", ТИД ДС, 2003 — 208 с.

2. Сергієнко А.М. Генератор рекурсивних фільтрів без блоків множення. 2014. [електронний ресурс]

http://kanyevsky.kpi.ua/GEN_MODUL/APgen/APMF_help_ukr.php

3. Сергієнко А. М., Сергієнко А.А. Методика проектування цифрових фільтрів з застосуванням VHDL. //Праці 3 міжнародної конференції InfoCom'2016, 1 – 2 грудня 2016 р. -К.:НТУУ “КПІ”, ВПІ “Політехніка”. – 2016. -С. 56-57. [електронний ресурс]

<https://iconfs.net/w.infocom2016/metodyka-proektuvannya-tsyfrovykh-filtriv-z-zastosuvannyam-vhdl>

4. Сергієнко А.М., Виноградов Ю.М., Лесик Т.М. Цифрова обробка сигналів. Комп'ютерний практикум мовою VHDL. – Київ. – 2012. – 106 с. [електронний ресурс]

http://kanyevsky.kpi.ua/wp-content/uploads/2017/11/DSP_LabS.pdf

5. Schlichthärle D. Digital Filters: Basics and Design. – Springer. Berlin Heidelberg, –2011. – 527 p.

6. Khan S. A. Digital Design of Signal Processing Systems. A Practical Approach. – Wiley. – 211. – 586 p. – Available at

http://dSPACE.bhos.edu.az/jspui/bitstream/123456789/1146/1/%5BShoab_Ahmed_Khan%5D_Digital_Design_of_Signal_Process.pdf