

Міністерство освіти і науки України
ДВНЗ «Прикарпатський національний університет імені Василя Стефаника»
Кафедра комп'ютерної інженерії та електроніки
(повна назва кафедри)

Павлишин Андрій Васильович
Pavlyshyn Andrii

УДК _____ 004:681.5 _____

Спеціальність 123 «комп'ютерна інженерія»
(шифр та назва спеціальності)

Кваліфікаційна робота
на здобуття освітньо-кваліфікаційного рівня _____ бакалавр _____
(бакалавр, спеціаліст, магістр)

Розробка мікроконвертора аналогових сигналів на базі
мікроконтролера
Development of microconverter of analog signals based on
microcontroller

Науковий керівник:
доктор технічних наук,
професор кафедри комп'ютерної
інженерії та електроніки
Новосядлий С.П.

Рецензент:
доктор фіз.-мат. наук, професор
кафедри фізики та хімії твердого
тіла
Салій Я.П.

Івано-Франківськ
2021

Форм.	Зона	Поз.	Позначення	Найменування	К-ть	Прим.
			123. УДК 004:681.5	Принципова ел. схема	1	
			123. УДК 004:681.5	Пояснювальна записка	77	

						123.КІ-41.11		
<i>Змн</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підп.</i>	<i>Дата</i>				
Розроб.		Павлишин А.В.			Специфікація	<i>Літ.</i>	<i>Арк.</i>	<i>Аркушів</i>
Перев.		Новосядлий С.П					2	78
Н.конт.								
Затверд								

АНОТАЦІЯ

В бакалаврській роботі розглянуто обробку аналогових сигналів за допомогою мікроконверторів. Описано принципи роботи мікроконвертора та його елементів, а також алгоритми обробки сигналів.

Розроблено принципову електричну схему та прототип пристрою. Прототип базується на мікроконтролері STM32F103C8, аналого-цифровому перетворювачі МСР3201 та цифро-аналоговому перетворювачі МСР4725.

Розроблено програму для пристрою яка містить в собі: конфігурацію роботи мікроконтролера, основний цикл та реалізацію алгоритмів роботи сигналів.

					123.KI-41.11			
Змін.	Арк.	№ докум.	Підпис	Дата				
Розробив		Павлишин А.В.			Анотація	Арк.	Аркуш	Аркушіє
Перевірів		Новосядлий С.П					3	78
Н. Контр.								
Затверд.								

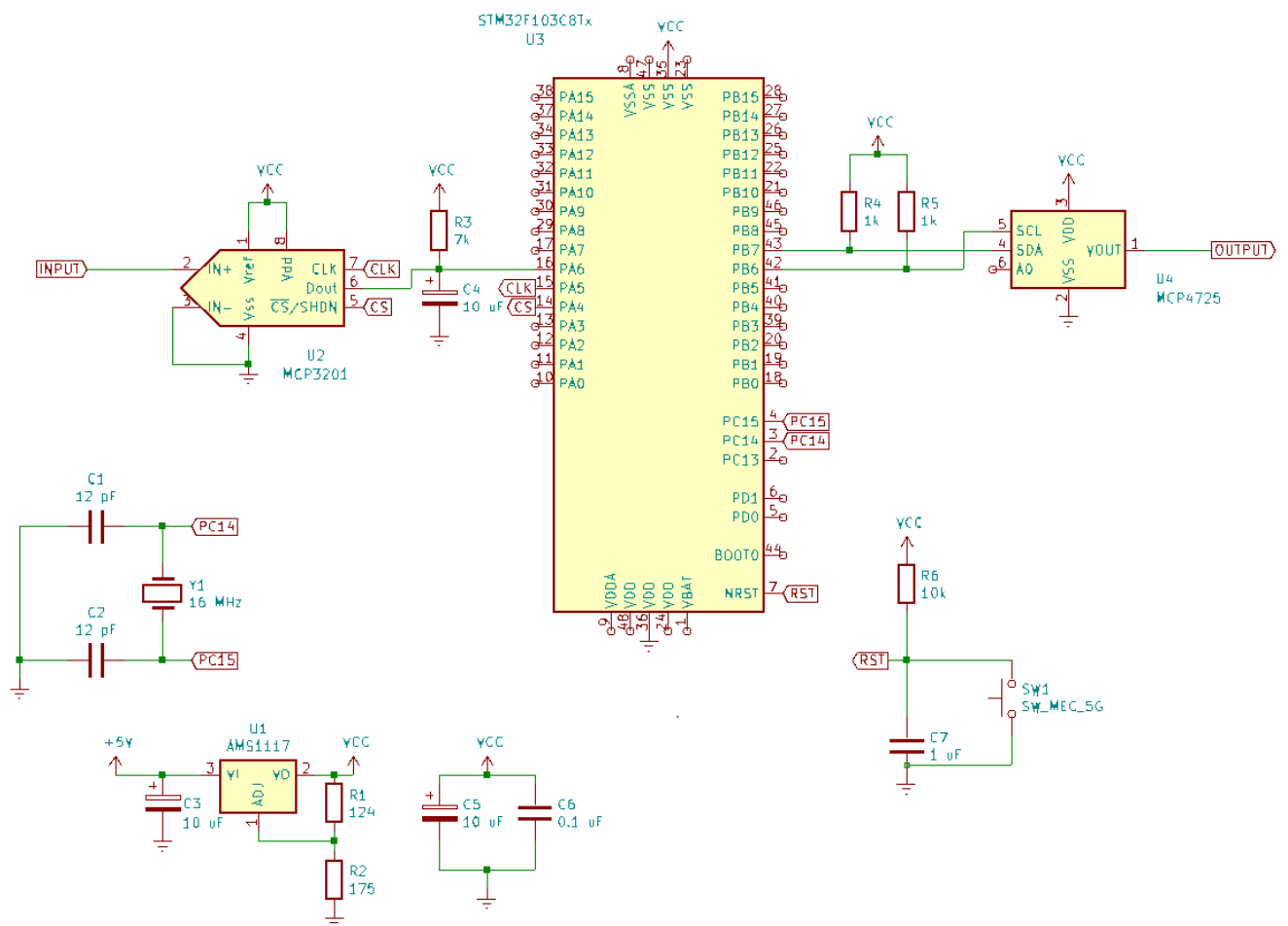
SUMMARY

In the diploma project considered analog signal processing with microconverters. Described working principles of microconverters, its elements and signal processing algorithms.

An electrical wiring diagram of the components and device prototype have been developed. Prototype based on STM32F10C8 microcontroller, analog-digital convertor MCP3201, and digital-analog convertor MCP4725.

Firmware that was developed for device includes: microcontroller configuration, main loop of work, implementation of signal processing algorithms.

					<i>123.KI-41.11</i>					
<i>Змін.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>	<i>Abstract</i>					
<i>Розробив</i>		<i>Павлишин А.В</i>						<i>Арк.</i>	<i>Аркуш</i>	<i>Аркушіє</i>
<i>Перевірів</i>		<i>Новосядлий С.П</i>							<i>4</i>	<i>78</i>
<i>Н. Контр.</i>										
<i>Затверд.</i>										



123.KI-41.11

Змін.	Арк.	№ докум.	Підпис	Дата
Розробив		Павличин А.В.		
Перевірів		Новосядлий С.П.		
Н. Контр.				
Затверд.				

Принципова
електрична схема

Арк.	Аркуш	Аркушів
	5	78

Міністерство освіти і науки України
 Державний вищий навчальний заклад
 «Прикарпатський національний університет імені Василя Стефаника»
 Фізико-технічний факультет
 Кафедра «Комп'ютерної інженерії та електроніки»

Пояснювальна записка

до кваліфікаційної роботи на тему:

Розробка мікроконвертора аналогових сигналів на базі мікроконтролера

					123.KI-41.11			
<i>Змін.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>				
<i>Розробив</i>		<i>Павлишин А.В</i>			Пояснювальна записка	<i>Арк.</i>	<i>Аркуш</i>	<i>Аркушів</i>
<i>Перевірів</i>		<i>Новосядлий С.П</i>					6	78
<i>Н. Контр.</i>								
<i>Затверд.</i>								

ЗМІСТ

ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ	8
ВСТУП.....	9
1. АНАЛІЗ ОСНОВНИХ АСПЕКТІВ ТА ПРИНЦИПІВ РОБОТИ МІКРОКОНВЕРТОРІВ.	11
1.1. Загальні відомості про пристрій	11
1.2. Основні елементи мікроконвертора	11
1.3. Види сигналів та методи їх обробки	21
1.4. Области застосування мікроконвертерів	28
Висновки до розділу.....	31
2. ВИБІР ЕЛЕМЕНТНОЇ БАЗИ ДЛЯ РЕАЛІЗАЦІЇ МІКРОКОНВЕРТОРА	32
2.1. Вибір аналого-цифрового перетворювача	32
2.2. Вибір цифро-аналогового перетворювача	37
2.3. Вибір мікроконтролера	42
Висновки до розділу.....	44
3. РОЗРОБКА МІКРОКОНВЕРТОРА	45
3.1. З'єднання основних елементів та розробка принципової електричної схеми пристрою.....	45
3.2. Розробка програми та опис роботи пристрою.....	49
Висновки до розділу.....	56
4. ЕКОНОМІЧНА ЧАСТИНА ТА ОХОРОНА ПРАЦІ	57
ВИСНОВКИ.....	62
ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ	63
ДОДАТКИ	65

					123.KI-41.11	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		7

ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ

АЦП — аналого-цифровий перетворювач.

ЦАП — цифро-аналоговий перетворювач.

МК — мікроконтролер.

АЛП – арифметико-логічний пристрій.

ПЗП – постійний запам'ятовуючий пристрій.

ОЗП – оперативний запам'ятовуючий пристрій.

ІС – інтегральна схема.

ВІС – велика інтегральна схема.

					123.КІ-41.11	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		8

ВСТУП

Електронні пристрої поділяються на два види: аналогові та цифрові.

Аналогові пристрої призначені для роботи з неперервними сигналами які можуть змінювати свій рівень на будь-яку величину в будь-який момент часу. Вони складаються з певних базових електричних кіл: фільтрів, підсилювачів, модуляторів, демодуляторів, компараторів і т.д.

Цифрові мікроелектронні пристрої являють собою дискретні цифрові автомати, виконані на інтегральних мікросхемах і призначені для обробки інформації, що представлена у вигляді цифрового коду. Вони використовуються для створення цифрових інформаційних, вимірювальних систем та систем керування.

Проте більшість реальних фізичних величин: звук, колір, світловий потік, тиск, температура, можуть бути представлені тільки у вигляді аналогового сигналу і не можуть бути чітко поділені на певні рівні. Аналогові пристрої більш складні, дорогі у виготовленні і менш надійні, в той час як цифрові пристрої надійніші, більш універсальні і простіші у виготовленні. Тому часто аналогові та цифрові схеми поєднують разом, обробка, перетворення та збереження інформації часто виконується у цифровому вигляді, а відтворення або, навпаки, зчитування відбувається у аналоговому вигляді.

Важливу роль в цьому підході відіграють пристрої аналого-цифрового та цифро-аналогового перетворення які виконують перехід між аналоговими та цифровими сигналами. Обробку та зберігання даних може здійснюватись за допомогою комбінаційних схем або, частіше, мікропроцесорів чи мікроконтролерів. Особливими рисами пристроїв побудованих на базі мікропроцесорів є можливість розширення, зміни чи корекції роботи пристрою шляхом зміни програми пристрою, без зміни апаратної частини пристрою. Це дозволяє зменшувати апаратні витрати при проектуванні, оскільки програмна частина може виконувати різні функції реальних пристроїв. Іншою особливістю є те, що такі пристрої можуть бути виконані на одному кристалі як ВІС. Такий підхід дозволяє забезпечити високу швидкодію, гнучкість та продуктивність.

					123.KI-41.11	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		9

В цій роботі розглядаються пристрої які здійснюють обробку аналогових сигналів в цифровому вигляді – мікроконвертори. В таких пристроях вхідний сигнал спочатку надходить на АЦП після цього на мікроконтролер в цифровому вигляді, мікроконтролер здійснює обробку значень згідно програми і передає дані на ЦАП який відтворює оброблений аналоговий сигнал. Таким чином сигнал пройшовши “наскрізно” зазнає обробки.

					123.KI-41.11	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		10

1. АНАЛІЗ ОСНОВНИХ АСПЕКТІВ ТА ПРИНЦИПІВ РОБОТИ МІКРОКОНВЕРТОРІВ

1.1. Загальні відомості про пристрій

Особливий тип пристроїв з АЦП-ЦАП – це мікроконвертори. На початку розвитку субмікронної кремнієвої технології була спроба створення ВІС на аналогових програмованих матриць, тобто пристроїв, які включають в себе операційні підсилювачі, суматори, компаратори, фільтри та також комірки, зв'язки між якими б встановлювались програмним шляхом. Але такі спроби не отримали комерційного успіху.

Нещодавно такі фірми як Analog Devices, почали серійний випуск програмованих пристроїв для перетворення аналогових сигналів в цифрові, які містять у своєму складі багатоканальний АЦП, мікроконтролер і одно- чи двоканальний ЦАП. Такий мікроконвертор приймає аналогові сигнали і перетворює їх у цифрові коди, обробка цих кодів є записаною в ПЗП мікроконтролера програмою і за допомогою ЦАП знову перетворює результат в аналоговий сигнал. Поступаючись перед суто аналоговою системою тільки швидкодією, така схема відрізняється дуже великою функціональною гнучкістю і точністю.[1]

1.2. Основні елементи мікроконвертора

В загальному мікроконвертер має три основні модулі:

- Аналого-цифровий перетворювач
- Мікропроцесор(ядро мікроконтролера)
- Цифро-аналоговий перетворювач

До складу пристрою можуть бути додані і інші модулі для додаткових функцій чи допоміжних до основної функції, обробки сигналу. Але вище наведені елементи є основними, жоден з них не може бути вилученим, і саме за допомогою них відбувається обробка аналогових сигналів.

					123.KI-41.11	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		11

На початку аналоговий сигнал який повинен бути оброблений мікроконвертором потрапляє на вхід АЦП.

АЦП здійснює перетворення вхідного аналогового сигналу в цифрову форму, з якою може працювати мікропроцесор.

Через те що аналоговий сигнал є неперервний в часі, а цифровий має конкретні значення в певні моменти часу, для перетворення аналогового сигналу в цифровий здійснюється дискретизація в часі(вибірка). Вибірка сигналу полягає в тому, що АЦП фіксує значення напруги аналогового сигналу на вході в певний момент часу. Це здійснюється за допомогою пристрою вибірки і збереження аналогового сигналу (ПВЗ) або аналоговою пам'яттю (Simple Hold Amplifier - SHA).[2]

В більшості схемотехнічних рішень для цього використовують певні поєднання накопичувального конденсатора та аналогових ключів з операційними підсилювачами (ОП). В режимі вибірки вихідна напруга повністю відповідає вхідному сигналу, а в режимі збереження – миттєвому значенню вхідного сигналу в момент закінчення вибірки.

Маючи конкретне значення напруги вхідного сигналу знаходиться відповідний йому рівень напруги шкали АЦП. Така операція називається квантуванням. Чим більше рівнів квантування має АЦП, тим більше розрядів матиме вихідний код і тим точніше АЦП буде передавати рівень аналогової напруги.

Кількість рівнів квантування визначається кількістю розрядів вихідного коду як: 2^n , де n – кількість розрядів вихідного коду. Тобто, 12-бітний має 4096 рівнів квантування.

Розподіл рівнів квантування по діапазону значень вхідної напруги, може бути лінійним та нелінійним(напр. логарифмічним). Найчастіше використовуються лінійні АЦП. В них різниця напруги між двома рівнями квантування або вага наймолодшого розряду вихідного коду, визначається як різниця між максимальним та мінімальним рівнями напруги, поділена на кількість рівнів квантування. Мінімальний рівень напруги, зазвичай, дорівнює 0

					123.KI-41.11	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		12

V(“землі”), а максимальний рівень визначається типом логіки АЦП(5, 3.3, 1.8 В). Наприклад 12-бітний АЦП з логікою 5В матиме різницю між двома сусідніми рівнями квантування: $(5-0)/4096 = 1.22$ мВ.

Похибка квантування визначається формулою: $\frac{1}{\sqrt{12}} LSB = 0.289 LSB$,

де LSB(Least significant bit) – значення ваги наймолодшого розряду у вольтях або в процентному відношенні до цілого діапазону.

Помилка квантування АЦП зі значенням ваги наймолодшого розряду 1.22 мВ буде рівною: $\frac{1}{\sqrt{12}} LSB = 0.289 \cdot 1.22 \text{ мВ} = \pm 0.35 \text{ мВ}$

Незалежно від типу АЦП, основним елементом квантування є компаратор. Компаратор порівнює значення напруг на його двох входах і залежно, від того на якому вході напруга більша на вхід подає лог. значення “0” або “1”.

Реалізація процесу квантування та його параметри(швидкодія, завадостійкість, спотворення, апаратна складність), залежать від типу АЦП. За типами АЦП зазвичай поділяють за реалізацією квантування, бо ця операція визначає більшість характеристик АЦП.[3]

Виділяють наступні типи АЦП:

- АЦП прямого перетворення
- АЦП з порозрядним зважуванням
- Інтегруючий АЦП
- Конверсні АЦП
- Сигма-Дельта АЦП

Після квантування АЦП виводить напругу у цифровому вигляді яку може сприймати мікроконтролер для подальшої обробки. Мікроконтролер отримавши дані з АЦП повинен здійснити обробку сигналу згідно записаної в пам’ять програми і після обчислень передати дані на вхід ЦАП.

Сьогодні при створенні інформаційно-телекомунікаційних систем використовують мікроконтролери з CISC або RISC архітектурою.

Перші мають досить розвинуту архітектуру, наприклад, МК серії КР1970ВЕ51 мають більше 111 команд. Аналіз програм показав, що тільки 20%

команд використовують у 80% випадків, а сам дешифратор команд займає майже 75% площі кристалу, що зменшує швидкодію та збільшує споживану потужність. Тому в розробників МК виникла ідея скорочення кількості команд і надання їм єдиного формату, що приводить до зменшення площі кристалу, тобто використовуючи вже RISC-архітектуру (Reduced Instruction Set Computer). Особливістю мікроконтролерів, виконаних на RISC- архітектурі, є те, що всі команди виконуються за 1-3 такти, тоді, як у CISC-мікроконтролерів за 1-3 машинних цикли, кожен з яких складається з декількох тактів. Тому RISC-контролери мають значно вищу швидкодію, меншу споживану потужність та площу кристалу.[4]

Також те що більшість команд RISC-контролерів виконуються за однаковий час спрощує їх конвеєризацію та використання паралельних обчислень.

Правда, повніша система команд CISC-контролерів в деяких випадках економить час виконання окремих фрагментів програми і пам'ять програм.

Тому доцільно використовувати мікроконтролери RISC-архітектури як з метою їх використання для побудови швидкісних мікроконверторів сигналів перетворення АЦП-ЦАП вже як сигнальних МК.

Одним з прикладів RISC-архітектури є мікроконтролери stm32 з архітектурою ARM-Cortex, на прикладі якого ми розглянемо їх будову.

Мікроконтролери складаються з мікропроцесора (ядра мікроконтролера) та периферії яка дозволяє взаємодіяти або керувати іншими електронними пристроями.

Для RISC-контролерів притаманний гарвардський тип процесора. Він реалізує повний логічний і фізичний розподіл не тільки адресних просторів, але і інформаційних шин для звертання до пам'яті програм і до пам'яті даних, які є розділеними, причому способи адресації та доступу до цих масивів також різні. Така будова вже є ближчою до структури до цифрових сигнальних процесорів і забезпечує істотне підвищення швидкодії. Процесор працює одночасно як із пам'яттю програм, так і з пам'яттю даних.

					123.KI-41.11	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		14

Процесор складається з наступних основних модулів:

- Арифметико-логічного пристрою(АЛП)
- Пристрій керування
- Регістри

АЛП виконує операції арифметичних обчислень(додавання, віднімання, множення, ділення) та логічні операції(І, АБО, НЕ, порівняння) над вхідними даними. АЛП має два входи, дані на один вхід надходять завжди з реєстрового файлу, а дані на другий вхід можуть надходити як з реєстрового файлу так і з шини даних чи з програмних інструкцій.

Пристрій керування координує роботу процесора, згідно складеної програми і є найважливішим його елементом. Він складається з реєстра команд, лічильника команд та дешифратора. За допомогою лічильника команд пристрій керування зчитує наступну команду з пам'яті та зберігає в реєстрі команд. Потім дешифратор перетворює інструкцію з реєстру на послідовність керуючих сигналів які дозволяють виконати команду. Пристрої керування можуть бути зроблені як комбінаційні схеми або ж як автомат з гнучкою логікою який має свою програму та набір інструкцій

Процесор напряму не працює з загальною пам'яттю, тому для роботи з даними та зберігання станів він використовує реєстри. Кількість, розрядність та призначення реєстрів є різним залежно від конкретного процесора. Наприклад, мікроконтролер з ядром ARM cortex-M3 має 37 реєстрів, з яких 31 є реєстри загального призначення. Регістри загального призначення доступні для зчитування та запису будь-яких даних програмою користувача. Регістри спеціального призначення використовуються процесором для конкретних функцій і користувач не має до них прямого доступу. До реєстрів спеціального призначення належать: вказівник стеку, лічильник команд, реєстр стану і т.д.

Крім основних елементів мікропроцесор може мати додаткові елементи, які визначаються виробником залежно від його призначення. Такими модулями можуть бути апаратні реалізації операцій множення та ділення або блок роботи з числами з плаваючою комою. Такі модулі здійснюють відповідні операції

					123.KI-41.11	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		15

швидше ніж вони б робились програмним чином. Наприклад операції множення та ділення на основі апаратної реалізації які доступні майже у всіх ARM процесорах здійснюються за один такт.

Мікропроцесор є найбільш складним елементом мікроконтролера. Розробка конкурентного мікропроцесора потребує дуже значних матеріальних та часових ресурсів. Також в розробку входить не тільки фізична частина, але й програмної частини: наборів інструкцій, компілятори для низькорівневих та високорівневих мов. Тому деякі мікроконтролерні компанії замість створення власного мікропроцесора купують ліцензію на виготовлення існуючих мікропроцесорів або використовують архітектури які поширюються вільно. На сьогодні ARM є найбільш поширеною комерційною архітектурою яку компанії купують для створення власних пристроїв. Серед вільно поширених архітектур найбільш вживана є RISC-V, проте вона все одно є менш вживаною виробниками порівняно з ARM. Також недавно стали набувати популярності рішення на архітектурі 8051 яка давно можна вважати застарілою. Причиною зростання популярності є закінчення ліцензійних прав, що дозволяє вільно використовувати цю архітектуру, відлагодженість та поширеність програмних засобів та можливість збільшити продуктивність за рахунок сучасніших технологій які дозволяють встановлювати високі робочі частоти процесора.

Сам по собі мікропроцесор не є самостійним модулем та не може працювати без додаткової периферії яка дозволяє йому працювати та керувати зовнішніми пристроями.

Такі елементи як кварцовий генератор та стабілізатор живлення дозволяють процесору просто функціонувати. Також потрібна пам'ять для зберігання інструкції програм та даних. Мікроконтролери зазвичай мають два типи пам'яті: постійна та тимчасова. Постійна пам'ять використовується для зберігання інструкцій програми та зберігання деяких змінних які є константами і є енергонезалежною. Основна постійна пам'ять зазвичай реалізується як Flash-пам'ять та не може бути перезаписана самим мікроконтролером в процесі роботи. Для зміни даних постійної пам'яті(зміна програми мікроконтролера)

					123.KI-41.11	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		16

використовуються зовнішні пристрої програматори. Також мікроконтролери мають додаткову енергонезалежну пам'ять типу EEPROM. EEPROM пам'ять є меншою за об'ємом, проте дані можуть бути перезаписані мікроконтролером без зовнішніх пристроїв. EEPROM зазвичай використовується для зберігання певних конфігураційних даних які можуть змінюватись в програмі і їхні значення необхідно зберігати навіть після виключення мікроконтролера. Другий тип пам'яті це оперативна. В оперативній пам'яті зберігаються дані які мікроконтролер може змінювати, зазвичай, це змінні над якими здійснюються обчислення чи дані які отримуються з периферії. Оперативна пам'ять є енергозалежною і всі дані які були занесені втрачаються після того як від мікроконтролера було відключено живлення.

Також для роботи мікроконтролер потребує пристрої вводу/виводу. Без них він не зможе взаємодіяти з зовнішнім світом, передавати дані іншим пристроям щоб здійснювати керування чи виводити результати своєї роботи.

Зв'язок між мікропроцесором та пристроями вводу-виводу зазвичай реалізується за допомогою регістрів. Мікропроцесор може працювати з регістрами, зчитувати та записувати в них дані. Виходи регістрів під'єднані до входів керуючих елементів периферії. Для прикладу вихід одного розряду регістра може бути під'єднаний до керуючого входу ключової схеми яка на виході може керувати напругою. Цей приклад є спрощеним, але саме за таким принципом здійснюється керування портами загального вводу і виводу у режимі цифрового виходу.

Тому керування периферією мікропроцесором на фізичному рівні відноситься до операцій роботи з пам'яттю.

До периферійних пристроїв належать порти введення-виведення, послідовні порт, таймери-лічильники, контролер переривань, кількість незалежних ліній портів введення-виведення – від 3 до 53. Ключовий розряд порту можна запрограмувати на режим входу або виходу. Потужні вихідні драйвери забезпечують струмову навантажувальну здатність 20mA на лінію порту. МК мають в своєму складі від декілька таймерів/лічильників загального

					123.KI-41.11	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		17

призначення з різною роздільною здатністю, які можуть працювати як від внутрішнього джерела так і від зовнішніх джерел частоти, подій.

Інтерфейс SPI призначений для послідовного введення і виведення даних, його використовують для керування пристроїв які теж підтримують SPI. Пристрої які взаємодіють через SPI поділяються на головні та другорядні, які їм підпорядковуються. Інтерфейс використовує чотири лінії: SCK – тактуючий сигнал, MOSI – дані з головного пристрою до другорядного MISO – дані з другорядного до головного пристрою, GND – земляна шина для проходження сигналів. Головний пристрій може посилати дані до другорядного не очікуючи на відповідь, або послати запит і зчитати відповідь для отримання даних від пристрою. Сам другорядний пристрій не може почати передачу даних без запиту головного пристрою.

Декілька другорядних пристроїв можуть бути підключеними до одного головного пристрою паралельно до одних і тих же інформаційних ліній. Оскільки інтерфейс SPI є послідовним, а пристрої підключені паралельно, то використовується додаткова лінія керування CS(Chip select) яка визначає який пристрій в певний час здійснювати комунікацію з головним пристроєм. Головний пристрій має мати для кожного другорядного пристрою окрему лінію CS, і під час комунікації на одній вибраній CS лінії встановлювати активний рівень, а на інших тримати неактивний. Таким чином вибирається один з декількох пристроїв для послідовної передачі даних.

Інтерфейс I2C є також послідовним інтерфейсом поширеним в електронних пристроях. Як у випадку з SPI пристрої діляться на головні та другорядні. Для передачі даних використовуються дві лінії: SDA – лінія даних і SCL – синхронізуюча лінія. Початок передачі даних завжди відбувається від головного пристрою до другорядного. Другорядний пристрій після отримання даних відправляє відповідь на запит. Всі пристрої під'єднані до одних і тих же ліній передачі. Вибір другорядного пристрою здійснюється за допомогою адресації. На початку відправки даних перші 7-біт означають адрес пристрою до якого звертаються і після цього йдуть самі дані. Всі другорядні пристрої

					123.KI-41.11	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		18

зчитують цей адрес і порівнюють з власним, і тільки той пристрій в якого отриманий адрес співпадає з власним зчитає наступні дані і відправляє відповідь-підтвердження, решта пристроїв ігнорують наступні дані. Тому всі другорядні пристрої повинні між собою повинні мати унікальні адреси. І оскільки адрес кодується 7 бітами, то до одного головного пристрою може бути підключено до 127 пристроїв.

У складі мікроконтролерів є і інші інтерфейси, проте SPI і I2C є в складі майже у всіх мікроконтролерів та використовуються для підключення до мікроконтролера зовнішніх АЦП та ЦАП, що може використовуватись в мікроконверторі.

Після здійснення обробки сигналу згідно записаної програми мікроконтролер повинен перетворити оброблений сигнал знов у аналогову форму.

Цифро-аналоговий перетворювач(ЦАП) виконує функцію обернену до АЦП, тому багато їх характеристик є спільними. Точність відтворення аналогової напруги залежить від розрядності ЦАП. Кількість рівнів ЦАП визначається як 2^n , де n – розрядність ЦАП. Та різниця між двома рівнями(вага наймолодшого розряду) лінійного ЦАП визначається як діапазон напруги шкали розділений на кількість рівнів. Тобто вихідна напруга може бути розрахована як вага наймолодшого розряду помножена на значення вхідного двійкового коду. Одним з важливих параметрів як і у випадку АЦП є швидкодія, оскільки від цього залежить з сигнали яких частот може відтворювати ЦАП.

Цифро-аналогове перетворення може відбуватись за допомогою ресурсів мікроконтролера(програмний) або через окремий модуль(апаратний).

Програмний метод цифро-аналогового перетворення включає в себе використання широтно-імпульсної модуляції(ШІМ).

Він полягає в зміні пропорційному цифровому значенню коефіцієнту заповнення імпульсів прямокутної напруги, що поступають з постійною частотою. Тут коефіцієнт заповнення визначається як відношення тривалості імпульсу до тривалості проміжків між ними.

					123.KI-41.11	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		19

Тому можна реалізувати просте цифро-аналогове перетворення за допомогою ШІМ сигналу з МК. В такому випадку для формування аналогової вихідної напруги використовується безпосередньо вихідний канал порту з послідовно підключеним RC-фільтром нижніх частот. Для підвищення точності і стабільності можна використати RC-фільтр(ФНЧ Батерворта другого порядку), сформованого на основі операційного підсилювача.

Цей метод є доступним, оскільки всі сучасні мікроконтролери мають в наявності ШІМ-виходи. Проте одним з його недоліків є те що в цьому випадку використовуються обчислювальні ресурси мікроконтролера для перетворення сигналу. Також складно чітко виділити параметри такого ЦАП через те що багато залежить від конкретної реалізації зовнішнього фільтра.

Інший метод полягає у використанні окремого модуля ЦАП.

Найпростішим варіантом є ЦАП який базується на методі зважування.

Такий ЦАП складається з певної кількості послідовно з'єднаних однакових резисторів і така сама кількість ключів. Кожному біту відповідає резистор який виступає в ролі джерела струму і всі вони за допомогою ключів можуть бути під'єднані до спільної точки "сумування". Ключами керує дешифратор який перетворює вхідний двійковий код на унарний який поступає на ключі. Цей метод є одним з найшвидших, проте характеризується найменшою точністю та містить велику кількість ключів та резисторів.

Однією з найбільш поширених структур є ЦАП виконаний на матриці R-2R. Тут використовуються дві величини опорів з відношенням 2 до 1. Кількість резисторів в два рази більша за кількість розрядів. Резистивна матриця може бути складена для додавання струмів(прямий режим) або додавання струмів(інверсний режим). У випадку структури додавання струмів перед виходом підключають ще ОП який може перетворити струм в напругу.

Таким чином вхідний аналоговий сигнал пройшовши "наскрізно" через весь ланцюжок перетворень на виході стає обробленим.

Одним з найважливіших характеристик всього пристрою є швидкодія. Тому що саме вона визначає які сигнали з якими частотами може обробляти

					123.KI-41.11	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		20

пристрій. Час обробки сигналу цілого пристрою складається з суми: часу перетворення аналого-цифрового пристрою, часу обробки сигналу МК, часу перетворення цифро-аналогового пристрою. Для того щоб сигнал після обробки на виході не мав втрат цей сумарний час повинен бути менший ніж половина періоду коливання сигналу, з теореми Найквіста-Шенона.

1.3. Види сигналів та методи їх обробки

Термін сигнал часто використовують в різних значеннях, без строгого формулювання. Під цим терміном розуміють засіб передавання інформації за допомогою певних фізичних явищ. Сигнали можуть бути поділені за видом фізичних явищ яким вони переносять інформацію – електромагнітні, оптичні, механічні. Зміна певної величини носія інформації(амплітуда, частота електромагнітних сигналів) є відображенням інформації. Сигнал можна визначити як фізична величина яка містить певну інформацію.

Сигнали є дуже різноманітні, за певними критеріями можна поділити їх на види. Проте реальні сигнали можуть і не вклатись в конкретну категорію. Нижче наведені характеристики за якими можуть ділитись сигнали за їхньою формою.

Детерміновані – значення сигналу відоме в будь-якій точці часу.

Випадкові – можуть набувати довільних значень в будь-який момент часу.

Неперервні – сигнал існує в будь-якій точці часу і може бути представлений як неперервна функція яка не має розривів першого роду.

Імпульсні – сигнал існує лише в певні моменти часу, тобто його функція матиме розриви першого роду.

Періодичні – сигнал який повторює свою форму за деякий період часу, може бути представлений як періодична функція.

Неперіодичний – сигнал не повторює свою форму сигналу, можна сказати що його період прямує до нескінченості.

Кінцевої довжини – період існування такого сигналу є певним визначеним числом, можна визначити початок і кінець сигналу.

					123.KI-41.11	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		21

Нескінченної довжини – сигнал існує завжди, не можливо виділити початок та кінець.

З цих вище перерахованих критеріїв на практиці виділяють два основних види сигналів це є цифрові та аналогові.

Цифровий сигнал – є детермінованим, тобто займає лише визначені значення, та дискретним, тобто зміна сигналу існує лише у відповідні моменти часу, інакше кажучи такий сигнал може бути поданий як функція відліків у часі. Навіть якщо при передачі відбудеться помилка, то маючи можна відновити первинний неспотворений сигнал програмним чином якщо це передбачає формат даних. Такі формати називаються коригувальними кодами.

Аналоговий сигнал – є неперервним в часі та може змінювати своє значення на будь-яку величину, такі сигнали ще називають випадковими(стохаїстичними). Через те що аналоговий сигнал немає обмежень за значенням амплітуди та часу в який він може змінюватись, виникають труднощі у передачі та прийому такого роду сигналів. Тут на відміну від цифрових кодів не можливо чітко відділити помилкові складові сигналу від корисної інформації.

Незважаючи на більшу надійність та легкість роботи цифровий сигнал не може повністю замінити аналоговий. Цифровий сигнал є штучним явищем, він використовується лише в цифрових пристроях. В той час як за допомогою аналогових сигналів можна представити природні явища такі як: температура, звук, світло, тиск та інші. Можна сказати, що реальний світ є “аналоговим”.

Аналогові сигнали через такі явища як шуми, перешкоди чи наведення змінюються відносно початкової форми, тобто зазнають спотворень.

Шуми – це хаотичні слабкі сигнали які виникають в самому електронному пристрої.

Наведення та перешкоди – це паразитичні сигнали що впливають на корисний сигнал ззовні пристрою найпоширеніші серед них це електромагнітні випромінювання від інших пристроїв. Пристрої які викликають наведення, зазвичай, відносяться до категорії пристроїв силової електроніки.

					123.KI-41.11	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		22

Тому при проектуванні систем які працюють з аналоговим сигналом передбачуються які чинники спотворюють сигнал та встановлюють додаткові елементи які компенсують шкідливі явища або відсіюють небажані складові сигналу. Також можуть бути враховані які характеристики та складові притаманні корисному сигналу у конкретному застосуванні.

Ці пристрої виконують функцію обробки сигналу і аналогові системи не можуть обійтися без них.

Мікроконвертори здійснюють обробку аналогових сигналів у цифровому вигляді за допомогою програмних обчислень.

Основними засобами цифрової фільтрації є цифрові фільтри зі скінченною імпульсною характеристикою та нескінченною імпульсною характеристикою (СІХ та НІХ), та дискретне перетворення Фур'є.[5]

Дискретне перетворення Фур'є(ДПФ) є одним з основних методів цифрової обробки сигналів. Дискретне перетворення Фур'є використовується для функцій цифрової фільтрації та спектрального аналізу сигналів.

Для дискретного сигналу який заданий як послідовність значень $S(n)$, пряме й обернене перетворення мають вигляд:

$$S(k) = \sum_{n=0}^{N-1} S(n) \exp \left[-j \frac{2\pi nk}{N} \right], \quad k = \overline{0, N-1} \quad (1.1)$$

$$S(n) = \frac{1}{N} \sum_{k=0}^{N-1} S(k) \exp \left[j \frac{2\pi nk}{N} \right], \quad n = \overline{0, N-1} \quad (1.2)$$

Де k – номер гармоніки із частотою f_k , $N - j$, обсяг вибірки. $S(k)$, визначений як комплексний спектр сигналу, можна виразити

$$S(k) = A(k) - jB(k) = C(k) e^{-j\varphi(k)} \quad (1.3)$$

Де амплітудно-частотна(АЧХ) і фазочастотна(ФЧХ) характеристики сигналу відповідно визначаються

$$C(k) = \sqrt{A^2(k) + B^2(k)} \quad (1.4)$$

$$\varphi(k) = \arctg \frac{B(k)}{A(k)} + 2\pi m \quad (1.5)$$

Обернене перетворення можна також виконувати за допомогою виразу:

$$S(n) = \frac{1}{N} \sum_{k=0}^{N-1} C(k) \cos \left[\varphi(k) + \frac{2\pi}{N} \cdot n \cdot k \right] \quad (1.6)$$

Відновлення сигналів можна покращити додатково використовуючи вагові функції. Тоді вираз матиме наступний вигляд:

$$\hat{S}(n) = \frac{1}{N} \sum_{k=0}^{N-1} W(k) \cos \left(\varphi(k) + \frac{2\pi}{N} \cdot n \cdot k \right) \quad (1.7)$$

, де $W(k)$ – введена вагова функція

Іншим записом ДПФ рівняння є:

$$F(\Delta\xi) = \sum_{k=0}^{N-1} x(k\Delta x) \delta(x - k\Delta \cdot x) e^{-j2\pi k\Delta\xi\Delta x} = k \cdot \sum_{k=0}^{N-1} x_k \cdot e^{-j\frac{2\pi}{N}kn} \quad (1.8)$$

$$\text{де } \begin{cases} \frac{2x_{max}}{N} = \Delta x \\ \frac{1}{2x_{max}} = \Delta \xi \end{cases}$$

З цього можна отримати

$$\Delta x \cdot \Delta \xi = \frac{1}{N} \quad (1.9)$$

Таким же чином можна отримати для оберненого перетворення

$$x_m = k \sum F_n \cdot e^{j\frac{2\pi}{N}nm} \quad (1.10)$$

В матричній формі

$$F = \frac{1}{\sqrt{N}} E_N X \quad (1.11)$$

$$E_N = \begin{bmatrix} W^0 & W^0 & W^0 & \dots & W^0 \\ W^0 & W^1 & W^2 & \dots & W^{N-1} \\ W^0 & W^2 & W^4 & \dots & W^{2(N-1)} \\ W^0 & W^3 & W^6 & \dots & W^{3(N-1)} \\ \dots & \dots & \dots & \dots & \dots \\ W^0 & W^{N-1} & W^{2(N-1)} & \dots & W^{(N-1)(N-1)} \end{bmatrix} \quad (1.12)$$

де $W^k = e^{-j2\pi k/n}$

Представлена матриця називається матрицею дискретних експонентних функцій.

При здійсненні перетворення Фур'є ряд рядки матриці задають набір ортогональних функцій, за допомогою яких виконується розкладання вихідного сигналу. Для перетворення Фур'є, як і для всіх ортогональних перетворень, визначник перетворення E_N завжди є ненульовим, завдяки чому можна здійснювати як пряме так і обернене перетворення.

$$\begin{cases} F = \frac{1}{\sqrt{N}} E_N X \\ X = \frac{1}{\sqrt{N}} E_N^{-1} F \end{cases} \quad (1.13)$$

Обчислення перетворення Фур'є потребує великої кількості множень і обчислень синусів.

ДПФ часто використовується для аналізу спектру сигналу. При цьому часто найбільш цікавими є амплітуди окремих гармонік. У цьому випадку спектр зображується як графік залежності амплітуди від частоти. Для обчислення спектру сигналу необхідно вибрати певний відрізок сигналу. І до обраного відрізка застосовують ДПФ.

При вираховуванні спектру може виникнути небажаний ефект. При розкладанні сигналу в ряд Фур'є вважається, що сигнал є періодичним з періодом який дорівнює розміру ДПФ. На межах таких періодів сигнал може мати розриви. А розриви суттєво впливають на спектр сигналу, спотворюючи його.

Для усунення цього ефекту застосовують “вагові вікна”. Вони зменшують значення сигналу поблизу меж обраного відрізка. Значення сигналу з вибраного для аналізу відрізка множать на вагові вікна. Існує багато вагових вікон, найбільш вживані вікна: Хеммінга і Блекмана.

$$W_{Hamming}(n) = 0.54 - 0.46 \cdot \cos \frac{2\pi n}{N} \quad (1.14)$$

$$W_{Blackman}(n) = 0.42 - 0.5 \cdot \cos \frac{2\pi n}{N} + 0.08 \cdot \cos \frac{4\pi n}{N} \quad (1.15)$$

Також важливим моментом є те що не має єдиного правильного спектру сигналу. Спектр можна обчислювати задаючи різні відрізки ДПФ і різні вагові вікна. Для кожного конкретного випадку потрібно застосовувати свої способи.

Іншими методами обробки сигналів є цифрові фільтри. Цифровим фільтром називають фізичну або програмну реалізацію алгоритму який змінює форму, амплітуду або фазові характеристики вхідного сигналу певним чином. В аналогових системах фільтр це пристрій зі спеціальною частотною характеристикою який перетворює вхідний сигнал на вихідний, підсилюючи або послаблюючи певні частоти в спектрі вхідного сигналу.

Відповідно до цього цифровий фільтр – пристрій що перетворює послідовність значень вхідного сигналу у числову послідовність вихідного сигналу. Для цифрових фільтрів теж існує поняття імпульсної характеристики, що є реакцією на одиничний імпульс.[6]

Імпульсна характеристика цифрового фільтра може бути визначена як результат дискретизації неперервної імпульсної характеристики відповідного аналогового фільтра.

Якщо взяти кінцеве число відліків, тоді можна отримати цифровий фільтр зі кінцевою імпульсною за характеристикою (КІХ)

Якщо взяти нескінченне число відліків, то отримаємо цифровий фільтр з нескінченною імпульсною характеристикою (НІХ).

Якщо в загальному під фільтром розуміють систему, що певні частоти пропускає, а інші затримує, то в цифровій обробці це поняття застосовується

					123.KI-41.11	Арк.
						26
Зм.	Арк.	№ докум.	Підпис	Дата		

більш широко. Дискретним фільтрами називають систему обробки сигналу, яка має властивості лінійності та стаціонарності.

В загальному випадку фільтр змінює в спектрі сигналу, амплітуди гармонік і їхні фази. Проте існують фільтри які не змінюють фазу сигналу. Такі фільтри називаються фільтрами з лінійною фазою. Це означає, що такі фільтри не спотворюють(модифікують) фазу сигналу, а лише зсувають весь сигнал в часі.

Основні характеристики будь-якого фільтра це частотна і фазова характеристики. Вони показують як фільтр змінює фази та амплітуди різних гармонік вхідних сигналів. У фільтрах з лінійною фазою розглядають лише частотну характеристику. Частотна характеристика зображується як графік залежності амплітуди(коэф. підсилення) від частоти. Значення амплітуди 0дб означає, що фільтр пропускає сигнал без змін. Частоти, амплітуди яких послаблюється в двічі, значення амплітуди на характеристиці становить -6 дб. Якщо фільтр є активним, то він може підсилювати деякі частоти і їхнє значення амплітуди буде більшим нуля.

Вихідний сигнал фільтра, що має частотну характеристику яка не задається аналітичними функціями(нетривіальна), залежить від певної кількості відліків вхідного сигналу. Для таких фільтрів рівняння фільтрації наступне:

$$y(k) = \sum_{i=0}^m b_i x(k - i) \quad (1.16)$$

Фільтри такого типу називаються нерекурсивними. Кількість відліків вхідного сигналу(значення m) називається порядком фільтра. Імпульсна характеристика визначається такого фільтра визначається його коефіцієнтами b_i . Оскільки в реальних пристроях кількість відліків(коефіцієнтів) обмежена певним числом, то такі фільтри відносять до КІХ-фільтрів.[7]

Фільтр у якому для фільтрації використовуються не тільки вхідні, але й вихідні відліки, називають рекурсивним. Рівняння фільтрації матиме наступний вигляд:

					123.KI-41.11	Арк.
						27
Зм.	Арк.	№ докум.	Підпис	Дата		

$$y(k) = \sum_{i=0}^m b_i x(k-i) - \sum_{i=0}^m a_i x(k-i) \quad (1.17)$$

Наявність зворотних зв'язків у рекурсивному фільтрі дозволяє одержати нескінченну імпульсну характеристику, тому такі фільтри належать до категорії БІХ-фільтрів. Проте такі фільтри в певних випадках є нестійкими.

Проектування цифрового фільтра полягає у підборі коефіцієнтів які задовільняють вимоги конкретного застосування.

Цифрова фільтрація найчастіше використовується для виділення сигналу та відновлення сигналу. Виділення сигналу полягає у відсіюванні різного роду шумів які надходять із зовнішнього середовища. Відновлення сигналу застосовується у випадку спотворень сигналу які викликані роботою самих пристроїв.

Тому широко застосовуються методи проектування цифрових фільтрів за прототипами аналогових фільтрів. У прямих методах синтезу де не використовується аналогові прототипи припускається, що ДПФ можна вважати як обробку сигналу фільтром з відповідною імпульсною характеристикою.

При створенні цифрових фільтрів один з важливих моментів пов'язаний з особливостями обчислювальних пристроїв та їх обмежень:

- Шум квантування
- Спотворення характеристик при квантуванні коефіцієнтів цифрових фільтрів
- Переповнення розрядної сітки при обчисленнях
- Округлення проміжних результатів

1.4. Області застосування мікроконвертерів

Мікроконвертери завдяки універсальності алгоритмів які вони можуть реалізувати застосовуються в різних областях де потрібна обробка аналогових

					123.KI-41.11	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		28

сигналів. Короткий перелік областей де мікроконвертори можуть ефективно виконувати свої функції наведений нижче.

Одна з основних задач мікроконверторів це – обробка вихідних сигналів давачів. Аналогові сигнали широко використовуються в електронних пристроях для вимірювання фізичних величин.[8]

Для того щоб представити такі фізичні явища як температуру, світло, механічні коливання використовують елементи які здатні реагувати на ці явища та змінювати свої параметри відповідно до їхніх величин. Ці елементи називаються давачами чи сенсорами. Електричний сигнал який йде з сенсорних елементів, зазвичай має малу амплітуду і містить шуми або флуктуації напруги. Тому між сенсорним елементом і виходом ставлять фільтруючі кола які відсіюють деякі шуми і усувають коливання напруги(усереднюють значення) щоб на виході отримати деяке стабільне значення. Мікроконвертори можуть здійснювати функції таких фільтрів, та також змінювати алгоритм обробки залежно від конкретного застосування.

Радіолокація – знаходження координат певного об'єкта, за допомогою радіохвиль. В радіолокаційній системі передавач генерує імпульси тривалістю декілька мікросекунд. Ці сигнали подаються на передавальну антену яка випромінює радіохвилі у повітряний простір. Повітря може відбивати деяку енергію від випроміненої хвилі на приймаючу антену. Коли об'єкт у повітряному просторі буде в зоні видимості радару, то він відбиватиме більше енергії, порівняно з однорідним повітрям і ця різниця буде зафіксована радаром. Відстань до об'єкту визначається різницею в часі між відправленим та прийнятим сигналом. Напрямок до об'єкту визначають виходячи з напрямку передавальної антени. Діапазон дії радіолокаційної системи визначається двома параметрами: енергія початкового сигналу передавача і значенням шуму приймача. Збільшення енергії імпульсу зазвичай здійснюється збільшенням його тривалості, що приводить до зменшення точності. Це відображається на виявленні об'єктів на дальніх відстанях і точності виявлення відстані до предмету. Мікроконвертори можуть покращити виконання різних функцій.

					123.KI-41.11	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		29

Мікроконвертори можуть фільтрувати прийняті сигнали щоб зменшити шум та формувати імпульси різної частоти та форми. Що дозволяє оптимізувати систему для конкретних завдань виявлення.

Смуга пропускання таких сигналів 100кГц до 1МГц. Застосовуються наступні алгоритми: спектральний аналіз, узгоджуюча фільтрація, двовимірне перетворення Фур'є.

Гідролокація – це знаходження місцезнаходження об'єкта за допомогою звукових хвиль. Процес гідролокації має ті самі принципи, що і радіолокація, різниця лише у використанні звукових хвиль. Використання звукових хвиль зумовлено тим, що вода майже або взагалі не пропускає електромагнітні хвилі. Пристрої які виконують гідролокацію називаються – сонарами. Виділяють два типи гідролокації: активна та пасивна. В активному звуковому локаторі, звукові імпульси передаються в водяне середовище, потім відбиті сигнали отримуються та обробляються. Активний сонар може здійснювати функції: виявлення та фіксація об'єктів, навігація, картографування. В пасивній гідролокації активний передавач відсутній, пристрій просто зчитує звуки які створюються іншими об'єктами. Особливість пасивного сонару є те що він не може бути виявлений через те що не посилає ніяких імпульсів у водяний простір. Діапазон імпульсів сонарів варіюється від 2 до 40 КГц. Може здатись, що гідролокаційна система є простішою через те що вона оперує сигналами нижчих частот. Проте сонар є більш складним через те що водяне середовище є менш однорідним та стабільним ніж повітряне.

Функції які може виконувати мікроконвертор в таких системах та методи обробки сигналів є такими самими як і в радіолокації.

Радіозв'язок – забезпечення справної передачі систем зв'язку за рахунок покращених алгоритмів відсіювання шумів, виділення та кодування сигналів. Смуга пропускання – до 100 кГц. Застосовуються алгоритми: спектральний аналіз, перетворення Фур'є, узгоджуюча фільтрація.

Обробка звукових сигналів – це робота з сигналами які отримуються з коливань повітря. Серед задач обробки звукових сигналів покращення якості

					123.KI-41.11	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		30

звукозапису, відсіювання шумів, компенсування спотворень, виділяти певні частини звуку які мають малу амплітуду порівняно з загальни сигналом. Необхідна смуга пропускання: 50 кГц. Використовувані алгоритми: перетворення Фур'є та цифрова фільтрація.

Медичні та біологічні сигнали – обробка сигналів кардіограми та енцефалограм, томографічні дослідження. Зчитування біологічних сигналів
Смуга пропускання таких сигналів до 400кГц.

Висновки до розділу

У даному розділі було описано загальні принципи роботи мікроконверторів та їхніх основних елементів. Також було розглянуто види сигналів та їхні параметри, а також методи обробки сигналів.

					123.KI-41.11	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		31

2. ВИБІР ЕЛЕМЕНТНОЇ БАЗИ ДЛЯ РЕАЛІЗАЦІЇ МІКРОКОНВЕРТОРА

2.1. Вибір Аналого-цифрового перетворювача

Аналогово-цифрові перетворювачі більшості типів мікроконтролерів в основному є ідентичними. Різниця полягає в тому, що є певні деталі, деякі МК крім аналогового потенціалу землі (AGND) відносно аналогових входів мають також і диференціальні входи, за допомогою цього сигнал на вході АЦП може бути представлений у вигляді різниці між двома вхідними каналами. Така різниця може бути додатково помножена (підсилена) на деякий коефіцієнт. Такий режим роботи підходить, зокрема, для вимірювання синфазних напруг. В деяких МК є окреме внутрішнє джерело опорної напруги.

Зупинимось на параметрах і характеристиках АЦП, які є основою формування мікроконверторів для аналого-цифрових сигналів. Такі АЦП мікроконтролерів мають параметри і характеристики:

- До восьми вхідних каналів;
- Можливість працювати в режимі одноразових перетворень або в режимі автономного живлення;
- В більшості мікроконтролерів реалізовано внутрішнє джерело опорної напруги, що робить їх більш універсальними;
- Після завершення перетворення може бути викликане переривання;
- В режимі очікування можна значно підвищити ефективність відсіювання шумів і завад.

Аналого-цифрові перетворювачі працюють за принципом послідовного наближення. Завдяки цьому вдалося досягнути відносно невеликої тривалості перетворення при досить високій точності.

На початку циклу перетворення схема керування АЦП через ключ S схеми вибірки і зберігає подає на внутрішній накопичувальний конденсатор C схеми

					123.KI-41.11	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		32

ПВЗ вихідну напругу, яка вибрана мультиплексором. Потім схема керування залишає ключ S, а напруга на вході залишається на конденсаторі протягом всього перетворення.

Метод зважування Відноситься до типу послідовних. На початку циклу схема управління АЦП записує його у внутрішній регістр число $U_z=1000000000_b$. Це число подається вже на внутрішній ЦАП, який перетворює його в аналогову напругу, що відповідає половині діапазону вимірювання. Компаратор порівнює це значення з аналоговою напругою на вході (U_e), що зберігається в схемі вибірки і зберігання (ПВЗ). Якщо значення вхідної напруги є меншим ніж значення U_z , то старший розряд Z скидається в "0" в іншому випадку в ньому зберігається логічна одиниця. На наступному кроці встановлюється в "1" другий за вагою розряд і це значення знову порівнюється з вхідною напругою за допомогою компаратора, якщо значення вхідної напруги менше то розряд стає "0", а якщо більшим, то залишається "1".

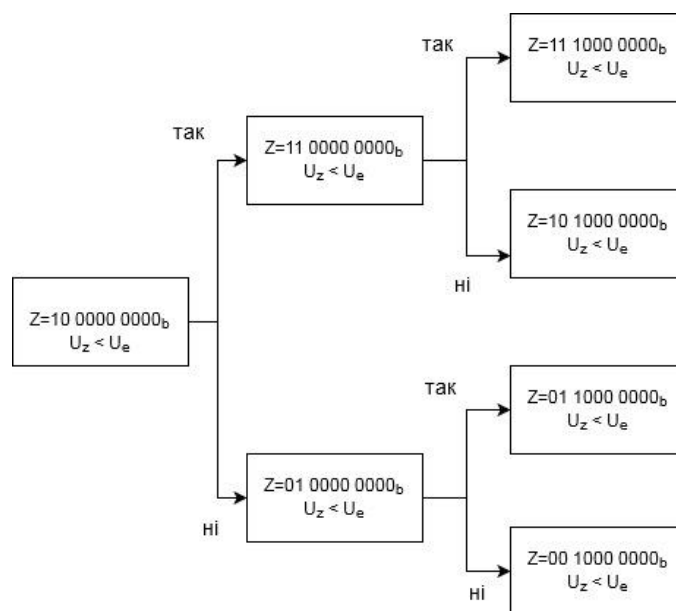


Рис. 2.1 Процес зважування для перших трьох розрядів.

На рис. 2.1 Подані перші 3 кроки. Як можна бачити із цього рисунку, встановлення чи скидання поточного розряду відбувається незалежно від раніше встановлених розрядів.

Ця процедура проводиться для решти 9 розрядів 12-бітного АЦП. В результаті цього перетворення область певного визначеного значення звужується наближаючись до вхідної напруги.

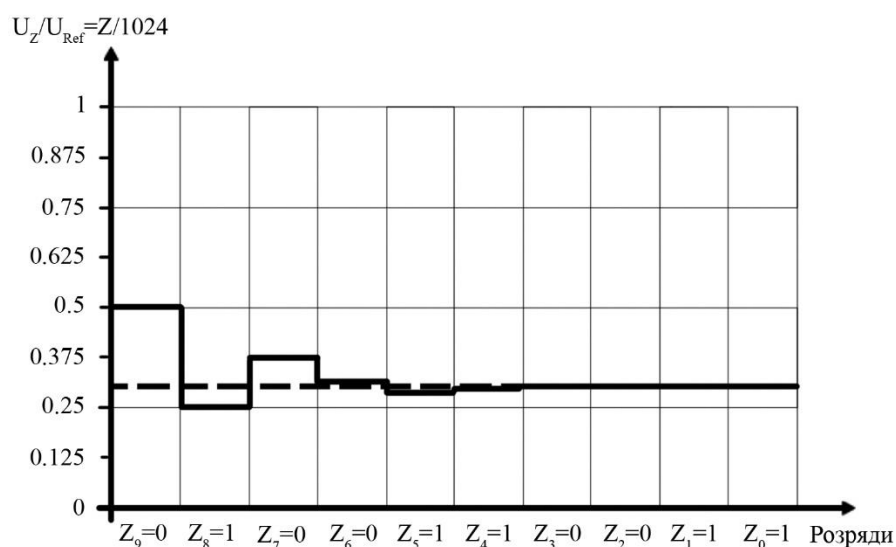


Рис. 2.2 Графічне представлення процесу зважування.

Тут до десяти етапів порівнювання ще додається час необхідний для опитування значення вхідної напруги. Таким чином для отримання результату аналого-цифрового перетворення необхідно тринадцять кроків.

Допустимі тактові частоти ЦАП лежать в діапазоні 50-200 кГц. Необхідна тактова частота встановлюється за допомогою внутрішніх регістрів і подільника АЦП.

АЦП може працювати в режимі одиничного перетворення або в автономному режимі, тобто в неперервному режимі через регістр. Вибір одного з двох режимів здійснюється за допомогою розрядів регістру управління і стану.

В залежності від типу МК в розробленій ВІС може бути представлена різна кількість вхідних каналів. В будь-якому випадку вхідний канал може змінюватись шляхом під'єднання до входу АЦП через мультиплексор MUX. Крім зовнішньої напруги які подаються на вхідні сигнали ADC0 через MUX в полі вхідної величини перетворювача також можуть бути подані величини

AGND аналогового блоку мікроконтролера і внутрішня порогова напруга номіналом 1.23 В, яке використовується для реєстрування провалів напруги на внутрішньому джерелі МК. Саме на його основі і формується внутрішня опорна напруга перетворювача.

Інтегральна схема MCP3201 від Texas Instruments – представляє метод дешевого швидкого і точного конвертування з розрядністю АЦП до 12-біт, і внутрішньою схемою пам'яті, а також з використанням внутрішнього джерела опорної напруги. Мікроконтролери можуть працювати з цією схемою через інтерфейс SPI.[9]

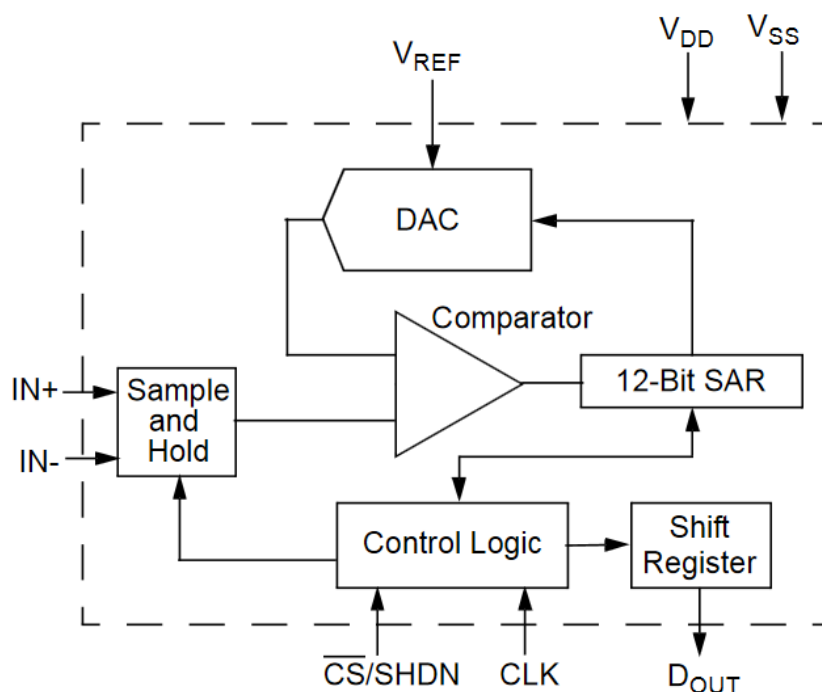


Рис. 2.3 Внутрішня структура MCP3201 від компанії Microchip.

Розглянемо параметри і характеристики MCP3201:

- роздільна здатність 12 біт;
- до 150000 перетворень в секунду;
- послідовний інтерфейс SPI;
- споживання струму при роботі до 400 мкА;

- споживання струму в режимі очікування до 2 мкА;
- температурний діапазон роботи від -40 °С до +85 °С;
- внутрішнє джерело опорної напруги + 2.5В;
- працює на уніполярні напрузі живлення від 2.7 до 5 В;

Так само як і вбудований АЦП мікроконтролерів, мікросхема МСР3201 працює принципом послідовного наближення. В кристал інтегровані необхідні для цього пристрій програмного управління, цифро-аналоговий перетворювач і компаратор.

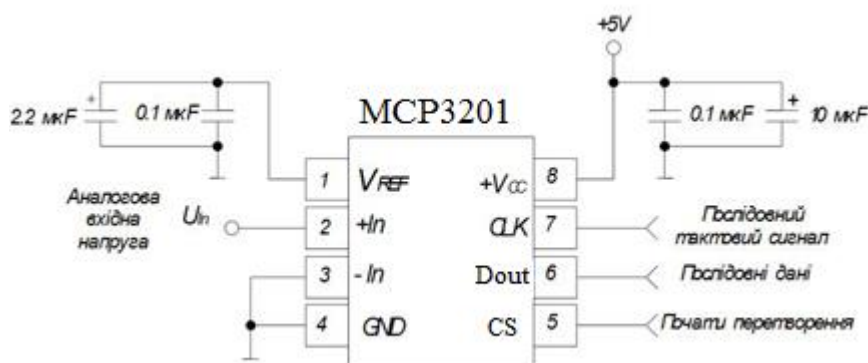


Рис. 2.4 Підключення мікросхеми МСР3201 до мікроконтролера.

Типова схема підключення мікросхеми МСР3201 до мікроконтролера представлена на рис. 2.4. Вхідна аналогова напруга VIN навантажується ємнісним струмом конденсатора внутрішньої схеми вибірки і зберігання. Коли конденсатор виявляється повністю зарядженим, вхідний струм МСР3201 майже нульовий, оскільки вхідний опір аналогової частини перевищує 1ГΩ. Джерело вхідної напруги повинне забезпечувати конденсатор схеми вибірки і зберігання ємнісним струмом такого рівня, щоб він був у стані зарядити цей конденсатор протягом часу вибірки 10мкс (у найгіршому випадку, коли продуктивність складає 100 000 перетворень в секунду).

Вхід CS забезпечує перемикання між станом вибірки і збереження, а також використовується для подачі сигналу про початок перетворення (рис. 4.1). Для

здійснення перетворення пристрою MCP3201 потрібний тактовий сигнал CLK який йде від мікроконтролера, що синхронізує послідовну передачу даних, а також визначає швидкість перетворення. Останнє обумовлено тим, що сигнал CLK задає робочу частоту пристрою керування послідовним наближенням. Надсилає послідовно вихідний сигнал одразу ж являє результат поточного перетворення. Частота сигналу CLK знаходиться в діапазоні 200кГц. Коефіцієнт заповнення некритичний до тих пір, поки тривалість імпульсу високого або низького рівня більше 50 нс. Тактова частота не повинна бути менше 200 кГц або ж період $T=1/f$ не повинен перевищувати 5мкс, оскільки в даному випадку падіння напруги на внутрішньому конденсаторі схеми вибірки і зберігання, обумовлене струмами витоку, може призвести до спотворення результатів.

12-тирозрядний результат аналого-цифрового перетворення по спадаючому фронту сигналу CLK виводиться через вихід DOUT і може бути отриманий мікроконтролером за наростаючим фронтом наступного імпульсу CLK. При цьому перша передається старший значущий розряд (MSB). Мікросхема MCP3201 має ряд властивостей, що виділяють її серед інших АЦП:

- перехід зі стану вибірки в стан зберігання відбувається за спадаючий фронт сигналу CS і не залежить від тактового сигналу CLK;
- тактовий сигнал CLK не обов'язково повинен подаватися безперервно в часі всього процесу перетворення, що дозволяє записувати 12 біт результату в два етапи за допомогою восьми розрядного апаратного інтерфейсу SPI мікроконтролера;
- після завершення перетворення сигнал CLK може залишатися як на рівні лог. 0, так і на рівні лог. 1.

2.2. Вибір цифро-аналогового перетворювача

Для відтворення аналогової форми вже обробленого сигналу потрібен цифро-аналоговий перетворювач. Цифро-аналоговий перетворювач в цьому

					123.KI-41.11	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		37

пристрої повинен мати таку саму роздільну здатність як і АЦП, інакше було б необхідно здійснювати масштабування значень засобами мікроконтролера. Також ЦАП повинен мати не меншу частоту перетворень ніж АЦП, інакше ЦАП буде обмежувати швидкодію системи.

Для цієї роботи був обраний цифро-аналоговий перетворювач МСР4725 компанії Microchip.

Його характеристики і параметри є наступними:

- роздільна здатність 12 біт;
- час встановлення ЦАП 6 мкс;
- I2C інтерфейс з стандартним(100 кбіт/с) та швидким(400 кбіт/с) режимами роботи;
- доступні для вибору вісім адрес I2C;
- вбудована енергонезалежна пам'ять
- зовнішня опорна напруга;
- доступний режим очікування в якому знижується енергоспоживання;
- температурний діапазон напруги від -40 °С до +125 °С;
- компактний корпус SOT-23 з шістьма виводами;

МСР4725 це енергоефективний, високоточний, одноканальний, 12-бітний цифро-аналоговий перетворювач з енергонезалежною пам'яттю EEPROM.[10]

Як можна бачити з вище наведених характеристик вибраний ЦАП має схожі характеристики з АЦП(МСР3201) тому він не буде обмежувати швидкодію пристрою чи потребувати додаткових адаптацій. Наявність різних інтерфейсів в АЦП та ЦАП не є недоліком, навпаки таке розділення дозволяє зменшити навантаження на одну шину і усунути конфліктні ситуації коли два пристрої підключені до одної шини.

Входи ЦАП та конфігурація його роботи здійснюється через енергонезалежну пам'ять EEPROM через команди I2C інтерфейсу. Наявність вбудованої енергонезалежної пам'яті дозволяє пристрою ЦАП зберігати вхідний код протягом часу коли живлення відключене і після включення живлення вихідна напруга встановиться моментально. Ця властивість є корисною якщо

					123.KI-41.11	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		38

ЦАП використовується як підтримуючий пристрій для інших пристроїв в мережі.

MCP4725 має вивід A0 для зовнішньої зміни I2C адреси. Вивід може бути під'єднаний до високого або до низького лог. рівня.

Інтерфейс MCP4725 це послідовний інтерфейс I2C який використовує дві лінії передачі: SDA та SCL.

Даний ЦАП добре підходить у випадках де бажані простота пристрою та висока компактність, та для випадків де для використання ЦАП його параметри роботи повинні зберігатись між виключеннями живлення пристрою.

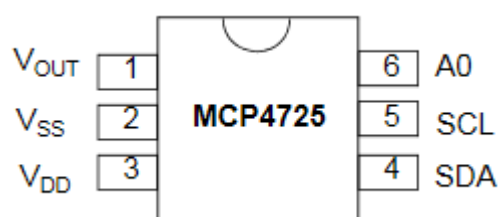


Рис. 2.5 Розміщення виводів MCP4725.

На рис.2.5 показано розміщення виводів пристрою ЦАП.

V_{OUT} – це вихідна аналогова напруга пристрою, вона її діапазон визначається значенням на входах V_{DD} та V_{SS} .

V_{DD} та V_{SS} – виводи призначені для живлення пристрою. V_{DD} використовується для підключення вхідної напруги і як опорна напруга ЦАП. Тому джерело живлення на цьому виводі повинне мати якомога менше шумів для високої ефективності ЦАП. Для цього до виводу необхідно під'єднати конденсатор ємністю 0.1 мкФ який йде на землю. Рекомендується підключити додатковий конденсатор паралельно для того щоб прибрати високочастотні шуми. Напруга живлення(V_{DD}) повинна бути в межах від 2.7 до 5.5 В.

V_{SS} вивід який використовується для підключення землі і повернення струму від пристрою. При застосуванні V_{SS} повинен бути підключеним до земляної шини через з'єднання з низьким імпедансом.

SDA – інформаційний вивід інтерфейсу I²C. SDA використовується для зчитування або запису в регістри ЦАП або EEPROM. Вивід SDA є N-канальним ключем з відкритим стоком. Тому потребує підтягуючого резистора з VDD лінії до SDA. Крім станів START і STOP, дані на SDA виводі повинні бути стабільними протягом періоду тактуючого сигналу. Зміна стану SDA з низького на високий лог. рівень або навпаки лише коли стан лінії SCL є низьким.

SCL – тактуючий вивід інтерфейсу I²C. MCP4725 може діяти лише як другорядний пристрій(Slave) тому вивід SCL може приймати лише зовнішні тактуючі сигнали. Вхідні дані з головного пристрою надходять на вивід SDA на передньому фронті сигналу SCL, а вихідні дані з MCP4725 відправляються на задньому фронті сигналу SCL. Вивід SCL є N-канальним ключем з відкритим стоком. Тому потребує підтягуючого резистора з V_{DD} лінії до SCL.

A0 – використовується для вибору першого біта I²C адресу пристрою. Може бути підключеним до V_{DD} або V_{SS} виводів.

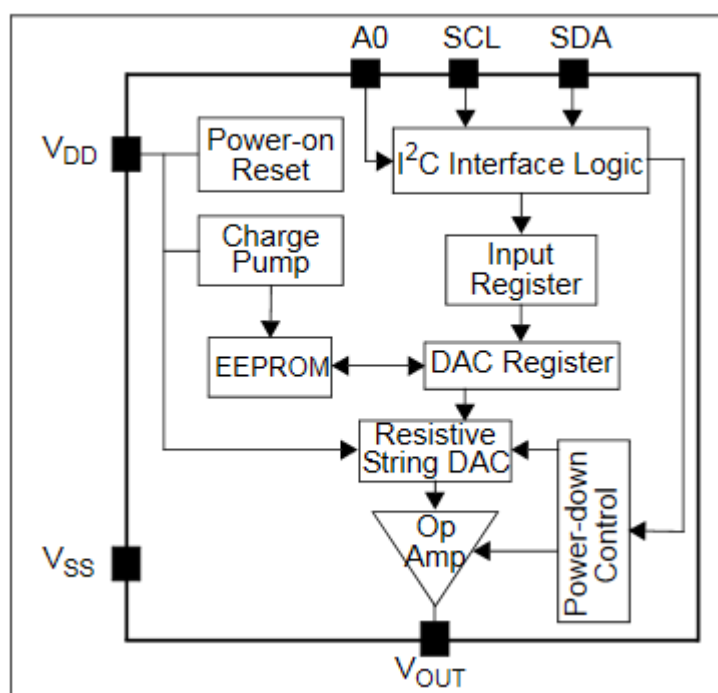


Рис. 2.6 Структура MCP4725.

Структура пристрою ЦАП та його внутрішні елементи зображена на рис. 2.6.

Power-on reset – цей блок забезпечує включення пристрою у визначеному стані. Якщо напруга живлення менша ніж порогова напруга V_{POR} (зазвичай 2 В), то всі елементи будуть відключені вихідної напруги на ЦАП не буде. Коли напруга живлення перевищить порогове значення V_{POR} пристрій перейде в стан відновлення. Під час періоду відновлення, пристрій відвантажує конфігураційні та вхідні коди ЦАП з EEPROM. Це дозволяє пристрою повернутись до стану в якому він був до відключення живлення.

EEPROM – 14 бітна пам'ять для збереження конфігураційних даних (2 біти) та вхідного коду ЦАП (12 біт). Ці дані можуть зчитуватись та перезаписуватись I²C інтерфейсом.

Charge Pump – модуль який підвищує напругу для перезапису даних в EEPROM не використовуючи зовнішню напругу програмування.

I²C Interface logic – модуль який здійснює роботу з I²C інтерфейсом. Приймає вхідні сигнали та передає їх на внутрішні регістри за допомогою яких здійснюється керування пристроєм. Або після запиту від головного пристрою відсилає дані свого стану.

DAC register – 12 бітний регістр який тримає в собі вхідний код ЦАП. Та його дані можуть бути продубльовані в EEPROM залежно від вхідної команди I2C. Вихід цього регістра подається на ланцюгову схему ЦАП.

Resistive string DAC – це ланцюгова схема резисторів де кожному біту перетвореного двійкового коду відповідає резистор чи джерело струму, підключене до спільної точки додавання. Сила струму джерела (провідність резистора) пропорційна вазі біта, якому він відповідає. Таким чином, всі ненульові біти коду додаються з вагою.

Op Amp – це вихідний операційний підсилювач ЦАП. Він має малу напругу зміщення та низький шум. Він приймає на вхід значення струму з ланцюгової схеми резисторів та перетворює її в вихідну напругу. Цей підсилювач може

					123.KI-41.11	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		41

керувати навантаженнями з високими опорами та ємністю без осциляцій.
Максимальний вихідний струм підсилювача 25мА

Power down Control – ця схема переводить пристрій ЦАП в режим зниженого енергоспоживання. Вона відключає майже всі внутрішні елементи крім І²С інтерфейсу.

Кількість рівнів вихідної напруги цього пристрою становлять – 4096.

Вага наймолодшого розряду(різниця між двома сусідніми рівнями) при опорній напрузі 5В буде становити 1.22 мВ, а при напрузі 3 В – 0.73 мВ.

Час встановлення є меншим ніж 10 мкс, зазвичай, дорівнює 6 мкс.

І²С адреса пристрою це перший отримуваний байт після якого ЦАП може почати комунікацію якщо його адреса співпадає. Перші 4 біти позначають код пристрою МСР4725 і завжди дорівнюють 1100. Після цього йдуть три біти А2, А1, А0 які програмовані наступним чином. Біти А2 та А1 встановлюються виробником і кінцевий користувач не може їх змінити. Вибір цих бітів може бути зроблений під час замовлення пристрою ЦАП. Біт А0 може бути змінений підключенням відповідного виводу мікросхеми А0 до лог. 1 або 0. В цьому випадку 3 біти адреси можуть бути змінені, що дозволяє підключити 8 пристроїв ЦАП до одного І²С інтерфейсу, або можливість змінити адресу яка б не конфліктувала з іншими пристроями.

2.3. Вибір мікроконтролера

В якості елемента який здійснює обчислення та контроль периферії було вибрано мікроконтролер stm32f103c8.

Його характеристики і параметри є наступними:

- Архітектура мікропроцесора ARM Cortex-M3
- максимальна тактова частота 72 МГц
- апаратні реалізації операцій множення та ділення які виконуються за 1 цикл
- до 128 кбайт ROM пам'яті

					123.KI-41.11	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		42

- 20 кбайт RAM пам'яті
- Режими низького енергоспоживання
- Внутрішній годинник реального часу
- 7-канальний DMA(Direct Memory Access) контролер
- Режим відладки через інтерфейси SWD та JTAG.
- До 2 інтерфейсів SPI, в залежності від модифікації
- До 2 інтерфейсів I2C, в залежності від модифікації
- Інтерфейс USB 2.0
- Обчислювальний модуль CRC
- 2 аналого-цифрові перетворювачі
- 26/37/51/80 портів входу/виходу загального призначення

Серед цих характеристик, особливо, важливі для застосування в мікроконверторах можна виділити наступні.

Максимальна тактова частота є параметром який визначає швидкодію системи, те настільки швидко мікроконтролер здатен приймати, передавати дані на периферію та здійснювати обчислення. Частота 72 МГц є достатньо високою щоб здійснювати обробку великої кількості сигналів та не буде обмежувати швидкодію вибраних АЦП та ЦАП.

Апаратна реалізація операції множення яка виконується за 1 цикл дозволяє суттєво підвищити швидкість виконання дискретного перетворення Фур'є де необхідна кількість операцій множення є квадратичною залежністю. Апаратна реалізація операції ділення яка теж виконується за 1 цикл може використовуватись у методах обробки сигналів які здійснюють "усереднення" значення сигналу з кількох вибірок.

Наявність декількох I²C та SPI інтерфейсів, що робить доступними багато зовнішніх пристроїв для підключення до мікроконтролера. Такі пристрої можуть бути введені для додаткових функцій(дисплеї, додаткова периферія) або ж для основної функції обробки сигналу. Більшість зовнішніх АЦП та ЦАП підтримують один з цих інтерфейсів, що дозволяє підібрати відповідні модулі

					123.KI-41.11	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		43

АЦП/ЦАП або зробити заміну з малими затратами обравши модуль з таким самим інтерфейсом.

Доступні 3 16-бітні таймери загального призначення які можуть використовуватись для часових переривань та системний 24-бітний таймер SysTick.[11]

Також в складі мікроконтролера доступний модуль DMA(Direct Memory Access). DMA контролер є окремим модулем який може здійснювати маніпуляції з даними незалежно від мікропроцесора. DMA може передавати дані в регістри інтерфейсів(SPI, I²C, UART) з пам'яті мікроконтролера або навпаки. Наприклад передача даних з АЦП в RAM пам'ять мікроконтролера, мікроконтролер в цьому випадку не здійснює роботи з периферією і працює лише зі значеннями в пам'яті. Іншим використанням DMA може бути оновлення зображення дисплею. В такому випадку DMA здійснює відвантаження даних з певною частотою в інтерфейс SPI чи I²C де підключений дисплей. Мікроконтролер повинен зберігати дані всього дисплею, що займає більше часу проте він може працювати тільки з даними відображення і не витрачати ресурси для передачі даних. В деяких випадках використання DMA дозволяє підвищити швидкодію системи, розвантаживши мікропроцесор, або й здійснюючи передачу даних зі швидкостями яких мікроконтролер не може досягти.

З огляду на вище перераховані характеристики, вибраний мікроконтролер відповідає вимогам та особливостям застосування в мікроконверторах.

Висновки до розділу

В даному розділі було проведено аналітичний вибір основних елементів мікроконвертора, а також обґрунтовано причини відповідності вибраних елементів. Здійснено опис характеристик, структури та особливостей елементів.

					123.KI-41.11	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		44

3. РОЗРОБКА МІКРОКОНВЕРТОРА

В цьому розділі описано підключення основних елементів, створення принципової електричної схеми, а також налаштування та алгоритми роботи пристрою. Принципова електрична схема була зроблена в програмі KiCAD. Налаштування мікроконтролера було здійснене в програмі CubeMX, а програма мікроконтролера була написана з використанням бібліотек HAL (Hardware Abstraction Library) від STMicroelectronics.

3.1. Підключення основних елементів до мікроконтролера

В цьому приладі мікроконтролер є центральним елементом до якого підключаються АЦП та ЦАП, та який виконує керування іншими модулями. Раніше вибраний мікроконтролер stm32f103c8, виготовляється у форматі QFP та має 48 виводів.

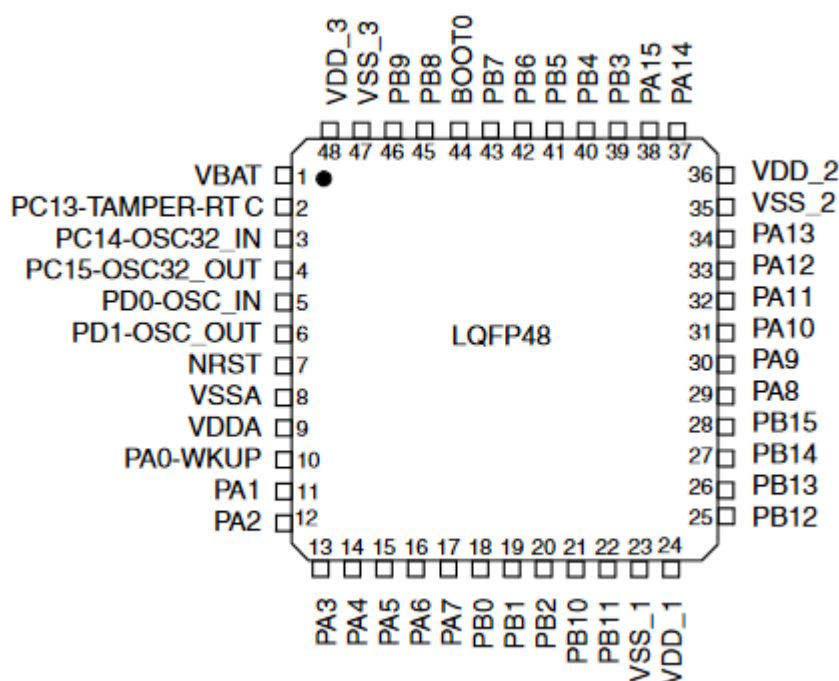


Рис. 3.1 Вигляд мікросхеми мікроконтролера stm32f103c8 та позначення його виводів.

Мікроконтролер не може працювати як самостійна мікросхема без додаткових компонентів. Нижче описані мінімальний набір елементів які дозволяють запуснути мікроконтролер в робочому режимі.

Для стабільної роботи мікроконтролера живлення подається через лінійний стабілізатор живлення АМ1117. Вихідна напруга цього регулятора становить 3.3В, а максимальний струм становить 1А. Він усуває різні зміщення та коливання напруги які можуть привести до нестабільної роботи пристрою.

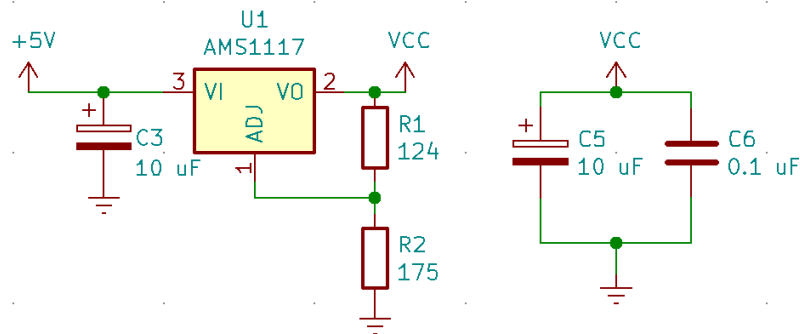


Рис. 3.2 Підключення лінійного регулятора живлення.

Задання швидкості роботи мікроконтролера та синхронізація всіх його компонентів здійснюється за допомогою тактового сигналу. Джерело тактового сигналу може бути внутрішнім або зовнішнім. Внутрішнім джерелом тактового сигналу виступають, RC-ланки мікроконтролера які можуть генерувати коливання з частотою 8МГц. Зовнішнім джерелом тактового сигналу, зазвичай, виступає кварцовий резонатор частотою 4-16 МГц. Зовнішнє джерело тактового сигналу є більш стабільним та точним, тому в цій роботі буде використано цей підхід. Мікроконтролер приймає зовнішній тактуючий сигнал на виводи PC14(OSC32_IN) та PC15(OSC32_OUT). Після цього сигнал надходить на входи внутрішніх подільників та змішувачів сигналів які змінюють частоту відповідно до конфігурації мікроконтролера. Для стабільності до кварцового генератора підключено два конденсатори.

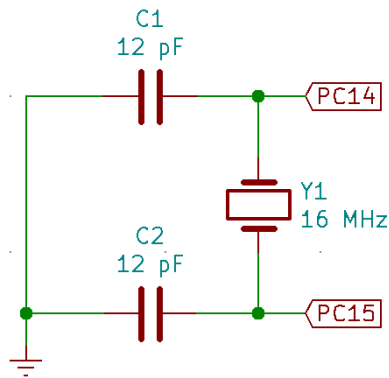


Рис. 3.3 Підключення кварцового резонатора.

Для правильного початку роботи мікроконтролера керуючий сигнал Reset повинен надходити тоді коли живлення вже подане на всі внутрішні блоки мікроконтролера. Для цього до виводу Reset підключають RC-ланку яка затримує сигнал.

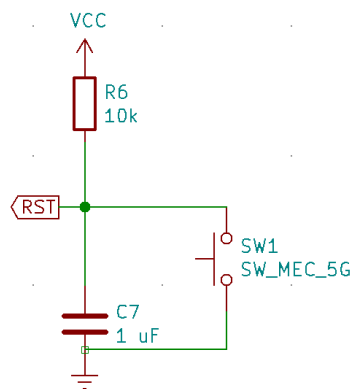


Рис. 3.4 Підключення затримуючої RC-ланки до виводу Reset.

Цих елементів достатньо щоб мікроконтролер міг працювати і переходити в робочий режим при подачі на нього живлення. Наступним кроком є підключення пристроїв АЦП, ЦАП.

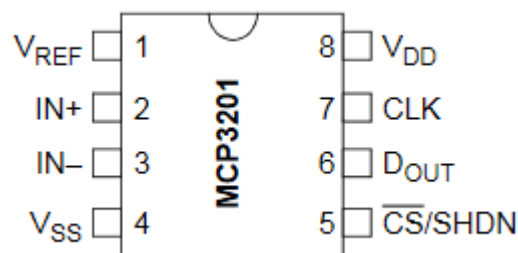


Рис. 3.5 Вигляд мікросхеми АЦП MCP3201 та позначення виводів.

MCP3201 використовує інтерфейс SPI для взаємодії з іншими пристроями. Тактуючий вивід CLK підключається до виводу PA5 мікроконтролера. Вивід CS буде підключеним до виводу PA4, а DOUT(MISO) до PA6. MCP3201 не має виводу який призначений для отримання даних з головного пристрою по SPI. Тому тут замість отримання запиту на дані, АЦП починає відправляти дані після того як напруга з входу CS перейде з високого рівня на низький. Виводи VDD та VSS підключаються до шини живлення(VCC) та шини землі(GND) відповідно. Вивід опорної напруги VREF підключається також до шини VCC відносно якої буде порівнюватись напруга. На додатній диференціальний вхід буде надходити вхідний аналоговий сигнал. Оскільки вхідний сигнал не є диференційним, від'ємний диференціальний вхід IN- буде підключений до землі. Також, виробник рекомендує під'єднати резистор та конденсатор до виводу DOUT.

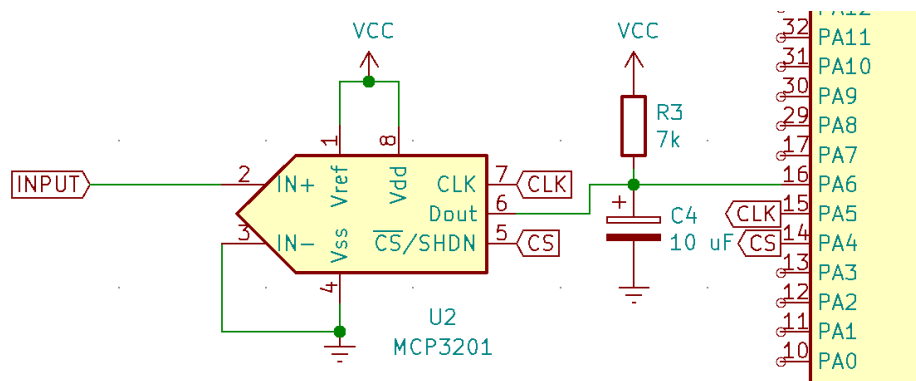


Рис. 3.6 Підключення АЦП до мікроконтролера.

ЦАП MCP4725 для отримання/передачі даних використовує інтерфейс I2C.

Вивід SCL підключається до виводу PB6, а SDA до виводу PB7 мікроконтролера. Виводи VDD та VSS підключаються до шини живлення(VCC) та шини землі(GND) відповідно. Опорна напруга яка визначатиме макс. значення вихідної напруги та різницю напруги між двома сусідніми рівнями, береться з виводу VDD. На виводі DOUT буде вихідна напруга ЦАП.

Оскільки, виводи SDA та SCL є виходами з відкритими колекторами, тому вони потребують підтягуючих (pull-up) резисторів номіналом 1-10 кОм.

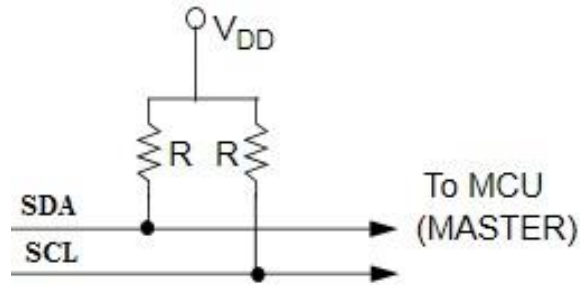


Рис 3.7 Схема підключення підтягуючих резисторів до ліній SDA і SCL.

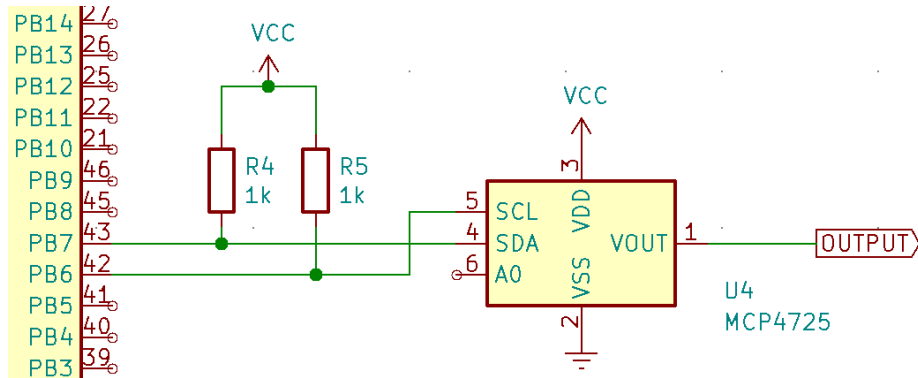


Рис. 3.8 Підключення ЦАП до мікроконтролера.

Повна електрична схема пристрою виглядатиме наступним чином:

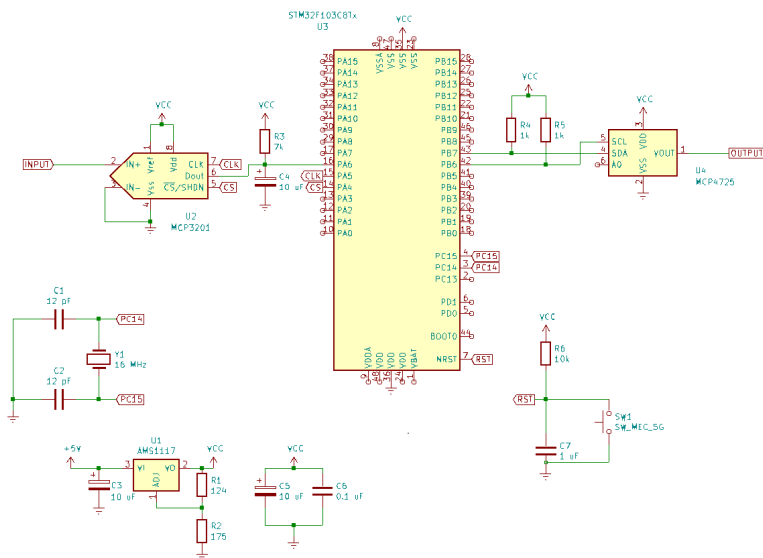


Рис. 3.9 Повна електрична схема.

3.2. Розробка програми та опис роботи пристрою.

Отримання даних з АЦП

Першим кроком обробки сигналу є отримання значення вхідного сигналу від АЦП. АЦП підключений до інтерфейсу SPI1 мікроконтролера. MCP3201 не

має виводу який призначений для отримання даних чи запитів. Тому початок передачі даних починається в момент коли рівень сигналу CS переходить з високого рівня на низький. Після цього мікроконтролер може зчитувати послідовно дані з АЦП на передньому фронті тактуючого сигналу CLK. Після цього сигнал CS переводиться з низького на високий рівень.

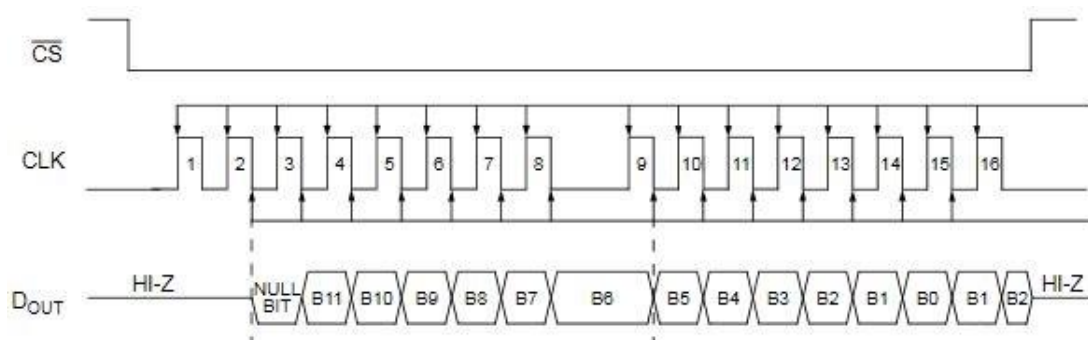


Рис 3.7 Діаграма передачі даних MCP3201.

Оскільки АЦП є 12-бітним, а зчитування даних по SPI відбувається пачками по 8-біт, тому мікроконтролер зчитує 16-біт і потім застосовує маскування для видалення старших чотирьох біт. Зчитування SPI відбувається за допомогою виклику функції “HAL_SPI_Receive(&hspi1, &spi_buf, 2, 100)”.

Передача даних на ЦАП

Модуль ЦАП, MCP4725, підключений до шини I2C мікроконтролера. Початок передачі через шину I2C починається коли рівень сигналу SDA переходить з високого рівня на низький при високому рівні SCL – цей перехід називається початковою умовою. Далі дані можуть послідовно передаватись по переключенню тактового сигналу, стан на лінії SDA може змінюватись при низькому рівні на лінії SCL. Для закінчення передачі даних сигнал на рівні SDA переходить з низького стану у високий при високому стані SCL – це і є кінцевою умовою передачі.

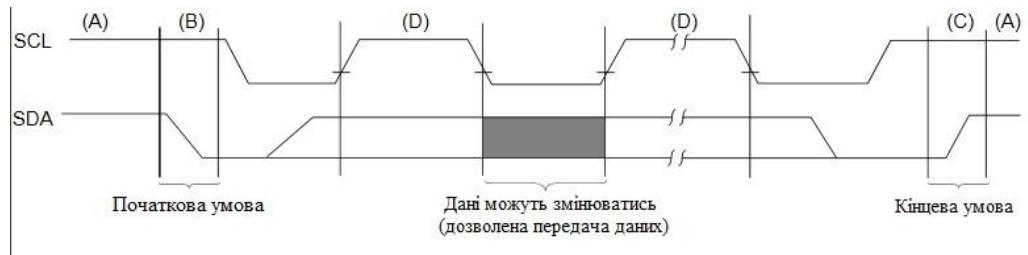


Рис. 3.8 Діаграма роботи шини I²C.

Для передачі коду вихідної напруги на вхід ЦАП, мікроконтролер спочатку посилає байт в якому 7-біт означають адресу пристрою і біт який позначає тип команди запис/зчитування. У випадку цієї роботи, адреса пристрою становить 0x60. Далі йдуть 2 байти передачі даних. В другому байті, перші два біти повинні бути нулями для режиму швидкої передачі. Далі йдуть два біти які визначають чи ЦАП працює в активному чи відключеному режимі. Останні 4 біти другого байту та третій байт – це 12-ти бітний вхідний код ЦАП.

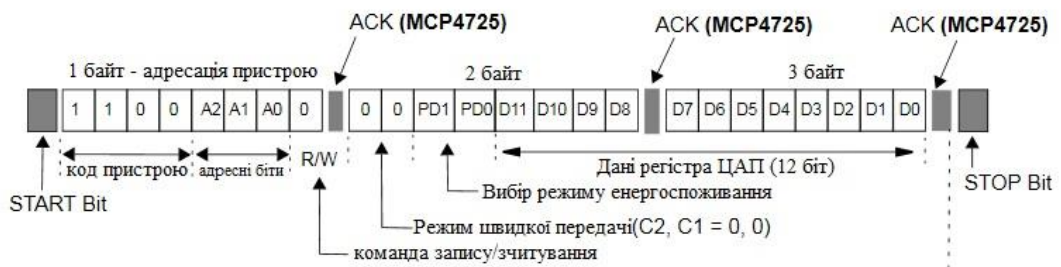


Рис. 3.8 Діаграма передачі даних MCP4725.

Передача даних на I2C відбувається через виклик функції “HAL_I2C_Master_Transmit(&hi2c1, mcp4725_address, buffer, 3, 1000);”, де buffer – це двобайтовий масив, mcp4725_address – I2C адреса mcp4725 (0x60), &hi2c1 – вказує на використання шини I2C1.

Базова структура програми

Маючи функції отримання даних з АЦП, та виведення даних на ЦАП можна створити базову програму яка визначатиме основний алгоритм роботи пристрою.



Рис. 3.9 Діаграма загального алгоритму роботи пристрою.

Діаграма загального алгоритму роботи пристрою показана на рис. 3.9. Тут послідовність запускається від переривання таймера який налаштований на частоту 10 кГц. Після чого виконується отримання даних з АЦП, їхня обробка та передача оброблених даних на ЦАП. Алгоритм обробки може бути різним залежно від потреб.

Базовою програмою мікроконвертора може бути пряма передача даних без обробки. Тобто блок обробки в такому випадку буде пустим. Вихідний сигнал буде повторювати форму вхідного сигналу з деякою затримкою.

Решта програм будуть походити від цієї програми і відрізнятися лише блоком обробки даних.

Фільтр низьких частот

Один з методів обробки є програмний фільтр низьких частот(ФНЧ). Такий фільтр використовується для згладжування показів різних датчиків та сигналів керування. Щоб згладити невеликі коливання використовують ФНЧ на основі інтегратора який виглядає наступним чином:

$$val = (1 - K) \cdot val + K \cdot val_{raw} \quad (3.1)$$

Тут val – оброблений фільтром сигнал, val_{raw} – дані отримані з АЦП, K – коефіцієнт фільтра який варіюється в межах від 0 до 1.

Коефіцієнт K визначає які коливання є занадто великими і потребують згладження, а також які будуть пропускатись з мінімальними змінами. Чим більше значення K , тим слабшим є фільтр і тим більші коливання він може пропустити.

При виборі значення K потрібно визначити які зміни сигналу є корисною інформацією, а які будуть вважатися шумом. Підбір цього значення можна виконати за допомогою наступного рівняння:

$$t = (1 - K) \cdot \frac{dt}{K} \quad (3.2)$$

де t – період часу який відділяє занадто швидкі зміни від потрібних; dt – час ітерації.

Наприклад якщо $K=0.1$, а час між двома вимірами (зчитування АЦП) $dt=20$ мс, то час $t=(1-0.1)*0.02/0.1=0.18$ с. Тобто всі зміни сигналу які тривають менше 0.18 секунд будуть відсіюються.

Друге значення часу t – це відставання обробленого фільтром сигналу від початкового.

Фільтр ковзного середнього

Фільтр ковзного середнього є широкоживим та простим фільтром зі скінченною імпульсною характеристикою. Результат фільтрації такого фільтра це середнє арифметичне останніх N значень вхідного сигналу. Рівняння фільтра виглядає наступним чином:

$$y_n = \sum_{k=0}^{N-1} x_{n-k} \quad (3.3)$$

В розгорнутому виді , для $N=5$:

$$y_n = \frac{1}{5} (x_n + x_{n-1} + x_{n-2} + x_{n-3} + x_{n-4}) \quad (3.4)$$

Такий фільтр має широке застосування завдяки своїй простоті, проте найголовнішою його особливістю – лінійна фазо-частотна характеристика і, відповідно, постійний час затримки сигналу по всій полосі частот. Цей фільтр

змінює амплітудний спектр не чіпаючи фазовий, що робить його зручним при використанні в системах регулювання. Фільтр ковзного середнього, завдяки своїй лінійній перехідній характеристиці, широко застосовується при лінійній інтерполяції, передискретизації сигналу і.т.д.

Головний недолік такого фільтра це обчислювальні затрати, які пропорційні розміру фільтра N . Для вирішення цієї проблеми існує рекурсивна версія фільтра. Тобто фільтр який має такі самі властивості, але реалізований за рекурсивною схемою.

Рекурсивна реалізація фільтра виходить зі звичайної форми.

Як було вище сказано повний вираз значення фільтра для $N=5$ виглядає так:

$$y_n = \frac{1}{5}(x_n + x_{n-1} + x_{n-2} + x_{n-3} + x_{n-4}) \quad (3.5)$$

А вираз попереднього значення:

$$y_{n-1} = \frac{1}{5}(x_{n-1} + x_{n-2} + x_{n-3} + x_{n-4} + x_{n-5}) \quad (3.6)$$

Якщо відняти від першого рівняння друге, то отримаємо:

$$y_n = y_{n-1} + \frac{1}{5}(x_n - x_{n-5}) \quad (3.7)$$

Для довільного розміру фільтра N загальний вираз матиме вигляд:

$$y_n = y_{n-1} + \frac{1}{N}(x_n - x_{n-N}) \quad (3.8)$$

Така реалізація фільтра має такі самі характеристики як і не рекурсивний фільтр, тобто вони є ідентичними. В програмній реалізації через цілочисельну арифметику при діленні на N виникає втрата значущих розрядів. Для уникнення цього явища, з рівняння вилучається операція ділення через домноження обох сторін рівняння на N , а ділення виконується в самому кінці.

Медіанний фільтр

Будь-який лінійний фільтр створений, щоб пропускати сигнали в заданій полосі частот і затримувати інші. Проте такі фільтри не можуть фільтрувати короткотривалі потужні викиди(скачки) значень сигналу.

Такий тип шуму виникає від якогось випадкової події: електростатичний розряд, наведення від включення близьких пристроїв, і.т.д. Наприклад серед даних отриманих з АЦП: 500, 511, 700, 520, 508, 512, значення 700 є помилковим. При використанні звичайного фільтра велике хибне число має великий вплив на вихідний сигнал. В такому випадку застосовується медіанний фільтр.

Медіанний фільтр – це фільтр який пропускає середнє значення з масиву(вікна) значень сигналу. В загальному, медіана це значення яке знаходиться посередині відсортованого масиву. У випадку якщо розмір парний, то медіаною є середнє арифметичне двох середніх величин. Зазвичай, розмір вікна вибирається непарним, щоб уникнути додаткових обчислень.

За реалізацією медіанні фільтри можна розділити на два типи:

- Розмірність = 3
- Розмірність > 3

Реалізація медіанного фільтру розмірністю три виділена окремо, через те що три – це найменша розмірність та знаходження результату може здійснюватись лише за допомогою конструкцій розгалуження if(якщо).

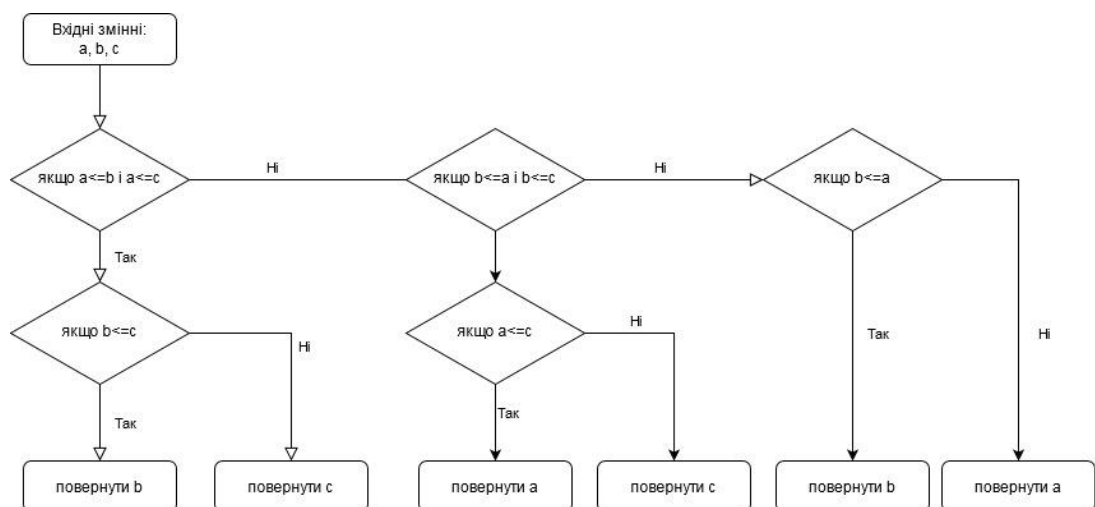


Рис. 3.11 Блок схема алгоритму медіанного фільтра

Зм.	Арк.	№ докум.	Підпис	Дата

для розмірності 3.

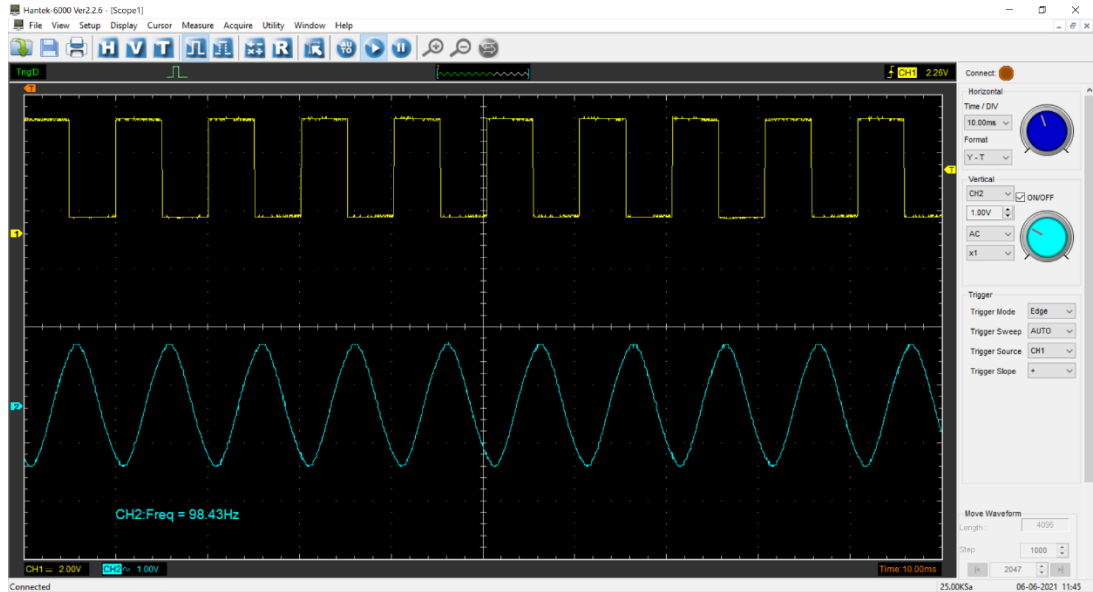


Рис. 3.12 Осцилограма роботи пристрою.

Трьохланковий фільтр низьких частот.

Висновки до розділу

У поточному розділі було послідовно описано реалізацію мікроконвертора. Було створено принципову електричну схему та за нею створено прототип пристрою на макетній платі. Розроблено програму пристрою яка включає: конфігурацію роботи мікроконтролера; функції зчитування даних з АЦП та передачі даних на ЦАП; основний цикл роботи пристрою; реалізація алгоритмів обробки даних.

Зм.	Арк.	№ докум.	Підпис	Дата

123.KI-41.11

Арк.

56

4. ЕКОНОМІЧНА ЧАСТИНА ТА ОХОРОНА ПРАЦІ

В цій роботі розглянуто обробку аналогових сигналів за допомогою мікроконтролера/мікропроцесора з використанням пристроїв АЦП-ЦАП, створено прототип пристрою та реалізовані алгоритми обробки сигналів. Такі пристрої здійснюють обробку аналогових сигналів та можуть замінювати аналогові схеми обробки сигналів.

Мікроконвертори можуть застосовуватись для:

- Обробка сигналів давачів
- Фільтрація та виділення сигналів
- Компенсування шкідливих впливів
- Заміна аналогових схем обробки сигналів для підтримки застарілого обладнання
- Обробка звуку(компенсація ехо, усунення шумів, і.т.д.)

Оскільки, в цій роботі був розроблений прототип пристрою який демонструє роботу та можливості мікроконверторів, то собівартість пристрою буде визначатись як сума вартості використаних комплектуючих виробів(табл. 4.1) без витрат на розробку та виробництво.

					123.KI-41.11	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		57

Таблиця 4.1 Вартість комплектуючих виробів прототипу мікроконвертора

Найменування	Кількість, шт.	Ціна за од., \$.	Сума
STM32F103C8	1	1.49	1.49
MCP3201	1	1.07	1.07
MCP4275	1	1.2	1.2
AMS1117	1	0.2	0.2
Кварцовий резонатор 16МГц	1	0.1	0.1
Конденсатор 10 мкФ	3	0.1	0.3
Конденсатор 12 пкФ	2	0.03	0.06
Конденсатор 1 мкФ	1	0.07	0.07
Конденсатор 0.1 мкФ	1	0.07	0.07
Резистор 1 кОм	2	0.02	0.04
Резистор 7 кОм	1	0.02	0.02
Резистор 124 Ом	1	0.02	0.02
Резистор 120 Ом	1	0.02	0.02

Загальні витрати на прототип становлять 4.66\$. Для порівняння мікроконтролер STM32G431 який має в наявності АЦП та ЦАП на одному кристалі, має дещо більші можливості та здатен працювати з сигналами більшої частоти. Проте сама мікросхема коштує 8.91\$ без врахування мінімального набору компонентів необхідних для роботи мікроконтролера.

Охорона праці

До охорони праці входять вимоги до: проектування, будівництва, монтажу обладнання та електроприладів, а також аналіз потенційних небезпек та шкідливих виробничих чинників. Виробничі чинники та потенційні небезпеки не є прямою загрозою для життя працівників, проте він може проявитися при порушенні правил охорони праці під час діяльності.

Серед потенційних небезпек належать:

- Неприйнятні умови робочого місця
- Негативний вплив факторів праці на фізичний та психологічний стан працівника
- Недостатня освітленість робочих місць та робочих зон
- Неприйнятні значення температури та вологості, тобто мікроклімату, в робочих приміщеннях.
- Порушення правил пожежної безпеки, що збільшує ймовірність загорянь.

Для підтримування безпеки необхідно виявляти та усувати потенційні небезпеки за допомогою підтримування вимог стандартів які стосуються відповідних аспектів роботи.

Вимоги та норми щодо робочого місця описані в НПАОП 0.00-1.28-10 «Правила охорони праці під час експлуатації електронно-обчислювальних машин». Розміри робочого місця та розташування елементів робочого простору повинні відповідати характеру роботи та фізіологічним особливостями працівника. Для цього положення крісла, висота, розміри столу(робочої поверхні) повинні бути відрегульовані відповідно до конкретного працівника та

					123.KI-41.11	<i>Арк.</i>
<i>Зм.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>		59

виду роботи. Трудові операції повинні здійснюватись в зоні оптимальної досяжності.

Робоче місце працівника повинно забезпечувати вільність рухів та стійке положення працівника. Робоче місце повинне містити лише необхідні предмети і забезпечувати необхідну оглядовість.

Для зменшення негативного впливу напруженості праці на фізичний та психологічний стан працівника рекомендується:

- Складати оптимальний режим та розклад трудового процесу.
- Запроваджувати перерви для виконання фізичних вправ, зниження зорового напруження, запобігання втомі.

Для підвищення продуктивності роботи рекомендується проводити тренінги, навчання керівного складу методам організації, інформування працівники, забезпечення стабільності колективу та гласності політики компанії.

Для виключення травмування чи ушкодження на виробництві проводяться:

- організаційні заходи – вивчення та контроль знань правил безпеки
- технічні заходи – аналіз потенційно небезпечних місць на виробництві; ізоляція проводів та струмопровідних частин під напругою від випадкового дотику; встановлення систем заземлення та автоматичного відключення; профілактика та перевірка справності захисних систем.

Для зменшення зорового напруження та електромагнітного випромінювання рекомендується замінювати моніторів з електро-променевими трубками на ЖК-монітори чи OLED-монітори.

Вимоги та норми параметрів повітряного середовища описано в ДСН 3.3.6-042-99 «Санітарні норми мікроклімату виробничих приміщень».

					123.KI-41.11	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		60

Таблиця 4.2 Оптимальні значення температури, вологості та швидкості переміщення повітряних мас.

Параметри	Оптимальні	Допустимі
Температура °С	20-22	26
Вологість %	40-60	75
Швидкість перен. повітр. мас м/с	0,1-0,3	0,5

Вимоги та норми освітленості робочих приміщень описані в ДБН В.2.5-28-2006 «Інженерне обладнання будинків і споруд. Природне і штучне освітлення». Рівень освітленості згідно норм цього документу повинен становити 200 люкс. Для цього передбачені пристрої бічного природного освітлення та пристрої штучного загального освітлення. У системах освітлення використовуються люмінесцентні лампи потужністю 40-80 Вт.

Протипожежні заходи та вимоги для приміщень з персональними комп'ютерами описані в НАПБ А.01.001-2014 «Правила пожежної безпеки в Україні».

Для запобігання пожеж в приміщеннях з наявністю персональних комп'ютерів на силовому обладнанні та освітлювальних колах встановлені захисні системи, які вимикають напругу від ділянки кола в якій виникло коротке замикання.

Також для виявлення теплових та димових ознак пожежі встановлюються системи пожежних та охоронних сигналізацій з датчиками диму та вогню.

В приміщеннях з наявністю персональних комп'ютерів чи іншого електрообладнання повинні бути встановлені вуглекислотні вогнегасники. Відстань від вогнегасника до місця можливого загоряння повинна не перевищувати 10 м.

Висновки до розділу

В цьому розділі розглянуто витрати на прототип, правила охорони праці, види потенційних небезпек які виникають при роботі з електропристроями. Розглянуто вимоги та норми організації робочих місць та приміщень згідно вимог охорони праці.

					123.KI-41.11	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		61

ВИСНОВКИ

В дипломному проекті розроблено прототип мікроконвертора для обробки аналогових сигналів цифровими засобами.

Описано загальні принципи роботи мікроконверторів та їхніх основних елементів. Також було розглянуто види сигналів та їхні параметри, а також методи обробки сигналів.

Розроблено принципову електричну схему та за нею створено прототип пристрою на макетній платі.

Розроблено програму пристрою в середовищі CubeMX з використанням бібліотек HAL (Hardware Abstraction Library) від STMicroelectronics. Програма пристрою включає: конфігурацію роботи мікроконтролера; функції зчитування даних з АЦП та передачі даних на ЦАП; основний цикл роботи пристрою; реалізація алгоритмів обробки даних.

Розглянуто витрати на прототип, правила охорони праці, види потенційних небезпек які виникають при роботі з електропристроями. Розглянуто вимоги та норми організації робочих місць та приміщень згідно вимог охорони праці.

					123.KI-41.11	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		62

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. О.В. Дробик, В.В. Кідалов, В.В. Коваль, Б.Я. Костік, В.С. Лазебний, Г.М. Розорінов, Г.О. Сукач. «Цифрова обробка аудіо- та відеоінформації у мультимедійних системах: навчальний посібник». - 2008.
2. Рицар Б.Є. Цифрова техніка / Рицар Б.Є.–К.: НМК ВО, 1990. – 371 с.
3. Баскаков С.И. Радиотехнические цепи и сигналы: Учебник для вузов. – М.: Высшая школа, 1988. – 448 с.
4. Грищук Ю. С. Мікропроцесорні пристрої: навч. посібник.–Х: НТУ «ХПІ», 2008.–348с.
5. Наконечний А.Й., Наконечний Р.А., Павлиш В.А. Цифрова обробка сигналів, Львів: Видавництво Львівської політехніки, 2010. – 368 тор.
6. Антонию А. Цифровые фильтры: анализ и проектирование. – М.: Радио и связь, 1983. – 320 с.
7. Блейхут Р. Быстрые алгоритмы цифровой обработки сигналов. – М.: Мир, 1989. – 448 с.
8. Гутников В.С. Фильтрация измерительных сигналов. – Л.: Энергоатомиздат, 1990. – 192 с.
9. MCP3201 Datasheet [Електронний ресурс]: <http://ww1.microchip.com/downloads/en/devicedoc/21290f.pdf>
10. MCP4275 Datasheet [Електронний ресурс]: <https://www.sparkfun.com/datasheets/BreakoutBoards/MCP4725.pdf>
11. STM32F103C8 Datasheet [Електронний ресурс]: <https://www.st.com/resource/en/datasheet/stm32f103c8.pdf>
12. Вишенчук И. М. Выполнение операции усреднения в измерительных приборах методом весовых функций // Измерения, контроль, автоматизация. 1980. № 3, 4. С. 17—22.Сергієнко А. Б. Цифровая обработка сигналов / А. Б. Сергиенко —СПб. : Питер, 2003. —608 с. —ISBN 5-318-00666-3.

					123.KI-41.11	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		63

13. Волович Г.И. Схемотехника аналоговых и аналого-цифровых электронных устройств / Г. И. Волович —М. : Издательский дом «Додэка-XXI», 2005. —528 с. —ISBN 5-94120-074-9.

14. Автоматизація оброблення технічної інформації [Електронний ресурс] : методичні вказівки до виконання реферативних робіт для студентів радіотехнічного факультету напряму 6.050902 всіх спеціальностей / НТУУ «КПІ» ; уклад. В. О. Адаменко. – Електронні текстові данні (1 файл: 505,62 Кбайт). – Київ : НТУУ «КПІ», 2015. – 15 с.

					123.KI-41.11	Арк.
						64
Зм.	Арк.	№ докум.	Підпис	Дата		

ДОДАТОК А

ЛІСТИНГ ПРОГРАМИ МІКРОКОНТРОЛЕРА

Основна програма мікроконтролера main.c:

```
/* USER CODE END Header */
/* Includes -----*/
#include "main.h"

/* Private includes -----*/
/* USER CODE BEGIN Includes */

/* USER CODE END Includes */

/* Private typedef -----*/
/* USER CODE BEGIN PTD */

/* USER CODE END PTD */

/* Private define -----*/
/* USER CODE BEGIN PD */
/* USER CODE END PD */

/* Private macro -----*/
/* USER CODE BEGIN PM */

/* USER CODE END PM */

/* Private variables -----*/
I2C_HandleTypeDef hi2c1;

SPI_HandleTypeDef hspi1;

TIM_HandleTypeDef htim1;
TIM_HandleTypeDef htim2;
```

					123.KI-41.11	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		65

```

UART_HandleTypeDef huart2;

/* USER CODE BEGIN PV */
uint8_t mcp4725_address = 0xC0; //0x60 - 7 bit address
uint8_t mcp4725_read = 0x01;
uint8_t mcp4725_dac = 0x40;      // Writes data to the DAC
uint8_t mcp4725_dac_fast = 0x0;  // Writes data to the DAC fast
uint8_t mcp4725_dac_eeprom = 0x60; // Writes data to the DAC and the EEPROM (persisting
the assigned value after reset)
uint16_t pre = 0;
uint16_t pre1 = 0;
uint16_t pre2 = 0;
uint16_t output = 0;
uint8_t buffer[2] = {0x00, 0x00}; //, 0x00};
uint8_t x = 0;
float k=0.07;
/* USER CODE END PV */

/* Private function prototypes -----*/
void SystemClock_Config(void);
static void MX_GPIO_Init(void);
static void MX_SPI1_Init(void);
static void MX_USART2_UART_Init(void);
static void MX_I2C1_Init(void);
static void MX_TIM1_Init(void);
static void MX_TIM2_Init(void);
/* USER CODE BEGIN PFP */

/* USER CODE END PFP */

/* Private user code -----*/
/* USER CODE BEGIN 0 */
uint8_t spi_buf[2];
/* USER CODE END 0 */

```

					123.KI-41.11	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		66

```

/**
 * @brief The application entry point.
 * @retval int
 */
int main(void)
{
    /* USER CODE BEGIN 1 */

    /* USER CODE END 1 */

    /* MCU Configuration-----*/

    /* Reset of all peripherals, Initializes the Flash interface and the Systick. */
    HAL_Init();

    /* USER CODE BEGIN Init */

    /* USER CODE END Init */

    /* Configure the system clock */
    SystemClock_Config();

    /* USER CODE BEGIN SysInit */

    /* USER CODE END SysInit */

    /* Initialize all configured peripherals */
    MX_GPIO_Init();
    MX_SPI1_Init();
    MX_USART2_UART_Init();
    MX_I2C1_Init();
    MX_TIM1_Init();
    MX_TIM2_Init();
    /* USER CODE BEGIN 2 */
    HAL_TIM_OC_Start(&htim1,TIM_CHANNEL_2);

```

					123.KI-41.11	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		67

```

HAL_TIM_Base_Start_IT(&htim2);
GPIOC->BSRR = GPIO_BSRR_BS13;
//buffer[0] = mcp4725_dac_fast;
/* USER CODE END 2 */

/* Infinite loop */
/* USER CODE BEGIN WHILE */
while (1)
{
    /* USER CODE END WHILE */

    /* USER CODE BEGIN 3 */
}
/* USER CODE END 3 */
}

/**
 * @brief System Clock Configuration
 * @retval None
 */
void SystemClock_Config(void)
{
    RCC_OscInitTypeDef RCC_OscInitStruct = {0};
    RCC_ClkInitTypeDef RCC_ClkInitStruct = {0};

    /** Initializes the RCC Oscillators according to the specified parameters
     * in the RCC_OscInitTypeDef structure.
     */
    RCC_OscInitStruct.OscillatorType = RCC_OSCILLATORTYPE_HSE;
    RCC_OscInitStruct.HSEState = RCC_HSE_ON;
    RCC_OscInitStruct.HSEPredivValue = RCC_HSE_PREDIV_DIV1;
    RCC_OscInitStruct.HSIState = RCC_HSI_ON;
    RCC_OscInitStruct.PLL.PLLState = RCC_PLL_ON;
    RCC_OscInitStruct.PLL.PLLSource = RCC_PLLSOURCE_HSE;
    RCC_OscInitStruct.PLL.PLLMUL = RCC_PLL_MUL9;

```

					123.KI-41.11	<i>Арк.</i>
<i>Зм.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>		68

```

if (HAL_RCC_OscConfig(&RCC_OscInitStruct) != HAL_OK)
{
    Error_Handler();
}
/** Initializes the CPU, AHB and APB buses clocks
*/
RCC_ClkInitStruct.ClockType                                =
RCC_CLOCKTYPE_HCLK|RCC_CLOCKTYPE_SYSCLK
                    |RCC_CLOCKTYPE_PCLK1|RCC_CLOCKTYPE_PCLK2;
RCC_ClkInitStruct.SYSCLKSource = RCC_SYSCLKSOURCE_PLLCLK;
RCC_ClkInitStruct.AHBCLKDivider = RCC_SYSCLK_DIV1;
RCC_ClkInitStruct.APB1CLKDivider = RCC_HCLK_DIV2;
RCC_ClkInitStruct.APB2CLKDivider = RCC_HCLK_DIV1;

if (HAL_RCC_ClockConfig(&RCC_ClkInitStruct, FLASH_LATENCY_2) != HAL_OK)
{
    Error_Handler();
}
HAL_RCC_MCOConfig(RCC_MCO,                                RCC_MCO1SOURCE_PLLCLK,
RCC_MCODIV_1);
}

/**
 * @brief I2C1 Initialization Function
 * @param None
 * @retval None
 */
static void MX_I2C1_Init(void)
{
    /* USER CODE BEGIN I2C1_Init 0 */

    /* USER CODE END I2C1_Init 0 */

    /* USER CODE BEGIN I2C1_Init 1 */

```

					123.KI-41.11	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		69

```

/* USER CODE END I2C1_Init 1 */
hi2c1.Instance = I2C1;
hi2c1.Init.ClockSpeed = 1000000;
hi2c1.Init.DutyCycle = I2C_DUTYCYCLE_2;
hi2c1.Init.OwnAddress1 = 0;
hi2c1.Init.AddressingMode = I2C_ADDRESSINGMODE_7BIT;
hi2c1.Init.DualAddressMode = I2C_DUALADDRESS_DISABLE;
hi2c1.Init.OwnAddress2 = 0;
hi2c1.Init.GeneralCallMode = I2C_GENERALCALL_DISABLE;
hi2c1.Init.NoStretchMode = I2C_NOSTRETCH_DISABLE;
if (HAL_I2C_Init(&hi2c1) != HAL_OK)
{
    Error_Handler();
}
/* USER CODE BEGIN I2C1_Init 2 */

/* USER CODE END I2C1_Init 2 */

}

/**
 * @brief SPI1 Initialization Function
 * @param None
 * @retval None
 */
static void MX_SPI1_Init(void)
{

/* USER CODE BEGIN SPI1_Init 0 */

/* USER CODE END SPI1_Init 0 */

/* USER CODE BEGIN SPI1_Init 1 */

```

					123.KI-41.11	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		70

```

/* USER CODE END SPI1_Init 1 */
/* SPI1 parameter configuration*/
hspi1.Instance = SPI1;
hspi1.Init.Mode = SPI_MODE_MASTER;
hspi1.Init.Direction = SPI_DIRECTION_2LINES;
hspi1.Init.DataSize = SPI_DATASIZE_8BIT;
hspi1.Init.CLKPolarity = SPI_POLARITY_HIGH;
hspi1.Init.CLKPhase = SPI_PHASE_1EDGE;
hspi1.Init.NSS = SPI_NSS_HARD_OUTPUT;
hspi1.Init.BaudRatePrescaler = SPI_BAUDRATEPRESCALER_64;
hspi1.Init.FirstBit = SPI_FIRSTBIT_MSB;
hspi1.Init.TIMode = SPI_TIMODE_DISABLE;
hspi1.Init.CRCCalculation = SPI_CRCCALCULATION_DISABLE;
hspi1.Init.CRCPolynomial = 10;
if (HAL_SPI_Init(&hspi1) != HAL_OK)
{
    Error_Handler();
}
/* USER CODE BEGIN SPI1_Init 2 */

/* USER CODE END SPI1_Init 2 */

}

/**
 * @brief TIM1 Initialization Function
 * @param None
 * @retval None
 */
static void MX_TIM1_Init(void)
{
    /* USER CODE BEGIN TIM1_Init 0 */

    /* USER CODE END TIM1_Init 0 */

```

					123.KI-41.11	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		71

```

TIM_ClockConfigTypeDef sClockSourceConfig = {0};
TIM_MasterConfigTypeDef sMasterConfig = {0};
TIM_OC_InitTypeDef sConfigOC = {0};
TIM_BreakDeadTimeConfigTypeDef sBreakDeadTimeConfig = {0};

/* USER CODE BEGIN TIM1_Init 1 */

/* USER CODE END TIM1_Init 1 */
htim1.Instance = TIM1;
htim1.Init.Prescaler = 100;
htim1.Init.CounterMode = TIM_COUNTERMODE_UP;
htim1.Init.Period = 3600;
htim1.Init.ClockDivision = TIM_CLOCKDIVISION_DIV1;
htim1.Init.RepetitionCounter = 0;
htim1.Init.AutoReloadPreload = TIM_AUTORELOAD_PRELOAD_ENABLE;
if (HAL_TIM_Base_Init(&htim1) != HAL_OK)
{
    Error_Handler();
}
sClockSourceConfig.ClockSource = TIM_CLOCKSOURCE_INTERNAL;
if (HAL_TIM_ConfigClockSource(&htim1, &sClockSourceConfig) != HAL_OK)
{
    Error_Handler();
}
if (HAL_TIM_OC_Init(&htim1) != HAL_OK)
{
    Error_Handler();
}
sMasterConfig.MasterOutputTrigger = TIM_TRGO_RESET;
sMasterConfig.MasterSlaveMode = TIM_MASTERSLAVEMODE_DISABLE;
if (HAL_TIMEx_MasterConfigSynchronization(&htim1, &sMasterConfig) != HAL_OK)
{
    Error_Handler();
}

```

					123.KI-41.11	<i>Арк.</i>
<i>Зм.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>		72


```

sConfigOC.OCMode = TIM_OCMode_TOGGLE;
sConfigOC.Pulse = 0;
sConfigOC.OCpolarity = TIM_OCPolarity_HIGH;
sConfigOC.OCNPolarity = TIM_OCNPolarity_HIGH;
sConfigOC.OCFastMode = TIM_OCFAST_DISABLE;
sConfigOC.OCIdleState = TIM_OCIdleState_RESET;
sConfigOC.OCNIdleState = TIM_OCNIdleState_RESET;
if (HAL_TIM_OC_ConfigChannel(&htim1, &sConfigOC, TIM_CHANNEL_2) !=
HAL_OK)
{
    Error_Handler();
}
sBreakDeadTimeConfig.OffStateRunMode = TIM_OSSR_DISABLE;
sBreakDeadTimeConfig.OffStateIDLEMode = TIM_OSSI_DISABLE;
sBreakDeadTimeConfig.LockLevel = TIM_LOCKLEVEL_OFF;
sBreakDeadTimeConfig.DeadTime = 0;
sBreakDeadTimeConfig.BreakState = TIM_BREAK_DISABLE;
sBreakDeadTimeConfig.BreakPolarity = TIM_BREAKPolarity_HIGH;
sBreakDeadTimeConfig.AutomaticOutput = TIM_AUTOMATICOUTPUT_DISABLE;
if (HAL_TIMEx_ConfigBreakDeadTime(&htim1, &sBreakDeadTimeConfig) != HAL_OK)
{
    Error_Handler();
}
/* USER CODE BEGIN TIM1_Init 2 */

/* USER CODE END TIM1_Init 2 */
HAL_TIM_MspPostInit(&htim1);

}

/**
 * @brief TIM2 Initialization Function
 * @param None
 * @retval None
 */

```

					123.KI-41.11	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		73

```

static void MX_TIM2_Init(void)
{

    /* USER CODE BEGIN TIM2_Init 0 */

    /* USER CODE END TIM2_Init 0 */

    TIM_ClockConfigTypeDef sClockSourceConfig = {0};
    TIM_MasterConfigTypeDef sMasterConfig = {0};

    /* USER CODE BEGIN TIM2_Init 1 */

    /* USER CODE END TIM2_Init 1 */
    htim2.Instance = TIM2;
    htim2.Init.Prescaler = 0;
    htim2.Init.CounterMode = TIM_COUNTERMODE_UP;
    htim2.Init.Period = 7200;
    htim2.Init.ClockDivision = TIM_CLOCKDIVISION_DIV1;
    htim2.Init.AutoReloadPreload = TIM_AUTORELOAD_PRELOAD_ENABLE;
    if (HAL_TIM_Base_Init(&htim2) != HAL_OK)
    {
        Error_Handler();
    }
    sClockSourceConfig.ClockSource = TIM_CLOCKSOURCE_INTERNAL;
    if (HAL_TIM_ConfigClockSource(&htim2, &sClockSourceConfig) != HAL_OK)
    {
        Error_Handler();
    }
    sMasterConfig.MasterOutputTrigger = TIM_TRGO_RESET;
    sMasterConfig.MasterSlaveMode = TIM_MASTERSLAVEMODE_DISABLE;
    if (HAL_TIMEx_MasterConfigSynchronization(&htim2, &sMasterConfig) != HAL_OK)
    {
        Error_Handler();
    }

    /* USER CODE BEGIN TIM2_Init 2 */

```

					123.KI-41.11	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		74

```

/* USER CODE END TIM2_Init 2 */

}

/**
 * @brief USART2 Initialization Function
 * @param None
 * @retval None
 */
static void MX_USART2_UART_Init(void)
{

/* USER CODE BEGIN USART2_Init 0 */

/* USER CODE END USART2_Init 0 */

/* USER CODE BEGIN USART2_Init 1 */

/* USER CODE END USART2_Init 1 */
huart2.Instance = USART2;
huart2.Init.BaudRate = 115200;
huart2.Init.WordLength = UART_WORDLENGTH_8B;
huart2.Init.StopBits = UART_STOPBITS_1;
huart2.Init.Parity = UART_PARITY_NONE;
huart2.Init.Mode = UART_MODE_TX_RX;
huart2.Init.HwFlowCtl = UART_HWCONTROL_NONE;
huart2.Init.OverSampling = UART_OVERSAMPLING_16;
if (HAL_UART_Init(&huart2) != HAL_OK)
{
    Error_Handler();
}
/* USER CODE BEGIN USART2_Init 2 */

/* USER CODE END USART2_Init 2 */

```

					123.KI-41.11	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		75

```

}

/**
 * @brief GPIO Initialization Function
 * @param None
 * @retval None
 */
static void MX_GPIO_Init(void)
{
    GPIO_InitTypeDef GPIO_InitStructure = {0};

    /* GPIO Ports Clock Enable */
    __HAL_RCC_GPIOC_CLK_ENABLE();
    __HAL_RCC_GPIOD_CLK_ENABLE();
    __HAL_RCC_GPIOA_CLK_ENABLE();
    __HAL_RCC_GPIOB_CLK_ENABLE();

    /*Configure GPIO pin Output Level */
    HAL_GPIO_WritePin(GPIOC, GPIO_PIN_13|GPIO_PIN_14, GPIO_PIN_RESET);

    /*Configure GPIO pins : PC13 PC14 */
    GPIO_InitStructure.Pin = GPIO_PIN_13|GPIO_PIN_14;
    GPIO_InitStructure.Mode = GPIO_MODE_OUTPUT_PP;
    GPIO_InitStructure.Pull = GPIO_NOPULL;
    GPIO_InitStructure.Speed = GPIO_SPEED_FREQ_LOW;
    HAL_GPIO_Init(GPIOC, &GPIO_InitStructure);

    /*Configure GPIO pin : PA8 */
    GPIO_InitStructure.Pin = GPIO_PIN_8;
    GPIO_InitStructure.Mode = GPIO_MODE_AF_PP;
    GPIO_InitStructure.Speed = GPIO_SPEED_FREQ_LOW;
    HAL_GPIO_Init(GPIOA, &GPIO_InitStructure);
}

```

					123.KI-41.11	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		76

```

/* USER CODE BEGIN 4 */
void HAL_TIM_PeriodElapsedCallback(TIM_HandleTypeDef* htim)
{
    GPIOC->BSRR = GPIO_BSRR_BR13;
    HAL_SPI_Receive(&hspi1, spi_buf, 2, 100);
    GPIOC->BSRR = GPIO_BSRR_BS13;
    pre2=((spi_buf[0] << 8) | spi_buf[1]) & 0x0fff;
    pre1=(uint16_t)((1-k)*pre1+k*pre2);
    pre=(uint16_t)((1-k)*pre+k*pre1);
    output=(uint16_t)((1-k)*output+k*pre);
    buffer[0] = (output / 256);    // Upper data bits    (D11.D10.D9.D8.D7.D6.D5.D4)
    buffer[1] = (output % 256);// << 4; // Lower data bits    (D3.D2.D1.D0.x.x.x.x)
    HAL_I2C_Master_Transmit(&hi2c1, mcp4725_address, buffer, 2, 1000);
}

#ifdef __GNUC__
int _write(int fd, char *ptr, int len)
{
    HAL_UART_Transmit(&huart2, (uint8_t *)ptr, len, HAL_MAX_DELAY);
    return len;
}
#elif defined(__ICCARM__)
#include "LowLevelIOInterface.h"
size_t __write(int handle, const unsigned char *buffer, size_t size)
{
    HAL_UART_Transmit(&huart1, (uint8_t *)buffer, size, HAL_MAX_DELAY);
    return size;
}
#elif defined(__CC_ARM)
int fputc(int ch, FILE *f)
{
    HAL_UART_Transmit(&huart1, (uint8_t *)&ch, 1, HAL_MAX_DELAY);
    return ch;
}

```

```

#endif

/* USER CODE END 4 */

/**
 * @brief This function is executed in case of error occurrence.
 * @retval None
 */
void Error_Handler(void)
{
    /* USER CODE BEGIN Error_Handler_Debug */
    /* User can add his own implementation to report the HAL error return state */

    /* USER CODE END Error_Handler_Debug */
}

#ifdef USE_FULL_ASSERT
/**
 * @brief Reports the name of the source file and the source line number
 * where the assert_param error has occurred.
 * @param file: pointer to the source file name
 * @param line: assert_param error line source number
 * @retval None
 */
void assert_failed(uint8_t *file, uint32_t line)
{
    /* USER CODE BEGIN 6 */
    /* User can add his own implementation to report the file name and line number,
    tex: printf("Wrong parameters value: file %s on line %d\r\n", file, line) */
    /* USER CODE END 6 */
}
#endif /* USE_FULL_ASSERT */

```

					123.KI-41.11	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		78