

Прикарпатський національний університет імені Василя Стефаника

Фізико-технічний факультет

Кафедра комп'ютерної інженерії та електроніки

Стахів Андрій Ігорович

Andrii Stakhiv

УДК 004:42

Спеціальність 123 «Комп'ютерна інженерія»

Кваліфікаційна робота

на здобуття освітнього ступеня бакалавра

Розробка і реалізація цифрового фільтру на мікроконтролері AVR
Development and implementation of a digital filter on an AVR microcontroller

Науковий керівник:

доктор фізико-математичних наук,

професор кафедри

комп'ютерної інженерії та

електроніки Володимир

МАНДЗІЮК

Рецензент:

Кандидат к.ф.-м.н, професор

кафедри фізики і хімії твердого

тіла, Любомир НИКИРУЙ.

Івано-Франківськ

2024

АНОТАЦІЯ

Дана кваліфікаційна робота присвячена розробці та реалізації цифрового фільтру на базі мікроконтролера AVR. Цифрові фільтри широко застосовуються в різних галузях, таких як обробка сигналів, телекомунікації, мультимедіа та інші. Метою роботи є створення ефективного та оптимізованого цифрового фільтру, який можна реалізувати на недорогих мікроконтролерах сімейства AVR.

У теоретичній частині роботи розглядаються основні концепції цифрової обробки сигналів та принципи проектування цифрових фільтрів. Вивчаються різні типи фільтрів, їх характеристики та методи реалізації. Також аналізуються особливості мікроконтролерів AVR та їх архітектура.

Практична частина роботи зосереджена на розробці алгоритму цифрового фільтру та його реалізації на мікроконтролері AVR. Детально описується процес програмування та оптимізації коду для забезпечення високої продуктивності та ефективності фільтру. Наводяться результати тестування та аналізу роботи фільтру, а також його порівняння з іншими існуючими рішеннями.

Розроблений цифровий фільтр може бути використаний у різних застосунках, де потрібна обробка сигналів у реальному часі, наприклад, в системах керування, моніторингу, телекомунікаціях тощо.

Ключові слова: цифровий фільтр, мікроконтролер AVR, обробка сигналів, цифрова обробка сигналів, оптимізація коду.

					123.КІ-41.19			
<i>Змн.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>				
Розробив		Стахів А.І.			Анотація	<i>Літ.</i>	<i>Арк.</i>	<i>Аркушіє</i>
Перевірив		Мандзюк В.І.					3	1
Н. Контр.								
Затвердив								

ABSTRACT

This qualification work is devoted to the development and implementation of a digital filter based on the AVR microcontroller. Digital filters are widely used in various fields such as signal processing, telecommunications, multimedia, and others. The aim of the work is to create an effective and optimized digital filter that can be implemented on inexpensive microcontrollers of the AVR family.

The theoretical part of the work deals with the basic concepts of digital signal processing and the design principles of digital filters. Different types of filters, their characteristics and implementation methods are studied. The features of AVR microcontrollers and their architecture are also analyzed.

The practical part of the work is focused on the development of the digital filter algorithm and its implementation on the AVR microcontroller. The process of programming and optimizing the code to ensure high performance and efficiency of the filter is described in detail. The results of testing and analysis of the filter's operation are presented, as well as its comparison with other existing solutions.

The developed digital filter can be used in various applications where real-time signal processing is required, for example, in control systems, monitoring, telecommunications, etc.

Keywords: digital filter, AVR microcontroller, signal processing, digital signal processing, code optimization..

					123.KI-41.19			
<i>Змн.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>				
Розробив		Стахів А.І.			Abstract	<i>Лім.</i>	<i>Арк.</i>	<i>Аркушіє</i>
Перевірів		Мандзюк В.І.					4	1
Н. Контр.								
Затвердив								

ПЕРЕЛІК ОСНОВНИХ СКОРОЧЕНЬ

АЦП - аналого-цифровий перетворювач

ЦОС - цифрова обробка сигналів

КІХ - кінцева імпульсна характеристика

НІХ - нескінченна імпульсна характеристика

ДПФ - дискретне перетворення Фур'є

РПЗП - регістровий пакет загального призначення

ОЗП - оперативний запам'ятовуючий пристрій

МК - мікроконтролер

ПЗ - програмне забезпечення

АЧХ - амплітудно-частотна характеристика

					123.КІ-41.19			
<i>Змн.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>				
Розробив		Стахів А.І.			Abstract	<i>Літ.</i>	<i>Арк.</i>	<i>Аркушіє</i>
Перевірів		Мандзюк В.І.					4	1
Н. Контр.								
Затвердив								

Пояснювальна записка
до кваліфікаційної роботи
на тему:
«Розробка і реалізація цифрового фільтру на мікроконтролері AVR»

					123.KI-41.19			
<i>Змн.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>				
Розробив		Стахів А.І.			Пояснювальна записка	<i>Літ.</i>	<i>Арк.</i>	<i>Аркушів</i>
Перевірив		Мандзюк В.І.					7	66
Н. Контр.								
Затвердив								

ЗМІСТ

ВСТУП	3
РОЗДІЛ 1. ТЕОРЕТИЧНІ ВІДОМОСТІ.....	5
1.1. Характеристики цифрових фільтрів.....	5
1.2. Переваги і недоліки цифрових фільтрів перед аналоговими.....	10
1.3. Фільтр зі скінченною імпульсною характеристикою.....	11
1.4. Фільтр з нескінченною імпульсною характеристикою.....	12
РОЗДІЛ 2. РОЗРОБКА І РЕАЛІЗАЦІЯ.....	15
2.1. Синтез передавальної функції і параметрів фільтру.....	15
2.2. Межі стійкості фільтра.....	21
2.3. Вибір структурної схеми реалізації фільтра.....	23
2.4. Аналіз помилок квантування.....	31
2.5. Вибір мікроконтролера.....	34
2.6. Цифро-аналоговий перетворювач.....	44
2.7. Програма реалізації фільтру.....	47
2.8. Принципова схема та друкована плата фільт-ру.....	50
РОЗДІЛ 3. ОХОРОНА ПРАЦІ ПРИ РОБОТІ З ПК.....	52
РОЗДІЛ 4. ВИМОГИ БЕЗПЕКИ ПЕРЕД ПОЧАТКОМ РОБОТИ.....	55
ВИСНОВКИ.....	59
СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ.....	61
ДОДАТОК.....	64

					123.KI-41.19	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		2

ВСТУП

Цифрова обробка сигналів (ЦОС) є важливою галуззю, що знаходить застосування у багатьох сферах, таких як телекомунікації, мультимедіа, біомедична інженерія, радіолокація та інші. Одним з ключових компонентів ЦОС є цифрові фільтри, які використовуються для видалення небажаних складових сигналу або виділення корисної інформації.

Цифрові фільтри мають низку переваг порівняно з аналоговими, включаючи високу стабільність, точність, повторюваність характеристик та можливість легкої модифікації. Вони широко застосовуються для обробки звукових та відеосигналів, зв'язку, контролю та керування процесами тощо. Проте реалізація ефективних цифрових фільтрів вимагає ретельного проектування та оптимізації з урахуванням обмежених обчислювальних ресурсів вбудованих систем.

Мікроконтролери AVR є популярними та доступними пристроями, які широко використовуються у вбудованих системах через їх низьку вартість, енергоефективність та простоту програмування. Водночас, їх обмежені ресурси пам'яті та обчислювальна потужність створюють виклики при реалізації складних алгоритмів ЦОС, таких як цифрові фільтри.

Метою даної кваліфікаційної роботи є розробка та реалізація ефективного цифрового фільтру на базі мікроконтролера AVR. Робота охоплює наступні завдання:

Вивчення теоретичних основ ЦОС та проектування цифрових фільтрів, включаючи різні типи фільтрів, їх характеристики та методи реалізації.

Аналіз архітектури та можливостей мікроконтролерів AVR для реалізації алгоритмів ЦОС.

Розробка алгоритму цифрового фільтру з урахуванням обмежених ресурсів мікроконтролера AVR.

Оптимізація алгоритму та коду для забезпечення високої продуктивності та ефективності фільтру.

Реалізація розробленого цифрового фільтру на мікроконтролері AVR з використанням відповідного програмного забезпечення та середовища розробки.

					123.KI-41.19	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		3

Тестування та аналіз роботи фільтру, порівняння з іншими існуючими рішеннями.

Результати даної роботи можуть бути застосовані у різноманітних галузях, де необхідна обробка сигналів у реальному часі на вбудованих системах з обмеженими ресурсами, наприклад, в системах керування, моніторингу, телекомунікаціях тощо..

					<i>123.KI-41.19</i>	<i>Арк.</i>
<i>Зм.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>		4

РОЗДІЛ 1. ТЕОРЕТИЧНІ ВІДОМОСТІ

1.1. Характеристики цифрових фільтрів

Цифрові фільтри є важливою складовою електроніки, що дозволяє маніпулювати цифровими сигналами шляхом вибіркового підсилення або послаблення певних частотних діапазонів. На відміну від аналогових фільтрів, які працюють з безперервними сигналами та залежать від фізичних властивостей компонентів, цифрові фільтри оперують дискретними значеннями і їх характеристики визначаються математичними алгоритмами.

Ключовою характеристикою цифрового фільтра є його передатна функція, яка описує співвідношення між вхідним та вихідним сигналами у частотній області за допомогою z-перетворення. Ця функція дозволяє точно налаштувати бажану частотну характеристику фільтра, забезпечуючи ефективне виділення або придушення потрібних частот цифрового сигналу:

$$H(z) = \frac{Y(z)}{X(z)} = \frac{\sum_{n=0}^N b_n \cdot z^{-n}}{\sum_{m=0}^M a_m \cdot z^{-m}} = \frac{\sum_{n=0}^N d_n \cdot z^{-n}}{1 + \sum_{m=1}^M c_m \cdot z^{-m}}, \quad \text{де } c_m = \frac{a_m}{a_0}; \quad d_n = \frac{b_n}{a_0}, \quad a_0 \neq 0.$$

Для аналізу цифрових фільтрів використовується z-перетворення, яке є дискретним аналогом перетворення Лапласа для неперервних систем. Воно отримується шляхом відображення комплексної s-площини на одиничне коло z-площини за допомогою співвідношення $z = e^{(sT)}$, де T - період дискретизації вхідного та імпульсної характеристики фільтра. Зазвичай, для спрощення розрахунків приймається $T = 1$.

Підставляючи це значення у вираз для передатної функції цифрового фільтра $H(z)$, ми можемо отримати відповідну передатну функцію $H(s)$ у області перетворення Лапласа. Цей зв'язок дозволяє використовувати методи аналізу неперервних систем для дослідження властивостей цифрових фільтрів.

Комплексний коефіцієнт передачі цифрового фільтра $H(e^{j\omega})$ який описує його частотні характеристики, можна знайти шляхом підстановки $s = j\omega$ у вираз

									Арк.
Зм.	Арк.	№ докум.	Підпис	Дата					5

для $H(s)$. Таким чином, аналіз цифрових фільтрів часто виконується у частотній області за допомогою зручних математичних перетворень.:

$$H(e^{j\omega}) = \frac{\sum_{n=0}^N b_n \cdot e^{-j \cdot n \cdot \omega}}{\sum_{m=0}^M a_m \cdot e^{-j \cdot m \cdot \omega}} = \frac{\sum_{n=0}^N d_n \cdot e^{-j \cdot n \cdot \omega}}{1 + \sum_{m=1}^M c_m \cdot e^{-j \cdot m \cdot \omega}}, \quad \text{де } c_m = \frac{a_m}{a_0}; \quad d_n = \frac{b_n}{a_0}, \quad a_0 \neq 0.$$

Амплітудно-частотна характеристика (АЧХ) цифрового фільтру може бути отримана як модуль $H(e^{j\omega})$, а фазо-частотна (ФЧХ) як аргумент:

$$|H(\omega)| = |H(e^{j\omega})|;$$

$$\Phi(\omega) = \arg[H(e^{j\omega})] = \arctan \left(\frac{\Im[H(e^{j\omega})]}{\Re[H(e^{j\omega})]} \right).$$

Також вводиться поняття групової затримки як похідної від ФЧХ:

$$\tau(\omega) = -\frac{d}{d\omega} \Phi(\omega).$$

Амплітудно-частотна (АЧХ) та фазо-частотна (ФЧХ) характеристики, а також групова затримка цифрового фільтра є безперервними функціями частоти. Однак, комплексний коефіцієнт передачі $H(e^{j\omega})$ є періодичною функцією з періодом 2π , оскільки $H(e^{j\omega}) = H(e^{j(\omega+2\pi k)})$ для будь-якого цілого k . Ця рівність випливає з підстановки $\omega = \omega + 2\pi k$ у вираз для $e^{j\omega}$.

Завдяки періодичності, для повного аналізу характеристик цифрового фільтра достатньо розглянути їх на інтервалі $[-\pi, \pi]$ або $[0, 2\pi]$.

Крім передатної функції, цифровий фільтр також повністю визначається своєю імпульсною характеристикою $h[n]$. Застосування перетворення Фур'є до $h[n]$ дає комплексний коефіцієнт передачі $H(e^{j\omega})$. Якщо $H(e^{j\omega})$ є періодичною функцією частоти, то дискретну імпульсну характеристику $h[n]$ можна представити як ряд Фур'є:

$$H(e^{j\omega})$$

Таким чином, імпульсна характеристика та передатна функція цифрового фільтра пов'язані перетворенням Фур'є і повністю визначають його властивості:

$$h(k) = \int_0^{\pi} H(e^{j\omega}) \cdot \exp(j\omega \cdot k) d\omega, \quad k=0,1,2,\dots$$

Імпульсна характеристика $h[n]$ та передатна функція $H(e^{j\omega})$ цифрового фільтра пов'язані перетворенням Фур'є і повністю визначають його властивості:

$$h[n] = (1/2\pi) \int_{-\pi \rightarrow \pi} H(e^{j\omega}) e^{j\omega n} d\omega$$

Однак, розраховувати імпульсну характеристику через інтеграл досить незручно, особливо для фільтрів з нескінченною імпульсною характеристикою (БІХ). Проте, якщо фільтр стійкий, то $h[n]$ спадає зі збільшенням n , і можна розрахувати задану кількість відліків імпульсної характеристики за допомогою швидкого перетворення Фур'є (ШПФ).

Нехай потрібно розрахувати N перших відліків імпульсної характеристики фільтра, заданого передатною функцією $H(z)$.

Спочатку необхідно розрахувати комплексний коефіцієнт передачі $H_k = H(e^{j\omega_k})$ заданого фільтра. Для чисельного розрахунку задається сітка частот $\omega_k = k \cdot \Delta\omega$, $k=0..N-1$, $\Delta\omega = 2\pi/N$. Тоді на цій сітці частот розраховується $H_k = H(e^{j\omega_k})$, отримуючи N відліків комплексного коефіцієнта передачі фільтра. Після цього імпульсну характеристику можна знайти як $h(k) = \text{IFFT}(H_k)$, де ІДФТ позначає операцію зворотного дискретного перетворення Фур'є. Таким чином, ми розрахували характеристики фільтра із заданою передатною функцією. Цей шлях розрахунку приводить до комплексного коефіцієнта передачі у частотній області з подальшим перетворенням у часову область.

На рисунку 1.1 і 1.2 показані розраховані характеристики фільтру при $N = 64$;

$$H(z) = \frac{0.3}{1 - 0.7 \cdot z^{-1}}$$

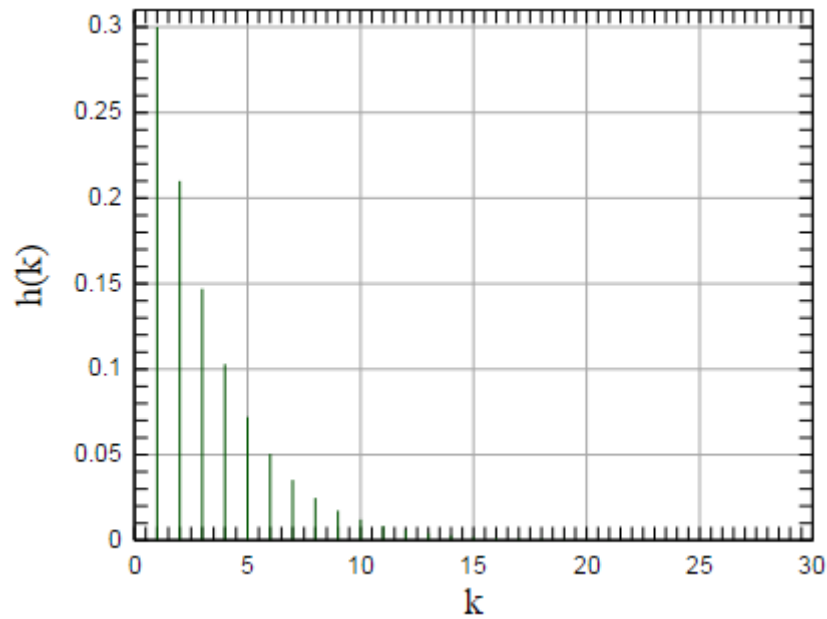


Рис.1.1 -Імпульсна характеристика фільтру

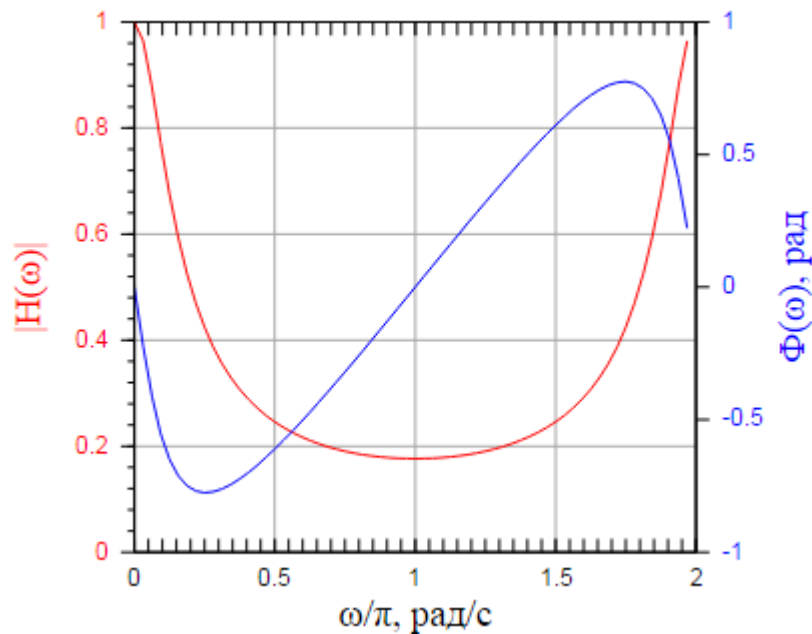


Рис.1.2 -Один період АЧХ і ФЧХ фільтру

Зверніть увагу, що на рисунку 1.2 по осі абсцис показана нормована частота ω/π , таким чином, амплітудно-частотна (АЧХ) і фазо-частотна (ФЧХ) характеристики представлені для нормованих частот від 0 до 2. Крім того, можна помітити, що АЧХ фільтра є симетричною щодо нормованої частоти 1, або $\omega/\pi = 1$, тобто $|H(\omega)| = |H(\omega + \pi)|$ а ФЧХ є антисиметричною.

Розглянемо тепер інший спосіб розрахунку характеристик фільтру - розрахунок у часовій області. Для цього наведемо структурну схему фільтру, заданого передатною функцією $H(z)$ (Рисунок 1.3).

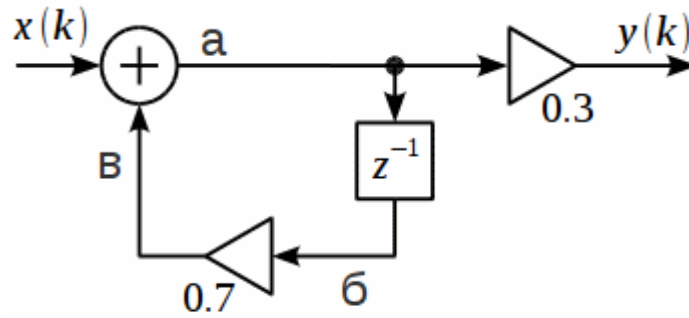


Рис.1.3 -Структурна схема фільтру

Щоб отримати імпульсну характеристику цифрового фільтру, необхідно подати на його вхід одиничний імпульсний сигнал $\delta[n]$:

$$x(k) = \delta(k) = \begin{cases} 1, & \text{при } k=0 \\ 0, & \text{при } k \neq 0 \end{cases}$$

Тоді на виході фільтру буде його імпульсна характеристика $h[n]$. Розрахуємо $h[n]$ за структурною схемою фільтру, заданого передатною функцією $H(z)$.

Нехай на вході при $n = 0$ сигнал $\delta[0] = 1$. Тоді на вузлах "а", "б" і "в" матимемо значення 1, 0 і 0 відповідно, отже, $h[0] = 1$. При $n = 1$ на вузлі "б" буде значення 1 (затримане з вузла "а"), на "в" - 0.7 (з ваговим коефіцієнтом 0.7), а на "а" - 0.7 (з ваговим коефіцієнтом 0.7), тому $h[1] = 0.7$. При $n = 2$ маємо: "а" = 0.49, "б" = 0.7, "в" = $0.7 \cdot 0.7 = 0.49$, тож $h[2] = 0.49$. Таким чином, можна продовжувати розрахунок для подальших відліків $h[n]$.

Обмеживши розрахунок N відліками, як і в попередньому випадку, отримаємо імпульсну характеристику $h[n]$, що збігається з наведеною на малюнку. Потім комплексний коефіцієнт передачі $H(e^{j\omega})$ фільтру можна обчислити, взявши швидке перетворення Фур'є від $h(k)$

Обидва способи розрахунку характеристик фільтру (у частотній або часовій області) мають приблизно однакову обчислювальну складність, і вибір залежить від конкретної ситуації та переваг користувача.

1.2. Переваги і недоліки цифрових фільтрів перед аналоговими

Цифрові фільтри мають низку переваг порівняно з аналоговими:

- Висока точність, оскільки точність аналогових фільтрів обмежена допусками на компоненти.
- Стабільність, адже на відміну від аналогових, передатна функція цифрових фільтрів не залежить від дрейфу характеристик елементів.
- Гнучкість налаштування та легкість зміни параметрів.
- Компактність реалізації, особливо для фільтрів з дуже низькими частотами зрізу, де аналогові фільтри потребують громіздких конденсаторів чи індуктивностей.

Однак цифрові фільтри також мають певні недоліки порівняно з аналоговими:

- Складність роботи з високочастотними сигналами, оскільки смуга частот обмежена частотою Найквіста, яка дорівнює половині частоти дискретизації. Тому для високочастотних сигналів застосовують аналогові фільтри або спочатку пригнічують високі частоти аналоговим фільтром перед цифровою обробкою.
- Складність роботи в реальному часі, оскільки обчислення повинні бути завершені протягом одного періоду дискретизації.
- Для забезпечення високої точності та швидкої обробки сигналів потрібні потужні процесори та додаткове високоточне і швидке апаратне забезпечення, як-от ЦАП та АЦП, що може бути дорогим.

Таким чином, вибір між цифровими та аналоговими фільтрами залежить від конкретних вимог додатку, таких як необхідна смуга частот, точність, стабільність, вартість та інші фактори.

					123.KI-41.19	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		10

1.3. Фільтр зі скінченною імпульсною характеристикою

Фільтр зі скінченною імпульсною характеристикою (СІХ-фільтр або FIR-фільтр, де FIR - скорочення від "finite impulse response") належить до класу лінійних цифрових фільтрів, особливістю яких є обмежена тривалість імпульсної характеристики. Після певного моменту часу значення імпульсної характеристики стають точно рівними нулю. Такі фільтри також називають нерекурсивними через відсутність зворотного зв'язку. Знаменник їхньої передатної функції є константою.

Різницеве рівняння, що описує зв'язок між вхідним $x[n]$ та вихідним $y[n]$ сигналами FIR-фільтра, має вигляд:

$$y(n) = b_0x(n) + b_1x(n-1) + \dots + b_Px(n-P)$$

Тут N - порядок фільтра, а $b[k]$ - його коефіцієнти. Іншими словами, значення будь-якого відліку вихідного сигналу визначається сумою зважених значень N попередніх відліків вхідного сигналу. Можна сказати, що значення виходу фільтра в будь-який момент часу є сумою відгуку на поточне значення входу та поступово згасаючих відгуків на N попередніх відліків сигналу, які ще впливають на вихід.

Після N відліків імпульсна характеристика стає рівною нулю.

Запишемо попереднє різницеве рівняння в компактному вигляді:

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

Для того, щоб знайти ядро фільтра, покладемо

$$x(n) = \delta(n)$$

де $\delta(n)$ — дельта-функція. Тоді імпульсна характеристика СІХ-фільтру може бути записана як:

$$h(n) = \sum_{i=0}^P b_i \delta(n-i)$$

Z-перетворення імпульсної характеристики дає нам передаточну функцію

СІХ-фільтру:

$$H(z) = \sum_{i=0}^P b_i z^{-i}$$

									Арк.
									11
Зм.	Арк.	№ докум.	Підпис	Дата				123.KI-41.19	

ФІР-фільтри (фільтри з скінченною імпульсною характеристикою) мають низку корисних властивостей, які роблять їх переважними у використанні порівняно з ПР-фільтрами (фільтрами з нескінченною імпульсною характеристикою) в деяких випадках:

- Стійкість. Усі ФІР-фільтри є стійкими за своєю природою, оскільки їхня імпульсна характеристика обмежена в часі.
- Відсутність зворотного зв'язку. Реалізація ФІР-фільтрів не потребує наявності ланок зворотного зв'язку в їхніх структурах, що спрощує їх проектування та реалізацію.
- Лінійність фазової характеристики. Фазову характеристику ФІР-фільтрів можна зробити лінійною, що є перевагою для певних застосувань, наприклад, в обробці мовних та аудіосигналів, де нелінійні спотворення фази є небажаними.

Таким чином, завдяки цим властивостям ФІР-фільтри часто є кращим вибором порівняно з ПР-фільтрами у випадках, коли критичними є стійкість, відсутність зворотного зв'язку та лінійність фазової характеристики. Однак, слід зазначити, що ПР-фільтри також мають свої переваги, наприклад, можливість реалізації більш гострих частотних характеристик при тому ж порядку фільтра.

1.4. Фільтр з нескінченною імпульсною характеристикою

Фільтр з нескінченною імпульсною характеристикою (НІХ-фільтр або ПР-фільтр, де ПР означає "infinite impulse response") належить до класу лінійних електронних фільтрів, які використовують один або кілька своїх попередніх виходів як вхідні сигнали, утворюючи тим самим зворотний зв'язок. Основною властивістю таких фільтрів є те, що їхня імпульсна характеристика має нескінченну тривалість у часовій області, а передатна функція має дрібно-раціональний вигляд. Такі фільтри можуть бути реалізовані як в аналоговій, так і в цифровій формі.

Прикладами НІХ-фільтрів є фільтри Чебишева, Баттерворта, Калмана та Бесселя.

					<i>123.KI-41.19</i>	<i>Арк.</i>
<i>Зм.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>		12

Таким чином, значення поточного вихідного відліку $y[n]$ залежить як від поточного та попередніх відліків вхідного сигналу $x[n]$, так і від попередніх відліків самого вихідного сигналу $y[n]$, що і зумовлює нескінченну тривалість імпульсної характеристики НІХ-фільтрів:

$$y(n) = b_0x(n) + b_1x(n-1) + \dots + b_Px(n-P) - a_1y(n-1) - a_2y(n-2) - \dots - a_Qy(n-Q)$$

де P порядок вхідного сигналу, b_i — коефіцієнти вхідного сигналу, Q — порядок зворотного зв'язку, a_i — коефіцієнти зворотного зв'язку, $x(n)$ — вхідний, а $y(n)$ — вихідний сигнали. Більш компактний запис різницевого рівняння:

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

Для того, щоб знайти ядро фільтра, покладемо

$$x(n) = \delta(n)$$

де $\delta(n)$ — дельта-функція. Тоді імпульсна перехідна функція (ядро фільтра) записується як

$$h(n) = \sum_{i=0}^P b_i \delta(n-i) - \sum_{k=1}^Q a_k h(n-k)$$

Z-перетворення імпульсної перехідної функції дає передавальну функцію НІХ-фільтру:

$$H(z) = \frac{\sum_{i=0}^P b_i z^{-i}}{1 + \sum_{k=1}^Q a_k z^{-k}}$$

Оцінка стійкості фільтра з нескінченною імпульсною характеристикою (НІХ-фільтра) проводиться на основі аналізу його передатної функції. Для дискретних НІХ-фільтрів необхідною і достатньою умовою стійкості є те, що всі полюси передатної функції за модулем мають бути менші за одиницю, тобто повинні лежати всередині одиничного кола на z -площині. Усі критерії стійкості, які застосовуються в теорії лінійних стаціонарних систем, такі як критерій стійкості Найквіста або критерій стійкості Рауса, також можуть бути використані для аналізу стійкості НІХ-фільтрів.

На відміну від фільтрів зі скінченною імпульсною характеристикою (СІХ-фільтрів або FIR-фільтрів), які завжди є стійкими, стійкість НІХ-фільтрів не є гарантованою і потребує перевірки шляхом аналізу розташування полюсів їхньої передатної функції. Якщо хоча б один полюс лежить поза одиничним колом на z -площині, то такий НІХ-фільтр буде нестійким.

Забезпечення стійкості є одним з ключових аспектів при проектуванні та реалізації НІХ-фільтрів, оскільки нестійкий фільтр може призвести до небажаної поведінки системи, наприклад, необмеженого наростання амплітуди вихідного сигналу.

					<i>123.KI-41.19</i>	<i>Арк.</i>
<i>Зм.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>		14

РОЗДІЛ 2. РОЗРОБКА ТА РЕАЛІЗАЦІЯ

2.1. Синтез передавальної функції і параметрів фільтру

Вихідні дані для розрахунку:

- Нижня смуга пропускання: 0 - 50 Гц
- Верхня смуга пропускання: 450 - 500 Гц
- Смуга придушення: 200 - 300 Гц
- Рівень пульсацій в смузі пропускання: 3 дБ
- Рівень загасання в смузі придушення: 20 дБ
- Частота дискретизації: 4 кГц

Для синтезу частотно-вибіркових рекурсивних цифрових фільтрів зручно скористатися методами перетворення аналогового фільтра в цифровий.

Такий синтез включає:

- Вибір методу відображення p -площини (аналогової області) в z -область (цифрову область).
- Розрахунок аналогового фільтра-прототипу за заданими вимогами до цифрового фільтра.
- Застосування до аналогового прототипу обраного методу відображення з p -площини в z -область.

Аналоговий фільтр-прототип, розрахований за вимогами до цифрового фільтра, називається фільтром-прототипом або просто прототипом.

Основні обмеження для методів синтезу цифрових фільтрів за прототипом:

- Збереження істотних властивостей амплітудно-частотної характеристики прототипу в АЧХ цифрового фільтра, що вимагає відображення уявної осі p -площини на одиничне коло z -області.
- Стійкий прототип повинен відображатися на стійкий цифровий фільтр, що вимагає відображення полюсів стійкого прототипу з лівої p -напівплощини всередину одиничного кола z -області.

					123.KI-41.19	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		15

- 's' – ознака аналогового розрахунку;

$$a = [1; 2548,3; 8,345 \cdot 10^6; 6,0361 \cdot 10^9; 5,6108 \cdot 10^{12}];$$

$$b = [1; 0; 5,0982 \cdot 10^6; 0; 5,6108 \cdot 10^{12}];$$

$$G(p) = \frac{p^4 + 5,0982 \cdot 10^6 p^2 + 5,6108 \cdot 10^{12}}{p^4 + 2548,3 p^3 + 8,345 \cdot 10^6 p^2 + 6,0361 \cdot 10^9 p + 5,6108 \cdot 10^{12}}$$

частотні характеристики (рис. 2.1)

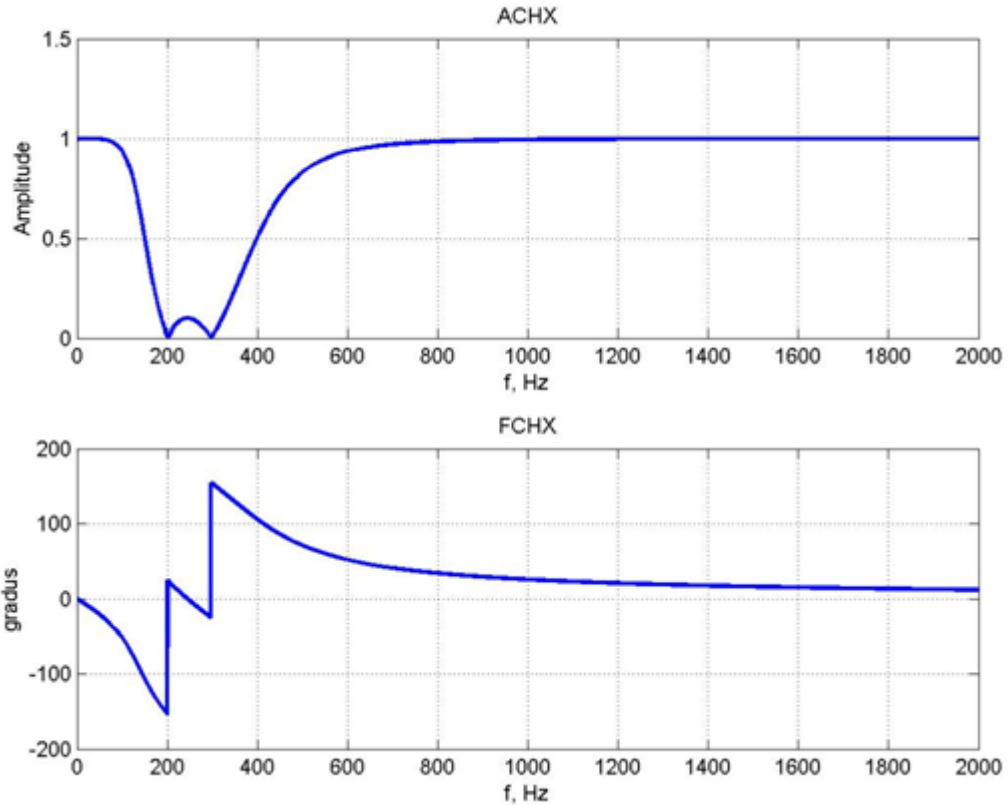


Рис.2.1 – Частотні характеристики

Зм.	Арк.	№ докум.	Підпис	Дата

Графік заданим вимогам можна побачити на ЛАЧХ (рис.2.2):

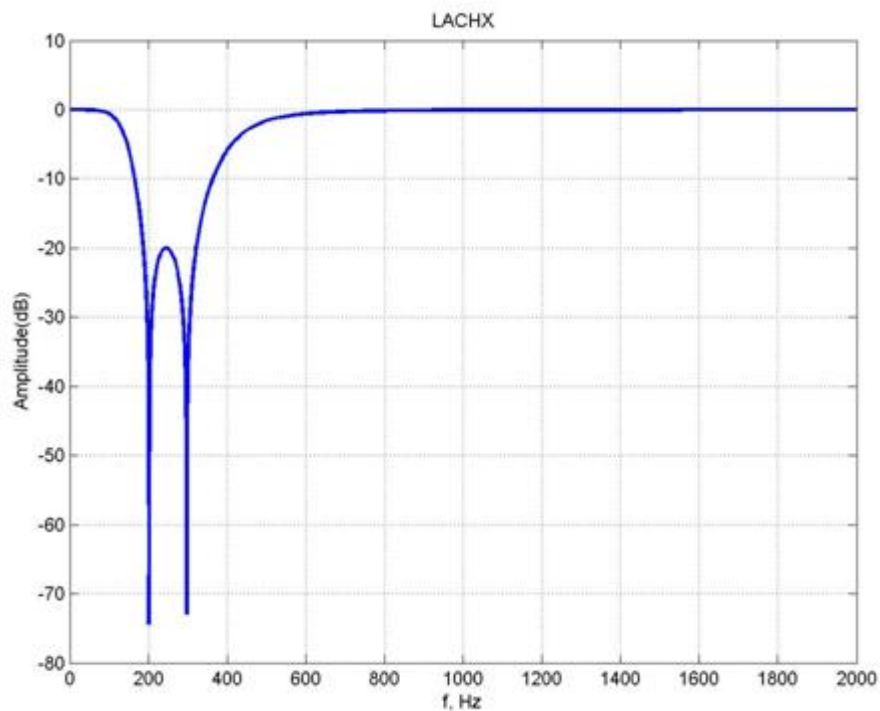


Рис.2.2 (а) – ЛАЧХ за заданими вимогами

У збільшеному вигляді:

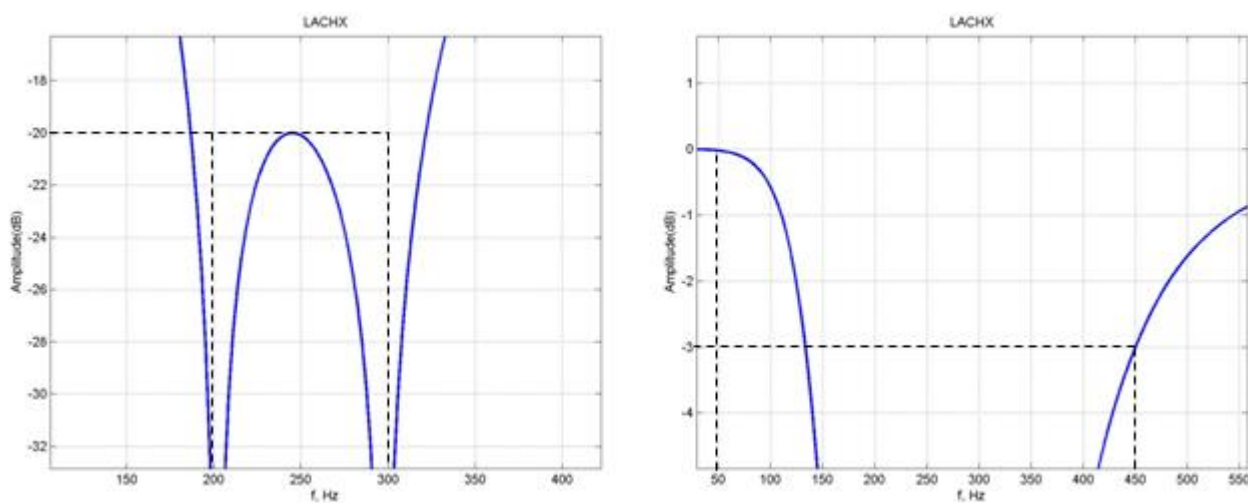


Рис.2.2 (б) – Збільшений вигляд ЛАЧХ

Зм.	Арк.	№ докум.	Підпис	Дата

Нулі і полюси АФ - прототипу (рис. 2.3):

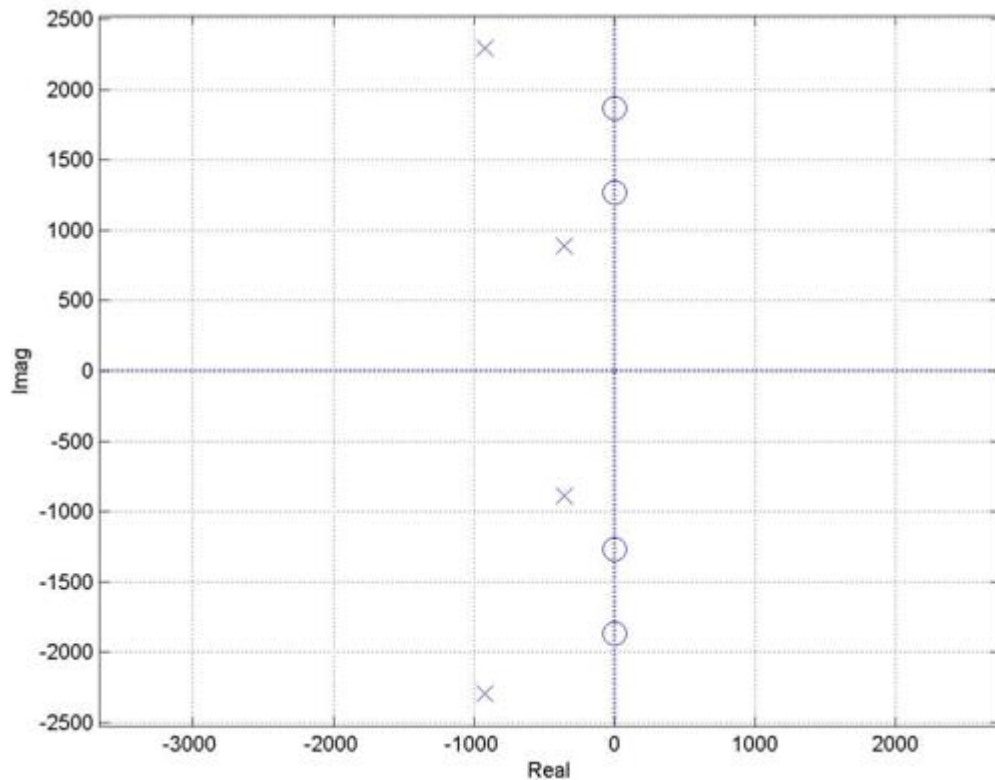


Рис.2.3 - О – нулі; X – полюси;

Оскільки всі полюси аналогового фільтра-прототипу лежать у лівій p-напівплощині, то цей прототип є стійким.

Тепер здійснимо синтез цифрового фільтра (ЦФ) з аналогового прототипу за допомогою методу білінійного z-перетворення, використовуючи стандартні засоби MATLAB. Скористаємось функцією `bilinear` та отримаємо коефіцієнти чисельника і знаменника передатної функції синтезованого цифрового фільтра:
 $[bz, az] = \text{bilinear}(b, a, fd);$

Де bz і az - коефіцієнти поліномів чисельника і знаменника передатної функції синтезованого цифрового фільтра;
 fd - частота дискретизації в герцах.

Метод білінійного z-перетворення відображає лівий напівплощині p-площини (стійкий аналоговий фільтр) всередину одиничного кола z-площини, забезпечуючи тим самим стійкість отриманого цифрового фільтра. Крім того, цей метод зберігає форму амплітудно-частотної характеристики в околиці нульової частоти.

Отримані коефіцієнти bz і az повністю визначають передатну функцію синтезованого цифрового фільтра з бажаними характеристиками згідно з вихідними вимогами до смуг пропускання, придушення, рівнів пульсацій та загасання.;

$$az = [1; -3,1517; 3,931; -2,3125; 0,54815];$$

$$bz = [0,73938; -2,7321; 4; -2,7321; 0,73938];$$

Запишемо передавальну функцію ЦФ

$$H(z) = \frac{0,73938z^4 - 2,7321z^3 + 4z^2 - 2,7321z + 0,73938}{z^4 - 3,1517z^3 + 3,931z^2 - 2,3125z + 0,54815}$$

Частотні характеристики ЦФ можна побачити на рисунку 2.4:

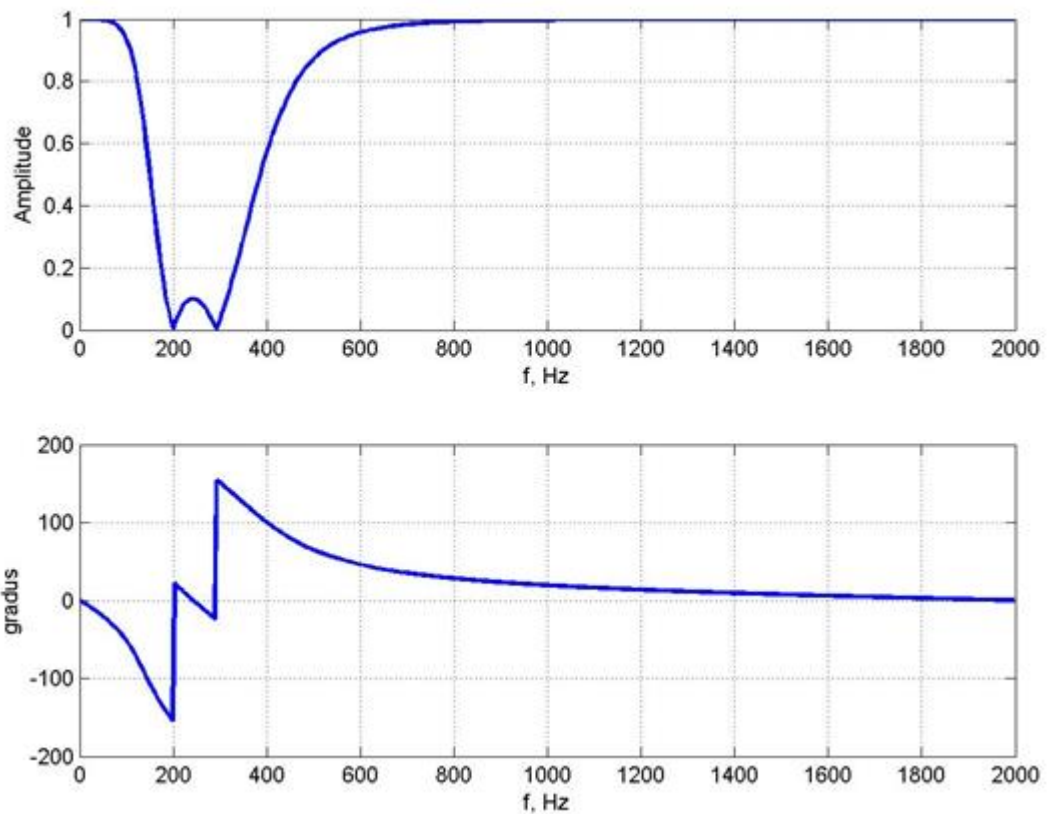


Рис.2.4 (а) - АЧХ и ФЧХ ЦФ

Зм.	Арк.	№ докум.	Підпис	Дата

На збільшеній ЛАЧХ ЦФ видно виконання заданих вимог:

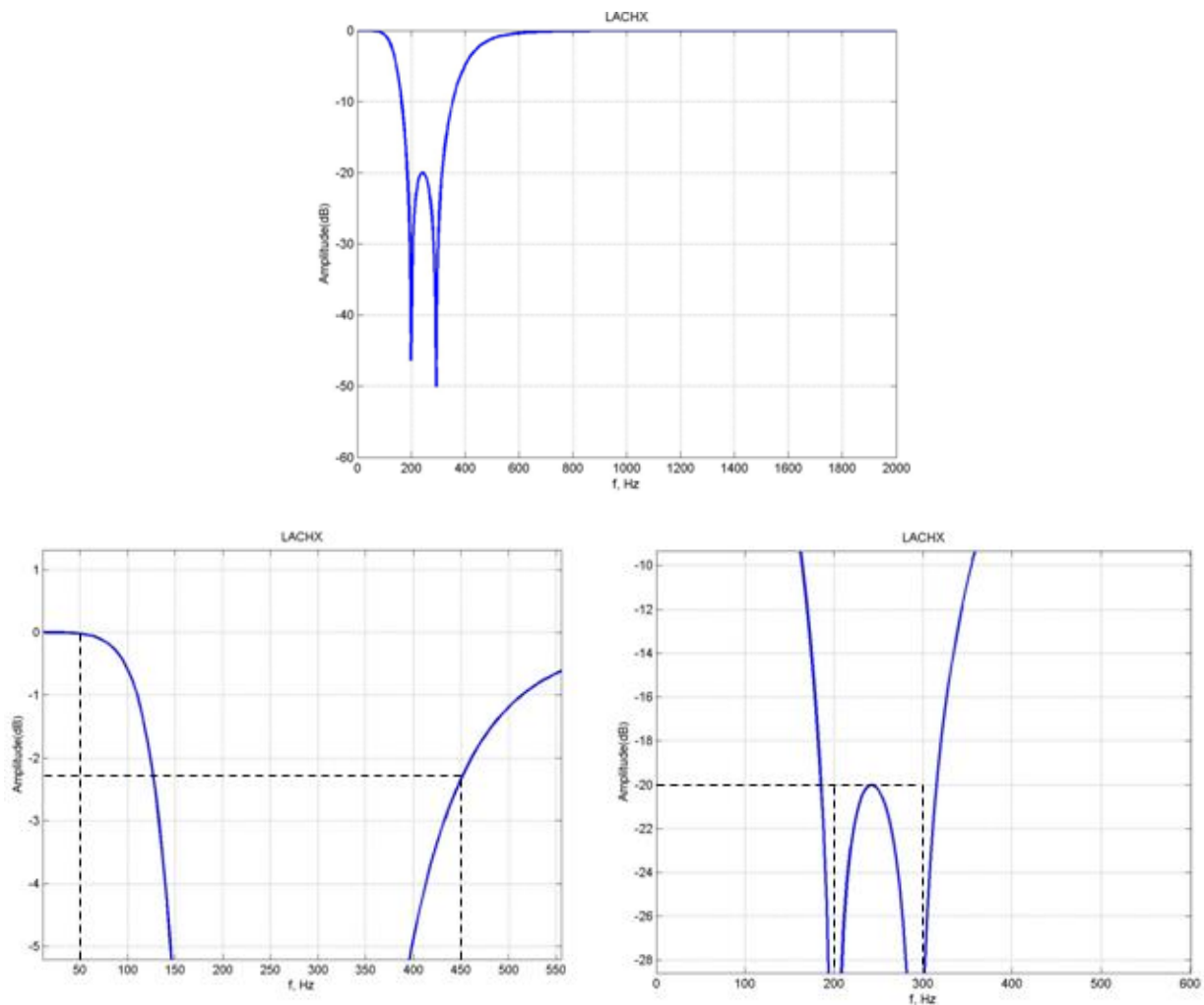


Рис.2.4 (б) – Збільшений вигляд АЧХ та ФЧХ ЦФ

Зм.	Арк.	№ докум.	Підпис	Дата

123.КІ-41.19

Арк.

20

Нулі і полюси ЦФ (рис.2.5):

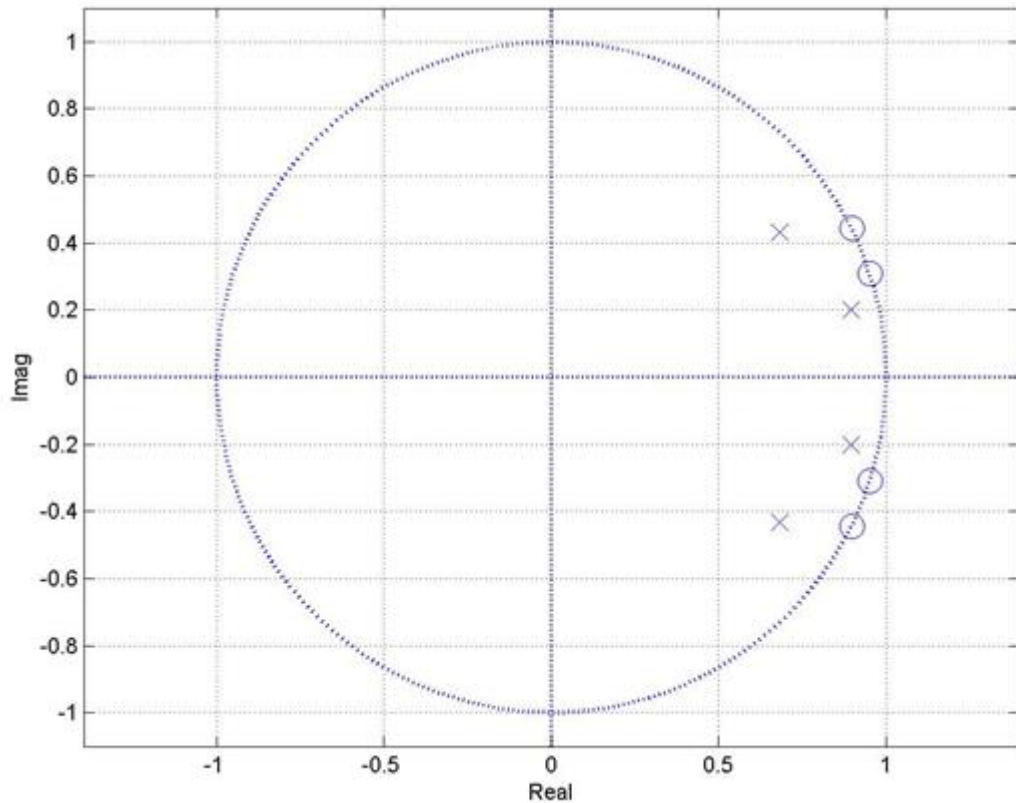


Рис.2.5 -О – нулі; X – полюси;

Отже, можемо побачити, що полюса лежать всередині кола одиничного радіуса на z - площині, значить ЦФ стійкий.

2.2. Межі стійкості фільтра

Розіб'ємо передавальну функцію ЦФ на дві біквдратні ланки

$$H(z) = \prod_{i=1}^2 \frac{b_{0i} + b_{1i}z^{-1} + b_{2i}z^{-2}}{1 + a_{1i}z^{-1} + a_{2i}z^{-2}}$$

Система буде стійкою.

Отже розглянемо характеристичне рівняння ланки:

$$z^2 + a_1z + a_2 = 0,$$

корені рівняння

$$z_{1,2} = \frac{-a_1}{2} \pm \frac{1}{2} \sqrt{a_1^2 - 4a_2}.$$

Зм.	Арк.	№ докум.	Підпис	Дата

При $a_1^2 - 4a_2 > 0$ маємо два різних дійсних корені;

$a_1^2 - 4a_2 = 0$ – два однакових дійсних кореня;

$a_1^2 - 4a_2 < 0$ – два комплексно - спряжених кореня.

Значить парабола $a_1^2 - 4a_2 = 0$ розділяє області дійсних і комплексних коренів.

Умови стійкості будуть наступні:

1) $1 - a_1 + a_2 > 0$

2) $1 + a_1 + a_2 > 0$

3) $1 - a_2 > 0$

На площині коефіцієнтів a_1, a_2 межі стійкості будуть виглядати так:

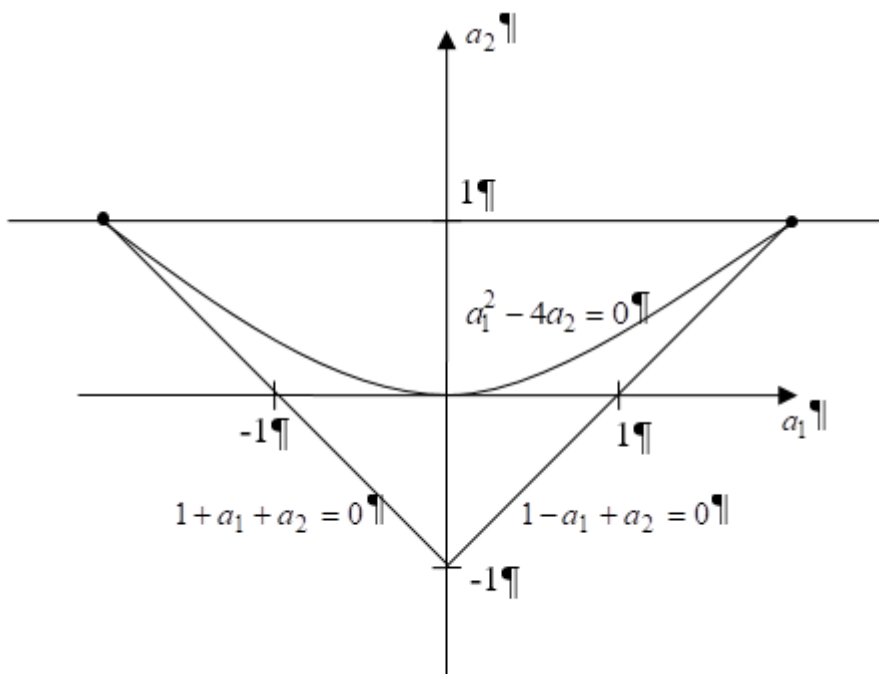


Рис.2.6 – Межі стійкості

Для отримання коефіцієнтів чисельників і знаменників біквдратних ланок звернемося до стандартної функції MatLab - *tf2sos*

$$[sos,g]=tf2sos(bz,az)$$

де *sos* – матриця, що містить речові коефіцієнти, виду:

$$sos = \begin{bmatrix} b_{01} & b_{11} & b_{21} & a_{01} & a_{11} & a_{21} \\ b_{02} & b_{12} & b_{22} & a_{02} & a_{12} & a_{22} \end{bmatrix};$$

g – параметр, що дозволяє врахувати в БЗ коефіцієнт передачі

$$H(z) = gH_1(z)H_2(z)$$

$$sos = \begin{bmatrix} 1 & -1,7931 & 1 & 1 & -1,3647 & 0,65341 \\ 1 & -1,902 & 1 & 1 & -1,787 & 0,8389 \end{bmatrix};$$

$$g = 0,73938.$$

Тоді отримаємо:

$$H_1(z) = \frac{1 - 1,7931z^{-1} + z^{-2}}{1 - 1,3647z^{-1} + 0,65341z^{-2}};$$

$$H_2(z) = \frac{1 - 1,902z^{-1} + z^{-2}}{1 - 1,787z^{-1} + 0,8389z^{-2}};$$

$$H(z) = 0,73938 \cdot \frac{1 - 1,7931z^{-1} + z^{-2}}{1 - 1,3647z^{-1} + 0,65341z^{-2}} \cdot \frac{1 - 1,902z^{-1} + z^{-2}}{1 - 1,787z^{-1} + 0,8389z^{-2}}.$$

Видно, що a_1, a_2 лежать всередині трикутника, отже фільтр є стійким.

2.3. Вибір структурної схеми реалізації фільтру

Цифрові фільтри можуть бути реалізовані як каскадні або паралельні сполучення різних стандартних елементів. Хоча теоретично ці фільтри є еквівалентними, їхня робота на мікропроцесорах з арифметикою як з фіксованою, так і з плаваючою комою значно відрізняється. На практиці рекурсивні фільтри часто втілюють за допомогою біквдратної ланки з передавальною функцією:

$$W(z) = \frac{b_0 + b_1z^{-1} + b_2z^{-2}}{1 + a_1z^{-1} + a_2z^{-2}}.$$

Різницеве рівняння фільтру записується у вигляді

$$y_n = b_0r_n + b_1r_{n-1} + b_2r_{n-2} - a_1y_{n-1} - a_2y_{n-2}.$$

Для реалізації на МП можуть бути використані структурні схеми біквдратної ланки прямої або канонічної форми.

					123.KI-41.19	Арк.
						23
Зм.	Арк.	№ докум.	Підпис	Дата		

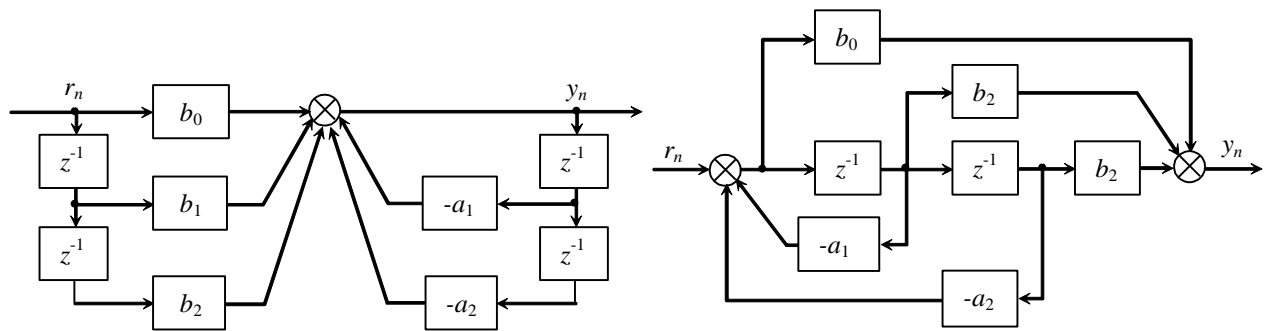


Рис. 2.7 - Пряма і канонічна схема бікватратної ланки

Зазначимо, що хоча структурні схеми мають однакову передавальну функцію, вони мають важливі відмінності. Наприклад, у структурі, реалізованій у прямій формі, компоненти прямого зв'язку з коефіцієнтами b_0 , b_1 , b_2 , що визначають нулі передавальної функції, розташовані перед доданками зворотного зв'язку з коефіцієнтами a_1 та a_2 , що відносяться до полюсів. У канонічній структурі все навпаки. Через це, на практиці полюси канонічної ланки схильні посилювати шум, що виникає під час обчислень. Реалізація у прямій формі включає один суматор і одну точку квантування для суми творів, тоді як канонічна ланка має два суматори та дві точки квантування. Вихід першого суматора канонічної ланки знаходиться у внутрішньому вузлі, що призводить до самопідтримуваного переповнення. Реалізація у прямій формі не містить внутрішнього вузла, і переповнення або самокоригується, або легко виправляється зовні. Крім того, вхід такої ланки масштабується, на відміну від канонічної ланки, де вхід не обмежується.

Розглянемо вплив помилок квантування на ланки. У прямій формі різницеve рівняння фільтру має такий вигляд.:

$$y_n = b_0 r_n + b_1 r_{n-1} + b_2 r_{n-2} - a_1 y_{n-1} - a_2 y_{n-2}.$$

Модель з урахуванням шуму квантування має вигляд:

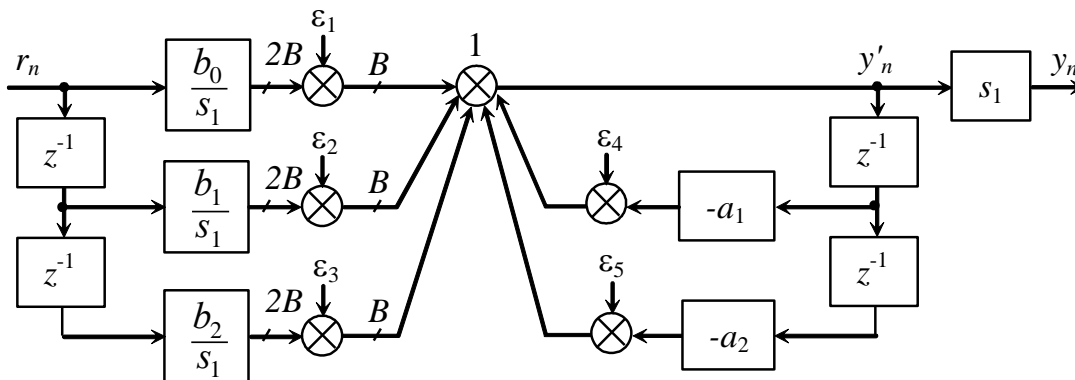


Рис.2.8 – Модуль із шумом

Вхідні та вихідні відліки представлені як B -бітові числа (включаючи біт знака), тобто вхідний сигнал квантований за допомогою АЦП. Після множення результату квантуються до B біт за допомогою округлення або усічення. Вихідні сигнали п'яти суматорів, де відбувається квантування добутоків, надходять на входи суматора 1. Тому всі джерела шуму квантування можна об'єднати $\varepsilon = \varepsilon_1 + \varepsilon_2 + \varepsilon_3 + \varepsilon_4 + \varepsilon_5$ і подати на вхід суматора 1.

Потужність шуму округлення добутоків визначається так:

$$D_{\varepsilon_{yn}} = \frac{5\delta_{\text{окр}}^2}{12} \left[\sum_{n=0}^{\infty} f^2(nT) \right] s_1^2 = \frac{5\delta_{\text{окр}}^2}{12} \left[\frac{1}{2\pi j} \oint_{\Gamma} F(z)F(z^{-1}) \frac{dz}{z} \right] s_1^2 =$$

$$= \frac{5\delta_{\text{окр}}^2}{12} \|F(z)\|_2^2 s_1^2,$$

де $F(z) = 1/(1 + a_1z^{-1} + a_2z^{-2})$ – передавальна функція фільтра, що зв'язує вхід ε з виходом фільтра, $f(nT) = Z^{-1}\{F(z)\}$ – імпульсна характеристика фільтра між входом ε і виходом фільтра, $\|F(z)\|_2^2$ – квадрат норми L_2 . Загальна потужність шуму на виході фільтра:

$$D_{y_n} = D_{\text{внх}} + D_{\varepsilon_{yn}} = \frac{\delta^2}{12} \left[\sum_{n=0}^{\infty} h^2(nT) + 5s_1^2 \sum_{n=0}^{\infty} f^2(nT) \right] =$$

$$= \frac{\delta^2}{12} \left[\|W(z)\|_2^2 + 5s_1^2 \|F(z)\|_2^2 \right],$$

$$\text{де } W(z) = \frac{b_0 + b_1 z^{-1} + b_2 z^{-2}}{1 + a_1 z^{-1} + a_2 z^{-2}}, F(z) = \frac{1}{1 + a_1 z^{-1} + a_2 z^{-2}}, D_{\text{ВЫХ}} - \text{потужність}$$

шуму квантування вхідного сигналу АЦП.

Вплив шумів квантування при канонічній формі реалізації фільтра. Модель з урахуванням шуму квантування має вигляд:

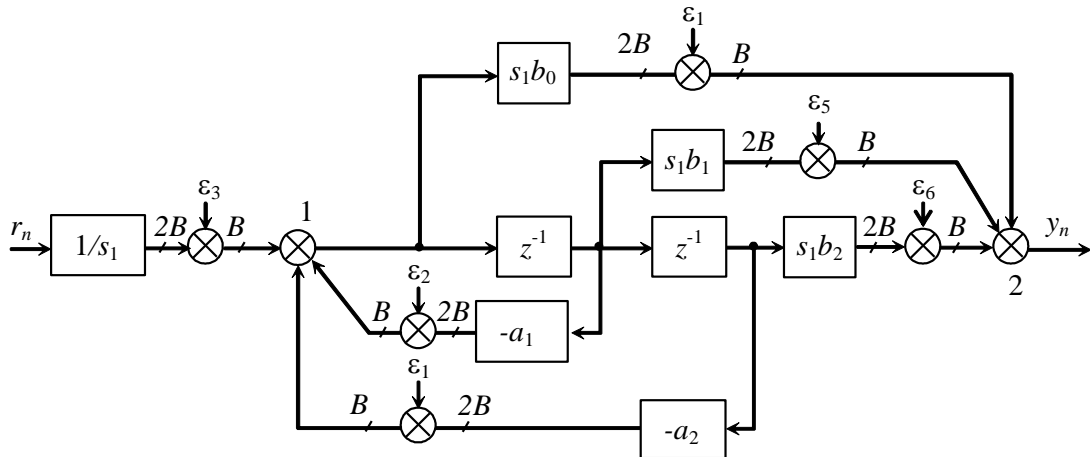


Рис.2.9 –Модель канонічної форми із шумом квантування

У схемі вхідні і вихідні відліки представлені як B -бітові слова. Виходи сумматорів шумів квантування добутоків $\varepsilon_1, \varepsilon_2, \varepsilon_3$ подаються на перший суматор (1), а виходи сумматорів шумів квантування добутоків $\varepsilon_4, \varepsilon_5, \varepsilon_6$ - на вхід сумматора 2. Джерела шуму квантування можна об'єднати $\varepsilon'_1 = \varepsilon_1 + \varepsilon_2 + \varepsilon_3$ і $\varepsilon'_2 = \varepsilon_4 + \varepsilon_5 + \varepsilon_6$.

Припускається, що джерела шуму від квантування добутоків не корелюють між собою. У такому випадку потужність шуму квантування добутоків визначається як сума окремих шумових компонентів.

$$D_{\varepsilon_{yn}} = \frac{3\delta_{\text{окр}}^2}{12} \sum_{n=0}^{\infty} h_1^2(nT) + \frac{3\delta_{\text{окр}}^2}{12} = \frac{3\delta_{\text{окр}}^2}{12} \left[\|W'(z)\|_2^2 + 1 \right],$$

де $h_1(nT)$ імпульсна характеристика фільтра, що зв'язує вхід джерела шуму ε'_1 з виходом фільтра.

$$W'(z) = s_1 \frac{b_0 + b_1 z^{-1} + b_2 z^{-2}}{1 + a_1 z^{-1} + a_2 z^{-2}} = s_1 W(z).$$

Тоді загальний шум квантування на виході фільтру має вигляд:

$$D_{y_n} = D_{\text{ВЫХ}} + D_{\varepsilon_{y_n}} = \frac{\delta_{\text{окр}}^2}{12} \left\{ 3 \left[1 + s_1^2 \sum_{n=0}^{\infty} h^2(nT) \right] + \sum_{n=0}^{\infty} h^2(nT) \right\} =$$

$$= \frac{\delta_{\text{окр}}^2}{12} \left\{ 3 \left[1 + s_1^2 \|W(z)\|_2^2 \right] + \|W(z)\|_2^2 \right\}.$$

Порівнюючи отримані формули для канонічної та прямої реалізацій, можна встановити, що загальна потужність шуму в канонічній реалізації вища, ніж у прямій. Використання масштабного множника в обох схемах призводить до збільшення вихідного шуму. Тому ми будемо дотримуватися прямої форми реалізації.

Визначимо розташування ланок. При каскадній реалізації виникають певні труднощі. Як утворювати пари "множник чисельника - множник знаменника"? Наприклад, фільтр четвертого порядку можна розкласти на дві ланки другого порядку і потім впорядкувати їх одним із чотирьох різних способів.:

$$1. W(z) = \frac{Y_1(z)}{R_1(z)} \frac{Y_2(z)}{R_2(z)}; \quad 2. W(z) = \frac{Y_2(z)}{R_2(z)} \frac{Y_1(z)}{R_1(z)};$$

$$3. W(z) = \frac{Y_1(z)}{R_2(z)} \frac{Y_2(z)}{R_1(z)}; \quad 4. W(z) = \frac{Y_2(z)}{R_1(z)} \frac{Y_1(z)}{R_2(z)}.$$

Очевидно, що число можливих розбиттів на пари для фільтру N-ого порядку одне $((N/2)!)^2$.

Щоб уникнути появи великої амплітудної характеристики на частоті, що відповідає полюсу, застосовується наступне емпіричне правило: об'єднувати в пару $Y_i(z)$ з $R_k(z)$, і розташовувати наприкінці каскаду ланки другого порядку з полюсами, найбільш близькими до одиничного кола.

Як розглядалося вище при каскадній структурі передавальна функція $H(z)$ має

вигляд:

$$H_1(z) = \frac{1 - 1,7931z^{-1} + z^{-2}}{1 - 1,3647z^{-1} + 0,65341z^{-2}};$$

$$H_2(z) = \frac{1 - 1,902z^{-1} + z^{-2}}{1 - 1,787z^{-1} + 0,8389z^{-2}};$$

$$g = 0,73938.$$

$$H(z) = 0,73938 \cdot \frac{1 - 1,7931z^{-1} + z^{-2}}{1 - 1,3647z^{-1} + 0,65341z^{-2}} \cdot \frac{1 - 1,902z^{-1} + z^{-2}}{1 - 1,787z^{-1} + 0,8389z^{-2}}$$

Позбудемося явного коефіцієнта g , для цього проведемо наступні дії:

$$H'_1(z) = H_1(z) \cdot \sqrt{g} = \frac{0,859872 - 1,541836z^{-1} + 0,859872z^{-2}}{1 - 1,3647z^{-1} + 0,65341z^{-2}}$$

$$H'_2(z) = H_2(z) \cdot \sqrt{g} = \frac{0,859872 - 1,635476z^{-1} + 0,859872z^{-2}}{1 - 1,787z^{-1} + 0,8389z^{-2}}$$

$$H(z) = H'_1(z) \cdot H'_2(z)$$

Шум квантування:

$$D_{\varepsilon_{yn}} = \frac{3\delta_{\text{окр}}^2}{12} \sum_{n=0}^{\infty} f_1^2(nT) + \frac{3\delta_{\text{окр}}^2}{12} \sum_{n=0}^{\infty} f_2^2(nT) + \frac{2\delta_{\text{окр}}^2}{12} \sum_{n=0}^{\infty} f_3^2(nT) + \frac{3\delta_{\text{окр}}^2}{12} =$$

$$= \frac{\delta_{\text{окр}}^2}{12} \left[3 \sum_{n=0}^{\infty} f_1^2(nT) + 5 \sum_{n=0}^{\infty} f_3^2(nT) + 3 \right] = \frac{\delta_{\text{окр}}^2}{12} \left[3 \|F_1(z)\|_2^2 + 5 \|F_3(z)\|_2^2 + 3 \right],$$

де $f_i(nT)$ – імпульсна характеристика, що зв'язує джерело шуму ε_i і вихід шумові компоненти ε_2 і ε_3 , проходять через одні ланки фільтра, тобто через передавальні функції ланок два і три, тому їх об'єднаємо.

					123.KI-41.19	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		28

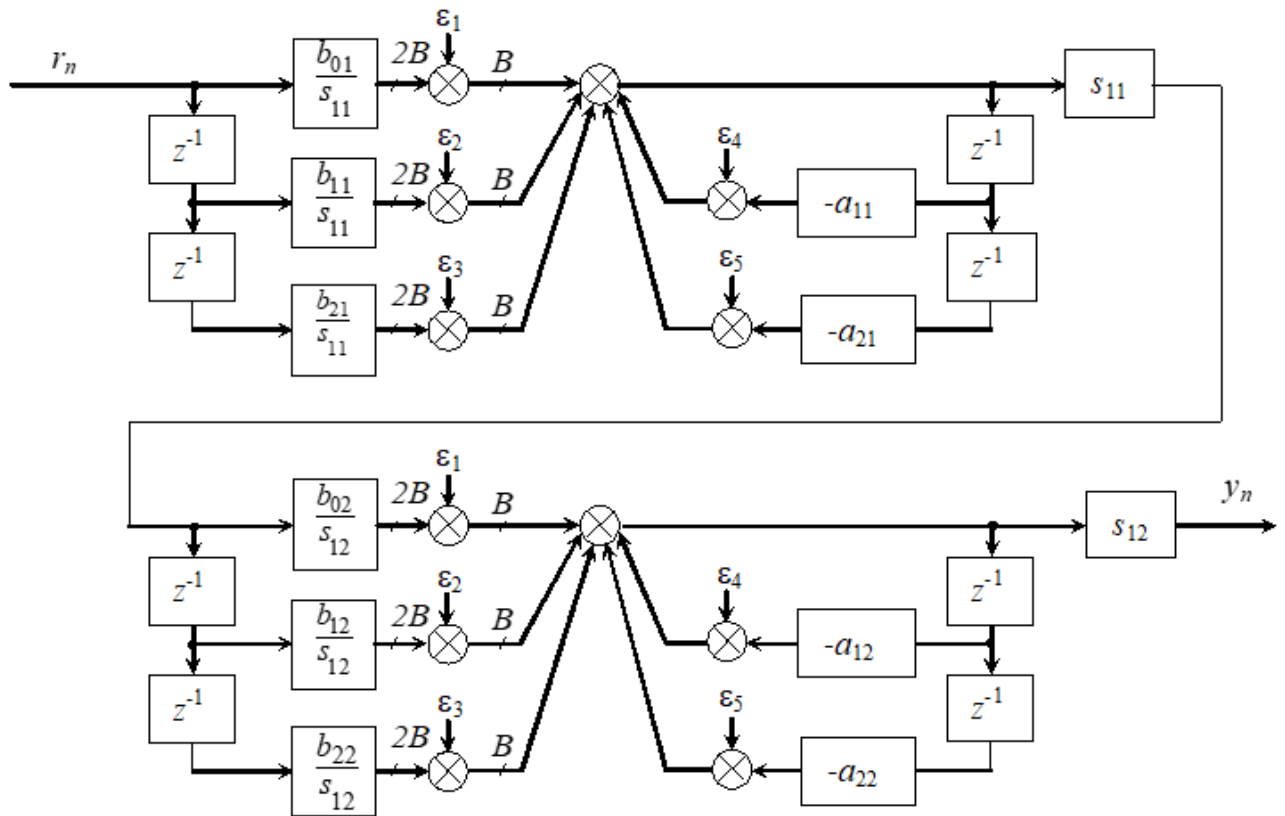


Рис.2.10 – Об'єднання ланок

При паралельній реалізації передатна функція цифрового фільтра розкладається на суму найпростіших дробово-раціональних доданків, де коефіцієнт біквдратного доданка при z^{-2} дорівнює нулю. У паралельній структурі порядок з'єднання цих елементарних ланок не має значення. Більш того, масштабування сигналу можна виконувати легше і незалежно для кожної ланки, а відношення сигнал-шум порівнювати з найкращими значеннями для каскадної реалізації. Однак слід зазначити, що нулі передатної функції паралельних структур більш чутливі до похибок квантування коефіцієнтів, особливо коли довжина коефіцієнтів зменшується до 5 біт або менше. Для коефіцієнтів більшої розрядності, наприклад 12 біт і вище, різниця між паралельною та каскадною реалізаціями для більшості фільтрів стає менш суттєвою. З цих причин на сьогодні каскадний метод реалізації цифрових фільтрів є більш поширеним і популярним..

Зм.	Арк.	№ докум.	Підпис	Дата

$$p = \begin{bmatrix} 0,89349 + 0,20143i \\ 0,89349 - 0,20143i \\ 0,68234 + 0,43338i \\ 0,68234 - 0,43338i \end{bmatrix}; \quad r = \begin{bmatrix} -0,079726 - 0,048398i \\ -0,079726 + 0,048398i \\ -0,22502 - 0,032588i \\ -0,22502 + 0,032588i \end{bmatrix}; \quad k = 1,3489.$$

Функція residuez виконує розкладання у вигляді

$$H(z) = \frac{r_1}{1 - p_1 z^{-1}} + \frac{r_2}{1 - p_2 z^{-1}} + \frac{r_3}{1 - p_3 z^{-1}} + \frac{r_4}{1 - p_4 z^{-1}} + k$$

Для отримання біквadratних ланок об'єднаємо дроби попарно, отримаємо:

$$H(z) = \frac{b_{01} + b_{11}z^{-1}}{1 + a_{11}z^{-1} + a_{21}z^{-2}} + \frac{b_{02} + b_{12}z^{-1}}{1 + a_{12}z^{-1} + a_{22}z^{-2}} + k;$$

де

$$\begin{aligned} b_{01} &= r_1 + r_2; & b_{02} &= r_3 + r_4; \\ b_{11} &= -r_1 p_2 - r_2 p_1; & b_{12} &= -r_3 p_4 - r_4 p_3; \\ a_{11} &= -p_2 - p_1; & a_{12} &= -p_4 - p_3; \\ a_{21} &= p_1 p_2; & a_{22} &= p_3 p_4; \end{aligned}$$

Підставивши числові значення отримуємо:

$$H_1''(z) = \frac{-0,15945 + 0,16197z^{-1}}{1 - 1,787z^{-1} + 0,8389z^{-2}};$$

$$H_2''(z) = \frac{-0,45003 + 0,33532z^{-1}}{1 - 1,3647z^{-1} + 0,65341z^{-2}};$$

$$k = 1,3489;$$

$$H(z) = H_1''(z) + H_2''(z) + k.$$

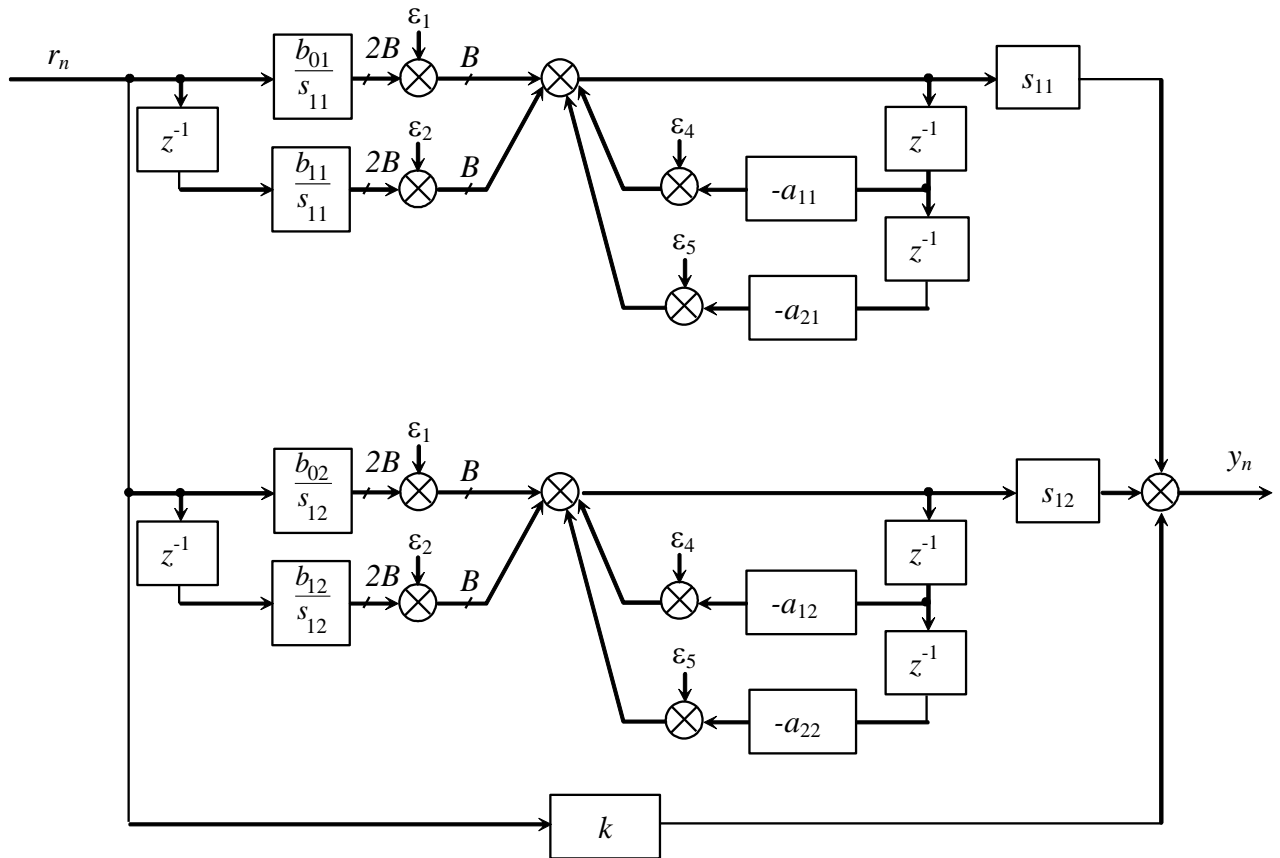


Рис.2.11 – Схема з'єднань із шумом квантування

$$D_{\varepsilon_{yn}} = \frac{\delta_{\text{окр}}^2}{12} \left\{ 5 + 2 \sum_{i=1}^2 \left[s_i^2 \sum_{n=0}^{\infty} h_i^2(nT) \right] \right\} = \frac{\delta_{\text{окр}}^2}{12} \left\{ 5 + 2 \sum_{i=1}^2 \left[s_i^2 \|W_i(z)\|_2^2 \right] \right\},$$

де $h_i(nT)$ – імпульсна характеристика i -ого джерела шуму.

Висновок: ЦФ реалізуємо каскадною структурою, а білінійну ланку в прямій схемі.

2.4. Аналіз помилок квантування

Кінцева розрядність представлення чисел при реалізації цифрових фільтрів призводить до зниження їхніх експлуатаційних характеристик. Особливо критичним цей вплив стає у випадках, коли полюси передатної функції розташовані поблизу межі стійкості, оскільки внаслідок похибок квантування коефіцієнтів полюси можуть переміститися за цю межу, спричинивши нестійкість фільтра.

Зм.	Арк.	№ докум.	Підпис	Дата

Для оцінки впливу квантування коефіцієнтів на розташування полюсів і стійкість фільтра розглянемо представлення чисел у форматі з фіксованою комою, де виділено 1 біт для знака, 1 біт для цілої частини та 14 біт для дробової частини. Таке представлення дозволяє відобразити числа в діапазоні від -2 до 2^{-14} з кроком 2^{-14} .

Проаналізувавши залежність розташування квантованих полюсів від їх початкового положення та порівнявши їх з межею стійкості (одичним колом на комплексній z-площині), можна оцінити ймовірність втрати стійкості при заданій розрядності коефіцієнтів. Це дозволить визначити необхідну розрядність для забезпечення заданого рівня ймовірності збереження стійкості фільтра після квантування коефіцієнтів.

Таблиця 2.1 – Аналіз квантування

Коефіцієнти до квантування	Коефіцієнти в двійковому вигляді, в доп. коді	в шістнадцятковому вигляді, в доп. коді	Коефіцієнти після квантування
$b_{01}=b_{02}= b_{21}=$ $b_{22}=$ $=0,859872$	0011 0111 0000 1000	3708	0,85986328125
$b_{11}= -1,541836$	1001 1101 0101 0011	9D53	-1,54180908203125
$b_{12}= -1,635476$	1001 0111 0101 0101	9755	-1,63543701171875
$a_{11}= -1,3647$	1010 1000 1010 1001	A8A9	-1,36468505859375
$a_{21}= 0,65341$	0010 1001 1101 0001	29D1	0,65338134765625
$a_{12}= -1,787$	1000 1101 1010 0010	8DA2	-1,7869873046875
$a_{22}= 0,8389$	0011 0101 1011 0000	35B0	0,8388671875

Діаграма полюсів і нулів квантованого фільтра:

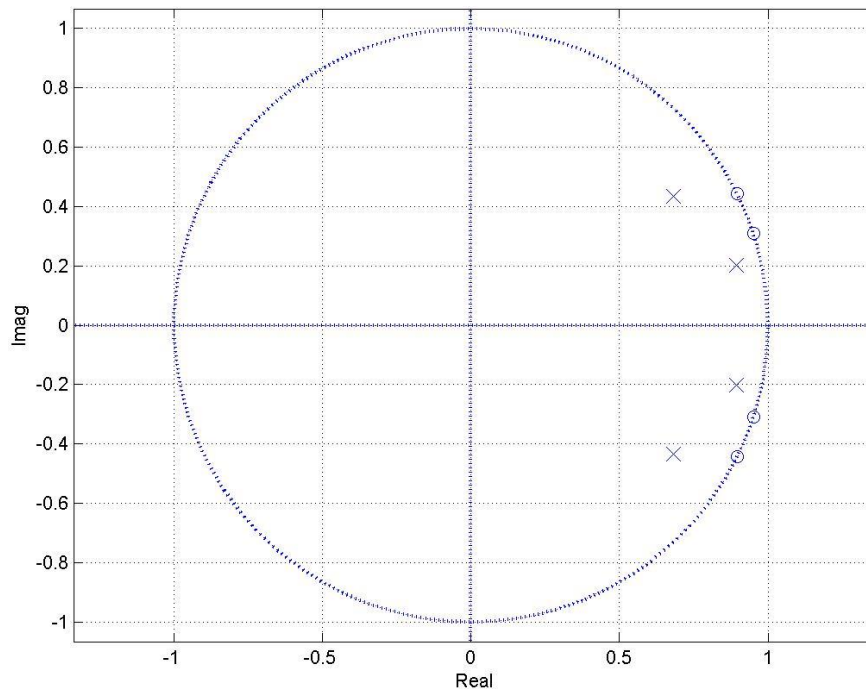


Рис.2.12 - Діаграма полюсів і нулів квантованого фільтра

Діаграма демонструє те, що полюса в одиничному колі, отже фільтр стійкий.

					<i>123.KI-41.19</i>	<i>Арк.</i>
<i>Зм.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>		33

2.5. Вибір мікроконтролера

Для апаратної реалізації цифрового фільтра буде обрано мікроконтролер із сімейства Mega компанії Atmel.

Як і всі мікроконтролери AVR від Atmel, мікроконтролери серії Mega є 8-бітними пристроями, призначеними для використання у вбудованих системах. Вони виготовляються за допомогою низькоспоживаючої КМОН-технології, яка в поєднанні з вдосконаленою RISC-архітектурою забезпечує оптимальне співвідношення між вартістю, продуктивністю та енергоспоживанням. Мікроконтролери цього сімейства є найбільш розвиненими представниками загального призначення в лінійці AVR від Atmel.

Основними перевагами мікроконтролерів серії Mega є:

- Висока продуктивність завдяки RISC-архітектурі
- Низьке енергоспоживання
- Вбудовані периферійні модулі (таймери, UART, SPI, I2C та ін.)
- Наявність вбудованої пам'яті програм та даних
- Розвинуте програмне забезпечення та підтримка компілятора Сі.

Відмінні здібності

Мікроконтролери AVR сімейства Mega від Atmel мають низку особливостей, які роблять їх привабливими для застосування у вбудованих системах:

- Вбудована Flash-пам'ять програм об'ємом від 8 до 256 Кбайт з високою стійкістю до циклів перезапису (не менше 10 000 циклів).
- Оперативна пам'ять (статичне ОЗП) об'ємом від 512 байт до 8 Кбайт.
- Енергонезалежна пам'ять даних на основі EEPROM об'ємом від 256 байт до 4 Кбайт з високою стійкістю до циклів перезапису (не менше 100 000 циклів).
- Можливість захисту пам'яті програм та даних від несанкціонованого читання та модифікації.
- Підтримка програмування безпосередньо в системі через послідовні інтерфейси SPI та JTAG, а також можливість самопрограмування.

					<i>123.KI-41.19</i>	<i>Арк.</i>
<i>Зм.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>		34

- Можливість внутрішньосхемного налагодження програм.
- Різноманітні способи синхронізації: вбудований RC-генератор з внутрішньою або зовнішньою RC-ланкою, вбудований генератор для зовнішнього кварцового або керамічного резонатора, зовнішній сигнал синхронізації.
- Наявність декількох режимів зниженого енергоспоживання.
- Вбудований детектор зниженої напруги живлення (Brown-Out Detector).
- Можливість програмного зниження частоти тактового генератора для економії енергії.

Ці особливості забезпечують гнучкість, енергоефективність та зручність розробки при використанні мікроконтролерів Atmel AVR Mega у вбудованих додатках..

Характеристики процесора

Основними характеристиками процесора мікроконтролерів AVR сімейства Mega від Atmel є:

- Повністю статична архітектура RISC, що дозволяє працювати на нульовій тактовій частоті.
- Арифметико-логічний пристрій (АЛП) безпосередньо підключений до банку регістрів загального призначення в кількості 32 штук.
- Більшість команд виконується за один машинний цикл (такт процесора).
- Векторна система переривань з підтримкою черги переривань.
- Велика кількість джерел переривань (до 45 внутрішніх та до 32 зовнішніх).
- Наявність апаратного модуля для операцій множення.
- Компактний набір високопродуктивних RISC-команд.
- Режим зниженого енергоспоживання.
- Програмована захисна система для коду та даних.

Ці особливості забезпечують високу продуктивність, енергоефективність, гнучкість та зручність програмування мікроконтролерів AVR Mega. RISC-

					<i>123.KI-41.19</i>	<i>Арк.</i>
<i>Зм.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>		35

архітектура із зменшеною кількістю циклів на команду та апаратна підтримка операцій множення дозволяють ефективно виконувати цифрову обробку сигналів та інші обчислювально-інтенсивні завдання. А статична реалізація процесора гарантує надійну роботу навіть на нульовій частоті..

Характеристики підсистеми вводу / виводу

Підсистема вводу/виводу мікроконтролерів AVR сімейства Mega має такі особливості:

- Програмна конфігурація та вибір режимів роботи портів вводу/виводу
- Можливість незалежного програмування кожного виводу порту як входу або виходу
- Вхідні буфери із тригерами Шмітта на всіх входах
- Можливість повного відключення цифрових портів від виводів мікросхеми
- Наявність програмовно керованих вбудованих підтягуючих резисторів (20-50 кОм) для входів

Архітектура

Мікроконтролери AVR Mega є 8-бітними пристроями з RISC-архітектурою. Вони містять електрично-перепрограмовувану пам'ять програм (Flash), енергонезалежну пам'ять даних (EEPROM) та різноманітні периферійні модулі. Набір периферії в Mega значно ширший, ніж у сімействі Tiny. Для більшості моделей Mega типовим є наявність декількох 8- та 16-бітних таймерів/лічильників, модулів USART, SPI, TWI (I2C), аналогового компаратора, сторожового таймера та портів вводу/виводу. Деякі моделі також мають АЦП, інтерфейси JTAG та DebugWire.

На структурній схемі показано мікроконтролер ATmega8x, який має:

- 3 порти вводу/виводу (B, D - 8-бітні, C - 7-бітний)
- Вхід скидання та виводи для підключення резонатора
- Два 8-бітні (T0, T2) та один 16-бітний (T1) таймер/лічильник
- 3 канали ШІМ
- Модулі USART, SPI, TWI

						123.KI-41.19	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата			36

- 6- або 8-канальний (залежно від корпусу) 10-бітний АЦП

Така архітектура з RISC-ядром та широким набором периферії забезпечує високу продуктивність, гнучкість та енергоефективність мікроконтролерів AVR Mega від Atmel..

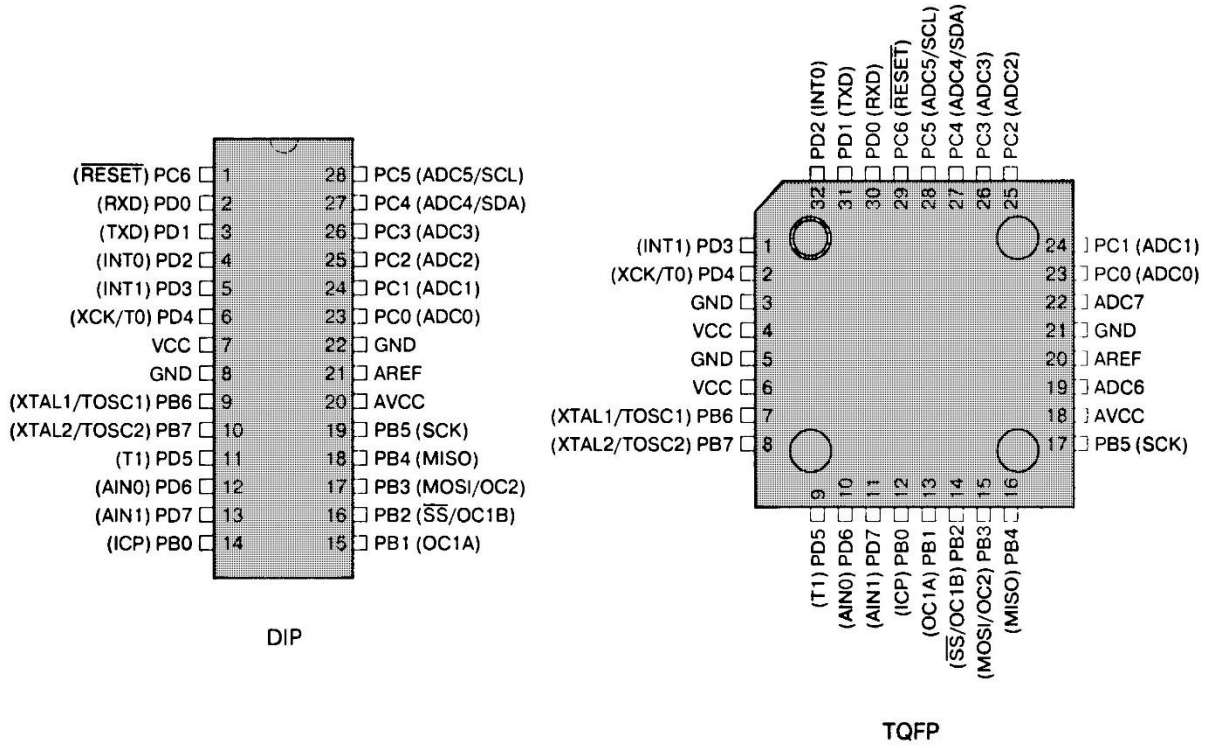


Рис.2.13 - Розташування виводів (вид зверху) моделей ATmega8x

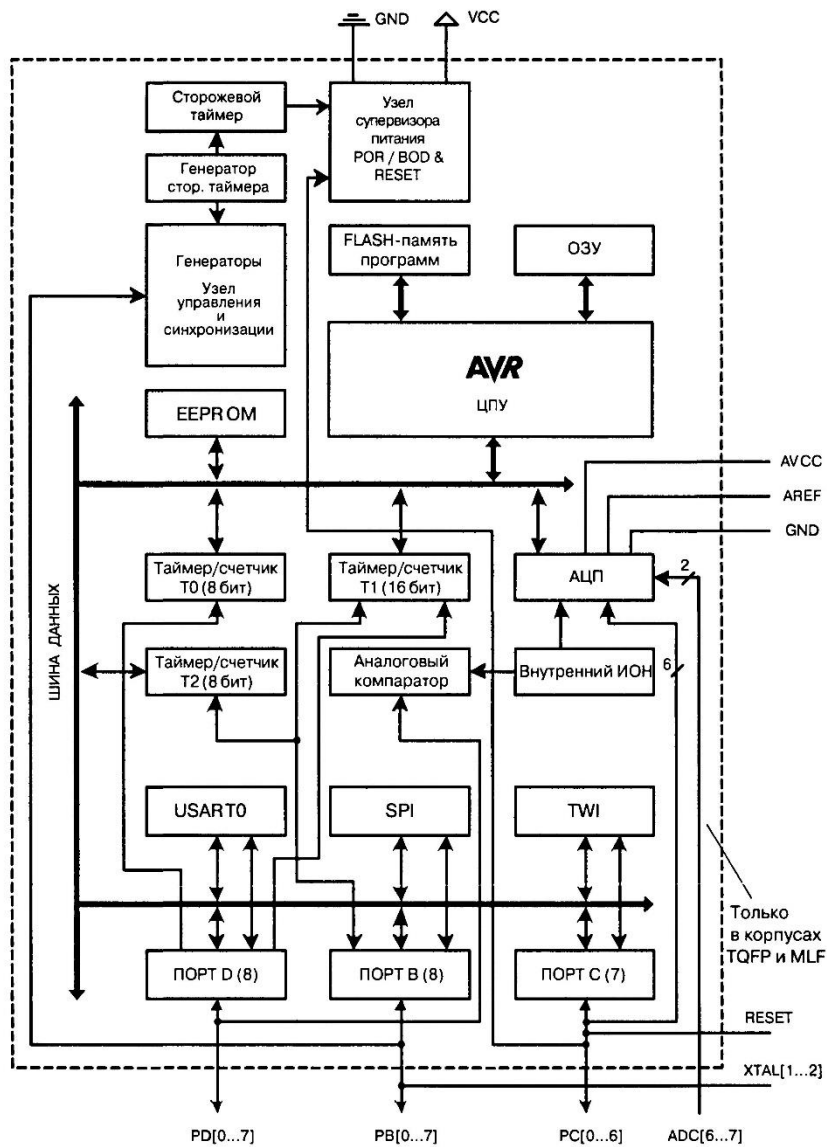


Рис.2.14 - Структурна схема мікроконтролерів ATmega8x

Організація пам'яті

У мікроконтролерах AVR сімейства Mega реалізована Гарвардська архітектура, в якій не лише розділені адресні простори пам'яті програм і пам'яті даних, а й використовуються окремі шини доступу до них. Способи адресації та доступу до цих областей пам'яті також різняться. Така архітектура дозволяє центральному процесору працювати одночасно як з пам'яттю програм, так і з пам'яттю даних, що істотно підвищує продуктивність системи. Кожна з областей пам'яті даних (оперативна пам'ять та EEPROM) також розташована у власному адресному просторі.

Зм.	Арк.	№ докум.	Підпис	Дата

На узагальненій карті пам'яті мікроконтролерів AVR сімейства Mega можна побачити розподіл адресних просторів:

- Пам'ять програм (Flash) розташована в адресному просторі \$000000-\$0xFFFF (64 Кбайт)
- Оперативна пам'ять даних (SRAM) займає адреси \$0x0000-\$0x10FF (4 Кбайт)
- Енергонезалежна пам'ять даних (EEPROM) mapped до адресного простору \$0x0000-\$0x03FF (1 Кбайт)
- Регістри вводу/виводу та спеціальних функцій mapped до адрес \$0x0000-\$0x003F

Такий розподіл адресних просторів забезпечує ефективний доступ як до пам'яті програм, так і до різних видів пам'яті даних, а також до регістрів периферійних пристроїв. Гарвардська архітектура в поєднанні з RISC-ядром процесора є запорукою високої продуктивності мікроконтролерів AVR Mega.

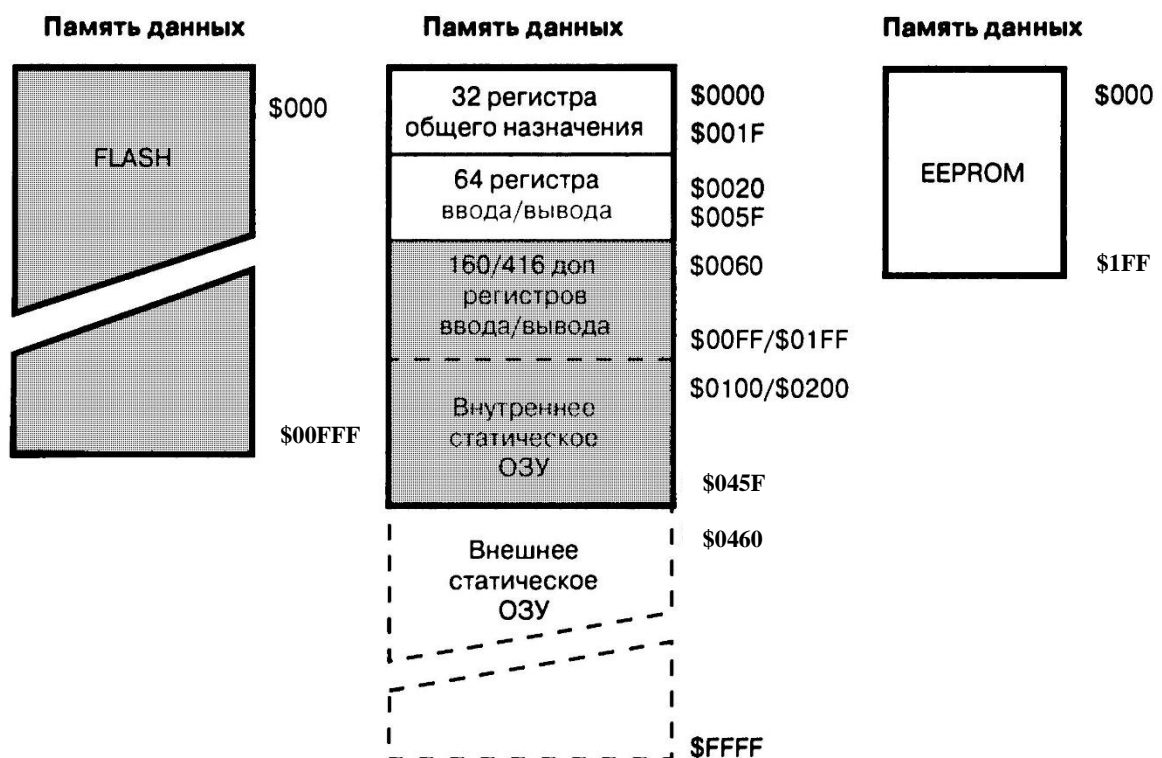


Рис.2.15 –Розподіл пам'яті в AVR Mega

Пам'ять програм

Пам'ять програм призначена для зберігання машинних кодів команд, які керують роботою мікроконтролера, а також таблиць констант, що не змінюються під час виконання програми. У мікроконтролерах AVR сімейства Mega пам'ять програм реалізована як електрично-перепрограмовувана флеш-пам'ять (FLASH). Через те, що всі команди мають фіксовану довжину у 16 біт, пам'ять програм має 16-бітну організацію. Обсяг пам'яті програм варіюється від 4К (4x1024) до 64К (64x1024) 16-бітних слів в різних моделях.

У більшості мікроконтролерів Mega пам'ять програм логічно розділена на дві нерівні частини: область прикладної програми та область завантажувача. В області завантажувача може розміщуватись спеціальна програма (завантажувач), яка дозволяє мікроконтролеру самостійно завантажувати та вивантажувати прикладні програми. Якщо можливість самопрограмування не використовується, прикладна програма може розташовуватись також і в області завантажувача.

Для адресації пам'яті програм використовується лічильник команд (Program Counter, PC). Розрядність лічильника команд варіюється від 11 до 17 біт в залежності від обсягу адресованої пам'яті програм в конкретній моделі мікроконтролера.

Програми та дані зберігаються в різних областях пам'яті відповідно до реалізованої в AVR Гарвардської архітектури, що забезпечує ефективний доступ як до коду, так і до даних під час виконання програм.

Пам'ять даних

Мікроконтролери сімейства Mega мають трирівневу структуру пам'яті: регістрова, оперативна (SRAM) та енергонезалежна (EEPROM).

Регістрова пам'ять включає 32 регістри загального призначення (РЗП) та службові регістри введення/виведення (РВВ). Деякі моделі мають додаткові регістри (ДРВВ) для керування розширеною периферією. РВВ займають 64 байти, а ДРВВ - 160 або 416 байтів залежно від моделі.

					<i>123.KI-41.19</i>	<i>Арк.</i>
<i>Зм.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>		40

Регістри РВВ та ДРВВ містять різні службові реєстри (керування, стану тощо) та реєстри для управління вбудованими периферійними пристроями. Кількість РВВ і ДРВВ варіюється в залежності від конкретної моделі.

Для зберігання змінних використовується статичний ОЗП (SRAM) місткістю від 512 байт до 8 Кбайт. Деякі моделі підтримують підключення зовнішнього SRAM до 64 Кбайт.

Для довготривалого зберігання даних, які можуть змінюватися (калібрувальні константи, серійні номери, ключі тощо), використовується вбудована EEPROM пам'ять місткістю від 256 байт до 4 Кбайт. Вона розташована в окремому адресному просторі з доступом через спеціальні РВВ.

Аналого-цифровий перетворювач

Більшість моделей цього сімейства оснащені 10-бітним АЦП послідовного наближення. Його основні характеристики: абсолютна похибка ± 2 LSB, інтегральна нелінійність $\pm 0,5$ LSB, швидкодія до 15 тис. вибірок/с.

На вході АЦП розташований 8-канальний (у деяких моделях 16-канальний) аналоговий мультиплексор для несиметричних входів. У певних корпусах кількість доступних каналів може бути меншою.

Більшість моделей дозволяють об'єднувати входи попарно для формування каналів з диференціальним входом. Деякі канали підтримують попереднє посилення сигналу в 10 або 200 разів, при цьому розрядність зменшується до 8 або 7 біт відповідно.

Як опорну напругу для АЦП можна використовувати напругу живлення, внутрішнє або зовнішнє джерело опорної напруги.

АЦП може працювати в двох режимах: одиночного перетворення (перетворення ініціюється користувачем) або безперервного перетворення (автоматичний запуск через певні інтервали часу).

Структурна схема модуля АЦП має аналоговий мультиплексор, підсилювач вхідного сигналу, схему вибірки-зберігання та власне АЦП. У деяких моделях відсутні певні елементи, показані сірим кольором..

					<i>123.KI-41.19</i>	<i>Арк.</i>
<i>Зм.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>		41

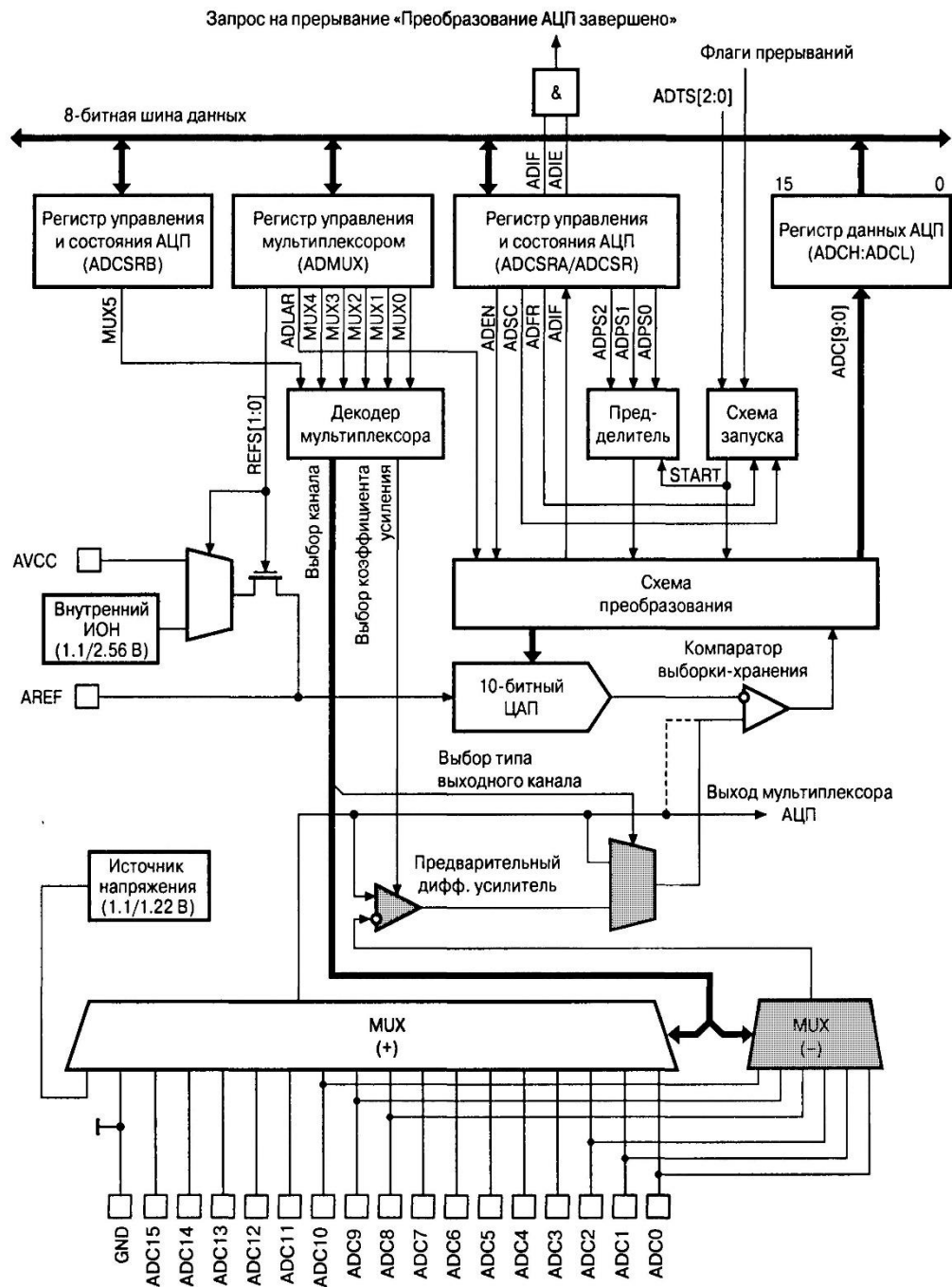


Рис.2.16 - Структурна схема модуля АЦП

Результат перетворення

Після завершення аналого-цифрового перетворення його результат записується у 16-бітний регістр даних АЦП. Оскільки АЦП 10-бітний, цей регістр логічно розділений на два 8-бітні регістри введення/виведення ADCH та ADCL, доступні тільки для читання.

Зм.	Арк.	№ докум.	Підпис	Дата

Адреси регістрів ADCH і ADCL залежать від конкретної моделі мікроконтролера. Для ATmega8535x, ATmega8x/16x/32x/64x/128x вони знаходяться за адресами 0x05:0x04 (0x25:0x24), а для інших моделей - 0x79:0x78. Після ввімкнення мікроконтролера вміст регістра даних АЦП дорівнює 0x0000. Для каналів з несиметричним входом числове значення результату перетворення визначається за допомогою виразу, наведеного нижче:

$$ADC = \frac{V_{IN} \cdot 1024}{V_{REF}}$$

де V_{IN} — значення вхідної напруги, а V_{REF} — величина опорної напруги.

Для каналів з диференціальним входом результат перетворення визначається виразом

$$ADC = \frac{(V_{POS} - V_{NEG}) \cdot K \cdot 512}{V_{REF}}$$

де V_{POS} — величина напруги на позитивному вході, V_{NEG} — величина напруги на негативному вході, а K — коефіцієнт посилення. Результат перетворення представляється в цьому випадку в додатковому коді, а його значення лежить в діапазоні від \$200 (-512) до \$ IFF (+512).

ATmega8 - це 8-бітний мікроконтролер з багатим набором вбудованої периферії та пам'яті. Основні його характеристики:

Пам'ять:

- 8 Кбайт енергонезалежної Flash-пам'яті для програм і даних
- 512 байт EEPROM пам'яті
- 1 Кбайт вбудованої SRAM
- Програмований захист пам'яті для безпеки користувацького коду

Периферійні модулі:

- Два 8-бітні таймери/лічильники з передподільниками
- Один 16-бітний таймер/лічильник з режимами захоплення та порівняння
- Лічильник реального часу з окремим генератором
- Три канали широтно-імпульсної модуляції (ШІМ)

					123.KI-41.19	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		43

- 6-канальний АЦП (у корпусі PDIP)
- 2-дротовий послідовний інтерфейс
- Програмований USART
- Інтерфейс SPI
- Програмований сторожовий таймер
- Вбудований аналоговий компаратор

Входи/виходи:

- 23 програмовані лінії введення/виведення

Робочі режими:

- Напруга живлення 4,5 - 5,5 В
- Тактова частота до 16 МГц

АТmega8 поєднує потужні обчислювальні можливості з низьким енергоспоживанням, роблячи його ідеальним для вбудованих додатків.

2.6. Цифро-аналоговий перетворювач

Цифро-аналоговий перетворювач (ЦАП) призначений для перетворення двійкового коду в пропорційну йому напругу та струм. Він використовується для узгодження комп'ютерних систем з аналоговими пристроями, а також як внутрішні вузли в аналогово-цифрових перетворювачах (АЦП). Схемотехніка ЦАП досить різноманітна, що дозволяє класифікувати їх за різними ознаками:

1. За способом перетворення вхідного коду: послідовні та паралельні.
2. За способом формування вихідного сигналу:
 - Послідовні: ШІМ, на конденсаторах, що перемикаються, сигма-дельта.
 - Паралельні: з підсумовуванням струмів, з підсумовуванням напруг, з підсумовуванням зарядів.
3. За типом вихідного сигналу: ЦАП зі струмовим виходом та виходом за напругою.
4. За типом цифрового інтерфейсу: з послідовним та паралельним виходом.

					<i>123.KI-41.19</i>	<i>Арк.</i>
						44
<i>Зм.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>		

Додаткова класифікація ЦАП може бути проведена за такими специфічними ознаками:

- За кількістю ЦАП на кристалі.
- За швидкодією.
- За розрядністю.
- За розрядністю вихідного сигналу.

Розглянемо ЦАП від фірми Texas Instruments, модель DAC900:

- Напруга живлення: +3 ... +5 В
- Швидкодія: 165 MSPS
- Час установки: 30 нс
- Кількість каналів: 1
- Споживана потужність: 170 мВт

DAC900 є високошвидкісним, високоякісним 10-бітним цифро-аналоговим перетворювачем зі струмовим виходом від 2 до 20 мА, виконаним за КМОН-технологією. Вихідний струм може бути налаштований за допомогою зовнішнього резистора.

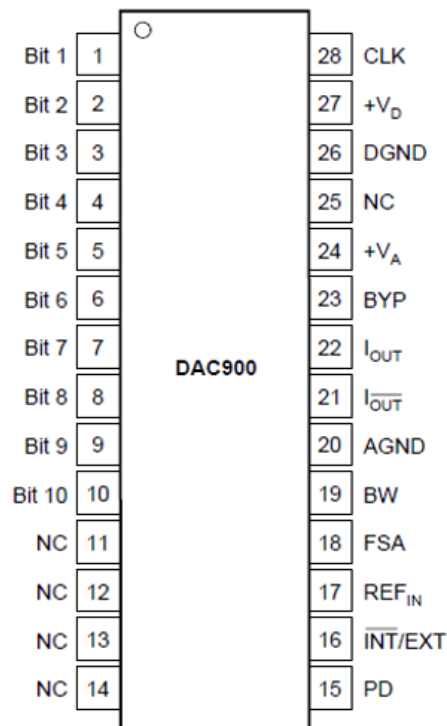


Рис.2.17 – Конфігурація виводів DAC900

Загальний вихідний струм ЦАП дорівнює сумі прямого і інверсного струму

$$I_{OUTFS} = I_{OUT} + I_{\overline{OUT}}$$

Вихідна напруга

$$V_{OUT} = I_{OUT} \cdot R_{LOAD}$$

$$V_{\overline{OUT}} = I_{\overline{OUT}} \cdot R_{LOAD}$$

Таблиця 2.2 – Результат вихідної напруги ЦАП

Вхідний код D9-D0	I_{OUT}	$I_{\overline{OUT}}$
11 1111 1111	20mA	0mA
10 0000 0000	10mA	10mA
00 0000 0000	0mA	20mA

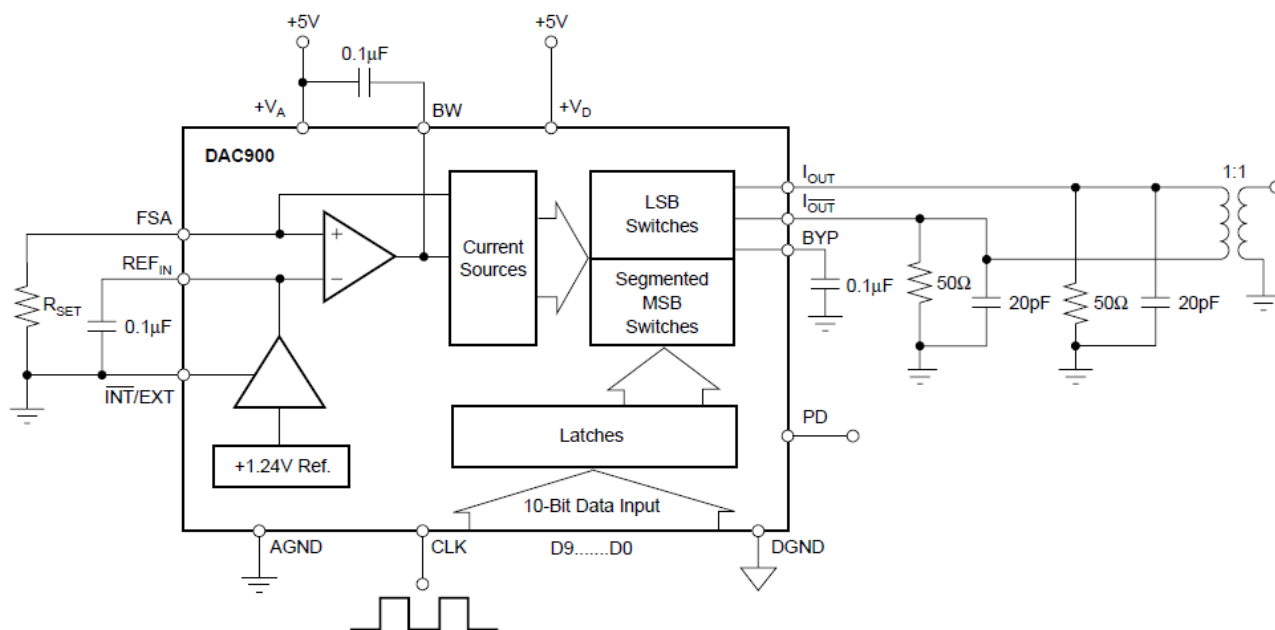


Рис.2.18 - Рекомендована схема включення DAC900

2.7. Програма реалізації фільтра

Каскадна структурна схема з прямим поданням біквдратних ланок описується наступними різницевиими рівняннями:

$$y_{1n} = b_{01}r_n + b_{11}r_{n-1} + b_{21}r_{n-2} - a_{11}y_{1n-1} - a_{21}y_{1n-2}$$

$$y_{2n} = b_{02}r_n + b_{12}r_{n-1} + b_{22}r_{n-2} - a_{12}y_{1n-1} - a_{22}y_{1n-2}$$

де r_n - задає вплив;

y_{1n} - вихідний результат першої біквдратної ланки (вхідний для другого);

y_{2n} - вихідний результат другої ланки (вихід фільтра).

Код для реалізації програми:

```
#include <mega8.h>

// Зберігаємо коефіцієнти в незалежній пам'яті EEPROM
eeprom float b01 = 0.859872, b11 = -1.541836, b21 = 0.859872,
            a11 = -1.3647, a21 = 0.65341,
            b02 = 0.859872, b12 = -1.635476, b22 = 0.85987,
            a12 = -1.787, a22 = 0.8389;

double rn, rn1, rn2, y1n, y1n1, y1n2, y2n, y2n1, y2n2;

// Функція для зчитування результату перетворення АЦП
unsigned int read_adc() {
    ADCSRA |= 0x40; // Запуск АЦП
    while ((ADCSRA & 0x10) == 0); // Чекати завершення перетворення АЦП
    ADCSRA |= 0x10; // Очищення прапора завершення
    return ADCW; // Повернути результат перетворення
}

// Функція для виводу результату
void out(double out_var) {
    int vix = (int)out_var;
    int a[10];
    for (int i = 0; i < 10; i++) {
        a[9 - i] = vix % 10; // Записати цифри у зворотному порядку
        vix = vix / 10;
    }
    PINB.2 = 1; // Початок виводу
    PIND.0 = a[0]; PIND.1 = a[1];
}
```

					123.KI-41.19	Арк.
						47
Зм.	Арк.	№ докум.	Підпис	Дата		

```

PIND.2 = a[2]; PIND.3 = a[3];
PIND.4 = a[4]; PIND.5 = a[5];
PIND.6 = a[6]; PIND.7 = a[7];
PINB.0 = a[8]; PINB.1 = a[9];
PINB.2 = 0; // Кінець виводу
}

void main(void) {
    // Ініціалізація портів
    PORTB = 0x00; // Програмування порту В на вивід
    DDRB = 0xFF;

    PORTC = 0x00; // Порт С на ввід
    DDRC = 0x00;

    PORTD = 0x00; // Порт D на вивід
    DDRD = 0xFF;

    MCUCR = 0x00; // Вимкнення переривань
    TIMSK = 0x00;

    ACSR = 0x80; // Вимкнення компаратора
    SFIOR = 0x00;

    ADMUX = 0x40; // Вхід АЦП - ADC0, опорна напруга - AVcc
    ADCSRA = 0x85; // Дозвіл роботи АЦП, частота перетворення 375 кГц

    // Початкове зчитування та обробка
    rn = read_adc();
    y1n = b01 * rn;
    y2n = b02 * y1n;
    out(y2n);

    y1n1 = y1n;
    y2n1 = y2n;
    rn1 = rn;

    rn = read_adc();
    y1n = b01 * rn + b11 * rn1 - a11 * y1n1;
    y2n = b02 * y1n + b12 * y1n1 - a12 * y2n1;
    out(y2n);
}

```

					<i>123.KI-41.19</i>	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		48

```

// Основний цикл обробки сигналів
while (1) {
    y1n2 = y1n1;
    y1n1 = y1n;
    rn2 = rn1;
    rn1 = rn;

    rn = read_adc();
    y1n = b01 * rn + b11 * rn1 + b21 * rn2 - a11 * y1n1 - a21 * y1n2;
    y2n2 = y2n1;
    y2n1 = y2n;

    y2n = b02 * y1n + b12 * y1n1 + b22 * y1n2 - a12 * y2n1 - a22 * y2n2;
    out(y2n);
}
}

```

					<i>123.KI-41.19</i>	<i>Арк.</i>
<i>Зм.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>		49

2.8. Принципова схема та друкована плата фільтру

Для розробки принципової схеми і друкованої плати фільтру скористаємося програмою *P-CAD*. Принципова схема: D1 – ATmega8, D2 – DAC900

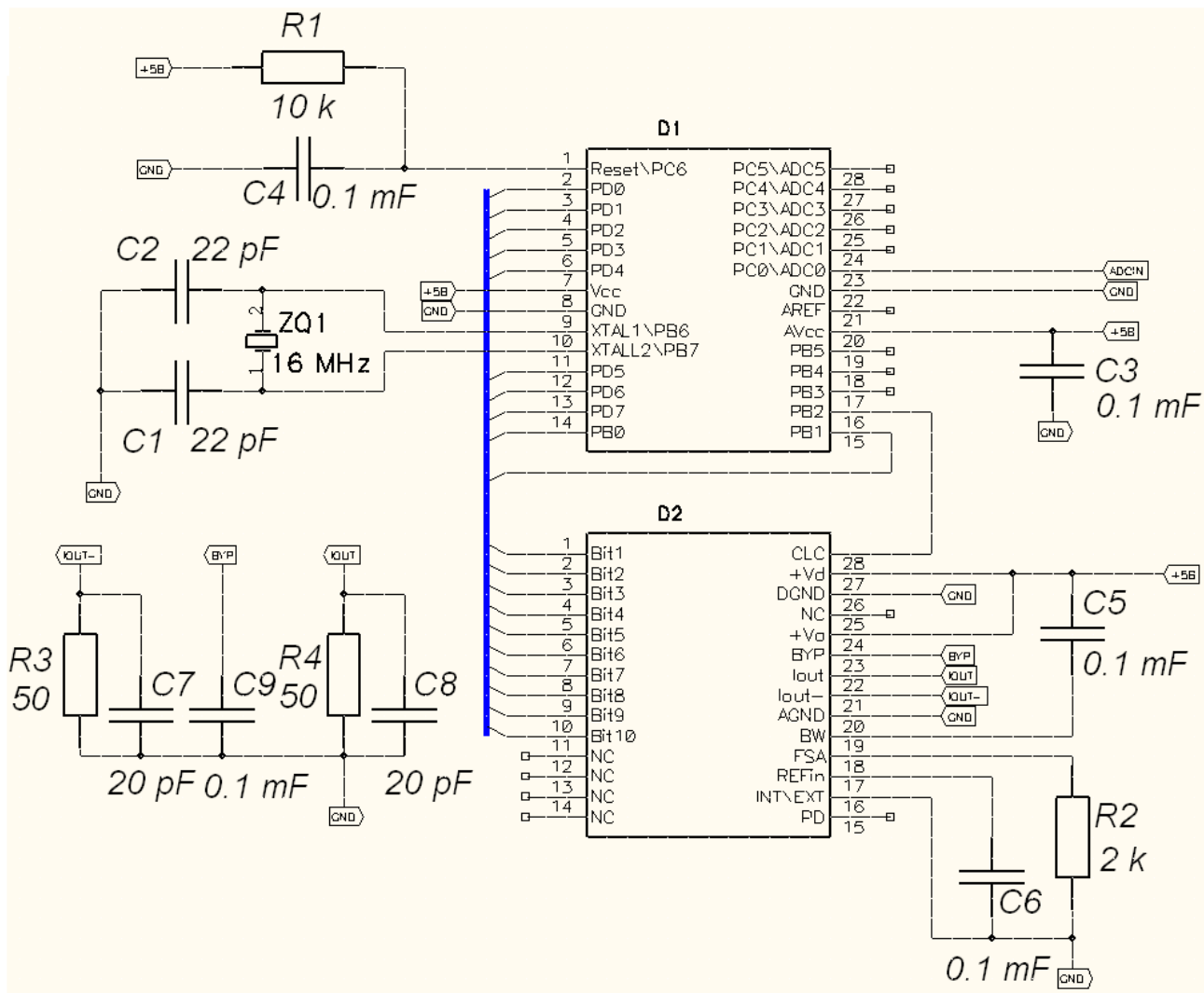


Рис 2.19 –Принципова схема DAC900 із ATmega8

Зм.	Арк.	№ докум.	Підпис	Дата

Друкована плата двостороння з шовкографією, розміри: 55x45 мм

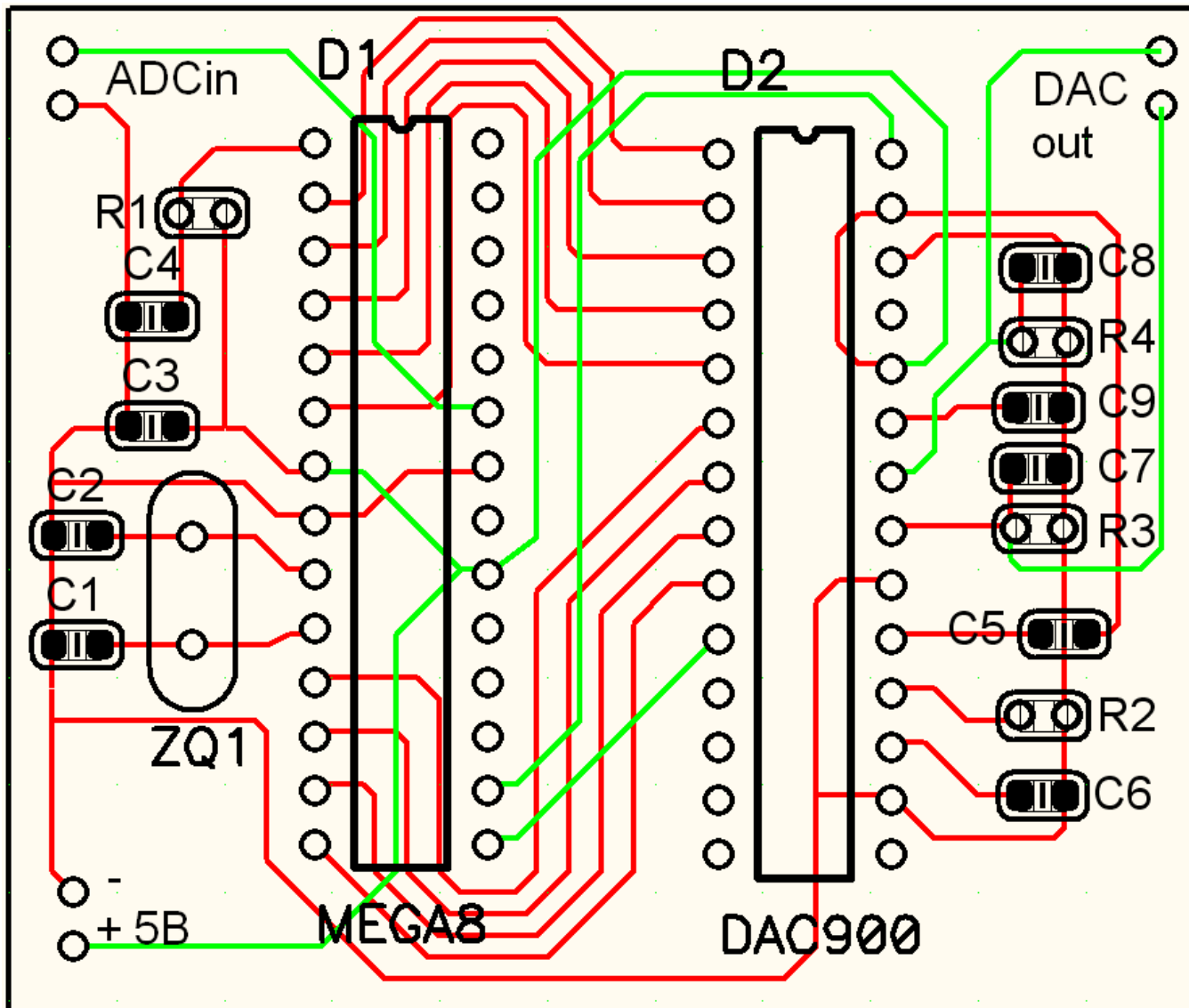


Рис.2.20 – Друкована плата DAC900

Зм.	Арк.	№ докум.	Підпис	Дата

123.KI-41.19

Арк.

51

РОЗДІЛ 3. ОХОРОНА ПРАЦІ ПРИ РОБОТІ З ПК

Робота з персональними комп'ютерами (ПК) є важливим аспектом для забезпечення здоров'я і безпеки працівників, які проводять значну частину свого робочого часу перед екраном. Основні аспекти охорони праці при роботі з ПК включають ергономіку, фізичні умови праці, організацію робочого часу, профілактику професійних захворювань та дотримання нормативних актів.

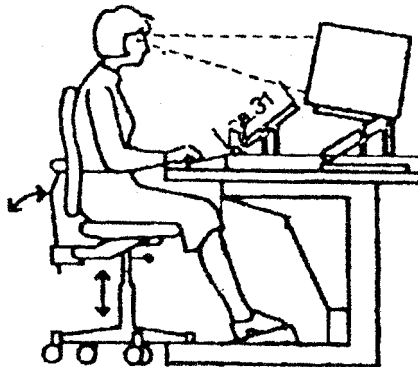
Ергономіка

1. Розташування робочого місця:

- Монітор повинен бути розташований на відстані 50-70 см від очей, верхня частина екрану на рівні очей або трохи нижче.
- Клавіатура і миша повинні знаходитися на такому рівні, щоб руки були зігнуті під кутом 90 градусів.
- Ноги повинні стояти рівно на підлозі або на підставці, коліна зігнуті під кутом приблизно 90 градусів.

2. Робоче крісло:

- Крісло має бути регульованим по висоті, з можливістю нахилу спинки.
- Підтримка попереку та достатня площа сидіння.



3. Освітлення:

- Пряме і відбите світло не повинні створювати відблисків на екрані.
- Рівномірне освітлення без різких тіней, оптимально – природне освітлення.

					123.KI-41.19	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		52

Фізичні умови праці

1. Мікроклімат:

- Температура у приміщенні повинна бути в межах 18-24 °С.
- Вологість повітря – 40-60%.
- Провітрювання робочого місця для забезпечення свіжого повітря.

2. Шум:

- Рівень шуму не повинен перевищувати 50 дБ, щоб не відволікати та не стомлювати працівників.

Організація робочого часу

1. Режим праці і відпочинку:

- Перерви кожні 2 години роботи на 10-15 хвилин.
- Комплекс вправ для очей і загальних фізичних вправ під час перерв.
- Розподіл робочого навантаження для зменшення монотонності та стомлюваності.

2. Ротація завдань:

- Зміна виду діяльності для зменшення стресу і фізичного напруження.

Профілактика професійних захворювань

1. Зорове навантаження:

- Використання антивідблискових фільтрів на моніторі.
- Вправи для очей (фокусування на різних відстанях, моргання для зволоження очей).
- Регулярне відвідування офтальмолога.

2. Опорно-руховий апарат:

- Вправи для зняття напруги з м'язів шиї, спини, рук.
- Регулярна зміна пози під час роботи.
- Правильне налаштування робочого місця для запобігання розвитку захворювань хребта.

Нормативні акти

1. Міжнародні стандарти:

- ISO 9241: Ergonomics of Human-System Interaction.

					123.KI-41.19	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		53

- Регуляції Міжнародної організації праці (МОП).

2. Національні нормативи:

- Законодавство про охорону праці.
- Національні стандарти ергономіки і безпеки праці.
- Рекомендації Міністерства охорони здоров'я.

Практичні рекомендації

1. Навчання працівників:

- Регулярні тренінги з ергономіки та здорового способу життя.
- Інструктажі щодо безпечного користування ПК.

2. Моніторинг і оцінка ризиків:

- Регулярна оцінка робочих місць на відповідність ергономічним вимогам.
- Оцінка стану здоров'я працівників, проведення медичних оглядів.

Забезпечення охорони праці при роботі з ПК є комплексним завданням, яке включає технічні, організаційні та медичні заходи. Виконання цих рекомендацій сприятиме зниженню ризиків професійних захворювань і підвищенню ефективності праці.

					123.KI-41.19	Арк.
						54
Зм.	Арк.	№ докум.	Підпис	Дата		

РОЗДІЛ 4. ВИМОГИ БЕЗПЕКИ ПІД ЧАС ВИКОНАННЯ РОБОТИ.

Вимоги безпеки під час виконання роботи над проектом є критично важливими для забезпечення здоров'я та безпеки працівників. Вони включають дотримання загальних правил безпеки, специфічних вимог для конкретних типів робіт, організаційних заходів і нормативних документів. Ось детальні рекомендації та вимоги:

Загальні вимоги безпеки

1. Інструктаж з охорони праці:

- Всі працівники повинні пройти первинний інструктаж з охорони праці перед початком роботи над проектом.
- Проведення повторних і позапланових інструктажів у разі змін в умовах праці або при впровадженні нових технологій.

2. Знання нормативних документів:

- Працівники повинні знати і дотримуватись вимог законодавства про охорону праці.
- Виконання норм і правил, встановлених в державних стандартах та інструкціях.

Організаційні заходи

1. Планування та оцінка ризиків:

- Проведення аналізу можливих ризиків, пов'язаних з виконанням проекту.
- Розробка заходів для мінімізації або усунення ризиків.

2. Організація робочих місць:

- Робочі місця повинні відповідати вимогам ергономіки і забезпечувати комфортні умови для роботи.
- Забезпечення належного освітлення, вентиляції та температурного режиму.

3. Розподіл обов'язків:

- Чіткий розподіл обов'язків серед членів команди.
- Призначення відповідальних за охорону праці осіб.

					<i>123.KI-41.19</i>	<i>Арк.</i>
<i>Зм.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>		55

Специфічні вимоги безпеки

1. Робота з електрообладнанням:

- Використання справного і сертифікованого електрообладнання.
- Дотримання правил електробезпеки, уникання роботи з мокрими руками або вологими інструментами.

2. Хімічні речовини:

- Правильне зберігання і використання хімічних речовин.
- Наявність засобів індивідуального захисту (ЗІЗ) для роботи з хімікатами.

3. Робота на висоті:

- Використання надійних засобів захисту від падіння (страхувальні пояси, мотузки тощо).
- Забезпечення стабільності та безпеки робочих платформ і сходів.

4. Робота з інструментами і технікою:

- Використання справних і відповідних інструментів.
- Дотримання правил безпечного використання ручного та електроінструменту.

Засоби індивідуального захисту (ЗІЗ)

1. ЗІЗ для очей та обличчя:

- Захисні окуляри, маски, щитки.

2. ЗІЗ для рук:

- Захисні рукавиці, рукавички.

3. ЗІЗ для дихання:

- Респіратори, маски.

4. ЗІЗ для тіла:

- Спецодяг, захисні костюми, взуття з твердим носком.

Поведінка та робочі процедури

1. Поведінка на робочому місці:

- Дотримання дисципліни та правил внутрішнього розпорядку.
- Уникання небезпечних дій, які можуть призвести до травм або аварій.

					123.KI-41.19	Арк.
						56
Зм.	Арк.	№ докум.	Підпис	Дата		

2. Екстрені процедури:

- Знання місцезнаходження аварійних виходів, пожежних гідрантів, аптечок першої допомоги.
- Дії у випадку пожежі, витоку хімічних речовин, аварійного відключення електрики.

Документальне забезпечення

1. Інструкції з охорони праці:

- Наявність затверджених інструкцій з охорони праці для кожного виду робіт.

2. Журнали обліку інструктажів:

- Ведення журналів обліку інструктажів з охорони праці.

3. Документи про технічний стан обладнання:

- Документація про регулярні перевірки та технічне обслуговування обладнання.

Медичні огляди

1. Попередні медичні огляди:

- Обов'язкові для нових працівників перед початком роботи.

2. Періодичні медичні огляди:

- Регулярні огляди для оцінки стану здоров'я працівників, які працюють в умовах підвищеної небезпеки.

					123.KI-41.19	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		57

Таблиця 4.1 - Час регламентованих перерв операторів (користувачів)
ВДТ залежно від категорії та групи робіт

Категорія робіт	Група роботи			
	А кількість знаків	Б кількість знаків	В годин	Час перерви при 8-годинній зміні, хв.
I	20000	15000	2	20
II	21000-40000	16000-30000	2,1-4	40
III	понад 40000	понад 30000	понад 4	60

Підсумок

Дотримання вимог безпеки під час виконання робіт над проектом допоможе запобігти нещасним випадкам, професійним захворюванням та забезпечить ефективну і безпечну роботу. Важливо, щоб всі працівники були проінформовані про ризики та засоби захисту, мали необхідні навички та знання для безпечного виконання своїх обов'язків.

ВИСНОВКИ

У цій кваліфікаційній роботі була успішно розроблена та реалізована система цифрового фільтра на базі мікроконтролера AVR. Досягнутий результат демонструє, що мікроконтролери AVR є ефективними інструментами для впровадження цифрових фільтрів у різноманітних додатках, що потребують обробки сигналів у реальному часі.

Основні етапи роботи включали:

1. Аналіз теоретичних основ цифрових фільтрів:

- Вивчено основні типи цифрових фільтрів, їх переваги та недоліки.
- Проведено порівняння каскадної та паралельної структур цифрових фільтрів.

2. Вибір та обґрунтування архітектури мікроконтролера AVR:

- Обґрунтовано вибір мікроконтролера AVR для реалізації цифрового фільтра з урахуванням його обчислювальних можливостей та енергоефективності.
- Проведено огляд характеристик мікроконтролера AVR, які сприяють його ефективному використанню у даному проекті.

3. Розробка програмного забезпечення для цифрового фільтра:

- Створено програмний код для реалізації біквадратного фільтра з використанням мови програмування C.
- Забезпечено збереження коефіцієнтів фільтра у незалежній пам'яті EEPROM мікроконтролера.
- Реалізовано функцію читання даних з АЦП мікроконтролера та обчислення вихідного сигналу фільтра.

4. Тестування та оптимізація:

- Проведено тестування розробленого цифрового фільтра на реальних даних.
- Виявлено та виправлено помилки, оптимізовано алгоритм для забезпечення стабільної та швидкої роботи системи.

5. Порівняння результатів:

					<i>123.KI-41.19</i>	<i>Арк.</i>
<i>Зм.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>		59

- Порівняно ефективність роботи фільтра, реалізованого на мікроконтролері AVR, з теоретичними очікуваннями.
- Продемонстровано відповідність результатів роботи фільтра заданим технічним вимогам.

У підсумку, реалізація цифрового фільтра на мікроконтролері AVR підтвердила можливість використання даної технології для створення високоефективних систем обробки сигналів. Розроблений фільтр може бути інтегрований у різні системи, що вимагають точної та швидкої обробки даних, наприклад, у медичних приладах, аудіообладнанні, системах зв'язку та інших електронних пристроях.

					<i>123.KI-41.19</i>	<i>Арк.</i>
<i>Зм.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>		60

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Алексєєв В. В. Цифрові фільтри: теорія та застосування. Київ: Наукова думка, 2010. - 245 с.
2. Білоус В. І., Козачок А. В. Основи теорії цифрових фільтрів. Київ: Видавництво КПІ, 2008. - 178 с.
3. Ватченко О. І. Мікропроцесорна техніка: навчальний посібник. Львів: Видавництво ЛНУ, 2011. - 220 с.
4. Гуцалюк М. Б. Основи цифрової обробки сигналів. Вінниця: ВНТУ, 2009. - 198 с.
5. Дем'яненко В. А. Проектування цифрових фільтрів: практичний посібник. Одеса: ОНУ, 2012. - 175 с.
6. Жуковський В. В. Мікроконтролери та їх застосування. Київ: Знання, 2013. - 260 с.
7. Зінченко С. В. Цифрова обробка сигналів. Харків: ХНУРЕ, 2007. - 210 с.
8. Іванов П. П., Кравець О. О. Мікроконтролери AVR: програмування та застосування. Київ: Видавництво КНЕУ, 2014. - 270 с.
9. Карпенко О. М. Цифрові фільтри в електронних системах. Київ: Видавництво НТУУ, 2009. - 188 с.
10. Корольчук В. П. Мікроконтролери AVR: від теорії до практики. Львів: Видавництво ЛП, 2015. - 210 с.
11. Кравченко І. В. Цифрова обробка сигналів: підручник. Київ: Видавництво НАУ, 2011. - 240 с.
12. Литвиненко М. А. Основи мікроконтролерної техніки. Дніпро: ДНУ, 2010. - 200 с.
13. Мельник М. І. Проектування цифрових систем з мікроконтролерами. Київ: Видавництво НАУ, 2008. - 220 с.
14. Мельниченко В. І. Цифрові фільтри: алгоритми та застосування. Львів: Видавництво ЛНУ, 2012. - 190 с.
15. Миколайчук С. О. Проектування цифрових фільтрів на основі мікроконтролерів. Київ: Видавництво КНЕУ, 2013. - 198 с.

					123.KI-41.19	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		61

16. Олійник Ю. І., Павленко І. С. Мікроконтролери AVR: архітектура та програмування. Київ: Видавництво НТУУ, 2009. - 230 с.
17. Пантелєєв В. Г. Цифрова обробка сигналів: практичні аспекти. Харків: ХНУРЕ, 2011. - 210 с.
18. Петров І. В. Мікроконтролери в системах керування. Київ: Видавництво НТУУ, 2010. - 225 с.
19. Поліщук О. О. Цифрові фільтри: основи теорії та практичні аспекти. Львів: Видавництво ЛНУ, 2014. - 220 с.
20. Пономаренко М. В. Сучасні цифрові фільтри. Харків: Видавництво ХНУРЕ, 2013. - 185 с.
21. Приймак В. М. Цифрова обробка сигналів: від теорії до практики. Київ: Видавництво НАУ, 2011. - 240 с.
22. Роговий О. П. Мікроконтролери AVR: від проектування до застосування. Київ: Видавництво КНЕУ, 2012. - 230 с.
23. Савчук В. Г. Цифрові системи обробки сигналів. Київ: Видавництво НТУУ, 2013. - 215 с.
24. Сидоренко А. В. Проектування цифрових систем на базі мікроконтролерів AVR. Львів: Видавництво ЛП, 2015. - 205 с.
25. Скиба М. М. Мікропроцесорні системи керування. Вінниця: ВНТУ, 2009. - 195 с.
26. Соловей В. В. Цифрові фільтри: проектування та застосування. Одеса: ОНУ, 2012. - 190 с.
27. Сопко В. М. Мікроконтролери AVR: практичний курс. Київ: Видавництво КПІ, 2010. - 225 с.
28. Ткаченко О. П. Основи цифрової обробки сигналів. Київ: Видавництво НАУ, 2011. - 215 с.
29. Трофимчук С. В. Цифрові системи обробки сигналів: підручник. Харків: ХНУРЕ, 2013. - 200 с.
30. Устименко В. І. Мікроконтролери та їх застосування. Київ: Видавництво НТУУ, 2012. - 230 с.

					<i>123.KI-41.19</i>	<i>Арк.</i>
<i>Зм.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>		62

- 31.Філіпович М. В. Цифрова обробка сигналів в системах керування. Львів: Видавництво ЛНУ, 2014. - 210 с.
- 32.Хоменко П. Г. Мікропроцесорна техніка: теорія і практика. Вінниця: ВНТУ, 2008. - 205 с.
- 33.Черненко С. П. Проектування цифрових фільтрів: від теорії до реалізації. Київ: Видавництво КНЕУ, 2013. - 198 с.
- 34.Шевченко О. М. Мікроконтролери AVR: навчальний посібник. Львів: Видавництво ЛП, 2011. - 215 с.
- 35.Яременко І. П. Цифрова обробка сигналів: теорія та застосування. Київ: Видавництво НАУ, 2010. - 240 с.

					<i>123.KI-41.19</i>	<i>Арк.</i>
						63
<i>Зм.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>		

ДОДАТОК

Код пристрою.

```
#include <mega8.h>

// Зберігаємо коефіцієнти в незалежній пам'яті EEPROM
eeprom float b01 = 0.859872, b11 = -1.541836, b21 = 0.859872,
             a11 = -1.3647, a21 = 0.65341,
             b02 = 0.859872, b12 = -1.635476, b22 = 0.85987,
             a12 = -1.787, a22 = 0.8389;

double rn, rn1, rn2, y1n, y1n1, y1n2, y2n, y2n1, y2n2;

// Функція для зчитування результату перетворення АЦП
unsigned int read_adc() {
    ADCSRA |= 0x40; // Запуск АЦП
    while ((ADCSRA & 0x10) == 0); // Чекає завершення перетворення
    АЦП
    ADCSRA |= 0x10; // Очищення прапора завершення
    return ADCW; // Повернути результат перетворення
}

// Функція для виводу результату
void out(double out_var) {
    int vix = (int)out_var;
    int a[10];
    for (int i = 0; i < 10; i++) {
        a[9 - i] = vix % 10; // Записати цифри у зворотному порядку
        vix = vix / 10;
    }
    PINB.2 = 1; // Початок виводу
    PIND.0 = a[0]; PIND.1 = a[1];
    PIND.2 = a[2]; PIND.3 = a[3];
    PIND.4 = a[4]; PIND.5 = a[5];
    PIND.6 = a[6]; PIND.7 = a[7];
    PINB.0 = a[8]; PINB.1 = a[9];
}
```

					<i>123.KI-41.19</i>	<i>Арк.</i>
<i>Зм.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>		64

```

    PINB.2 = 0; // Кінець виводу
}

void main(void) {
    // Ініціалізація портів
    PORTB = 0x00; // Програмування порту B на вивід
    DDRB = 0xFF;

    PORTC = 0x00; // Порт C на ввід
    DDRC = 0x00;

    PORTD = 0x00; // Порт D на вивід
    DDRD = 0xFF;

    MCUCR = 0x00; // Вимкнення переривань
    TIMSK = 0x00;

    ACSR = 0x80; // Вимкнення компаратора
    SFIOR = 0x00;

    ADMUX = 0x40; // Вхід АЦП - ADC0, опорна напруга - AVcc
    ADCSRA = 0x85; // Дозвіл роботи АЦП, частота перетворення 375 кГц

    // Початкове зчитування та обробка
    rn = read_adc();
    y1n = b01 * rn;
    y2n = b02 * y1n;
    out(y2n);

    y1n1 = y1n;
    y2n1 = y2n;
    rn1 = rn;

    rn = read_adc();
    y1n = b01 * rn + b11 * rn1 - a11 * y1n1;

```

					<i>123.KI-41.19</i>	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		65

```

y2n = b02 * y1n + b12 * y1n1 - a12 * y2n1;
out(y2n);

// Основний цикл обробки сигналів
while (1) {
    y1n2 = y1n1;
    y1n1 = y1n;
    rn2 = rn1;
    rn1 = rn;

    rn = read_adc();
    y1n = b01 * rn + b11 * rn1 + b21 * rn2 - a11 * y1n1 - a21 *
y1n2;
    y2n2 = y2n1;
    y2n1 = y2n;

    y2n = b02 * y1n + b12 * y1n1 + b22 * y1n2 - a12 * y2n1 - a22
* y2n2;
    out(y2n);
}
}

```

					<i>123.KI-41.19</i>	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		66