

Вінницький національний технічний університет
Факультет інформаційних технологій та комп'ютерної інженерії
Кафедра обчислювальної техніки

МАГІСТЕРСЬКА КВАЛІФІКАЦІЙНА РОБОТА

на тему:

**«Засоби апаратно-програмного опрацювання сигналів в аудіопроекторів з
дистанційним керуванням»**

08-23.МКР.003.00.000 ПЗ

Виконав: студент 2 курсу, групи 1КІ-20м

напряму підготовки (спеціальності)

123 — «Комп'ютерна інженерія»

_____ Гирін Д. Ю.

Керівник: к.т.н., доц.каф. ОТ

_____ к.т.н., доц. Крупельницький Л.В

(прізвище та ініціали)

Допущено до захисту

Завідувач кафедри ОТ

_____ д.т.н., проф. Азаров О. Д.

« » _____ 2021 р

Вінниця 2021

Вінницький національний технічний університет

Факультет інформаційних технологій та комп'ютерної інженерії

Кафедра обчислювальної техніки

Освітній рівень магістр

Спеціальність 123 Комп'ютерна інженерія

ЗАТВЕРДЖУЮ

Завідувач кафедри
обчислювальної техніки

_____ О.Д. Азаров.
« ____ » _____ 2021 року

З А В Д А Н Н Я

НА МАГІСТЕРСЬКУ КВАЛІФІКАЦІЙНУ РОБОТУ СТУДЕНТУ

Гиріну Денису Юрійовичу

1 Тема роботи «Засоби апаратно-програмного опрацювання сигналів в аудіопроекторах з дистанційним керуванням», керівник роботи Крупельницький Леонід Віталійович к.т.н., доцент, затверджені наказом вищого навчального закладу від 24.09.2021 року № 277

2 Строк подання студентом роботи 22.12.2021

3 Вихідні дані до роботи: спосіб обробки звукового сигналу – цифровий, аналіз цифрових сигнальних процесорів, вибір методів опрацювання аудіосигналів, вибір технологій бездротової передачі даних.

4 Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити): вступ, аналіз засобів перетворення й опрацювання звукових сигналів, методи й засоби обробки сигналів для аудіопроекторів, розробка методів цифрового опрацювання сигналів і схем аудіопроекторів з дистанційним керуванням, висновки.

5 Перелік графічного матеріалу (з точним зазначенням обов'язкових

креслень): технічне завдання, схема аудіопроцесора, схема пристрою гітарного ефекту, лістинг програми керуючого мікроконтролера.

6. Консультанти розділів роботи наведені в таблиці 1.

Таблиця 1 Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада Консульта нта	Підпис, дата	
		завдання видав	завдання прийняв
1-3	Крупельницький Л. В. к.т.н., доцент		
4	Кавецький В. В. к.е.н., доцент		

7 Дата видачі завдання 07.09.2021

8 Календарний план наведено в таблиці 2.

Таблиця 2 Календарний план

№ з/п	Назва етапів дипломного Проекту	Срок виконання етапів роботи	Примітка
1	Постановка задачі	01.09.2021	
2	Огляд існуючих рішень	01.10.2021	
3	Аналіз методів опрацювання аудіосигналів	11.10.2021	
4	Огляд цифрових сигнальних процесорів	25.10.2021	
5	Розробка методів і алгоритмів опрацювання сигнала	8.11.2021	
6	Розробка структурних і принципів схем	01.09.2021	
7	Розробка алгоритму і ПЗ	01.10.2021	
8	Розрахунок економічної частини	11.10.2021	
9	Оформлення пояснювальної записки та ілюстративного матеріалу	25.10.2021	
10	Виконання магістерської кваліфікаційної роботи	8.11.2021	
11	Перевірка якості виконання магістерської кваліфікаційної роботи та усунення недоліків	22.11.2021	

Магістрант _____ Гирін Д. Ю.

Керівник _____ к.т.н., доц. Крупельницький Л.В.

АНОТАЦІЯ

УДК 004.4

Гирін Д. Ю. Засоби апаратно-програмного опрацювання сигналів в аудіопроцесорах з дистанційним керуванням. Магістерська кваліфікаційна робота зі спеціальності 123 — комп'ютерна інженерія, освітня програма — комп'ютерна інженерія. Вінниця: ВНТУ, 2021. 114 с.

У магістерській кваліфікаційній роботі розглянуто апаратно-програмного опрацювання сигналів в аудіопроцесорах з дистанційним керуванням. Удосконалений метод передачі звукового сигналу за допомогою інтерфейсу безпроводної передачі даних, що забезпечує надійність та зручність передачі й опрацювання звукового сигналу. Удосконалений метод опрацювання звукових сигналів, що дозволяє використовувати його у повсякденному житті шляхом керування мікропроцесорну платформу Arduino та мобільний додаток.

З використанням середовища Arduino IDE, мовою програмування Swift та мікропроцесорної платформи Arduino UNO розроблено та виготовлено пристрій керування звуком: гітарний процесор, що керується віддалено через інтерфейс безпроводного з'єднання, і може використовуватись для керування звукозаписуючих та звуковідтворюючих пристроїв, що дозволяє керувати набором пристроїв через телефон завдяки інтерфейсу Bluetooth навіть віддалено.

Ключові слова: аудіопроцесор, аналогова обробка сигналів, цифрова обробка сигналів, звукові ефекти, бездротова передача сигналів, цифровий сигнальний процесор.

ABSTRACT

UDC 004.4

Hyrin D. Y. Means of hardware and software signal processing in audio processors with remote control. Master's thesis in the specialty 123 - computer engineering, educational program - computer engineering. Vinnytsia: VNTU, 2021. 114 p.

In the master's qualification work the hardware-software processing of signals in audio processors with remote control is considered. An advanced method of audio signal transmission using a wireless data transmission interface, which provides reliability and convenience of audio signal transmission and processing. An advanced method of processing audio signals that allows you to use it in everyday life by controlling the Arduino microprocessor platform with mobile application.

Using the Arduino IDE, the Swift programming language and the Arduino UNO microprocessor platform, a sound control device has been developed and manufactured: a guitar processor that is controlled remotely via a wireless interface and can be used to control audio and audio recording devices with the phone via Bluetooth interface.

Key words: audio processor, analog signal processing, digital signal processing, sound effects, wireless signal transmission, digital signal processor.

ЗМІСТ

ВСТУП	9
1 АНАЛІЗ ЗАСОБІВ ПЕРЕТВОРЕННЯ Й ОПРАЦЮВАННЯ ЗВУКОВИХ СИГНАЛІВ	11
1.1 Завдання обробки звукових сигналів	11
1.2 Базові методи в задачах інструментальної цифрової обробки сигналів ...	13
1.2.1 Аудіотрансляція	14
1.2.1 Аудіотрансляція	14
1.2.2 Звукові ефекти	15
1.2.3 Порівняння аналогової та цифрової обробки звукових сигналів	17
1.3 АЦП та ЦАП у інструментальних пристроях	26
1.3.1 Параметри та класифікація інструментальних пристроїв обробки звуку.....	27
2 АНАЛІЗ МЕТОДІВ І ЗАСОБИ ОБРОБКИ СИГНАЛІВ ДЛЯ АУДИОПРОЦЕСОРІВ З ДИСТАНЦІЙНИМ КЕРУВАННЯМ	33
2.1 Огляд архітектури	33
2.2 Розвиток сигнальних процесорів	34
2.3 Аналіз основних параметрів ЦСП.....	36
2.4 Апаратна реалізація ЦСП.....	39
2.4.1 Гарвардська архітектура	39
2.4.2 Стандартні ЦСП	40
2.4.3 Покращені стандартні ЦСП.....	42
2.4.5 Суперскалярні ЦСП.....	44
2.4.6 Гібридні ЦСП	45

					08-23.МКР.003.00.000 ПЗ						
мн.		докум.	Підпис	Дата	Засоби апаратно-програмного опрацювання сигналів в аудиопроекторів з дистанційним керуванням Пояснювальна записка						
Розробив		Гирін Д. Ю.							Літ.	Аркуш	Аркушів
Перевірив		Крупельницький Л.В.								6	121
Реценз.		Кондратенко Н.Р.							ВНТУ, гр. 1КІ-20м		
Н. контр.		Швець С.І.									
Затвердж.		Азаров О.Д.									

2.5 Програмування ЦСП.....	46
2.5.1 Особливості асемблерів ЦСП.....	46
2.5.2 Сумісність усередині сімейств ЦСП	47
2.5.3 Налагоджування програм	47
3 РОЗРОБКА МЕТОДІВ ЦИФРОВОГО ОПРАЦЮВАННЯ СИГНАЛІВ І	
СХЕМ АУДИОПРОЦЕСОРІВ З ДИСТАНЦІЙНИМ КЕРУВАННЯМ.....	49
3.1 Реалізація ефекту однократної затримки сигналу	49
3.2 Багаторазова затримка сигналу	51
3.2.1 Багаторазова затримка з використанням трансверсального фільтра..	52
3.2.2 Багаторазова затримка з використанням рекурсивного фільтра	53
3.3 Ефект детонації	54
3.4 Ефект хорусу.....	56
3.5 Ефект вібрато	57
3.6 Зміна динамічного діапазону: компресори й експандери	59
3.7 Архітектура та алгоритм роботи аудіопроцесора	61
3.8 Розробка принципової схеми процесора.....	66
3.9 Розробка пристрою для гітарного ефекту з дистанційним керуванням на мікроконтролері ARDUINO UNO	72
3.9.1 Призначення й технічні параметри	72
3.9.2 Структурна й принципова схеми пристрою	73
3.9.3 Загальний алгоритм роботи програми.....	74
4 ЕКОНОМІЧНИЙ РОЗДІЛ.....	77
4.1 Проведення комерційного та технологічного аудиту науково-технічної розробки.....	77
4.2 Визначення рівня конкурентоспроможності розробки	81
4.3 Розрахунок витрат на проведення науково-дослідної роботи.....	84
4.3.1 Витрати на оплату праці.....	84
4.3.2 Відрахування на соціальні заходи	87
4.3.3 Розрахунок витрат на комплектуючі.....	89

					08-23.МКР.003.00.000 ПЗ	Арк.
						7
ЗМН.	Арк.З	№ докум.	Лист	Підпис №	Дата	

4.3.4 Спецустаткування для наукових робіт	90
4.3.5 Програмне забезпечення для наукових робіт	91
4.3.6 Амортизація обладнання, програмних засобів та приміщень	92
4.3.7 Паливо та енергія для науково-виробничих цілей	93
4.3.8 Службові відрядження.....	94
4.3.9 Загальновиробничі витрати	94
4.4 Розрахунок економічної ефективності науково-технічної розробки при її можливій комерціалізації потенційним інвестором	97
ВИСНОВКИ	103
ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ	104
ДОДАТОК А Технічне завдання	106
ДОДАТОК Б Схема аудіопроцесора	109
ДОДАТОК В Схема пристрою гітарного ефекту з дистанційним керуванням.....	110
ДОДАТОК Г Лістинг іОs програми для керування мікроконтролером.....	111
ДОДАТОК Д Протокол перевірки навчальної (кваліфікаційної) роботи.....	121

					08-23.МКР.003.00.000 ПЗ	Арк.
						8
Змн.	Арк.	№ докум.	Підпис	Дата		

ВСТУП

Актуальність теми дослідження зумовлена темпом розвитку цифрові технологій, зокрема, у медійній сфері також. При цьому залишається невирішеним коло питань, пов'язаних з технічними аспектами виступів музичних артистів. Якщо якість та надійність сучасного звукозаписуючого та звуковідтворюючого обладнання не викликає жодних сумнівів, то комфорт музикантів під час роботи з таким обладнанням іноді досі залишається під питанням. Це зумовлено недостатньою увагою до даної проблеми та відсутністю відповідних технічних рішень.

Об'єктом дослідження є процес обробки звукових сигналів у професійних інструментальних аудіо процесорах.

Предметом дослідження є засоби апаратно-програмного опрацювання сигналів в аудіо процесорах з дистанційним керуванням.

Метою роботи є створення ефективних апаратно-програмних засобів для обробки звуку з використанням бездротової передачі сигналів керування.

Для досягнення цієї мети необхідно вирішити такі задачі:

— проаналізувати сучасний стан і тенденції розвитку зарубіжних і вітчизняних розробок обробки звукових сигналів;

— обрати і реалізувати вдосконалені методи опрацювання аудіо сигналів;

— запропонувати програмні-апаратні засоби обробки та передачі звуку використовуючи технології бездротового з'єднання.

Для досягнення поставленої в роботі мети використовуються такі методи дослідження:

— системний аналіз;

— об'єктно-орієнтовані методи проектування;

— методи натурного моделювання.

Наукова новизна роботи полягає у вдосконаленні спеціалізованих методів опрацювання звукових сигналів з використанням бездротові технології керування їх параметрами. Базовані на цьому принципі структурні рішення дозволяють значно покращити функціональні можливості сучасних систем інструментального звукозапису та звуковідтворення.

Практичне значення роботи полягає в тому, що в апаратній частині запропоновано оригінальне рішення побудови цифрових пристроїв інструментального запису та відтворення звуку, що підвищить споживчі властивості останніх, не збільшуючи виробничі витрати у значній мірі. Використані науково-технічні рішення дозволяють реалізувати та адаптувати їх для концертної діяльності музичних колективів.

Апробацію матеріалів роботи виконано під час доповіді на XLVII Науково-технічна конференція факультету інформаційних технологій та комп'ютерної інженерії (2018).

Публікація за темою роботи:

Гирін, Д.. Програмні засоби для експериментальних досліджень діаграм направленості мікрофонних решіток. НТКП ВНТУ. Факультет інформаційних технологій та комп'ютерної інженерії, Ukraine, mar. 2018. Available at: <<https://conferences.vntu.edu.ua/index.php/all-fitki/all-fitki-2018/paper/view/4851/4037>>. Date accessed: 19 Dec. 2021.

1 АНАЛІЗ ЗАСОБІВ ПЕРЕТВОРЕННЯ Й ОПРАЦЮВАННЯ ЗВУКОВИХ СИГНАЛІВ

1.1 Завдання обробки звукових сигналів

Звук — коливальний рух фізичних частинок середовища. Але цифрові пристрої, такі як комп'ютери, телефони та програвачі компакт-дисків, не можуть обробляти звук у аналоговій формі, що означає, що він повинен бути перетворений на цифровий сигнал — послідовних набір цілочислених елементів — для зберігання та обробки у цифровому вигляді. Таких перетворень досягають за допомогою аналого-цифрового перетворювача (АЦП), тоді як цифро-аналоговий перетворювач (ЦАП) використовується для реверсування процесу на іншому кінці.

Під обробкою звукового сигналу розуміється зміна частотної або фазової характеристики, звуження або розширення динамічного діапазону, застосування амплітудної, частотної або фазової модуляції, видалення шумів, а також створення затриманих за часом загасаючих копій цього сигналу. Метою обробки можуть бути як суто технічні завдання, такі як узгодження параметрів сигналу з характеристиками електроакустичного тракту, так і звукорежисером, що визначаються, зокрема, різні звукові ефекти (тремоло, вібрато, хор, луна, реверберація та інші).

Аналого-цифрове перетворення — це процес перетворення сигналу з аналогової області на цифрову. Цей процес може відбуватися в основній смузі частот, як у випадку АЦП з прямим перетворенням, високій або низькій частоті, залежно від вимог і, отже, від архітектури АЦП, якої дотримуються розробники. У роботі розглянуто апаратну архітектуру, в якій може бути реалізований аналого-цифровий перетворювач (АЦП) [1].

Аналого-цифровий перетворювач (АЦП) є одним з основних будівельних блоків сучасних збору даних систем (як систем збору даних). Основне призначення аналого-цифрових перетворювачів у системі збору даних — перетворювати аналогові сигнали на потік цифрових даних, щоб

система збору даних могла обробляти їх для відображення, зберігання та аналізу.

Перш ніж перейти до розгляду комп'ютерних засобів аналізу фонограм, що використовуються у фоноскопії, необхідно коротко зупинитися на основних відмінностях аналогових та цифрових звукових сигналів.

Аналоговий сигнал - це зміна напруги пропорційно змін стану джерела сигналу і змін навколишнього середовища. Термін "аналоговий" стосовно електричного сигналу позначає, що цей сигнал безперервний за часом і амплітудою. Він точно відображає форму звукової хвилі, яка поширюється у повітрі; і в цьому випадку аналоговий сигнал - це зміна напруги, пропорційна до зміни звукового тиску. Аналоговий звуковий сигнал є безперервним сигналом, представленим електричною напругою або струмом, який можна порівняти з звуковим коливанням у просторі та повітрі. Аналогова обробка сигналу потім включає фізіологічну обробку безперервного сигналу шляхом зміни напруги або струму або заряду через електричні ланцюги [1].

Цифрова обробка сигналів - важлива галузь електроніки та телекомунікаційної техніки, яка займається імпривізацією надійності та точності цифрового зв'язку з використанням кількох методів. Варто зазначити, що цифрова обробка сигналів включає розробку алгоритмів, які можна використовувати для покращення сигналу певним чином або вилучення деякої корисної інформації.

Основою для попередньої обробки сигналів є процедури швидких дискретних ортогональних перетворень, що реалізуються у різних функціональних базисах, процедури лінійної та нелінійної фільтрації, лінійної алгебри.

Найголовніше завдання обробки — це усунення перешкод та шумів. Вирішити це завдання повною мірою можна тільки в тому випадку, якщо сигнал надходить надлишковий з певними параметрами.

Внаслідок цього необхідно забезпечити правильний прийом сигналу. Чим більше надходить корисного сигналу і що менше перешкод, то більше ймовірність виконати завдання якісно.

До основних завдань цифрової обробки сигналу належить [2]:

- спектральний аналіз;
- лінійна фільтрація;
- згортка традиційних типів;
- частотно-часовий аналіз;
- нелінійне оброблення;
- адаптивна фільтрація;
- багатошвидкісна обробка;
- секційна згортка.
- цифрова обробка використовується у таких галузях, як:
- літакобудування, оборонні системи, космічне устаткування;
- електроніка для автомобілів;
- освітлювальні системи;
- мобільний, стаціонарний зв'язок, інтернет-телефонія та інше;
- електронні прилади та пристрої для дому;
- медичне обладнання;
- вимірювальні та інші прилади;
- системи управління.

1.2 Базові методи в задачах інструментальної цифрової обробки сигналів

При цифровій обробці сигналів використовуються різні методи, затребуваність яких сьогодні набагато вище, ніж у аналогових методів, що застосовувалися раніше. В основі більшості методів є застосування DSP-процесорів. Ці імпульсні цифрові пристрої забезпечують якісну обробку сигналів і поєднуються в дискретні керуючі системи.

Методи та алгоритми, що застосовуються у цифровій обробці, постійно удосконалюються.

Цифрова обробка має ряд переваг [4]:

- обробка сигналу проводиться з точністю від 75% та вище;
- дестабілізуючі фактори зводяться до мінімуму;
- обробка інформації може здійснюватися програмними засобами.

Просте налаштування та застосування спеціальних алгоритмів для обробки сигналів суттєво підвищують ефективність та продуктивність роботи.

Як приклади використання ЦОС можна навести:

- системи шумоподавлення для аудіозаписів та відео;
- обробка зображень — вирівнювання ступеня освітленості, антиаліасинг, накладання псевдотонування;
- різноманітні художні ефекти та способи покращення естетичних якостей зображень, фотографій та інше;
- відновлювальні роботи над зображеннями;
- компресія зображень;
- пошук певних фрагментів на графічних файлах.

Найчастіше обробка сигналів цифровим способом використовується в області графіки, але звук можна очищати від шумів за допомогою ЦОС, роблячи його чіткішим і правильнішим.

Способи обробки та області застосування включають зберігання, стиснення даних, повернення музичної інформації, обробку мови, локалізацію, акустичне виявлення, передачу, придушення шуму, акустичну дактилоскопію, розпізнавання звуку, синтез і посилення (наприклад, вирівнювання, фільтрація, стиснення рівня).

1.2.1 Аудіотрансляція

Обробка звукових сигналів використовується при ширококомовній передачі звукових сигналів для того, щоб підвищити їх чіткість або

оптимізувати ширину смуги пропускання або затримки. У цій галузі найважливіша обробка звуку відбувається безпосередньо перед трансмітером. Аудіопроцесор тут повинен запобігати або мінімізувати надмірну модуляцію, компенсувати нелінійні трансмітери (потенційна проблема із середньою хвилею та короткохвильовим мовленням) та регулювати загальну нещільність до бажаного рівня [2].

1.2.2 Звукові ефекти

Звукові ефекти — це системи, призначені зміни способу звукового сигналу. Необроблений звук метафорично називають сухим, тоді як оброблений звук називають сирим.

Затримка або відлуння - для імітації ефекту реверберації у великому залі до вихідного сигналу додається один або кілька відкладених сигналів. Щоб сприйматися як відлуння, затримка повинна бути близько 35 секунд або вище.

Для створення незвичайного звуку до вихідного сигналу додається затриманий сигнал з затримкою, що безперервно змінюється (зазвичай менше 10 мс). Цей ефект робиться в електронному вигляді з використанням DSP, але спочатку ефект створювався шляхом відтворення того ж запису на двох синхронізованих *tape-плеєрах*, а потім змішуючи сигнали разом.

Фазер [5] — ще один спосіб створення незвичайного звуку, сигнал розщеплюється, частина фільтрується повним змінним фільтром для отримання фазового зсуву, а потім нефільтрований і фільтрований сигнали змішуються для отримання гребінчастого фільтра. Фазерний ефект спочатку був реалізацією флер-ефекту, оскільки затримки було важко реалізувати з аналоговим обладнанням. До вихідного сигналу додається затримана версія сигналу. Затримка повинна бути короткою, щоб не сприйматися як луна, але вище за 5 мс, щоб бути чутною. Якщо затримка занадто коротка, вона активно взаємодітиме з несвоєчасним сигналом і створює ефект флангування.

Корекція [3] — частотна характеристика регулюється за допомогою аудіофільтра (фільтрів) для отримання бажаних спектральних характеристик. Частотні діапазони можуть бути підкреслені або скориговані за допомогою низькочастотних, високочастотних, смугових або смугових фільтрів. Смужна фільтрація мови може імітувати ефект телефону, оскільки телефони використовують смугові фільтри.

Ефекти надмірного збудження можуть використовуватися для створення звуків дисторсингу та посилення ситості. Найголовніший ефект перевизначення включає сигнал, коли його абсолютне значення має певний поріг.

Регіфікація шкали часу — цей ефект піднімає або опускає сигнал по висоті. Наприклад, сигнал може бути зсунуто на октаву вгору або вниз. Змішування вихідного сигналу з ліцензіатом може створити гармонізацію. Іншим застосуванням регіфікації є регіфікація, де музичний сигнал налаштовується поліпшення інтонації. Доповненням зсуву кроку є тимчасова шкала узгодження, тобто процес зміни швидкості звукового сигналу без впливу на його крок.

Робочі голосові ефекти використовуються для того, щоб голос актора звучав як синтезований людський голос.

Кільцева модуляція — це функція обробки сигналу, реалізація частотного змішування, що виконується шляхом генерування кількох частот із частот двох сигналів, один, як правило, є синусоїдною хвилею або іншим простим сигналом, а інший — сигналом, який підлягає модуляції.

Динамічний стиск діапазону — управління динамічним діапазоном звуку, щоб уникнути ненавмисної або нерозумної флюктуації гучності. Динамічне стиснення діапазону не слід плутати зі стиском аудіоданих, де обсяг даних зменшується без впливу на аплітудність звуку.

3D аудіо ефекти [5] — розміщення звуків поза просторовим діапазоном, доступним через стерео або об'ємне зображення.

Синтез хвильового поля — просторовий аудіо ефект для створення віртуальних акустичних середовищ.

1.2.3 Порівняння аналогової та цифрової обробки звукових сигналів

Щоб оцінити всі переваги цифрового оброблення звукових сигналів, слід розуміти різницю між аналоговою обробкою. Наприклад, розберемося, як відбувається обробка аналогового сигналу. Прикладом обробки аналогового сигналу є ключова RC-схема, показана на рисунку 1.1.

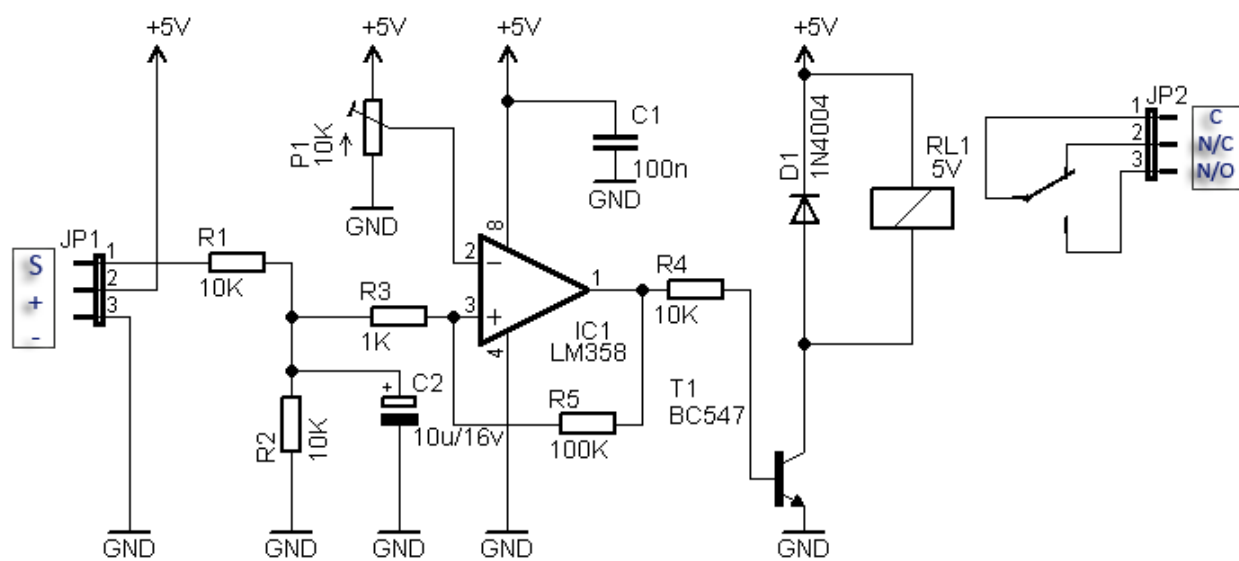


Рисунок 1.1 — Активний аналоговий RC-фільтр НЧ з ключовою схемою

Ця схема діє як фільтр нижніх частот. Він видаляє або відфільтровує частотні складові, які знаходяться вище за частоту зрізу схеми, і пропускає більш низькочастотні складові з невеликим згасанням. У цьому прикладі метою обробки сигналу є усунення високочастотного шуму та виділення потрібної частини сигналу.

Зверніть увагу, що і вхід, і вихід мають аналогову форму. Це велика перевага, тому що сигнали інтересу до науки та техніки мають аналогову природу. Отже, при аналоговій обробці сигналів відпадає потреба в інтерфейсних схемах (АЦП і ЦАП) на вході та виході блоку обробки сигналів.

Одним із основних недоліків обробки аналогових сигналів є зміна вартості електричних компонентів [5]. Аналогові схеми залежать від точності активних та пасивних компонентів (резисторів, конденсаторів, котушок індуктивності та підсилювачів). Наприклад, частота зрізу (f_c) фільтра нижніх частот першого порядку визначається як: $f_c = 1/2 * \pi * RC$.

Як бачите, реакція фільтра є функцією значень компонентів. Оскільки електричні компоненти не можуть бути виготовлені з ідеальною точністю, точність аналогових схем обмежена. Через допуск компонентів продуктивність не відтворюється на 100%, і ми очікуємо деякого відхилення від плати до плати в різних параметрах схеми.

Інший недолік у тому, що аналогові схеми негнучкі. Наприклад, щоб змінити частотну характеристику вищезгаданого фільтра, нам потрібно налаштувати значення компонентів (необхідно змінити обладнання). Це не стосується цифрової обробки сигналів. За допомогою Digital Signal Processor (DSP) можна навіть перетворити фільтр нижніх частот на фільтр верхніх частот, просто змінивши деякі програмовані коефіцієнти.

Понад те, аналогові схеми не підходять реалізації математичних функцій (множення, розподіл тощо. буд.). Це контрастує з цифровою областю, де можуть бути легко реалізовані навіть набагато складніші математичні операції.

Після того, як сигнал оброблений блоком DSP, зазвичай необхідно перетворити його в еквівалентний аналоговий сигнал. Це досягається за допомогою цифро-аналогового перетворювача.

У цьому випадку використовується система цифрової обробки сигналу, щоб додати відлуння або налаштувати темп і висоту голосу для отримання ідеального звуку. Потім оброблений сигнал надходить у ЦАП створення аналогового сигналу, який може виводитися через динаміки. Зверніть увагу, що є програми DSP, яким не потрібний ЦАП. Наприклад, алгоритм цифрової обробки сигналу, що використовується в радарі, може дати нам положення та швидкість літака. Цю інформацію можна просто надрукувати на папері.

Алгоритм DSP складається з низки математичних операцій. Наприклад, фільтр з кінцевою імпульсною характеристикою (КИХ) 4-го порядку вимагає п'яти цифрових помножувачів, чотирьох суматорів вздовж деяких елементів затримки, як показано на рисунку 1.2.

Отже, процесор цифрових сигналів є обчислювальним механізмом. Цей обчислювальний механізм може бути процесором загального призначення, FPGA або спеціально створеним чіпом DSP. Кожен варіант має свої переваги та недоліки з точки зору гнучкості, швидкості, простоти програмування та енергоспоживання.

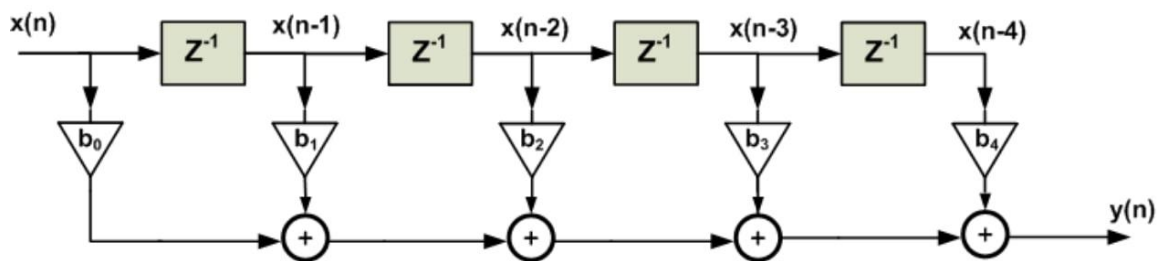


Рисунок 1.2 — Структура цифрового нерекурсивного фільтра, що реалізується в DSP

Оскільки обчислювальні ресурси дуже цінні, цифрова обробка сигналів намагається надати нам інструменти та методи, які дозволяють створювати швидкі та ефективні з обчислювальної точки зору алгоритми. Наприклад, є кілька різних структур, які можна використовувати для реалізації цього КИХ-фільтра.

DSP знаходить застосування у широкому спектрі додатків. Концепції та інструменти DSP знаходять застосування в будь-якій програмі, яка має керувати вхідним сигналом у цифровій області. Це включає, крім іншого, стиснення аудіо та відео, обробку та розпізнавання мовлення, цифрову обробку зображень та радарні програми.

Щоб зробити кар'єру в кожній із цих областей, необхідно володіти широким спектром спеціалізованих алгоритмів DSP, математики та методів. Фактично, дуже мало ймовірно, що хтось один зможе опанувати всю вже

розроблену технологію DSP. Однак деякі загальні концепції DSP, такі як фільтрація, кореляція та спектральний аналіз, використовуються майже у всіх програмах DSP. Отже, перший крок у навчанні DSP — це оволодіння базовими концепціями, а потім зосередження уваги на спеціалізованих методах, необхідних для цієї галузі інтересів.

Деякі базові концепції DSP, а також деякі спеціалізовані методи, пов'язані з цифровою обробкою зображень, описані в попередніх статтях. У мене також є серія статей про ПЛІС та реалізації алгоритмів ЦГЗ на основі ПЛІС, які можуть допомогти вам почати роботу з цим відносно складним предметом.

1.2.4 Особливості обробки аудіо сигналів

Цифрові схеми не страждають від вищезазначених обмежень. Наприклад, хоча варіації у значеннях компонентів та паразитних характеристиках можуть трохи змінити затримку затвора інвертора CMOS, загальна функціональність затвора буде збережена. Отже, на відміну від аналогових схем, цифрові схеми менш схильні до змін компонентів і паразитних впливів. Цифрові схеми також гнучкіші і підходять реалізації математичних функцій [6].

Як показано на рисунку 1.3, потрібні аналого-цифрові (A/D) та цифро-аналогові (D/A) перетворювачі на вході та виході блоку обробки сигналів для взаємодії нашої цифрової схеми з реальним світом. Потім квантує амплітуду кожної вибірки. На рисунку 1.4 показано, як 4-біт АЦП може квантувати аналоговий вхід. Перетворювач A/D здійснює вибірку аналогового входу через рівні проміжки часу, як показано на рисунку 1.4.

На цьому рисунку аналоговий вхід (синя крива) набуває різних значень у вхідному діапазоні АЦП. У 4-бітному АЦП є 16 дискретних рівнів для квантування амплітуди вхідного сигналу. Ці рівні показані малюнку кратними молодшим значним бітам. Отже, LSB (молодший значний біт) визначає мінімальну зміну значення аналогового входу, яке може бути

виявлено АЦП. Іншими словами, це мінімальна зміна входу, що призводить до зміни коду виходу АЦП.

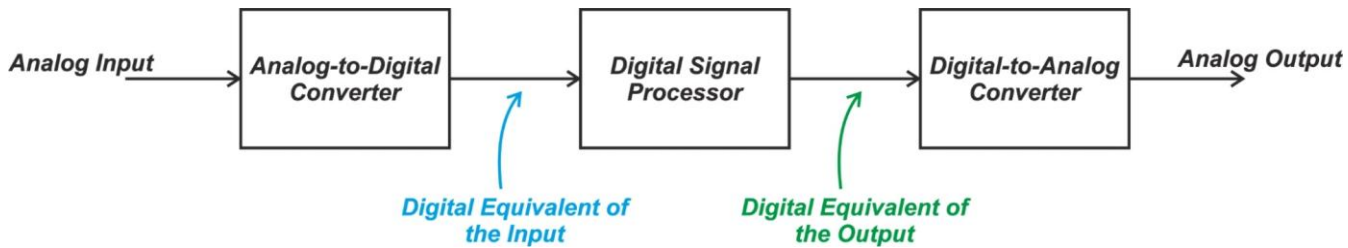


Рисунок 1.3 — Обробка аналогового сигналу засобами DSP

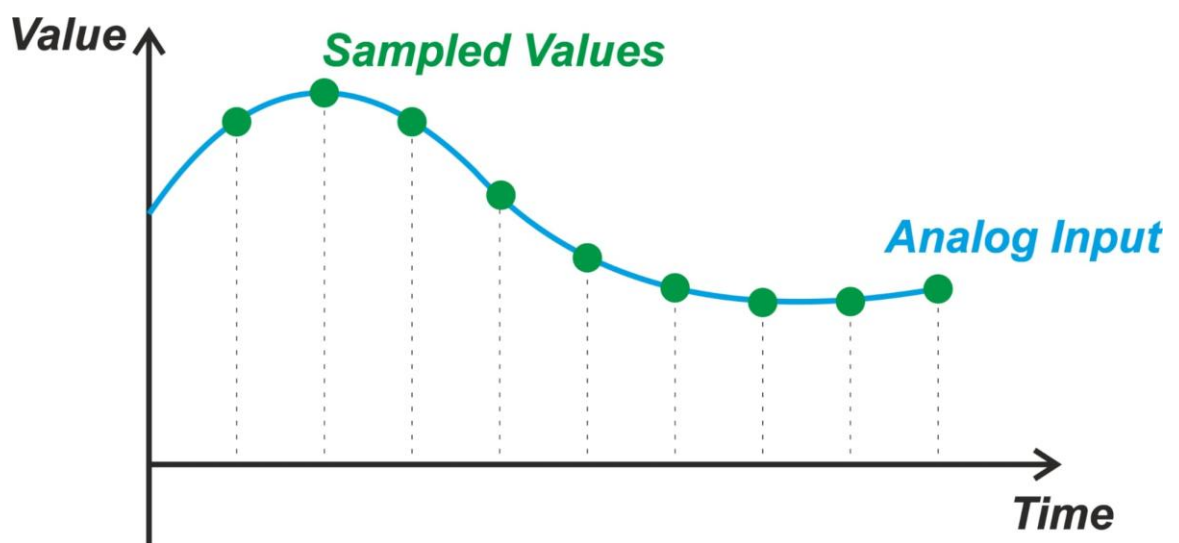


Рисунок 1.4 — Дискретизація сигналу в АЦ-перетворювачі

Розглянемо як АЦП генерує двійковий код кожної вибірки. АЦП порівнює амплітуду аналогового вхідного сигналу з 16 дискретними рівнями. На основі цього порівняння створюється цифрове уявлення введення. Наприклад, із синьою кривою, показаною на рисунку 1.5, процес порівняння вхідного сигналу з 16 дискретними рівнями АЦП може призвести до зображеної червоної кривої. Потім АЦП використовує двійковий код для представлення кожного рівня отриманого сходового наближення. Наприклад, коли крива квантованих вибірок сигналу вчетверо більше молодшого розряду, вихідний сигнал нашого чотирибітного АЦП буде 0100.

Слід зазначити, що блок цифровий сигнальний процесор на рисунку 1.2 приймає дискретну тимчасову послідовність, тому що АЦП бере вибірки з

кратністю заздалегідь заданого інтервалу вибірки. І амплітуда кожної вибірки квантується. Це відрізняється від обробки аналогового сигналу, коли вхідний сигнал є безперервним і може набувати будь-якого значення в заданому діапазоні.

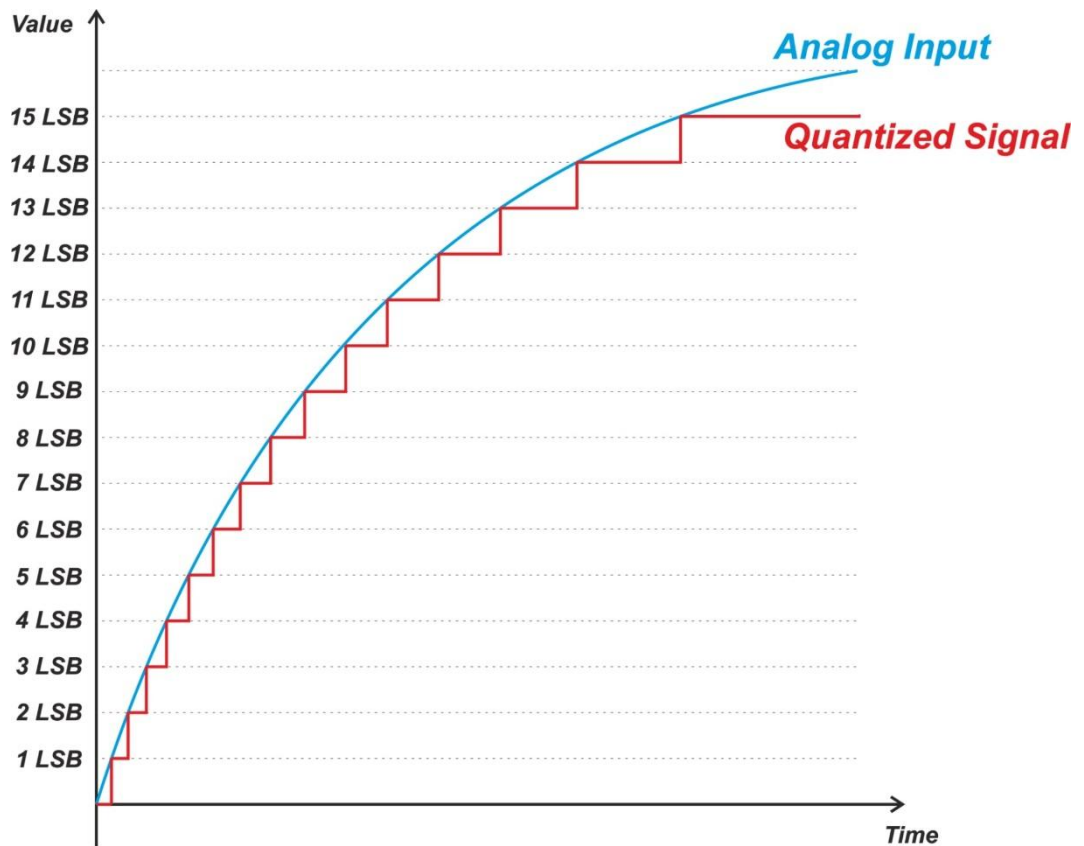


Рисунок 1.5 — Квантування входу 4-бітного АЦП

Сьогодні використовуються п'ять основних типів АЦП [6]:

- АЦП послідовного наближення (SAR);
- дельта-сигма ($\Delta\Sigma$) АЦП;
- двоканальний АЦП;
- конвеєрний АЦП;
- флеш-АЦП.

Кожна технологія має особливості та можливості, які сприяють її використанню на ринку. З АЦП вони включають [6]:

- частота дискретизації;
- бітова роздільна здатність.

У мікропроцесорних системах аналого-цифровий перетворювач (АЦП) виконує функцію імпульсного елемента, а цифро-аналоговий перетворювач (ЦАП) — екстраполятор.

Аналого-цифрове перетворення полягає в перетворенні інформації, що міститься в аналоговому сигналі, в цифровий код. Цифро-аналогове перетворення призначене для виконання зворотного завдання, тобто перетворити число, представлене у вигляді цифрового коду, в еквівалентний аналоговий сигнал.

На вхід АЦП подається сигнал як струму або напруги, який у процесі перетворення квантується за рівнем. Ідеальна статична характеристика 3-розрядного АЦП наведено на рисунку 1.7.

Існують різні методи аналого-цифрового перетворення, які відрізняються точністю і швидкістю. У більшості випадків ці властивості антагоністичні один одному. Нині широко застосовуються такі типи перетворювачів, як АЦП з послідовними наближеннями (порозрядне вирівнювання), інтегруючі АЦП, паралельні (флеш) АЦП, АЦП «сигма-дельта» та ін.



Рисунок 1.6 — Структурна схема корелятора сигналів з однаковою енергією

Блок-схема АЦП з послідовним наближенням показана на рисунку 1.8.

Основними елементами пристрою є компаратор (К), цифро-аналоговий перетворювач (ЦАП) і логічна схема управління. Принцип перетворення заснований на послідовному порівнянні рівня вхідного сигналу з рівнями сигналу, що відповідають різним комбінаціям вихідного коду, і формуванні

результуючого коду за результатами порівняння. Послідовність порівнюваних кодів відповідає правилу половини поділу. На початку перетворення вхідний код ЦАП встановлюється в стан, в якому всі біти, крім старшого, дорівнюють 0, а старший дорівнює 1. При такій комбінації на виході ЦАП створюється напруга, що дорівнює половині діапазону вхідної напруги. Ця напруга порівнюється з вхідною напругою на компараторі. Якщо вхідний сигнал більше, ніж сигнал, що надходить від ЦАП, старший біт вихідного коду встановлюється на 1, інакше він скидається на 0. Під час наступного тактового циклу частково сформований таким чином код повертається на вхід ЦАП, наступний біт в ньому встановлюється в одиницю і порівняння повторюється. Процес триває до тих пір, поки не буде порівняно найменший біт. Це для формування N -бітового вихідного коду необхідно N ідентичних елементарних циклів порівняння. Це означає, що за інших рівних умов швидкість такого АЦП зменшується зі збільшенням потужності. Внутрішні елементи АЦП послідовної апроксимації (ЦАП і компаратор) повинні мати точність, кращу за значення половини найменшого розряду АЦП.

Для порівняння, вхідна напруга подається на однойменні входи відразу на компаратори $N-1$. Сигнали від високоточного дільника напруги, який підключений до джерела опорної напруги, надходять на протилежні входи компараторів. При цьому напруги на виходах дільника рівномірно розподіляються по всьому діапазону зміни вхідного сигналу. Пріоритетний кодер формує цифровий вихідний сигнал, який відповідає найвищому компаратору з активованим вихідним сигналом. Таким чином Для забезпечення N -бітового перетворення необхідні дільники $2N$ і компаратор $2N-1$. Це один з найшвидших методів перетворення. Однак, коли розрядність велика, це вимагає високої вартості обладнання. Точність усіх резисторів дільників і компараторів, у свою чергу, повинна бути кращою половини найменшого розряду.

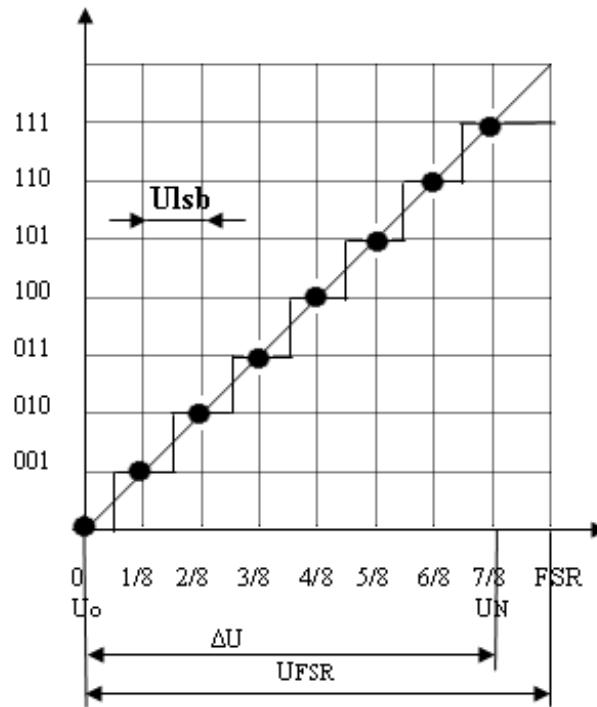


Рисунок 1.7 — Передатна характеристика 3-розрядного АЦП



Рисунок 1.8 — АЦП з послідовним наближенням

Блок-схема АЦП з подвійним інтегруванням показана на рисунку 1.9.

АЦП характеризується рядом параметрів, що дають можливість реалізувати вибір того чи іншого пристрою виходячи з вимог системи. Усі параметри АЦП можна розділити на дві групи: статичні та динамічні. Перші визначають характеристики точності приладу при роботі з постійним або повільно мінливим вхідним сигналом, а другі характеризують швидкість

роботи приладу як збереження точності при збільшенні частоти вхідного сигналу.

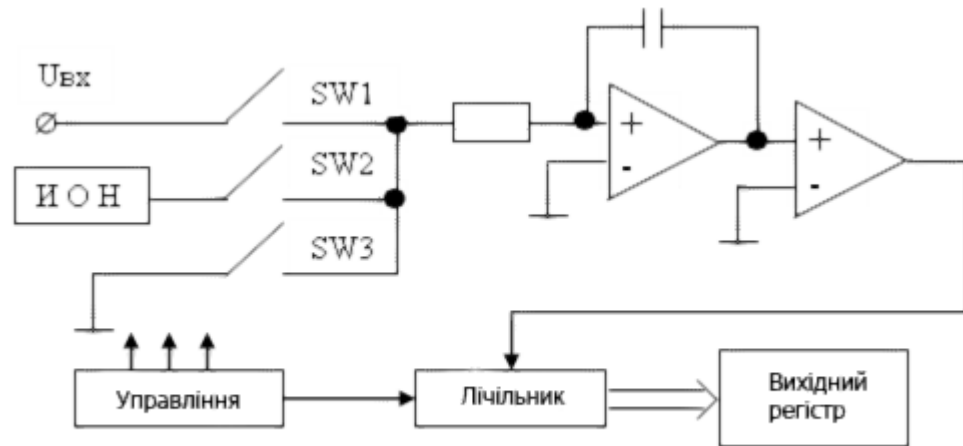


Рисунок 1.9 — АЦП з подвійним інтегруванням

1.3 АЦП та ЦАП у інструментальних пристроях

Прикладом обробки звуку в інструментальних процесорах може бути будь-який гітарний кабінет [4]. Перша та найголовніша відмінність від звичайного динаміка, це те, що він має дуже сильний завал на частотах понад 5КГц. Хоча звичайна колонка має діапазон до 20 КГц. Гітара – інструмент середньочастотний і не має жодного корисного музичного сигналу на частотах понад 5КГц. Типова частотна характеристика гітарного динаміка показана на рисунку 1.10. Під час гри гітарний сигнал створює піки та впадини в синусоїді.

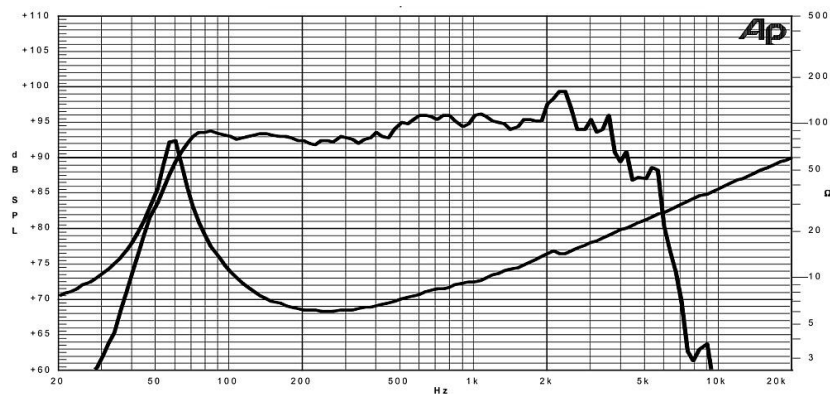


Рисунок 1.10 — Частотна характеристика динаміка гітарного кабінету

1.3.1 Параметри та класифікація інструментальних пристроїв обробки звуку

Всі пристрої обробки звуку можна умовно розділити на 3 групи [3]:

- пристрої, які не вносять у сигнал додаткових складових (аудіопроекторні блоки);
- пристрої, що вносять у сигнал додаткові складові (звукові ефекти);
- пристрої, що синтезують нові сигнали на основі параметрів вихідного сигналу (вокодери).

Аудіопроекторні блоки. До них відносяться блоки затримки, еквалайзери, кросовери та компресори.

Необхідність у блоках затримки з'явилася у 40-х роках ХХ століття, коли у кіно став використовуватися стереозвук. Як відомо, людина сприймає звук як сукупність сигналів, що надходять кожному з вух. Аналізуючи затримку звукової хвилі, що надходить до кожного вуха, наш мозок з легкістю визначає місце розташування джерела звуку.

За допомогою блоку затримки, що використовується, наприклад, в одному з каналів, можна імітувати зміну розташування джерела звуку щодо слухача. Вочевидь, що з формуванні просторових ефектів, слухач охоче втягується у звукову картину. Взагалі, затримка звуку – природне явище, пов'язане з тим, що швидкість поширення звукової хвилі є відносно невисокою.

Напевно всім знайомий ефект луни, що виникає при відображенні хвилі від перешкоди та подолання нею зворотного шляху. Відмінність ефекту, реалізованого з допомогою блоку затримки, у тому, що «відбитий» сигнал нічим відрізняється від вихідного. У реальних умовах спектр сигналу при відображенні суттєво змінюється, тому що різні його складові по-різному відбиваються від перешкод.

Нині блоки затримки широко використовуються у складі аудіопроцесорів. Вони служать для вирівнювання звукового поля у великих та складних приміщеннях, у конференц-залах, а також для створення звукових ефектів, таких як Echo, Delay, Reverberation та ін. Час затримки може регулюватися від одиниць мікросекунд до десятків секунд. Аналогові лінії затримки є досить складними і зараз не використовуються. Принцип дії цифрових блоків полягає у записі двійкового представлення сигналу на згадку і наступному відтворенні цієї інформації, але із затримкою, яка програмно регулюється та встановлюється [2].

Еквалайзери (від англ. equalize – вирівнювати) з'явилися досить давно, і їхня історія розвитку нерозривно пов'язана з розвитком електричних фільтрів. Вони призначені для вирівнювання амплітудно-частотних характеристик електричних сигналів. Які фільтрують властивості електричних ланцюгів до 30-х років ХХ століття вже широко використовувалися для компенсації втрат у каналах передачі сигналів. Першим, хто впровадив еквалайзер у звукопідсилювальну систему, був Джон Волкман [2].

EQ-251A використовувався в кіно для корекції звуку і мав лише два повзунки та перемикач вибору частоти для кожного з них. Вже тоді застосування еквалайзерів було необхідне, враховуючи низьку якість звукозаписної та звуковідтворювальної техніки. Після Другої світової війни було розроблено багатосмугові еквалайзери, що використовують фільтри з високою добротністю. В даний час ці пристрої широко використовуються як у побутових, так і професійних системах. Найпростіший, усім відомий еквалайзер – регулятор тембру, який включає 2 фільтри: НЧ та ВЧ. Схема роботи типового регулятора тембру показана на рисунку 1.11.

Принцип дії еквалайзера полягає у роздільному регулюванні рівнів складових сигналу на різних частотах. Звуковий сигнал має багатий спектр, який за допомогою смугових фільтрів поділяється на складові. Це дає можливість виділити певні частоти, що становлять основу звучання

музичного інструменту, або прибрати високочастотні складові, такі як звук торкання пальцем струни, зробити темброве фарбування м'якшим або, навпаки, жорсткішим [2]. Функціональна схема типового еквайзера показана на рисунку 1.12.

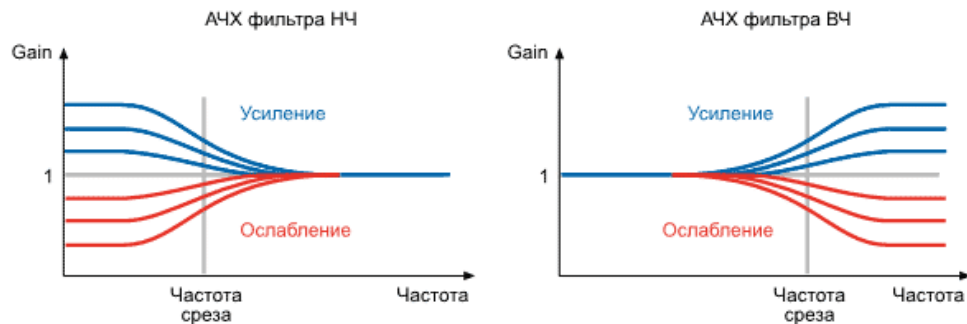


Рисунок 1.11 — Схема роботи регулятора тембру

Необхідно пам'ятати, що еквайзер ніколи не додасть до звучання того, чого немає у вихідному сигналі, а значить, він не вирішить проблеми самого сигналу. Тому професіонали вважають за краще такі пристрої не використовувати або використовують їх дуже обережно. Недбале поводження з еквайзером призводить до неприродності звучання. Наприклад, видалення з сигналу, що знімається з гітари, низьких частот, може покращити якість запису при зведенні його з іншими партіями (низькі частоти хоч і не притаманні гітарі, але присутні в спектрі сигналу і можуть маскувати інші інструменти, впливаючи на їх звучання).

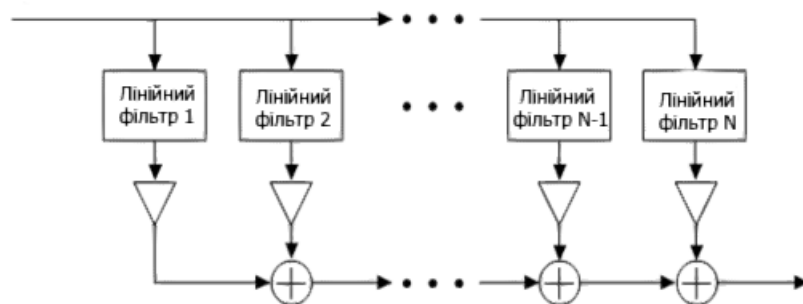


Рисунок 1.12 — Функціональна схема еквайзера

Однак, навіть незначне приглушення високочастотних складових (від 1 до 5 кГц) у тому ж сигналі, призведе до втрати чіткості та яскравості гри, зникнуть деталі. Візуально це виявляється у згладжуванні сигналу, але в слух сприймається як тьмяність і зайва тонкість звучання. Непрофесійний підхід до еквалізації вокальної партії може викликати не лише втрату виразності та енергійності голосу, а й навіть зміну літер та закінчень.

Розрізняють графічні та параметричні еквалайзери [2]. Графічні відрізняються більшою наочністю, користувач за допомогою численних повзунків (регуляторів), кожен з яких відповідає фіксованій частоті смугового фільтра, визначає рівень посилення або ослаблення сигналу у вузькій смузі частот. Центральні частоти фільтрів у графічних еквалайзерах зазвичай встановлюються за октавами. Для вибору центральних частот, до речі, є стандарт ISO.

При досить велику кількість смуг на панелі управління еквалайзера цілком чітко вимальовується графік його амлітудно-частотної характеристики. Тому такі еквалайзери називаються графічними. Їх недолік у тому, що користувач немає можливості самостійно налаштувати смугові фільтри, тобто. встановлювати їх центральні частоти та добротність.

Параметричний еквалайзер дозволяє регулювати параметри смугових фільтрів: центральну частоту, добротність (смуга пропускання) та власне посилення (ослаблення). Вони просто незамінні, коли треба вирізати вузьку ділянку спектра сигналу, на якому, наприклад, через акустичний зворотний зв'язок, система звуковідтворення починає збуджуватися. Кожен з нас, напевно, чув характерний свист з акустичних систем при піднесенні до них мікрофона. Параметричні еквалайзери складніші в налаштуванні та дещо дорожчі. Кількість смуг у них, як правило, набагато менша, ніж у графічних.

В даний час використовуються графічні та параметричні еквалайзери, причому як цифрові, так аналогові та гібридні. Ці пристрої можуть виконуватися як у вигляді окремих блоків, так і бути вбудовані в мікшери, попередні підсилювачі, програмне забезпечення для роботи зі звуком.

Компресор призначений для стиснення динамічного діапазону сигналу. Іншими словами, він зменшує різницю між найгучнішими і тихими звуками. У недавньому минулому функції компресора виконував звукооператор, передбачаючи зміни гучності, наприклад, вокаліста та регулюючи відповідним чином гучність відтворення.

Історія компресорів розпочалася під час Другої світової війни. Тоді, через різкі стрибки рівня гучності в переданому сигналі, часто виходили з ладу радіостанції. З того часу компресор використовується практично у всіх радіостанціях, у тому числі і в широкомовних. Крім того, пристрій знайшов застосування у звукових системах. Робота компресорів допомагає підвищити розбірливість мовлення під час роботи з мікрофонами, збільшити цифровий дозвіл, а значить, і відносини сигнал/шум, надати «щільність» звучання голосу та інструментам.

Результат обробки звуку компресором ледь помітний на слух. Але при правильній роботі з ним тьмянний звук можна зробити гострішим і більш насиченим, голос зробити жорсткішим, надати йому нехарактерні для виконавця риси, а дешеві музичні інструменти виставити у вигіднішому світлі. Однак, через нестачу кваліфікації, звук можна непоправно зіпсувати.

Різновидів компресорів, як та інших блоків обробки звуку, безліч – цифрові та аналогові, апаратні та програмні, ручні та автоматичні, оптико-електронні, лампові, транзисторні тощо.

Компресори також можуть бути частотно залежними або багато смуговими [5]. Багатосмуговий компресор, по суті, поєднує в собі кросовер та багатоканальний компресор. Обробка сигналу здійснюється окремо для різних частотних складових, що корисно при зведенні музичної композиції. Частотно-залежний компресор піддає сигнал обробці тільки заданому вузькому діапазоні частот. Окремим випадком частотно-залежного компресора є de-esser.

В даний час у чистому вигляді компресор не використовується, він доповнюється безліччю перерахованих функцій, пов'язаних з динамічною

обробкою сигналу. Наприклад, CN-9102 Inter-M, крім функцій 2-х канального компресора, виконує функції лімітера (limiter) та шумоподавлювача (noise gate).

Лімітер є окремим випадком компресора з встановленим ступенем компресії.

Шумоподавлювач використовується для створення комфортних пауз між сигналами. Він також закриває вхід при падінні рівня сигналу нижче за порогове значення, яке встановлюється вручну або автоматично виходячи з відношення сигнал/шум [6].

При установці надто високого порога noise gate threshold сигнал може суттєво втратити інформативність, що виражається у зменшенні тривалості, стрімкості звучання.

Серед інших пристроїв динамічної обробки звуку можна виділити компресор, що підвищує, і експандер. Підвищує компресор, на відміну розглянутого вище, посилює сигнал при падінні його рівня нижче встановленого порогового рівня. Експандер – це пристрій, який розширює динамічний діапазон сигналу, тобто протилежний компресору. Експандер дозволяє покращити макродинаміку твору, надати жвавості деяким інструментам, а також у ряді випадків виправити перекомпресію сигналу.

2 АНАЛІЗ МЕТОДІВ І ЗАСОБИ ОБРОБКИ СИГНАЛІВ ДЛЯ АУДИОПРОЦЕСОРІВ З ДИСТАНЦІЙНИМ КЕРУВАННЯМ

2.1 Огляд архітектури

Архітектура сигнальних процесорів, в порівнянні з мікропроцесорами настільних комп'ютерів, має деякі особливості:

- гарвардська архітектура (розділення пам'яті команд і даних), як правило модифікована;
- більшість сигнальних процесорів мають вбудовану оперативну пам'ять, з якої може здійснюватися вибірка декількох машинних слів одночасно
- не рідко вбудовано відразу декілька видів оперативної пам'яті, наприклад, через Гарвардську архітектуру буває окрема пам'ять для інструкцій і окрема — для даних;
- деякі сигнальні процесори володіють одним або навіть декількома вбудованими постійними пристроями, що запам'ятовують, з найбільш споживаними підпрограмами, таблицями і тому подібне;
- апаратне прискорення складних обчислювальних інструкцій, тобто швидке виконання операцій, характерних для цифрової обробки сигналів, наприклад, операція «множення з накопиченням» (MAC) ($Y := X + A \times B$) зазвичай виконується за один такт;
- «вільні» за часом цикли із заздалегідь відомою довжиною. Підтримка векторно-конвеєрної обробки за допомогою генераторів адресних послідовностей;
- детермінована робота з відомими часом виконання команд, що дозволяє виконувати планування роботи в реальному часі;
- порівняно невелика довжина конвеєра, так що незаплановані умовні переходи можуть займати менший час, чим в універсальних процесорах;

- спеціалізований набір регістрів і інструкцій, часто складних для компіляторів. Деяка архітектура використовує VLIW;
- в порівнянні з мікроконтролерами, обмежений набір периферійних пристроїв - втім, існують «перехідні» чіпи, що поєднують в собі властивості DSP і широку периферію мікроконтролерів;
- цифрові сигнальні процесори зазвичай споживають істотно менше потужності, чим еквівалентні по продуктивності процесори загального призначення.

2.2 Розвиток сигнальних процесорів

До 1980 року декілька компаній випустили пристрої, які можна вважати попередниками цифрових сигнальних процесорів (ЦСП). Так, в 1978 Intel випускає «процесор аналогових сигналів» 2120. У його склад входили АЦП, ЦАП і процесор обробки цифрових даних, проте апаратна функція множення була відсутня. У 1979 АМІ випускає S2811 - периферійний пристрій, керований основним процесором комп'ютера. Обидва виробів не досягли успіху на ринку.

Основну історію ЦСП прийнято відлічувати від 1979-1980 років, коли Bell Labs представила перший однокристальний ЦСП Mac 4, а також на «IEEE International Solid-State Circuits Conference '80» були показані μ MPD7720 компанії NEC і DSP1 компанії AT&T, які, проте, не набули широкого поширення. Стандартом де-факто став випущений ледве пізніше кристал TMS32010 фірми Texas Instruments, по багатьом параметрам і вдалим технічним рішенням що перевершує вироби конкурентів.

Ось деякі його характеристики АЛУ:

- розмір слова: 16 біт;
- розрядність обчислювача: 32 біт;
- швидкодію: 5 млн. операцій складання або множення в секунду;
- Тривалість командного циклу: 160-280 нс;
- ОЗУ: 144-256 слів;

- ПЗП програм: 1,5-4 До слів;
 - ППЗУ: до 4К слів (окремі моделі);
- Пристрої введення-виводу: 8 портів по 16 розрядів;

Завдяки прогресу в напівпровідникових технологіях в 1970-80 роках були випущені вироби, що мають розширені функції в порівнянні з першим поколінням. До характерних відмінностей можна віднести:

- збільшення об'єму ОЗУ до 0,5 До слів;
- додана можливість підключення зовнішньої пам'яті програм і зовнішньої пам'яті даних об'ємом до 128 До слів;
- швидкодія підвищена в 2-4 рази;
- покращені підсистеми переривань і введення-виводу.

Набагато пізніше також були випущені пристрої, що формально відносяться до другого покоління, але наступні удосконалення, що мають:

- збільшена розрядність даних;
- знижена напруга живлення і, як наслідок, енергоспоживання;
- введені режими економії енергії;
- апаратна підтримка мультипроцесора (система спільного доступу до зовнішньої пам'яті);
- апаратна підтримка кільцевих буферів;
- апаратна підтримка операцій циклів;
- розширені способи адресації;
- дві внутрішні шини даних, що дозволяє значно прискорити парну обробку даних (координати X/Y, дійсна і уявна частина і т. д.), або віртуально подвоїти розрядність оброблюваних даних;
- введена кеш-пам'ять.

Третє покоління ЦСП (1980-90 роки) прийнято пов'язувати з початком випуску виробів, що реалізують арифметику з плаваючою комою. Характерні особливості перших випущених зразків:

- продуктивність: приблизно 20-40 млн. операцій/сек. (MIPS);

- два блоки ОЗУ по 1 До 32-розрядних слів з можливістю одночасного доступу;
- кеш-пам'ять об'ємом 64 словом;
- розрядність регістрів: 32 біт;
- розрядність АЛУ: 40 біт;
- регістри для операцій з підвищеною точністю;
- вбудовані контролери ПДП;
- розрядність шин: 32 біт для команд і 24 біт для адреси.

Четверте покоління ЦСП характеризується значним розширенням наборів команд, створенням VLIW і суперскалярних процесорів. Помітно зросли тактові частоти. Так, наприклад, час виконання команди MAC ($Y := X + A \times B$) удалося скоротити до 3 нс.

Сучасні ЦСП. Кращі сучасні ЦСП можна характеризувати наступними параметрами:

- тактова частота - 1 ГГц і вище;
- багатоядерність;
- наявність дворівневого кешу;
- вбудовані багатоканальні контролери прямого доступу до пам'яті;
- швидкодія близько декількох тисяч MIPS і MFLOPS;
- виконання до 8 паралельних інструкцій за такт;
- сумісність із стандартними шинами (PCI і ін.).

2.3 Аналіз основних параметрів ЦСП

ЦСП діляться на процесори, оброблювальні дані з фіксованою крапкою і оброблювальні дані з плаваючою крапкою. Пристрої з плаваючою крапкою зручніше у вживанні, але вони помітно складніше по пристрою і більш за дорогу.

Більшість ЦСП з фіксованою крапкою обробляють дані з розрядністю 16 біт, процесори з плаваючою крапкою — 32 біта. Багато моделей можуть обробляти дані з подвійною точністю.

Швидкодія як інтегральну характеристику визначити досить складний, тому швидкість роботи характеризують декількома параметрами, а також часом вирішення деяких реальних завдань.

Тактова частота і час командного циклу. Для сучасних ЦСП тактова внутрішня частота може відрізнятися від зовнішньої, тому можуть вказуватися два значення. Час командного циклу вказує на час виконання одного етапу команди, тобто час одного циклу конвеєра команд. Оскільки команди можуть виконуватися за різну кількість циклів, а також з врахуванням можливості одночасного виконання декількох команд, цей параметр може характеризувати швидкодію ЦСП досить наближений.

Кількість виконуваних команд за одиницю часу. Різний час виконання команд, а також виконання декількох команд одночасно не дозволяють використовувати цей параметр для надійної характеристики швидкодії.

Кількість виконуваних операцій за одиницю часу (MIPS). Даний параметр враховує одночасну обробку декількох команд і наявність паралельних обчислювальних модулів, тому досить добре може вказувати на швидкодію ЦСП. Деякою проблемою тут залишається те, що поняття «операції» чітко не формалізоване.

Кількість виконуваних операцій з плаваючою крапкою за одиницю часу. Параметр аналогічний попередньому і використовується для процесорів з плаваючою крапкою.

Кількість виконуваних операцій MAC за одиницю часу. Дана команда, з одного боку, є базовою для багатьох обчислень, а з іншої — досить проста. Тому час її виконання можна використовувати у тому числі і для оцінки загальної продуктивності ЦСП.

Види і об'єм внутрішньої пам'яті. Об'єм внутрішньої оперативної пам'яті показує, скільки даних ЦСП може обробити без звернення до зовнішньої пам'яті, що може характеризувати загальну швидкодію системи а також можливість працювати «в реальному часі». Тип ПЗП визначає можливості по програмуванню пристрою. Моделі із звичайним ПЗП лічать

для серійного виробництва, ППЗУ (однократно програмоване) зручно для невеликих накладів, а вживання Flash-пам'яті дозволяє міняти програму пристрою багато разів під час експлуатації. На даний момент не дуже потужні DSP найчастіше забезпечені чималою Flash-пам'яттю (її ціна неухильно знижується) і помітним об'ємом RAM і тому можуть бути самодостатніми без додавання зовнішній пам'яті і на етапі розробки і на етапі виробництва, що підвищує конкурентоспроможність таких DSP у ряді сегментів ринку. Потужні DSP як правило покладаються на зовнішню пам'ять підключену по досить швидких шинах а розміщення там Flash-пам'яті може бути технічно проблематичним, наприклад, із-за помітного виділення тепла.

Об'єм пам'яті, що адресується. Об'єм зовнішньої пам'яті, що адресується, характеризується шириною зовнішньої шини адреси.

Кількість і параметри портів введення-виводу. Даний параметр показує можливості ЦСП по взаємодії із зовнішніми по відношенню до його пристроями.

Склад внутрішніх додаткових устроїв. У число внутрішніх можуть входити всілякі за призначенням пристрої, наприклад, загального вживання - таймери, контролери ПДП і т. д., а також проблемно-орієнтовані — АЦП, кодеки, компресори даних та інші.

Напруга живлення і споживана потужність. Дана характеристика особливо важлива для ЦСП, що вбудовуються в переносні пристрої. Зазвичай переважно низьковольтні пристрої (1,8-3,3В), які мають швидкодію аналогічно 5В процесорам, але помітно економніше в плані вжитку енергії. Багато пристроїв мають режими економії при просте, або дозволяють програмно відключати частину своїх пристроїв.

Склад і функціональність засобів розробки і підтримки:

- перелік мов програмування, для яких є компілятори під дану систему;
- наявність і можливості засобів відладки готових програм;
- доступність документації і технічної підтримки;

- наявність бібліотек стандартних підпрограм і математичних функцій;
- наявність, доступність і можливості сумісних пристроїв - АЦП, ЦАП, контролери живлення і так далі;
- допустимі параметри довкілля;
- інші, залежно від призначення.

Часто використовуються також інтегральні характеристики ЦСП, наприклад показник «потужність/струм/швидкодія», наприклад mA/MIPS (міліампер на 1 млн. інструкцій в секунду), що дозволяє оцінити реальну споживану потужність залежно від складності завдання, що вирішується процесором у вказаний момент.

Вибір ЦСП цілком визначається призначенням системи, що розробляється. Наприклад, для масових мобільних пристроїв важлива дешевизна процесора, низьке енергоспоживання, тоді як вартість розробки системи відходить на другий план. З іншого боку, для вимірювального устаткування, систем обробки звукової і відеоінформації важливі ефективність процесора, наявність розвинених інструментальних засобів, багатопроцесорність і так далі.

2.4 Апаратна реалізація ЦСП

2.4.1 Гарвардська архітектура

Цифрові сигнальні процесори будуються на основі т.з. «Гарвардської архітектури», відмітною особливістю якої є те, що програми і дані зберігаються в різних пристроях пам'яті - пам'яті програм і пам'яті даних. На відміну від архітектури фон Неймана, де процесору для вибірки команди і двох операндів потрібно мінімум три цикли шини, ЦСП може виробляти одночасні звернення як до пам'яті команд, так і до пам'яті даних, і вказана вище команда може бути отримана за два цикли шини. У реальності, завдяки продуманості системи команд і іншим заходам, цей час може бути

скорочений до одного циклу. У реальних пристроях пам'ять команд може зберігати не лише програми, але і дані. В цьому випадку говорять, що ЦСП побудований по модифікованій гарвардській архітектурі. Пам'ять команд і пам'ять даних зазвичай розташовуються на кристалі ЦСП. У зв'язку з тим, що ця пам'ять має відносно невеликий об'єм, виникає необхідність у використанні зовнішніх (відносно кристала процесора) пристроїв, що запам'ятовують. Для таких пристроїв роздільні шини команд і даних не використовуються, так це зажадало б значно збільшити кількість зовнішніх виводів кристала, що дорого і непрактично. Тому взаємодія ЦСП із зовнішніми пристроями, що запам'ятовують, відбувається по одному комплекту шин без розділення на команди і дані. Слід також відмітити, що звернення до зовнішньої пам'яті завжди займає значно більше часу, чим до внутрішньої, тому в додатках, критичних до часу виконання, такі звернення необхідно мінімізувати.



Рисунок 2.1 — Функціональна схема процесорів гарвардської архітектури.

2.4.2 Стандартні ЦСП

На рисунку показано два варіанти виконання команди МАС на стандартному ЦСП. У першому варіанті обидва операнда зберігаються в пам'яті даних, тому на їх вибірку потрібно два такти, тобто час виконання п

складань рівний $2n$. У другому випадку один з операндів зберігається в пам'яті програм, тому команда виконується за один такт, і загальний час виконання циклу буде рівний n тактів (слід уточнити, що в реальності для виконання за один такт МАС повинна виконуватися усередині спеціальної команди циклу для виключення повторної вибірки самої коди команди, що вимагає додаткового такту). Тут видно, що ефективна реалізація алгоритму вимагає використання пам'яті програм для зберігання даних, а максимальна швидкодія досягається лише в однокомандному циклі. Одним з варіантів, що дозволяє відмовитися від використання пам'яті програм для зберігання даних, є вживання т.з. «двопортової пам'яті», тобто пам'яті, що має два комплекти вхідних шин - двох шин адреси і даних. Така архітектура дозволяє виробити одночасне звернення по двох адресах (правда, при цьому вони повинні знаходитися в різних блоках, що адресуються). Дане рішення застосовується в ЦСП компаній Motorola (DSP56000) і Lucent (DSP1600).

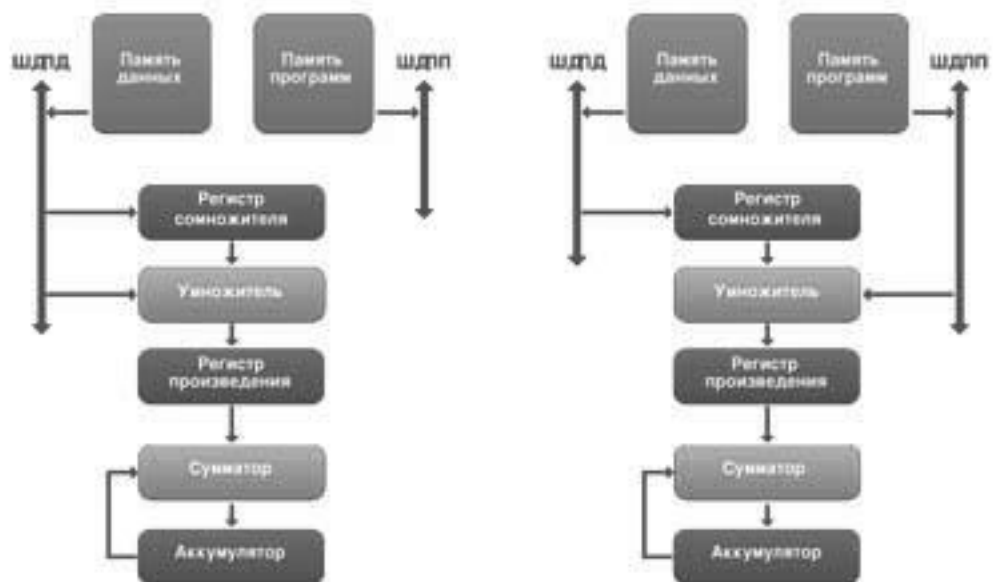


Рисунок 2.2 — Функціональна схема процесорів стандартної архітектури.

При вказаній архітектурі підвищити продуктивність можна лише збільшенням тактової частоти.

2.4.3 Покращені стандартні ЦСП

«Покращені стандартні ЦСП» для підвищення продуктивності системи, в порівнянні із стандартними ЦСП, використовують наступні методи підвищення паралелізму:

- збільшення кількості операційних і обчислювальних пристроїв;
- введення спеціалізованих співпроцесорів;
- розширення шин для збільшення об'єму передачі даних;
- використання пам'яті з багатократним доступом (декілька звернень за такт);
- ускладнення системи команд.

Багато хто з цих способів застосовувався вже починаючи з найперших процесорів, тому частенько їх неможливо однозначно класифікувати як «стандартні» або «покращенні».

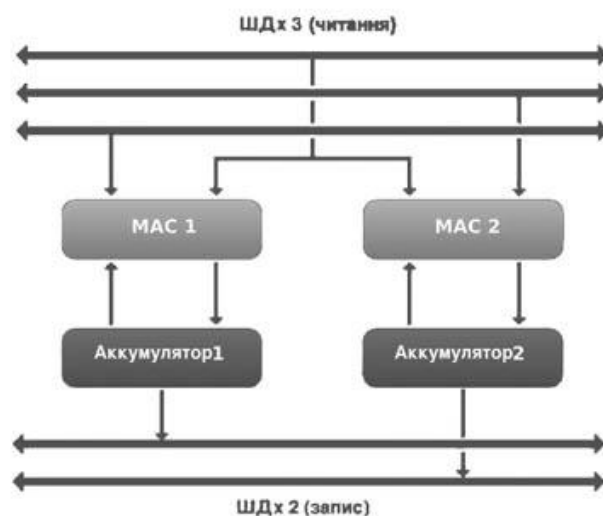


Рисунок 2.3 — Функціональна схема процесорів покращеної архітектури.

На рис. 2.3 показаний приклад реалізації обчислення двох паралельних команд МАС. Для цього в ЦСП є два модулі МАС і два акумулятори. Блоки МАС отримують дані по трьох шинах одночасно, причому одне із значень є

для них загальним. Таким чином, відбувається одночасного виконання двох команд:

$$AK1 := AK1 + D1 \times D2$$

$$AK2 := AK2 + D1 \times D3$$

Особливість показаного рішення полягає в тому, що до виконання двох паралельних команд з одним загальним співмножником можна звести багато алгоритмів ЦОС, наприклад: фільтр з симетричними коефіцієнтами. Як однакові співмножники використовуються коефіцієнти фільтру, а на роздільні шини подаються два різні набори відліків сигналу, тобто паралельно розраховуються дві половини фільтру, які потім підсумовуються.

На загальну шину подаються відліки сигналу, а на роздільних - набори коефіцієнтів. У деяких процесорах (Lucent DSP16xxx, ADI ASDP-2116x) використовуються два однакові ядра, кожен зі своєю пам'яттю, тобто одна команда виконується одночасно в двох ядрах з різними даними. Це дозволяє обійти обмеження на використання повністю незалежних даних. Характерним недоліком таких процесорів можна вважати необхідність у високій кваліфікації розробника, оскільки ефективне використання вказаних особливостей вимагає програмування на мові асемблера, хорошого знання архітектури і системи команд, тобто ці пристрої вважаються «недружніми» до мов високого рівня.

2.4.4 ЦСП з архітектурою VLIW

Основна відмінність VLIW-процесорів полягає в тому, що коди команд ще на етапі компіляції збираються у великі «суперкоманди» і виконуються паралельно. Зазвичай такі процесори використовують RISC-архітектуру з фіксованою довжиною команди, де кожна з них виконується в окремому операційному модулі. До характерних особливостей таких процесорів можна віднести: великий набір операційних модулів, що працюють незалежно один від одного. До складу таких модулів можуть входити:

арифметичних:

- модулі арифметичних операцій і операцій порівняння;
- модулі логічних операцій;
- модулі множення чисел з плаваючою і фіксованою крапкою;
- модулі генерації констант;
- модулі генерації адреси, у тому числі для лінійних і циклічних

буферів;

- необхідність оптимізації компілятора під кожен модель процесора, оскільки між моделями може мінятися склад і функції обчислювальних блоків, що спричиняє за собою зміну переліку команд, які можуть виконуватися одночасно;

- необхідність наявності надшироких шин даних (близько 128 біт), аби код операції, що складається з окремих команд (до 8-мі) міг бути отриманий з пам'яті за одне звернення;

- високі вимоги до об'єму пам'яті програм, що також пов'язане з великою довжиною операції.

Зазвичай, якщо процесор має декілька однакових модулів, то при створенні програми на асемблері є можливість вказівки лише типу необхідного операційного модуля, а конкретний пристрій буде призначений компілятором. З одного боку, це спрощує програмування таких пристроїв, а з іншого боку, дозволяє досить ефективно використовувати їх ресурси.

2.4.5 Суперскалярні ЦСП

Суперскалярні процесори також характеризуються великим набором паралельних операційних модулів і можливістю одночасного виконання декількох команд. Проте, в порівнянні з VLIW, вони мають дві характерні особливості:

- команди процесора не групуються в блоки, кожна їх них поступає в процесор незалежно;

— команди для паралельного виконання групуються усередині процесора на основі складу і поточної завантаженості операційних блоків, а також залежності між даними.

За допомогою описаного підходу можна обійти наступні недоліки VLIW:

— неефективне використання пам'яті через велику довжину групової операції;

— залежність скомпільованої коди від складу операційних модулів конкретного процесора.

Платою за вирішення цих проблем стає значне ускладнення схеми процесора, в якому з'являється модуль планування виконання команд.

Суперскалярні процесори планують виконання команд не лише на основі інформації про завантаженість операційних блоків, але і на основі аналізу залежностей між даними. Наприклад, команда збереження результату арифметичної операції не може бути виконана раніше самої операції обчислення, навіть якщо модуль звернення до пам'яті в даний момент вільний. Ця особливість наводить у тому числі до того, що один і той же набір команд може по-різному виконуватися в різних місцях програми, що робить неможливою точну оцінку продуктивності. Особливо це поважно для систем, що працюють в реальному часі, адже оцінка по найгіршому результату приведе до того, що ресурси процесора будуть використані не повністю. Таким чином, в цих системах завдання точної оцінки продуктивності суперскалярних ЦСП залишається відкритим.

2.4.6 Гібридні ЦСП

Під гібридними ЦСП зазвичай розуміють спеціалізовані пристрої, що поєднують в собі функцію мікроконтролера і цифрового сигнального процесора. Зазвичай такі вироби призначені для виконання однієї функції - наприклад, управління електричними двигунами, або іншими об'єктами в реальному часі. Іншою широкою сферою їх застосування останнім часом стає

мобільна телефонія, де раніше використовувалися два процесори - один звичайний для управління функціями апарату (дисплеєм, клавіатурою), а інший для обробки голосових сигналів (кодування і т. д.).

2.5 Програмування ЦСП

Для програмування ЦСП зазвичай використовують одну з двох мов - асемблер і С. Основні особливості асемблерів ЦСП збігаються з аналогічними мовами звичайних мікропроцесорів і, в цілому, можуть бути описані як:

- мова асемблера є машинно-орієнтованою, тобто кожне сімейство процесорів має мову, що відрізняється від мови інших сімейств;
- одна команда на асемблері зазвичай еквівалентна одній команді машинної мови;
- при програмуванні на асемблері програмістові доступні всі ресурси процесора і системи, що дозволяє використовувати їх максимально ефективно;
- від програміста вимагається хороше знання архітектури кожного конкретного процесора, з яким він працює, тобто необхідна кваліфікація персоналу має бути досить високою;
- створення і відладка програм на асемблері - тривалий трудомісткий процес, що також вимагає високої кваліфікації.

З іншого боку, при використанні мов високого рівня, зокрема, С, можна помітно спростити і прискорити створення програм, але при цьому ресурси системи використовуватимуться менш ефективно, в порівнянні з програмою, цілком написаною на асемблері. У реальності зазвичай використовуються підхід, що поєднує достоїнства як мов високого рівня, так і ефективності програм на асемблері. Виражається це в тому, що стандартні бібліотеки зазвичай створюються на асемблері, як і критичних до часу виконань і об'єму пам'яті частини коду. В той же час допоміжні модулі можуть створюватися

на мові високого рівня, прискорюючи і спрощуючи розробку програмної системи в цілому.

2.5.1 Особливості асемблерів ЦСП

До цікавих особливостей асемблерів ЦСП можна віднести наступне: наявність двох форм запису багатьох команд - мнемонічною і алгеброю. Мнемонічна форма аналогічна запису команд для звичайних мікропроцесорів, наприклад, `ADD dst, src`. Інша, алгебра, в асемблерах стандартних мікропроцесорів використовується рідше, тоді як на мові ЦСП згадана команда може бути записана у вигляді `dst = dst + src`. Засоби організації стандартних структур, наприклад, спеціальних апаратних команд повторення однієї команди або блоку коду. При цьому, на відміну від команд повторення звичайних процесорів, ЦСП може пропускати цикл вибірки коду повторюваної команди, що зменшує час виконання кожного повторення як мінімум на 1 цикл шини, що при двоцикловій команді дає подвійний виграш за часом.

2.5.2 Сумісність усередині сімейств ЦСП

Зазвичай ЦСП випускаються сімействами, і вироби усередині сімейств мають аналогічні мови асемблера, або навіть сумісні на рівні машинних код. Також усередині сімейства зазвичай використовуються однакові набори бібліотек підпрограм. Як і в звичайних мікропроцесорах, частенько старші моделі ЦСП можуть виконувати машинний код молодших моделей, або їх асемблер включає всі команди молодших моделей як підмножину власного набору команд.

2.5.3 Налагоджування програм

Зазвичай, налагоджування програм, написаних для ЦСП, виробляється за допомогою спеціальних засобів, що включають програмні імітатори і

емулятори. У їх склад також часто входять засоби профілізації (виміри швидкості виконання блоків коди).

3 РОЗРОБКА МЕТОДІВ ЦИФРОВОГО ОПРАЦЮВАННЯ СИГНАЛІВ І СХЕМ АУДИОПРОЦЕСОРІВ З ДИСТАНЦІЙНИМ КЕРУВАННЯМ

3.1 Реалізація ефекту однократної затримки сигналу

Одним з найпоширеніших класів є звукові ефекти, засновані на затримці сигналу. У реальній ситуації затримка звукового сигналу може виникнути при відбитті звукової хвилі від твердої перешкоди, наприклад, від стіни. Величину виникаючої при цьому затримки легко визначити, знаючи пройдений звуковою хвилею шлях і враховуючи, що швидкість звуку на рівні моря можна вважати рівної 330 м/с. Таким чином, за одну мілісекунду звукова хвиля проходить 33 сантиметра. Для розрахунків зручніше пам'ятати, що один метр вона проходить за 3 мс.

Розглянемо спочатку реалізацію однократного відбиття звукової хвилі. Цей простий ефект є базовим для реалізації більш складних ефектів, таких, як імітація реверберації й хору.

Як уже говорилося вище, відбиття звукової хвилі від твердої перешкоди приводить до появи в крапці реєстрації затриманої версії випромінюваного сигналу. До появи цифрових процесорів обробки сигналу й інших цифрових пристроїв для затримки основного сигналу використовувалися спеціальні магнітофони з декількома рухливими головками. При цифровій реалізації лінії затримки вхідний аналоговий сигнал перетворюється в цифрову форму, записується в циклічний або будь-який інший буфер затримки, і сигнал з виходу цього буфера затримки перетворюється назад в аналогову форму й відтворюється.

Блок-схема реалізації однократної затримки звукового сигналу наведено на рисунку 3.1.

Вхідний сигнал $x(p)$ одночасно надходить на лінію затримки й на суматор, що формує вихідний сигнал. Затриманий на величину D сигнал $x(p)$ множиться на масштабний коефіцієнт a й також надходить на суматор, що

формує вихідний сигнал. Перетворення, здійснювані даної схемою, можуть бути виражені наступною формулою:

$$y(n) = x(n) + ax(n - D), \quad (3.1)$$

Дана схема є лінійним фільтром з передатною функцією:

$$H(z) = 1 + az^{-D}, \quad (3.2)$$

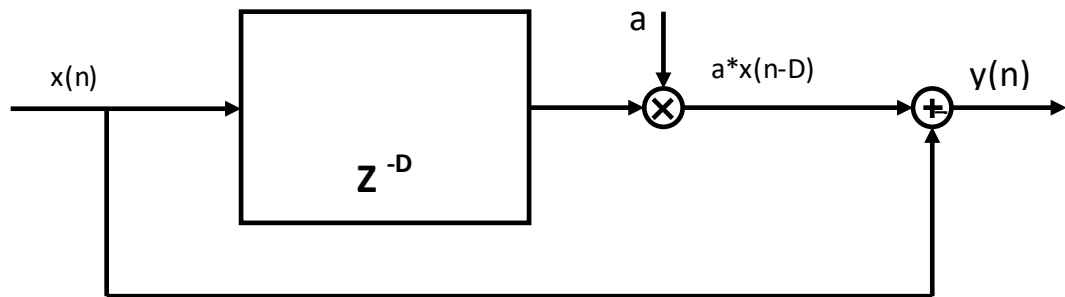


Рисунок 3.1 — Блок-схема реалізації однократної затримки звукового сигналу

Оскільки при будь-якому відбитті звуковий сигнал послабляється, то $-1 < a < 1$. Негативні значення відповідають відбиттю в протифазі. Ця модель є досить грубою, оскільки як поширення звукової хвилі, так і її відбиття є частотно-залежними процесами. Однак у більшості випадків ця модель є адекватною. Величина D визначається різницею часу між приходом прямої і відбитої звукової хвилі.

Передатна характеристика (3.2) є характеристикою гребінчатого фільтра. Відмінною рисою даного класу фільтрів є виконання для них рівності

$$H(j\omega) = H\left(j\omega + \frac{2\pi}{D}\right), \quad (3.3)$$

Однократна цифрова затримка часто використовується для пожвавлення "сухої" музики. У цьому випадку як основний, так і затриманий аудіосигнали складаються з однаковою амплітудою. Для того щоб при виконанні цієї операції не виникало переповнення розрядної сітки цифрового пристрою, обоє сигналів складаються з коефіцієнтом $1/2$. У цьому випадку використовується затримка порядку від 15 мс до 40 мс. Введення цієї затримки приводить до появи в слухача враження, що два виконавці відіграють в унісон. При зменшенні величини затримки нижче 10 мс у сигналі можуть виникнути небажані биття, оскільки сума двох гармонійних сигналів з однаковою амплітудою може бути представлена як амплітудна модуляція гармонійного сигналу із частотою, рівній сумі частот вихідних сигналів, гармонійним сигналом із частотою, рівної різниці частот вихідних сигналів.

Область використання однократних затримок сигналу дуже обмежена, оскільки в реальній житті рідко зустрічається ситуація, коли в акустичному полі присутнє тільки одна поверхня, що відбиває. Крім того, як відзначалося вище, навіть у цьому випадку для підвищення адекватності моделі необхідно використовувати формуючий фільтр, що враховує частотну характеристику поверхні, що відбиває, і, можливо, частотну характеристику середовища поширення сигналу. Для моделювання акустичних полів з декількома, що відбивають поверхнями й обліку частотних характеристик процесів відбиття й поширення звукової хвилі використовуються багаторазові затримки.

3.2 Багаторазова затримка сигналу

Багаторазова затримка сигналу може бути досягнута двома шляхами: доступом до різних елементів, що зберігаються в одній лінії затримки, або створенням декількох окремих ліній, що мають різний час затримки. Перший спосіб використовується в цифровій обробці сигналів і в магнітофонах з декількома зчитувальними головками, а другий — при використанні механічних і електричних ліній затримки.

Багаторазова затримка сигналу може бути реалізована як з використанням трансверсальних (з кінцевою імпульсною характеристикою), так і з використанням рекурсивних фільтрів (з нескінченною імпульсною характеристикою).

3.2.1 Багаторазова затримка з використанням трансверсального фільтра

Блок-схема реалізації багаторазової затримки сигналу з використанням трансверсального фільтра наведено на рисунку 3.2.

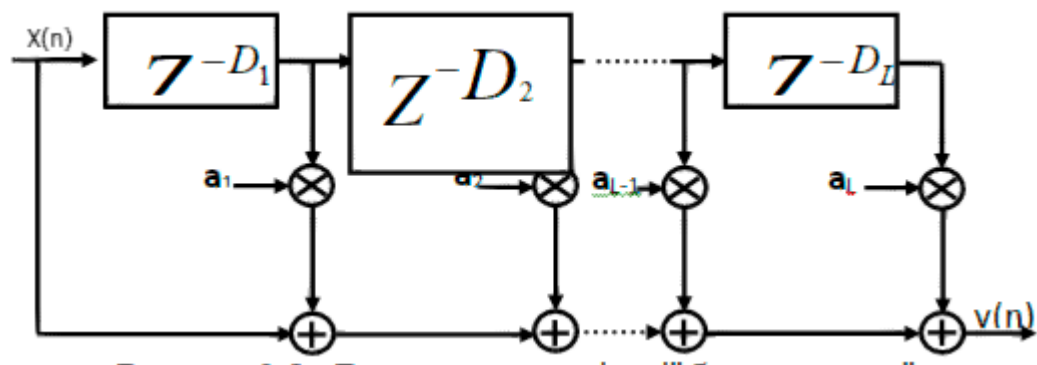


Рисунок 3.2 — Блок-схема реалізації багаторазової затримки з використанням трансверсального фільтра

Різницеве рівняння трансверсального фільтра може бути записане у вигляді:

$$y(n) = x(n) + \sum_{i=1}^L a_i x(n - \sum_{k=1}^i D_k) \quad (3.4)$$

У даному фільтрі використовуються різні величини затримок вхідного сигналу, що не дозволяє вважати цей фільтр гребінчатим. Використання кінцевого числа ліній затримки дає можливість урахувати вплив обмеженого числа поверхонь, що відбивають.

Оскільки в трансверсальному фільтрі може бути враховане тільки обмежене число відбиттів сигналу, те його не можна використовувати для

моделювання перевідбиттів звукової хвилі. Для цього слід використовувати рекурсивні фільтри.

3.2.2 Багаторазова затримка з використанням рекурсивного фільтра

Блок-схема реалізації багаторазової затримки з використанням найпростішого рекурсивного фільтра наведено на рисунку 3.3.

Цей фільтр може бути використаний для опису перевідбиттів звукової хвилі між двома нескінченними паралельними стінами, на одній з яких розташований випромінювач плоскої хвилі. Приймач може розташовуватися на кожній зі стін. При відбитті від стіни енергія звукової хвилі падає в a раз. Імпульсна характеристика даного фільтра являє собою послідовність одиночних імпульсів, розташованих на відстані D відліків друг від друга, причому імпульс із номером n має амплітуду a^n .

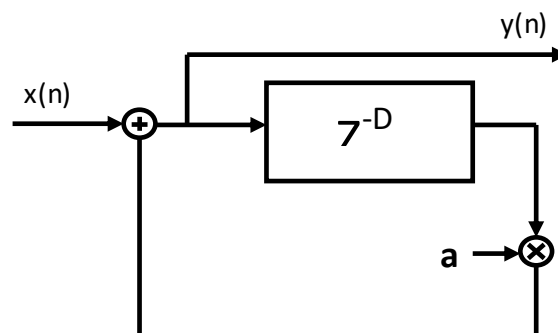


Рисунок 3.3 — Блок-схема реалізації багаторазової затримки з використанням рекурсивного фільтра

Даний фільтр описується різницевою рівнянням:

$$y(n) = ay(n - D) + x(n), \quad (3.5)$$

і має передатну функцію:

$$H(z) = \frac{1}{1 - az^{-D}}, \quad (3.6)$$

Крім описаних вище найпростіших звукових ефектів існують і більш складні, засновані на переміщенні спектральних складових вихідного сигналу. Такий ефект може бути, наприклад, будуть досягнуті при зміні величини затримки сигналу за певним законом у базовій блок-схемі, наведеної на рисунку 3.4.

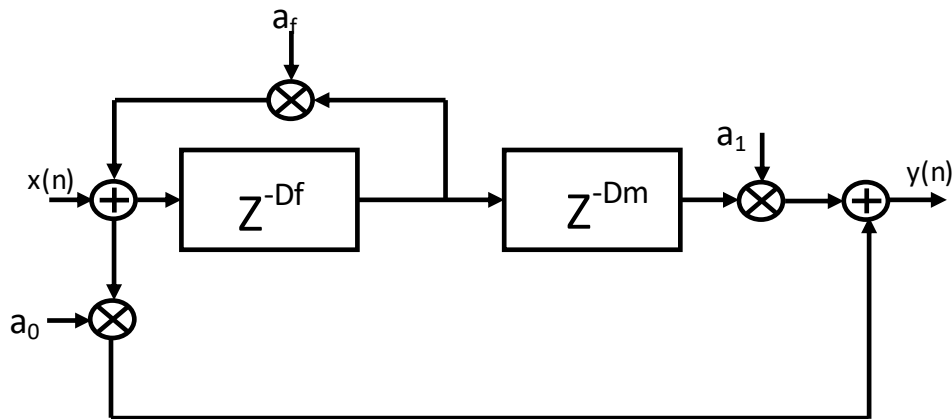


Рисунок 3.4 — Базова блок-схема.

У цій блок-схемі величина D_f є постійної, а величина $D_m = f_m(t)$ являє собою деяку функцію часу. Тому дана блок-схема реалізує гребінчатий фільтр із мінливої в часі частотною характеристикою. Ця блок-схема має занадто складну структуру, щоб описати її одним різницеvim рівнянням. Вибір закону зміни величини D_m , а також значення постійних величин a_0 , a_1 , a_f і D_f дозволяють реалізувати різні ефекти.

3.3 Ефект детонації

Ефект детонації був відкритий при спробі використання вихідних сигналів із двох котушкових магнітофонів для створення ефекту хору, який буде описаний нижче. У результаті виник ефект, що нагадує звук реактивного літака, що злітає.

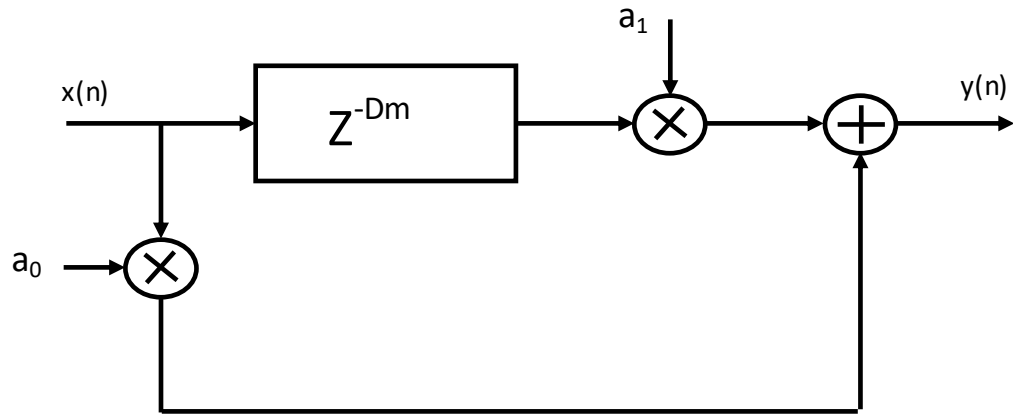


Рисунок 3.5 — Блок-схема пристрою, що реалізує ефект детонації

Для реалізації ефекту детонації в цифрових пристроях може бути використаний пристрій, блок-схема якого наведено на рисунок 3.5

Як видно з рис. 3.5 , ця блок-схема є часткою базової блок-схеми, зображеної на рис. 3.4. Даній блок-схемі відповідає наступне різницеве рівняння:

$$y(n) = a_0 x(n) + a_1 x(n + f(n)) , \quad (3.7)$$

$$f(n) = \frac{D}{2} \left[1 + \cos\left(\frac{2\pi}{N} n\right) \right] , \quad (3.8)$$

Для запобігання виникнення переповнень розрядної сітки при обчисленні вихідного значення коефіцієнт a_0 вибирається рівним 0,5, а коефіцієнт a_1 вибирається від -0,5 до 0,5. Значення параметра D , що визначає діапазон зміни затримки, звичайно вибирається від 0.25 до 25 мс. Значення параметра N , що визначає частоту детонації, вибирається виходячи з вимог, пропонованих до вихідного сигналу.

Зменшення довжини лінії затримки приводить до пропорційного підйому частот спектральних складових вихідного сигналу, а її зменшення — до зниження цих частот.

3.4 Ефект хорусу

Іншим ефектом, що збагачують звукову палітру добутку, є ефект хорусу. Він дозволяє створити в слухача ілюзію, що даний добуток одночасний виконується на декілька однакових музичних інструментах. Хоча музиканти намагаються відіграти синхронно, вони не можуть уникнути невеликих відхилень від загального темпу. Крім того, неминучі відмінності в настроюванні окремих інструментів і в гучності виконання.

Для створення ефекту одночасного виконання добутку на M інструментах може бути використаний пристрій, блок-схема якого наведено на рисунку 3.6.

Як правило, усі коефіцієнти при реалізації ефекту затримки вибираються рівними ($1/M$), а самі величини затримки від 15 мс до 35 мс. Природно, що кожна лінія затримки повинна змінювати свої параметри незалежно від усіх інших.

При такому виборі коефіцієнтів моделюються тільки відмінності в темпі виконання добутку й у настроюванні інструментів. Те що ці відмінності зв'язано один з одним, на слух не визначається, оскільки в реальній ситуації така залежність відсутня, і слух не пристосований для її виділення. Для моделювання різного рівня відтворення досить зробити змінними коефіцієнти фільтра, додавши до їхнього постійного значення деякий випадковий низькочастотний сигнал малого рівня, оскільки виконавці намагаються зберігати середній рівень гучності при відтворенні. Для простоти в якості випадкового низькочастотного сигналу може бути використаний той же самий сигнал, що й у лініях затримки.

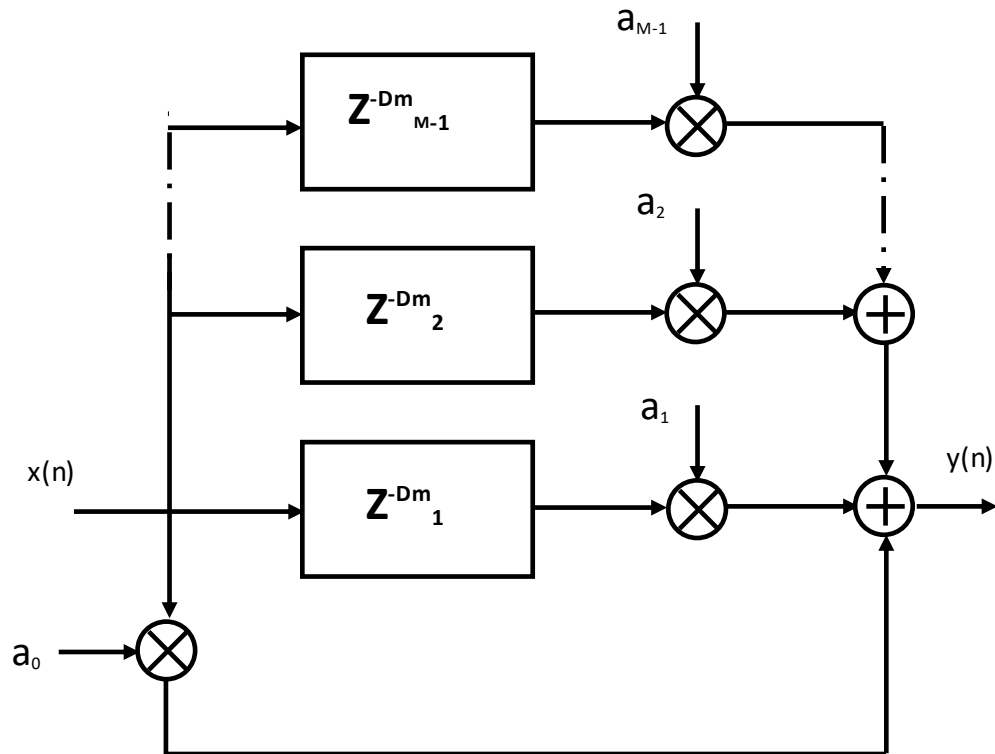


Рисунок 3.6 — Блок-схема пристрою, що реалізує ефект хору

Даній блок-схемі відповідає наступне різницеве рівняння:

$$y(n) = a_0 x(n) + \sum a_i x(n + f_i(n)) , \quad (3.9)$$

$$f(n) = \frac{D}{2} \left[1 + \cos\left(\frac{2\pi}{N} n\right) \right] , \quad (3.10)$$

3.5 Ефект вібрато

Ефект вібрато імітує вібрато людських голосів їх тембр, що визначає, теплоту й виразність приймання, що й відповідає, виконання на струнних музичних інструментах, що періодично змінює в невеликих межах висоту звуків. Ефект вібрато може бути отриманий у пристроях ефекти, що реалізують, детонації й хору при повному видаленні вхідного сигналу з вихідного. Величина затримки при цьому вибирається таким чином, щоб частотне зрушення основних спектральних складових сигналу становило

трохи герц. Цей ефект часто плутають із ефектом тремоло, де сигнал зазнає не частотної, а амплітудної модуляції. Обоє ефекту використовують однаковий сигнал, що модулює. Ефекти тремоло й вібрато можуть спільно використовуватися для створення ефекту обертового диктора.

Блок-схема пристрою, що реалізує ефект вібрато, наведено на рисунку 3.7.

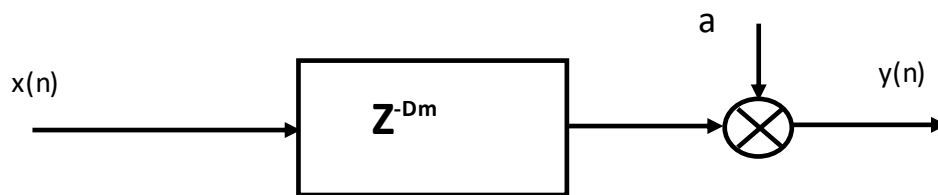


Рисунок 3.7 — Блок-схема пристрою, що реалізує ефект вібрато

Пристрій, блок-схема якого наведено на рисунку 3.7, може використовуватися також і для зміни основного тону диктора (наприклад, щоб зробити голос мовця невпізнаним). У цьому випадку величина затримки змінюється від нуля до деякої граничної величини за лінійним законом. По досягненню затримки своєї граничної величини вона скидається в нуль, і процес відновлюється. Зміна затримки за таким законом дозволяє забезпечити роботу пристрою в реальному часі, але в момент обнуління величини затримки чути клацання. Справа в тому, що саме в цей момент відбувається розрив відтвореного сигналу при переході до повторного відтворення деякої його ділянки.

Якщо необхідно не підвищити, а понизити частоту основного тону, то величина затримки постійно зменшується й при досягненні нуля встановлюється у своє максимальне значення. При такому переході губиться частина вхідного сигналу. Для усунення клацань, що виникають при зміні величини затримки, рекомендується використовувати фільтри нижніх частот.

Більш простим способом усунення клацань є використання двох подібних пристроїв, фази пилкоподібних сигналів, що модулюють, яких

зрушені на половину періоду. При цьому необхідно задати вагове вікно (наприклад, вікно Ханна) на періоді сигналу, що модулює. Вихідні сигнали кожного пристрою множаться на відлік вагової функції, відповідний до поточної довжини лінії затримки, і складаються один з одним, що забезпечує усунення клацань при переходах. Якщо позначити період сигналу, що модулює, як N , відлік сигналу, що модулює, як $f(n)$, а відлік вагової функції як $W(n)$, те, з використанням функції одержання залишку від цілочисленного розподілу числа M на число N , позначуваної як $\text{mod}_n(M)$, процедура перетворення основного тону може бути записана у формі наступного різницевого рівняння:

$$\begin{aligned} y(n) &= ax(n + f(n)) \\ f(n) &= \frac{D}{2} \left(1 + \cos \frac{2\pi n}{N} \right), \end{aligned} \quad (3.9)$$

3.6 Зміна динамічного діапазону, компресори й експандери

Компресори й експандери використовуються для зміни динамічного діапазону сигналу, тобто для зменшення відмінності рівнів самого тихого й найгучнішого сигналів. Для цього при перевищенні поточним значенням сигналу деякого порога встановлюється коефіцієнт передачі, менший, чому використовуваний до перевищення порога. Відношення старого коефіцієнта передачі до нового називається ослабленням. Усі коефіцієнти передачі задаються в логарифмічній шкалі. Тобто при ослабленні 2:1 збільшенню вхідного сигналу на 20 дб буде відповідати збільшення вихідного сигналу на 10 дб. Подібний компресор, що змінює свою передатну характеристику в крапці a , може бути реалізований по наступній формулі:

$$y = \begin{cases} x, & |x| \leq \alpha \\ \operatorname{sgn}(x) \sqrt{\frac{|x|}{\alpha}}, & |x| < \alpha \end{cases}, \quad (3.10)$$

Застосування компресорів дозволяє передавати по лініях зв'язку сигнали, що мають більший динамічний діапазон, чому передавальний канал. На прийомному кінці цієї лінії встановлюється експандер, що має передатну характеристику, зворотну характеристиці компресора.

Іншою областю застосування компресорів є звукозаписна індустрія й радіомовлення. Справа в тому, що при записі музичних добутоків для більш точної передачі звучання всіх груп інструментів оркестру встановлюється кілька мікрофонів. Звуковий тиск створює на виході мікрофона електричний сигнал, що має динамічний діапазон від 60 дБ до 70 дБ. Однак більшість каналів незручно передати весь діапазон зміни сигналу внаслідок обмежень, що накладаються перешкодами в каналі і його нелінійностями. Тому виникає завдання стиснути вихідний динамічний діапазон сигналу до динамічного діапазону каналу, що звичайно становить порядку 40 дБ.

При рівні ослаблення вище, чим 10:1, компресор перетворюється в обмежник. Його основна відмінність полягає в тому, що застосування компресора припускає наступне використання експандера, а після обмежника які-небудь розмови про відновлення сигналу, як правило, безглузді. Тому найбільше часто використовуються обмежники з рівнем ослаблення ($\alpha : 1$), що відповідає обмеженню сигналу на рівні порога.

Оскільки передатні характеристики компресорів і експандерів відрізняються великою нелінійністю, їх використання приводить до істотного викривлення сигналу. Так, наприклад, при використанні аналогових компресорів і експандерів неминуче виникає питання про забезпечення рівності амплітуд переданого й прийнятого сигналів, оскільки а якщо ні, то крапки зміни коефіцієнтів передачі компресора й експандера не збіжаться, що приведе до виникнення нелінійних викривлень. Як відзначалося вище, у

багатьох випадках необхідно робити компресію сигналу з випередженням, тобто до фактичного перевищення сигналом граничного рівня. У цьому випадку при повільному росту або спаді сигналу невідповідність крапок зміни коефіцієнтів передачі приведе до розтягування в часі описаних вище нелінійних викривлень.

3.7 Архітектура та алгоритм роботи аудіопроцесора

Арифметичний мікропроцесор в основі мікросхеми, яка отримує доступ до даних від каналів ADC, RAM, Delay Memory, що сполучають банк даних і входи пристрою POT, а так само блок LFO. ALU має доступ до DAC, пам'яті або Lfos. Дані для керування надходять від внутрішнього ROM або зовнішнього EEPROM, залежно від стану штирькового виводу T0. Роздільна здатність ADC і DAC становить 24 біта, і вся обробка виконано 24 бітним форматом лінійної IC. Вхід пристрою POT навмисно квантуються до 9 бітів, з додаванням гістерезису так, щоб дані потенціометра не мали «деренчання контактів» між суміжними кодами. Лічильник програми коливається від 0 до 127, дозволяючи 128 можливих циклів технології у звукову вибірку. Програма може включати умовні твердження пропуску однак, деякі дозволяють інструкціям «перепригувати» базований відкритий стан, таке як стан суматора, але також і якщо код виконав попередній семпл; це дозволяє інформації установки для Lfos бути вставленої в код програми, який виконаний тільки одного разу (як установка) після чого, Lfos залишають свobodнозапущенному на наступних семплах. Контроль запису буде містити єдину повну програму, завантажену або від внутрішнього ROM або від зовнішнього EEPROM. Кожна інструкція 32 біта шириною, вимагає 4 КБ даних при роботі програми. Кожна інструкція містить адресу пам'яті, коефіцієнти й інформацію керування як потрібно певною інструкцією. Діапазон інструкцій досить широкий, розроблений, для більш зручного управління даними для звукових ефектів. У результаті один режим може включати кілька одночасних режимів; множення й кілька доповнень або

вирахувань утримуються в межах єдиної інструкції. У результаті традиційний процесор обробки сигналів, який вимагав би виконання декількох інструкцій зробити роботу єдиної інструкції FV-1. Сигнали затримується в RAM часу затримки, яка здійснена в плаваючій комі для збільшеної економії. Перевірка адреси для пам'яті часу затримки обчислена від лічильника вибірки плюс відносна адреса, яка поставляється певною інструкцією, або автоматично від LFO під час LFO зв'язав інструкції. Це дозволяє вказувати що дані доступу в межах RAM часу затримки до автоматично декременту в кожному циклі.

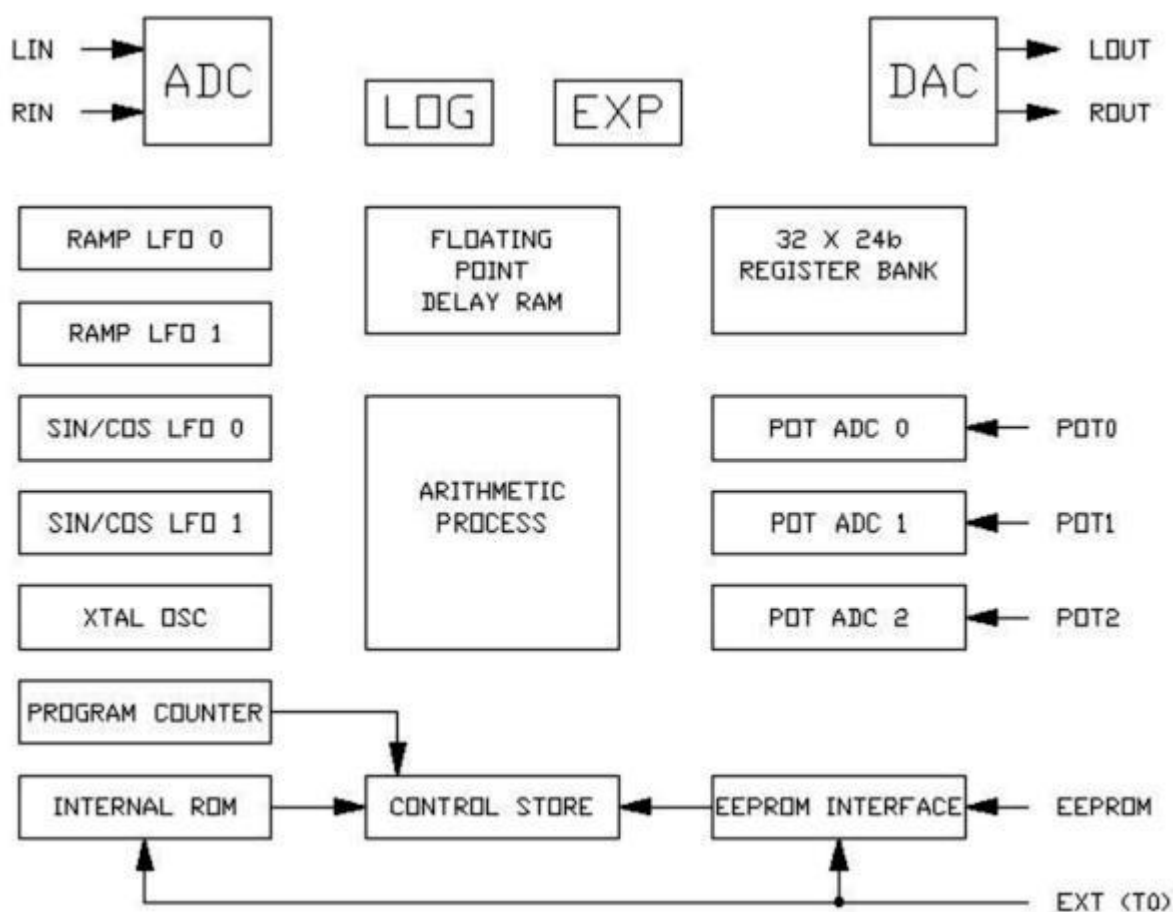


Рисунок 3.8 — Базова структурна схема

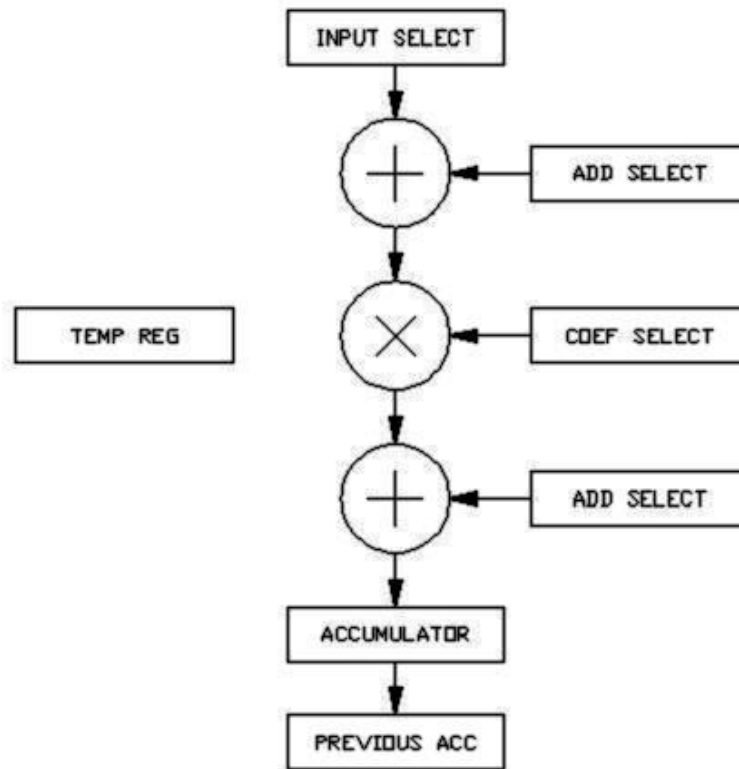


Рисунок 3.9 — Арифметична група елементів технології.

Мікропроцесор далі розширений включенням логарифма й групи елементів зразка. Єдині цикли можуть поставити логарифм або експоненту попереднього значення суматора, масштабуючи результат коефіцієнтом і підсумовуючись константу. Це особливо корисно у виконанні числових підрозділів і квадратного коріння, як може бути потрібно в обчисленні середньоквадратичних значень амплітуд сигналу. Селектор - інструкція, яка контролює, щоб зібрати кілька різних джерел сигналу, щоб підсумувати або відняти з відібраним вхідним джерелом, і колись і після множення коефіцієнта, що й множить, який може бути вставлений у межах інструкції або принесений від підключеного банку даних. Тимчасовий регістр запам'ятовує вхідне значення останнього циклу накопичуючий регістр, що й попередній, зберігає значення попередньої інструкції. Ці сигнали підсумують до ступеня інтеграції можливих інструкцій, і корисності заключних кодових послідовностей. Коефіцієнт тривалості імпульсу, представленого мультиплексору, буде залежати від джерела. Коефіцієнти, вставлені в межах

інструкції, від 11 до 16 бітів шириною, у межах від -2.0 до +1.999969. Коефіцієнт для вставки суміжних вибірок в інструкції СНО 14 бітів шириною, розташовуючись для 0 до 1.0. Значення підсумували до результату мультиплікування під час логарифма, експоненти або інструкції SOF 11 бітів шириною. Вистава даних у всіх перетворювачах і регістрах, 24 біта, 2's комплімент, найкраще виражений як: S.bbbbbbbbbbbbbbbbbbbbbb, у фракційній формі. Щоб запобігти обриву сигналу в межах програми, сигнали ADC повинні бути прочитані в мікропроцесор на зменшеному рівні, і виходу алгоритму потрібно тоді дати відповідний коефіцієнт підсилення й відіслати в DAC. Розмір амплітуди буде залежати від програми. Результат логарифмічної операції зрушується на 4 місця праворуч, а вхід переводиться в режим експоненти, як впливає, буде правильними переміщеними 4 місцями. Логарифмічна операція ухвалює або позитивний вхід або негативний, і обробляє їх обох як аргументи позитивного виду. Тому що максимальне вхідне значення до логарифмічної операції 1.0 або менше, результатом логарифма буде нуль або негатив тільки. Аналогічно, функція експоненти очікує нуль або негативний вхід, вхід з позитивним видом зробить обмежену продуктивність режиму насичення. Доступні Lfos у межах FV-1 входять у два види: SIN/COS і пилкоподібний, і два з кожного включені в структуру FV-1. Вони можуть використовуватися, щоб змодельовати час затримки й забезпечити синусоїдальний або, що лінійно змінює крапки адреси. Числова швидкість (частота) і амплітуди Lfos можуть бути змінені, записуючи їм нові значення безупинно зсередини програми, але звичайно встановлюються нагорі програми одного разу, і залишаються вільно-керованому при використанні режиму пропуску, щоб перестрибнути через команди ініціалізації. Lfos забезпечують більшу гнучкість у модуляції часу затримки програм реверберації, щоб підсумувати, що просторові зміни, або виконати подачу ефекту як транспозицію й флэнджинг. Два стилі LFO досить різні по своїй структурі, але обоє здатні до одночасної передачі відносної адреси, щоб затримати RAM, також забезпечуючи коефіцієнт інтерполяції

арифметичному мікропроцесору. Інструкція СНО (коротке замикання для хорусу) дозволяє прямо управляти пам'яттю часу затримки через LFO, дозволяючи двом послідовним інструкціям одержати доступ до дійсного пункту в межах RAM часу затримки, навіть *між* розміщеннями RAM, ефективно забезпечуючи автоматично, гладко *переміщаючи* крапки від часу затримки. Як може бути припущений, установка для такої технології обов'язково складна. До все-таки далі ускладнюють справи, додаткова гнучкість знайдена вибором синусоїди або косинусоїди виходу, щоб торкнутися часу затримки; ефект хорусу можливий, з далеко не всіма інструкціями. Неприємність із усім цим полягає в тому, що Ви повинні спочатку зрозуміти синтаксис установки LFO і використання скоріше складний, все-таки гнучка інструкція СНО.

Блок SIN/COS — пари інтеграторів, які забезпечують Синус сигналів квадратури й Косинус, по швидкості, що управляють швидкістю сигналу амплітуди, і в амплітуді, називаної управлінням амплітуди. Ці керуючі сигнали, записуються в регістр у блок SIN/COS яка може бути записана єдиною інструкцією на початку програми (і перескочити в наступному виконанні), або безупинно записаний інструкціями в межах програми, яка може бути результатом ROT або обчисленого значення сигналу. Вихід SIN/COS це широкі значення даних, які розгалужуються на фракції цілого числа, які можуть безпосередньо звернутися до пам'яті часу затримки, будучи підсумованим до адреси часу затримки інструкції СНО, і фракційної частини, яка одночасно харчується як коефіцієнт до арифметичної групи елементів виконати вставку сигналу фракційного розміщення. Тому в інструкції СНО є багато змінних, які з Lfos залучені в операцію, або синус або косинус використовується, і чи є операція в першому місці пам'яті або другому місці для інтерполяції.

LFO Пилкоподібного сигналу — єдиний інтегратор, який дозволяє «виток навколо», створюючи при цьому пилкоподібну форму хвилі. Ознака значення перевірки швидкості визначить, детермінується чи сигнал нагору

або вниз. Числове походження дозволяє, що обчислення «плавно накладає» коефіцієнти, які можуть бути відібрані під час режиму СНО, щоб інтерполювати між зафіксовані виходом результатів показчика RAMPA до результатів показчика RAMPB, дозволяючи метод переміщення подачі. Це підсумує ще одне можливе джерело керівництва для синтаксису інструкції СНО, той із плавно накладає вибір керування.

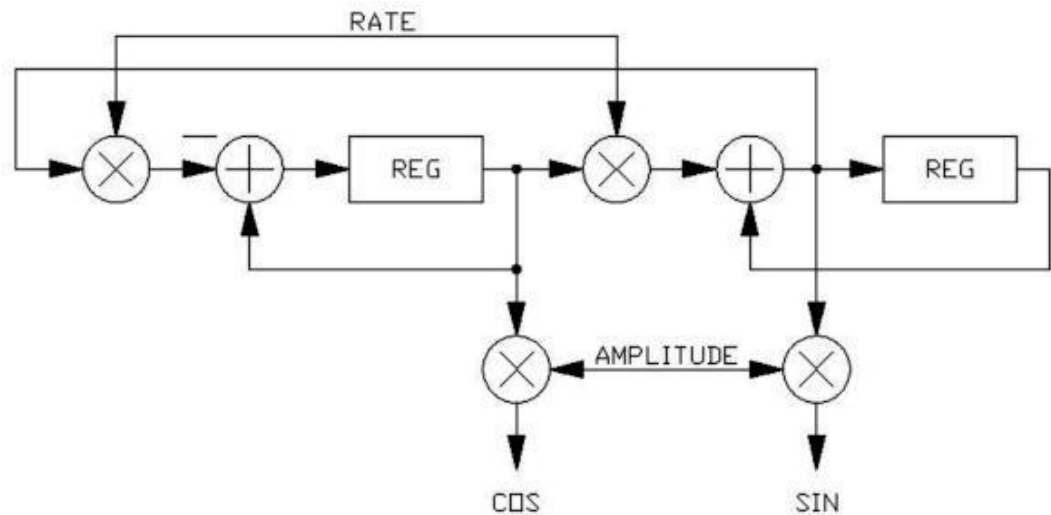


Рисунок 3.10 — Генератори SIN/COS.

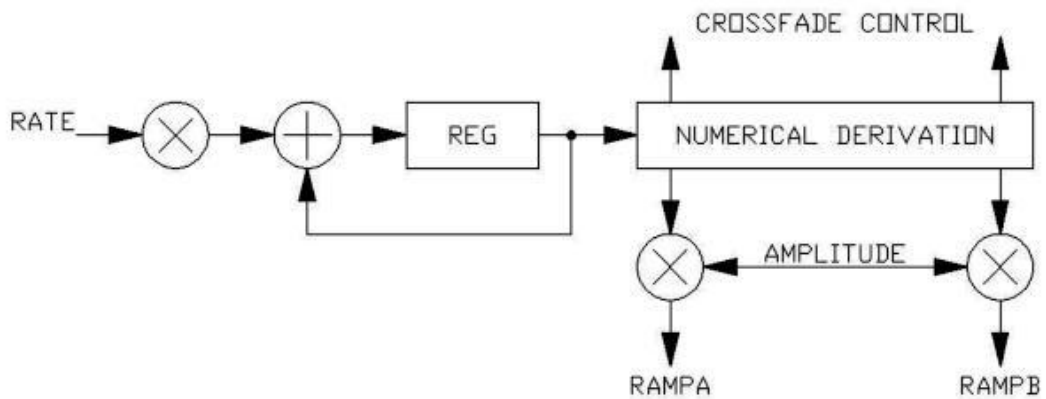


Рисунок 3.11 — Генератор пилкоподібного сигналу.

3.8 Розробка принципової схеми процесора

Spin Semiconductor FV-1 є повним рішенням для дослідження реверберації в одній мікросхемі. З інтегрованим ADC стерео і DAC, FV-1

може бути оброблений як будь-який інший аналоговий компонент у вашому сигнальному пристрої. FV-1 може одержати доступ до 16 програм у цілому, 8 програм демонстрації вбудовані на внутрішній ROM, і розробник може вибрати і з'єднати послідовний EEPROM з 8 додатковими програмами. При використанні розроблювачем зовнішнього EEPROM його продукт буде відрізнятися від інших, маючи спеціалізований набір програми без потреби в мікропроцесорі в системі. Маючи 3 вхідних потенціометра для в управління, програми мають 3 оперативні змінні параметри, такі як час згасання реверберації, швидкість і глибину хоруса або частоти у фільтрі. Ці вхідні пристрої доступні як коефіцієнти для Вашої програми й можуть використовуватися незалежно один від одного.

Широкий набір команд дозволяє користувачам програмувати ефекти всіх видів. З інструкціями як LOG і EXP, користувачі можуть легко програмувати звукові властивості й програми компресії. Інтегрований цифровий Lfos і генератор пилкоподібного сигналу використовується при програмування хоруса, фленжера й піч шифтера. Повна система для розробки включає асемблер й друковану плату для дослідження з інтерфейсом USB доступна від Spin Semiconductor.

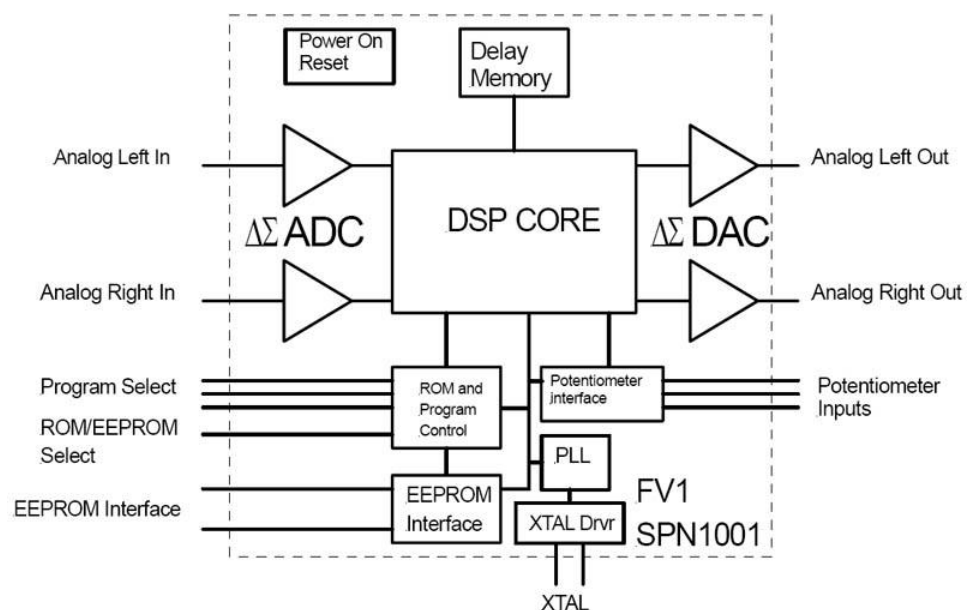


Рисунок 3.12 — Блок схема ЦСП процесора FV-1.

Ядро DSP FV-1 забезпечує багато покращених особливостей і інструкцій, які дозволяють Вам створювати складні програми, щоб змусити Ваш продукт виділитися. Ядро DSP складається з MAC/ALU, де втримуються всі математичні й логічні інструкції, блоки LOD і EXP, які виконують логарифмічні й експонентні функції, 32 - розрядний регістр даних, незалежний від RAM часу затримки й LFO і пилкоподібних генераторів, які можуть управляти адресацією пам'яті. Повний список і опис інструкцій доступний у керівництві асемблера, яке включено в компакт-диск у наборі розвитку або може бути завантажено від вебсайта Spin Semiconductor.

Змінні резистори підключаються через конденсатори зворотнього зв'язку, оскільки вхідні потенціометри внутрішньо зміщені щодо напруги VDD/2. Вхідний сигнал проходить через вхідні буфери, побудованих на р - канальних польових транзисторах Q1 і Q2. Резистори бажане заземлити як близько наскільки це можливо до вхідних виводів. Шунтувальні конденсатори повинні мати гарну якість, також можна використовувати плівкові конденсатори. Вхідний обхід повинен бути віддалений від будь-якого високочастотного шуму із вхідних сигналів, і забезпечити низький повний опір на високих частотах.

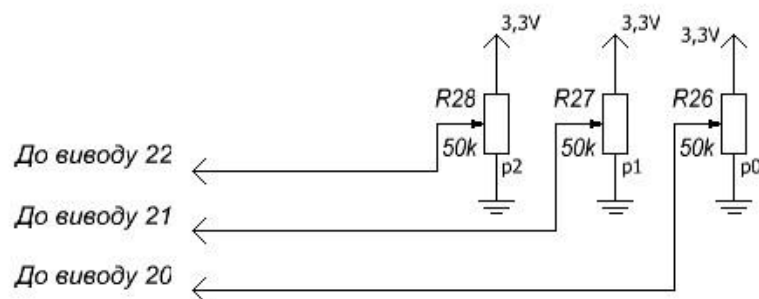


Рисунок 3.13 — Схема аналогового дистанційного керування процесором

Вихід пристрою можна захистити від шкідливого шуму, яке може бути більш повністю вилучений за допомогою активного фільтра із двома полюсами, але для більшості цей шум і не чуємо й ніякого наслідку до наступної схеми, тому простого RC фільтр досить, щоб вилучити будь-які високочастотні компоненти шуму, якщо це потрібно.

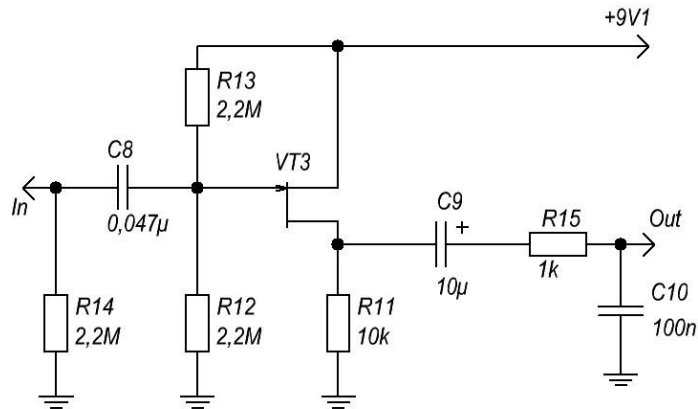


Рисунок 3.14 — Вхідний буфер з високим опором.

Опір вихідного навантаження не повинне бути менше ніж 10 КБ, щоб не перевантажити вихідні пристрої, що задають, і в конденсатора зв'язку повинна бути константа часу з опором навантаження приблизно від 20 мс до 50 мс.

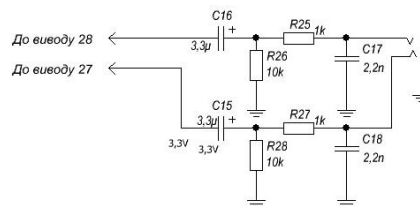


Рисунок 3.15 — Схема виходу.

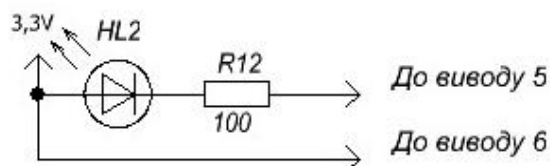


Рисунок 3.15 — Підключення світлової індикації перевантаження.

Можна підключити світлодіод, він загоряється приблизно на 30 мілісекунд, якщо до внутрішнього ADC або DAC підводить більш сильний сигнал чому потрібно. Напруги живлення цифрової частини повинне шунтуються керамічним конденсатором. В ідеалі, розробка буде на двосторонній друкованій платі з однієї стороною, що виконує роль заземлення. Це забезпечує зручне з'єднання котушки індуктивності для шунтувальних конденсаторів через vias. Однобічна друкована плата може використовуватися за умови, що велика заземлена ділянка буде перебувати під мікросхемою, і поверхневі конденсатори обходу будуть перебувати від виводів живлення й до цієї землі так близько до мікросхеми наскільки можливо. Аналогічно із виводам REFP і REFN. До них потрібно прикласти шунтувальну доріжку близько до частини, з резистором, щоб заживити від VDD. Струм, що проходить через REFP є маленьк, що не перевищує 30µа, у такий спосіб буде тільки невелике зниження напруги через резистор. Опір резистор фільтра між VDD і виводом REFP не повинне бути занадто більшим, інакше зміниться рівень сигналу. Вхідний опір керуючих потенціометрів перебуває в межах 10 мегомів, і ці значення є досить стійкими до шуму. Ніякі шунтувальні конденсатори не повинні бути застосовані в цих ланцюгах.

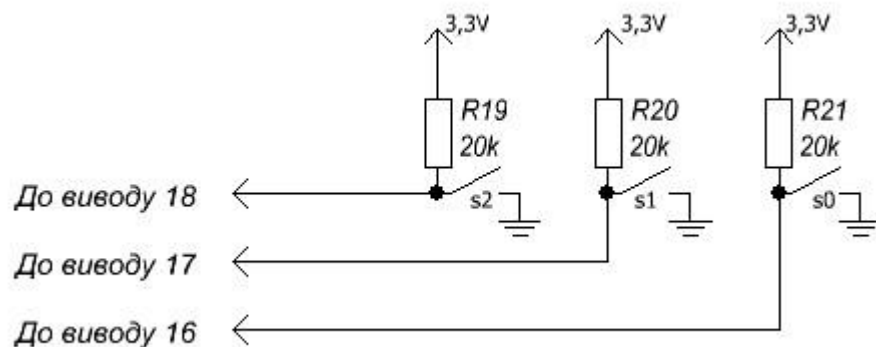


Рисунок 3.17 — Принципова схема дистанційного керування для перемикання програм.

Кварцовий генератор розроблений для стандартного кристала годин 32768 Гц, оскільки це особливо вигідно, але будь-яке джерело синхроімпулсів логічного рівня може бути приєднаний до безпосередньо мікросхеми X1. Норма вибірки системи буде по цій прикладній нормі. В 32768 Гц ADC і ширина смуги DAC складуть 15 КГц, при частоті синхроімпулсів 48 КГц, ширина смуги пропускання складає більше 20 КГц. Кристали годин мають дуже високий коефіцієнт Q, і вимагають часу, щоб досягтися повного рівня генерації. Усе компоненти для функціонування для кристала включені внутрішні в процесор, включаючи 10Meg резистор у ланцюзі зсуву й два конденсатори ємністю 12pf. Кварц необхідно розмістити максимально близько до мікросхеми. Якщо Ви бажаєте перевірити режим генератора, використовуйте вивід X2 як крапку для вимірів. Залежно від постачальника кварцу може бути проблема запуску із кристалом, рекомендується, щоб усі розробки включали конденсатор ємністю 15pf між виводом X2 і землею.

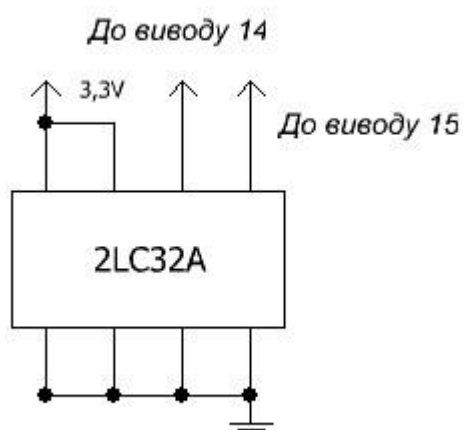


Рисунок 3.18 — Підключення зовнішньої пам'яті.

Використання зовнішнього EEPROM є додатковим, і у випадку розробок, які не використовують цей варіант розширення, виводи SCK і SDA можна залишити не замкненими. T1 однак повинен бути заземлений, а вивід

T0 повинен бути підключений до заземлення якщо використовуються внутрішні програми або до VDD (програми EEPROM).

3.9 Розробка пристрою для гітарного ефекту з дистанційним керуванням на мікроконтролері ARDUINO UNO

3.9.1 Призначення й технічні параметри

Пристрій, що розробляється – це цифро-аналоговий ділей з природним звучанням. Гітарний процесор (далі – педаль) використовує ту ж смугу частот, що і класичні пристрої на магнітній стрічці. Педаль може підключатися перед підсилювачем або в ланцюг ефектів.

У педалі не застосовуються системи шумозаглушення, в результаті чого зберігається природний затухаючий звук. Ланцюг використовують аналогове посилення без фільтрів. При дотриманні гучності на вході вихідний сигнал не спотворюється і не забарвлюється. Відбитий сигнал можна налаштовувати для отримання широкого спектру звучань. Ділей розроблявся таким чином, щоб його можна було використовувати в поєднанні з ефектом дісторшн: ця перевірка була потрібна, так як подібна комбінація часто призводить до проблем з ділеєм.

Педаль може підключатися в ланцюзі як до дісторшена, так і після нього. Особливо добре педаль працює з чистим звуком. Звук ділею ретельно продумувався від перших відбиттів до затухання. Педаль має набір основних параметрів: час затримки (Delay Time), рівень звуку (Delay Level) та повтор (Repeat). Коли час затримки перевищує 120 мс, його, як правило, синхронізують з темпом музики (встановлюючи час рівним або частці, або кратним її тривалості). Прототип же створює не ритмічні повтори, а налаштований відгук, що чудово звучить і без прив'язки до темпу.

Існують різні варіанти формування відбитого звуку. В схемі використовується цифровий, так як потрібно зробити педаль компактною, надійною, що підтримує раціональний час затримки. Крім того, при використанні цифрового ділею стають менш критичними вимоги до інших

компонентів ланцюга ефектів. Аналогічний аналоговий пристрій коштував би набагато дорожче. Для отримання такої ж якості звуку треба було використовувати компоненти, що перевищують за ціною ті, що були використані, в кілька разів.

Управління мікроконтролером:

— LEVEL — налаштування рівня відбитого сигналу — від відсутності ефекту (проти годинникової стрілки) до максимального ефекту (за годинниковою стрілкою);

— DELAY — регулювання часу затримки – від 25 до 450 мс;

— REPEAT — кількість повернень відбитого сигналу – від одного до нескінченності.

Технічні вимоги:

— споживання струму від 25 до 32 мА в залежності від налаштувань;

— напруга: від 8 В до 15 В (при напрузі менше 7,5 В сигнал буде зникати);

— вхідний опір 180 кОм;

— навантажувальна здатність 10 кОм;

— максимальна вхідна напруга 2 В;

— співвідношення сигнал/шум 80 дБ;

— час затримки: приблизно від 25 до 450 мс;

— роз'єми входу та виходу 1/4".

3.9.2 Структурна й принципова схеми пристрою

Принципова схема гітарного ефекту зображена на рисунку 3.1. За її основу була взята популярна мікросхема PT2399, що дозволило скомбінувати необхідні елементи для побудови двох гітарних ефектів – ділей та повторювач.

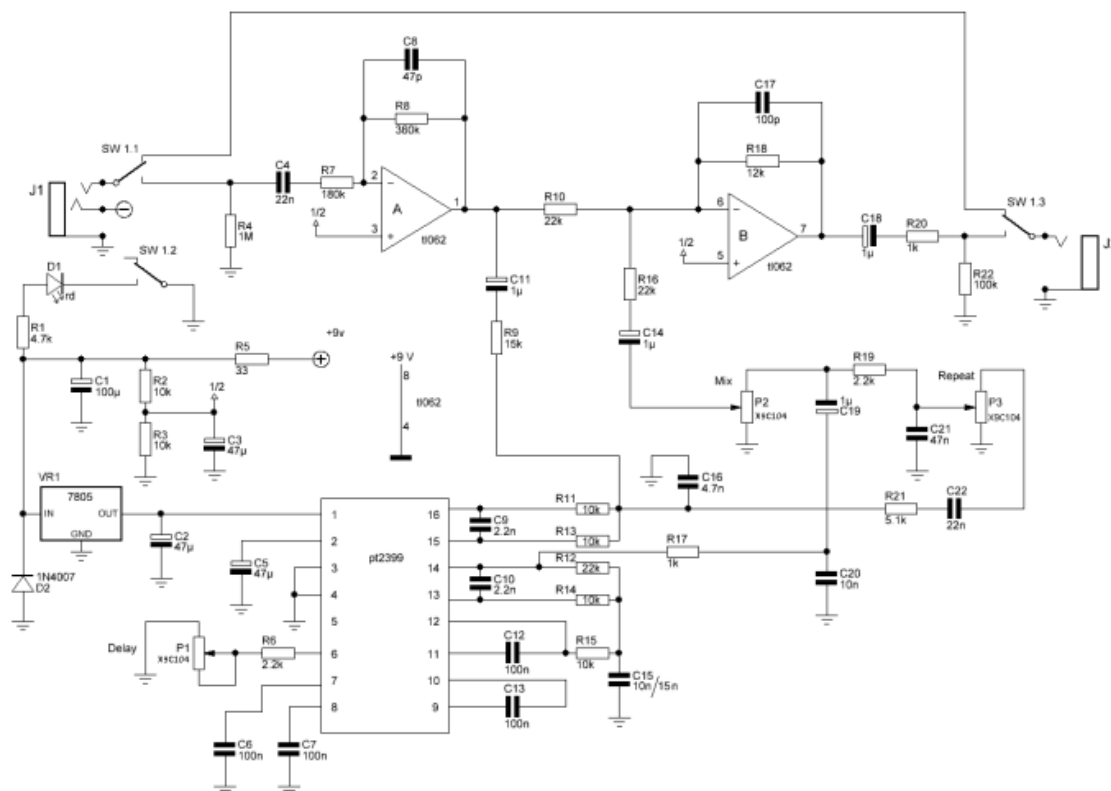


Рисунок 3.19 — Принципова електрична схема гітарного ефекту

Одним з основних елементів є мікроконтролер PT2399, який виконує основну роль у відтворенні ефекту «затримки» звуку у гітарному ефекті. Важливим елементом є цифрові потенціометри X9C104, які з'єднано з керуваною платою Arduino, яка, використовуючи написане програмне забезпечення, керує рівнем затримки, кількістю повторів звукового сигналу та його гучністю.

3.9.3 Загальний алгоритм роботи програми

Програмне забезпечення проекту складається з виконавчих команд мікроконтролера Arduino та мобільного додатку для iOS на мові програмування Swift.

Загальний алгоритм роботи програми для мікроконтролера Arduino UNO зображено на рисунку 3.20.

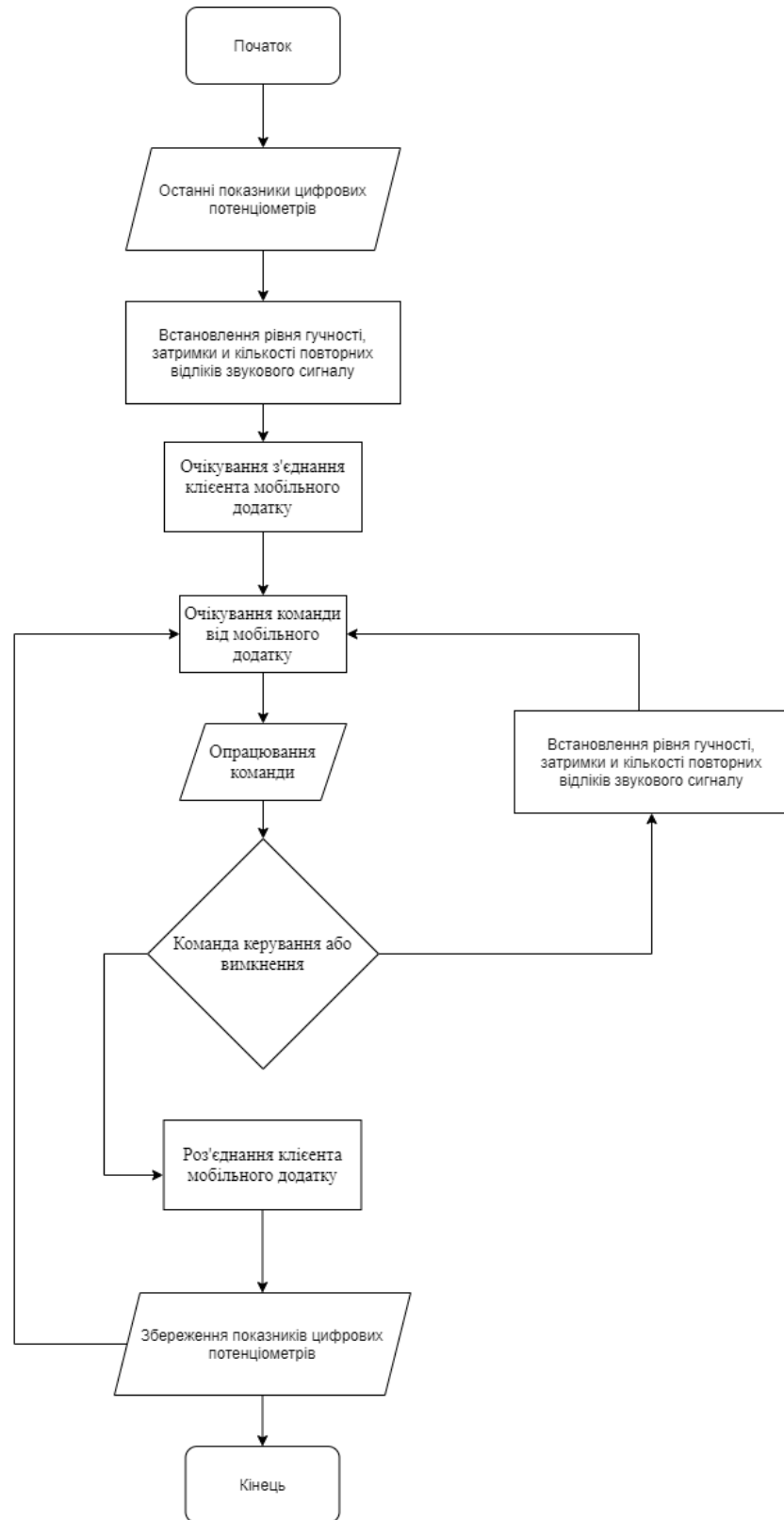


Рисунок 3.20 — Загальний алгоритм роботи гітарного ефекту

Основною задачею програми мікроконтролера є керування цифровими потенціометрами та збереження їх показників. Оскільки потенціометри не мають аналогового збереження значення опору резисторів які в них

використовуються, цифрове значення опору потенціометра, яке представлено у вигляді цілого числа у діапазоні від 0 до 10, зберігається у пам'яті мікроконтролера після роз'єднання з клієнтом мобільного додатку.

Алгоритм програми має циклічний характер:

- встановлення останніх значень показників потенціометрів;
- очікування клієнту мобільного додатку ;
- опрацювання отриманої Bluetooth команди;
- встановлення показників потенціометрів;
- роз'єднання клієнту або повернення до пункту 3;
- збереження останніх показників потенціометрів.

Лістинг iOS програми для керування мікроконтролером наведено в Додатку Г.

Основними складовими iOS програми є:

- модуль посередник між візуальним інтерфейсом та функціоналом передачі Bluetooth даних у програмі;
- модуль управління потенціометрами мікроконтролера Arduino;
- модуль пошуку доступних Bluetooth пристроїв для з'єднання.

4 ЕКОНОМІЧНИЙ РОЗДІЛ

Науково-технічна розробка має право на існування та впровадження, якщо вона відповідає вимогам часу, як в напрямку науково-технічного прогресу та і в плані економіки. Тому для науково-дослідної роботи необхідно оцінювати економічну ефективність результатів виконаної роботи.

Магістерська кваліфікаційна робота з розробки та дослідження «Засоби апаратно-програмного опрацювання сигналів в аудіопроцесорах з дистанційним керуванням» відноситься до науково-технічних робіт, які орієнтовані на виведення на ринок (або рішення про виведення науково-технічної розробки на ринок може бути прийнято у процесі проведення самої роботи), тобто коли відбувається так звана комерціалізація науково-технічної розробки. Цей напрямок є пріоритетним, оскільки результатами розробки можуть користуватися інші споживачі, отримуючи при цьому певний економічний ефект. Але для цього потрібно знайти потенційного інвестора, який би взявся за реалізацію цього проекту і переконати його в економічній доцільності такого кроку.

Для наведеного випадку нами мають бути виконані такі етапи робіт:

- проведено комерційний аудит науково-технічної розробки, тобто встановлення її науково-технічного рівня та комерційного потенціалу;
- розраховано витрати на здійснення науково-технічної розробки;
- розрахована економічна ефективність науково-технічної розробки у випадку її впровадження і комерціалізації потенційним інвестором і проведено обґрунтування економічної доцільності комерціалізації потенційним інвестором.

4.1 Проведення комерційного та технологічного аудиту науково-технічної розробки

Метою проведення комерційного і технологічного аудиту дослідження за темою «Засоби апаратно-програмного опрацювання сигналів в аудіопроцесорах з дистанційним керуванням» є оцінювання науково-

технічного рівня та рівня комерційного потенціалу розробки, створеної в результаті науково-технічної діяльності.

Оцінювання науково-технічного рівня розробки та її комерційного потенціалу рекомендується здійснювати із застосуванням 5-ти бальної системи оцінювання за 12-ма критеріями, наведеними в табл. 4.1 [13, 14].

Таблиця 4.1 – Рекомендовані критерії оцінювання науково-технічного рівня і комерційного потенціалу розробки та бальна оцінка

Бали (за 5-ти бальною шкалою)					
	0	1	2	3	4
Технічна здійсненність концепції					
1	Достовірність концепції не підтверджена	Концепція підтверджена експертними висновками	Концепція підтверджена розрахунками	Концепція перевірена на практиці	Перевірено працездатність продукту в реальних умовах
Ринкові переваги (недоліки)					
2	Багато аналогів на малому ринку	Мало аналогів на малому ринку	Кілька аналогів на великому ринку	Один аналог на великому ринку	Продукт не має аналогів на великому ринку
3	Ціна продукту значно вища за ціни аналогів	Ціна продукту дещо вища за ціни аналогів	Ціна продукту приблизно дорівнює цінам аналогів	Ціна продукту дещо нижче за ціни аналогів	Ціна продукту значно нижче за ціни аналогів
4	Технічні та споживчі властивості продукту значно гірші, ніж в	Технічні та споживчі властивості продукту трохи гірші, ніж в аналогів	Технічні та споживчі властивості продукту на рівні аналогів	Технічні та споживчі властивості продукту трохи кращі, ніж в	Технічні та споживчі властивості продукту значно кращі, ніж в
5	Експлуатаційні витрати значно вищі, ніж в аналогів	Експлуатаційні витрати дещо вищі, ніж в аналогів	Експлуатаційні витрати на рівні експлуатаційних витрат аналогів	Експлуатаційні витрати трохи нижчі, ніж в аналогів	Експлуатаційні витрати значно нижчі, ніж в аналогів
Ринкові перспективи					
6	Ринок малий і не має позитивної динаміки	Ринок малий, але має позитивну динаміку	Середній ринок з позитивною динамікою	Великий стабільний ринок	Великий ринок з позитивною динамікою
7	Активна конкуренція великих компаній на	Активна конкуренція	Помірна конкуренція	Незначна конкуренція	Конкурентів немає
Практична здійсненність					

Продовження таблиці 4.1

8	Відсутні фахівці як з технічної, так і з комерційної реалізації ідеї	Необхідно наймати фахівців або витратити значні кошти та час на навчання наявних фахівців	Необхідне незначне навчання фахівців та збільшення їх штату	Необхідне незначне навчання фахівців	Є фахівці з питань як з технічної, так і з комерційної реалізації ідеї
9	Потрібні значні фінансові ресурси, які відсутні. Джерела фінансування ідеї відсутні	Потрібні незначні фінансові ресурси. Джерела фінансування відсутні	Потрібні значні фінансові ресурси. Джерела фінансування є	Потрібні незначні фінансові ресурси. Джерела фінансування є	Не потребує додаткового фінансування
10	Необхідна розробка нових матеріалів	Потрібні матеріали, що використовуються у військово-промисловому комплексі	Потрібні дорогі матеріали	Потрібні досяжні та дешеві матеріали	Всі матеріали для реалізації ідеї відомі та давно використовуються у виробництві
11	Термін реалізації ідеї більший за 10 років	Термін реалізації ідеї більший за 5 років. Термін окупності інвестицій більше 10-ти років	Термін реалізації ідеї від 3-х до 5-ти років. Термін окупності інвестицій більше 5-ти років	Термін реалізації ідеї менше 3-х років. Термін окупності інвестицій від 3-х до 5-ти років	Термін реалізації ідеї менше 3-х років. Термін окупності інвестицій менше 3-х років
12	Необхідна розробка регламентних документів та отримання великої кількості дозвільних документів на виробництво та реалізацію продукту	Необхідно отримання великої кількості дозвільних документів на виробництво та реалізацію продукту, що вимагає значних коштів та часу	Процедура отримання дозвільних документів для виробництва та реалізації продукту вимагає незначних коштів та часу	Необхідно тільки повідомлення відповідним органам про виробництво та реалізацію продукту	Відсутні будь-які регламентні обмеження на виробництво та реалізацію продукту

Результати оцінювання науково-технічного рівня та комерційного потенціалу науково-технічної розробки потрібно звести до таблиці.

За результатами розрахунків, наведених в таблиці 4.2, зробимо висновок щодо науково-технічного рівня і рівня комерційного потенціалу розробки. При цьому використаємо рекомендації, наведені в табл. 4.3 [13, 14].

Таблиця 4.2 – Результати оцінювання науково-технічного рівня і комерційного потенціалу розробки експертами

Критерії	Експерт (ПІБ, посада)		
	1	2	3
	Бали:		
1. Технічна здійсненність концепції	4	4	4
2. Ринкові переваги (наявність аналогів)	1	1	1
3. Ринкові переваги (ціна продукту)	3	3	3
4. Ринкові переваги (технічні властивості)	2	2	2
5. Ринкові переваги (експлуатаційні витрати)	3	3	3
6. Ринкові перспективи (розмір ринку)	2	2	2
7. Ринкові перспективи (конкуренція)	1	1	1
8. Практична здійсненність (наявність фахівців)	3	3	3
9. Практична здійсненність (наявність фінансів)	3	4	4
10. Практична здійсненність (необхідність нових матеріалів)	4	4	4
11. Практична здійсненність (термін реалізації)	4	4	4
12. Практична здійсненність (розробка документів)	4	4	3
Сума балів	34	35	34
Середньоарифметична сума балів $СБ_c$	34,3		

Таблиця 4.3 – Науково-технічні рівні та комерційні потенціали розробки

Середньоарифметична сума балів $СБ_c$, розрахована на основі висновків експертів	Науково-технічний рівень та комерційний потенціал розробки
41...48	Високий
31...40	Вище середнього
21...30	Середній
11...20	Нижче середнього
0...10	Низький

Згідно проведених досліджень рівень комерційного потенціалу розробки за темою «Засоби апаратно-програмного опрацювання сигналів в аудіопроекторах з дистанційним керуванням» становить 34,3 бала, що, відповідно до таблиці 4.3, свідчить про комерційну важливість проведення даних досліджень (рівень комерційного потенціалу розробки вище середнього).

4.2 Визначення рівня конкурентоспроможності розробки

Одиничний параметричний індекс розраховуємо за формулою [13, 14]:

$$q_i = \frac{P_i}{P_{базі}}, \quad (4.1)$$

де q_i – одиничний параметричний індекс, розрахований за i -м параметром;

P_i – значення i -го параметра виробу;

$P_{базі}$ – аналогічний параметр базового виробу-аналога, з яким проводиться порівняння.

Загальні технічні та економічні характеристики розробки представлено в таблиці 4.4.

Таблиця 4.4 – Основні техніко-економічні показники аналога та розробки, що проектується

Показники (параметри)	Одиниця вимірювання	Аналог	Проектований пристрій	Відношення параметрів нової розробки до аналога	Питома вага показника
Напруга живлення	В	12	12	1	0,16
Кількість цифрових входів/виходів	шт	14	54	3,8	0,12

Продовження таблиці 4.4

Кількість аналогових входів	шт	6	16	2,67	0,24
Flash-пам'ять	КБ	32	128	4	0,3
SRAM	КБ	2	8	4	0,18
Експлуатаційні витрати	грн	45	32	0,71	0,45
Ціна	грн	2900	2150	0,74	0,55

Нормативні параметри оцінюємо показником, який отримує одне з двох значень: 1 – пристрій відповідає нормам і стандартам; 0 – не відповідає.

Груповий показник конкурентоспроможності за нормативними параметрами розраховуємо як добуток частинних показників за кожним параметром за формулою [13, 14]:

$$I_{\text{ГП}} = \prod_{i=1}^n q_i, \quad (4.2)$$

де $I_{\text{ГП}}$ – загальний показник конкурентоспроможності за нормативними параметрами;

q_i – одиничний (частинний) показник за i -м нормативним параметром;

n – кількість нормативних параметрів, які підлягають оцінюванню.

За нормативними параметрами розроблюваний пристрій відповідає вимогам ДСТУ, тому $I_{\text{ГП}} = 1$.

Значення групового параметричного індексу за технічними параметрами визначаємо з урахуванням вагомості (частки) кожного параметра [13, 14]:

$$I_{\text{ГП}} = \sum_{i=1}^n q_i \cdot \alpha_i, \quad (4.3)$$

де $I_{тп}$ – груповий параметричний індекс за технічними показниками (порівняно з виробом-аналогом);

q_i – одиничний параметричний показник i -го параметра;

α_i – вагомість i -го параметричного показника, $\sum_{i=1}^n \alpha_i = 1$;

n – кількість технічних параметрів, за якими оцінюється конкурентоспроможність.

Проведемо аналіз параметрів згідно даних таблиці 4.4.

$$I_{mn} = 1 \cdot 0,16 + 3,8 \cdot 0,12 + 2,67 \cdot 0,24 + 4 \cdot 0,3 + 4 \cdot 0,18 = 3,18.$$

Груповий параметричний індекс за економічними параметрами розраховуємо за формулою [13, 14]:

$$I_{EP} = \sum_{i=1}^m q_i \cdot \beta_i, \quad (4.4)$$

де I_{EP} – груповий параметричний індекс за економічними показниками;

q_i – економічний параметр i -го виду;

β_i – частка i -го економічного параметра, $\sum_{i=1}^m \beta_i = 1$;

m – кількість економічних параметрів, за якими здійснюється оцінювання.

Проведемо аналіз параметрів згідно даних таблиці .

$$I_{EP} = 0,71 \cdot 0,45 + 0,74 \cdot 0,55 = 0,73.$$

На основі групових параметричних індексів за нормативними, технічними та економічними показниками розрахуємо інтегральний показник конкурентоспроможності за формулою [13, 14]:

$$K_{\text{ИИТ}} = I_{\text{ИИП}} \cdot \frac{I_{\text{ИИП}}}{I_{\text{ЕП}}}, \quad (4.5)$$

$$K_{\text{ИИТ}} = 1 \cdot 3,18 / 0,73 = 4,37.$$

Інтегральний показник конкурентоспроможності $K_{\text{ИИТ}} > 1$, отже розробка переважає відомі аналоги за своїми техніко-економічними показниками.

4.3 Розрахунок витрат на проведення науково-дослідної роботи

Витрати, пов'язані з проведенням науково-дослідної роботи на тему «Засоби апаратно-програмного опрацювання сигналів в аудіопроекторів з дистанційним керуванням», під час планування, обліку і калькулювання собівартості науково-дослідної роботи групуємо за відповідними статтями.

4.3.1 Витрати на оплату праці

До статті «Витрати на оплату праці» належать витрати на виплату основної та додаткової заробітної плати керівникам відділів, лабораторій, секторів і груп, науковим, інженерно-технічним працівникам, конструкторам, технологам, креслярам, копіювальникам, лаборантам, робітникам, студентам, аспірантам та іншим працівникам, безпосередньо зайнятим виконанням конкретної теми, обчисленої за посадовими окладами, відрядними розцінками, тарифними ставками згідно з чинними в організаціях системами оплати праці.

Основна заробітна плата дослідників

Витрати на основну заробітну плату дослідників (Z_o) розраховуємо у відповідності до посадових окладів працівників, за формулою [13, 14]:

$$Z_o = \sum_{i=1}^k \frac{M_{ni} \cdot t_i}{T_p}, \quad (4.6)$$

де k – кількість посад дослідників залучених до процесу досліджень;

M_{ni} – місячний посадовий оклад конкретного дослідника, грн;

t_i – число днів роботи конкретного дослідника, дн.;

T_p – середнє число робочих днів в місяці, $T_p=21$ дні.

$$Z_o = 13280,00 \cdot 21 / 21 = 13280,00 \text{ грн.}$$

Проведені розрахунки зведемо до таблиці.

Таблиця 4.5 – Витрати на заробітну плату дослідників

Найменування посади	Місячний посадовий оклад, грн	Оплата за робочий день, грн	Число днів роботи	Витрати на заробітну плату, грн
Керівник проекту	13280,00	632,38	21	13280,00
Інженер-розробник електронних пристроїв	12650,00	602,38	10	6023,81
Інженер-системотехнік 1-ї кат.	12100,00	576,19	12	6914,29
Інженер-програміст	12400,00	590,48	5	2952,38
Технік	7210,00	343,33	10	3433,33
Всього				32603,81

Витрати на основну заробітну плату робітників (Z_p) за відповідними найменуваннями робіт НДР на тему «Засоби апаратно-програмного опрацювання сигналів в аудіопроекторах з дистанційним керуванням» розраховуємо за формулою:

$$Z_p = \sum_{i=1}^n C_i \cdot t_i, \quad (4.7)$$

де C_i – погодинна тарифна ставка робітника відповідного розряду, за виконану відповідну роботу, грн/год;

t_i – час роботи робітника при виконанні визначеної роботи, год.

Погодинну тарифну ставку робітника відповідного розряду C_i можна визначити за формулою:

$$C_i = \frac{M_M \cdot K_i \cdot K_c}{T_p \cdot t_{зм}}, \quad (4.8)$$

де M_M – розмір прожиткового мінімуму працездатної особи, або мінімальної місячної заробітної плати (в залежності від діючого законодавства), прийmemo $M_M=2379,00$ грн;

K_i – коефіцієнт міжкваліфікаційного співвідношення для встановлення тарифної ставки робітнику відповідного розряду (табл. Б.2, додаток Б) [13, 14];

K_c – мінімальний коефіцієнт співвідношень місячних тарифних ставок робітників першого розряду з нормальними умовами праці виробничих об'єднань і підприємств до законодавчо встановленого розміру мінімальної заробітної плати.

T_p – середнє число робочих днів в місяці, приблизно $T_p = 21$ дн;

$t_{зм}$ – тривалість зміни, год.

$$C_1 = 2379,00 \cdot 1,10 \cdot 1,65 / (21 \cdot 8) = 25,70 \text{ грн.}$$

$$З_{р1} = 25,70 \cdot 6,00 = 154,21 \text{ грн.}$$

Таблиця 4.6 – Величина витрат на основну заробітну плату робітників

Найменування робіт	Тривалість роботи, год	Розряд роботи	Тарифний коефіцієнт	Погодинна тарифна ставка, грн	Величина оплати на робітника грн
Установка обладнання	6,00	2	1,10	25,70	154,21
Підготовка робочого місця розробника	2,40	2	1,10	25,70	61,68
Інсталяція програмного забезпечення	2,20	5	1,70	39,72	87,39

Продовження таблиці 4.6

Перевірка вхідних компонентів	1,50	4	1,50	35,05	52,57
Налагодження програмних блоків	5,60	6	2,00	46,73	261,69
Налагодження апаратних блоків	4,00	5	1,70	39,72	158,88
Випробування моделі	2,00	2	1,10	25,70	51,40
Всього					827,83

Додаткову заробітну плату розраховуємо як 10 ... 12% від суми основної заробітної плати дослідників та робітників за формулою:

$$Z_{\text{дод}} = (Z_o + Z_p) \cdot \frac{H_{\text{дод}}}{100\%}, \quad (4.9)$$

де $H_{\text{дод}}$ – норма нарахування додаткової заробітної плати. Прийmemo 12%.

$$Z_{\text{дод}} = (32603,81 + 827,83) \cdot 12 / 100\% = 4011,80 \text{ грн.}$$

4.3.2 Відрахування на соціальні заходи

Нарахування на заробітну плату дослідників та робітників розраховуємо як 22% від суми основної та додаткової заробітної плати дослідників і робітників за формулою:

$$Z_n = (Z_o + Z_p + Z_{\text{дод}}) \cdot \frac{H_{\text{зн}}}{100\%}, \quad (4.10)$$

де $H_{\text{зн}}$ – норма нарахування на заробітну плату. Приймаємо 22%.

$$Z_n = (32603,81 + 827,83 + 4011,80) \cdot 22 / 100\% = 8237,56 \text{ грн.}$$

4.3.3 Сировина та матеріали

До статті «Сировина та матеріали» належать витрати на сировину, основні та допоміжні матеріали, інструменти, пристрої та інші засоби і предмети праці, які придбані у сторонніх підприємств, установ і організацій та витрачені на проведення досліджень за темою «Засоби апаратно-програмного опрацювання сигналів в аудіопроекторах з дистанційним керуванням».

Витрати на матеріали (M), у вартісному вираженні розраховуються окремо по кожному виду матеріалів за формулою:

$$M = \sum_{j=1}^n H_j \cdot C_j \cdot K_j - \sum_{j=1}^n B_j \cdot C_{ej}, \quad (4.11)$$

де H_j – норма витрат матеріалу j -го найменування, кг;

n – кількість видів матеріалів;

C_j – вартість матеріалу j -го найменування, грн/кг;

K_j – коефіцієнт транспортних витрат, ($K_j = 1,1 \dots 1,15$);

B_j – маса відходів j -го найменування, кг;

C_{ej} – вартість відходів j -го найменування, грн/кг.

$$M_1 = 2,00 \cdot 113,00 \cdot 1,1 - 0,000 \cdot 0,00 = 248,60 \text{ грн.}$$

Таблиця 4.7 – Витрати на матеріали

Найменування матеріалу, марка, тип, сорт	Ціна за 1 кг, грн	Норма витрат, кг	Величина відходів, кг	Ціна відходів, грн/кг	Вартість витраченого матеріалу, грн
Офісний папір SOFTUltra Plus	113,00	2,00	0,000	0,00	248,60
Папір для записів Parers SOFT Light A5	72,00	1,00	0,000	0,00	79,20
Органайзер офісний OFFICE SOFT	211,00	3,00	0,000	0,00	696,30

Продовження таблиці 4.7

Канцелярське приладдя (набір офісного працівника)	174,00	5,00	0,000	0,00	957,00
Картридж для принтера Canon LBP6500	850,00	1,00	0,000	0,00	935,00
Диск оптичний NewOPtice CD-RW	12,20	3,00	0,000	0,00	40,26
Flesh-пам'ять Kingston 16 GB	99,00	1,00	0,000	0,00	108,90
Тека для паперів BOSS PAPERS BOX	87,00	2,00	0,000	0,00	191,40
Всього					3256,66

4.3.3 Розрахунок витрат на комплектуючі

Витрати на комплектуючі (K_6), які використовують при проведенні НДР на тему «Засоби апаратно-програмного опрацювання сигналів в аудіопроекторів з дистанційним керуванням», розраховуємо, згідно з їхньою номенклатурою, за формулою:

$$K_6 = \sum_{j=1}^n H_j \cdot C_j \cdot K_j, \quad (4.12)$$

де H_j – кількість комплектуючих j -го виду, шт.;

C_j – покупна ціна комплектуючих j -го виду, грн;

K_j – коефіцієнт транспортних витрат, ($K_j = 1,1 \dots 1,15$).

$K_6 = 1 \cdot 255,00 \cdot 1,1 = 280,50$ грн.

Проведені розрахунки зведемо до таблиці.

Розрахунок витрат на комплектуючі представлено в таблиці 4.8.

Таблиця 4.8 – Витрати на комплектуючі

Найменування комплектуючих	Кількість, шт.	Ціна за штуку, грн	Сума, грн
Arduino UNO R3	1	255,00	280,50
Цифрові потенціометри Х9С104	3	15,00	49,50
Bluetooth-модуль HC-06	1	86,00	94,60
З'єднувальні дроти для Arduino	4	21,00	92,40
Конектори	4	5,00	22,00
Мікросхема Pt2399	1	42,00	46,20
Мікросхема TL062	1	36,00	39,60
Мікросхема 78L05	1	28,00	30,80
Світлодіод	1	8,00	8,80
Діод 1n4007	1	16,00	17,60
Гнізда Stereo jack 6.3	6	11,50	75,90
Кнопка 3PDT	1	7,50	8,25
Роз'єм для елемента живлення	1	4,80	5,28
Корпус Gainta 1590B	1	98,00	107,80

Продовження таблиці 4.8

Резистори 0,25 Вт	5	1,45	7,98
Конденсатори 16 В	15	2,60	42,90
Кабель USB	1	55,00	60,50
Всього			990,61

4.3.4 Спецустаткування для наукових робіт

До статті «Спецустаткування для наукових (експериментальних) робіт» належать витрати на виготовлення та придбання спецустаткування необхідного для проведення досліджень, також витрати на їх проектування, виготовлення, транспортування, монтаж та встановлення.

Балансову вартість спецустаткування розраховуємо за формулою:

$$B_{\text{спец}} = \sum_{i=1}^k C_i \cdot C_{\text{пр.і}} \cdot K_i, \quad (4.13)$$

де C_i – ціна придбання одиниці спецустаткування даного виду, марки, грн;

$C_{np.i}$ – кількість одиниць устаткування відповідного найменування, які придбані для проведення досліджень, шт.;

K_i – коефіцієнт, що враховує доставку, монтаж, налагодження устаткування тощо, ($K_i = 1,10 \dots 1,12$);

k – кількість найменувань устаткування.

$$B_{спец} = 5820,00 \cdot 1 \cdot 1,1 = 6402,00 \text{ грн.}$$

Отримані результати зведемо до таблиці:

Таблиця 4.9 – Витрати на придбання спецустаткування по кожному виду

Найменування устаткування	Кількість, шт	Ціна за одиницю, грн	Вартість, грн
Монтажна станція RADIOEFFECT-Z1200	1	5820,00	6402,00
Осцилограф цифровий ОЦ-1534	1	6230,00	6853,00
Всього			13255,00

4.3.5 Програмне забезпечення для наукових робіт

До статті «Програмне забезпечення для наукових (експериментальних) робіт» належать витрати на розробку та придбання спеціальних програмних засобів і програмного забезпечення, (програм, алгоритмів, баз даних) необхідних для проведення досліджень, також витрати на їх проектування, формування та встановлення.

Балансову вартість програмного забезпечення розраховуємо за формулою:

$$B_{прог} = \sum_{i=1}^k C_{инрг} \cdot C_{прог.i} \cdot K_i, \quad (4.14)$$

де $C_{инрг}$ – ціна придбання одиниці програмного засобу даного виду, грн;

$C_{прог.i}$ – кількість одиниць програмного забезпечення відповідного найменування, які придбані для проведення досліджень, шт.;

K_i – коефіцієнт, що враховує інсталяцію, налагодження програмного засобу тощо, ($K_i = 1,10 \dots 1,12$);

k – кількість найменувань програмних засобів.

$$B_{\text{пр}} = 10250,00 \cdot 1 \cdot 1,1 = 11275,00 \text{ грн.}$$

Отримані результати зведемо до таблиці:

Таблиця 4.10 – Витрати на придбання програмних засобів по кожному виду

Найменування програмного засобу	Кількість, шт	Ціна за одиницю, грн	Вартість, грн
Пакет імітаційного моделювання Arena Rockwell Automation	1	10250,00	11275,00
Всього			11275,00

4.3.6 Амортизація обладнання, програмних засобів та приміщень

В спрощеному вигляді амортизаційні відрахування по кожному виду обладнання, приміщень та програмному забезпеченню тощо, розраховуємо з використанням прямолінійного методу амортизації за формулою:

$$A_{\text{обл}} = \frac{Ц_{\text{б}}}{T_{\text{с}}} \cdot \frac{t_{\text{вик}}}{12}, \quad (4.15)$$

де $Ц_{\text{б}}$ – балансова вартість обладнання, програмних засобів, приміщень тощо, які використовувались для проведення досліджень, грн;

$t_{\text{вик}}$ – термін використання обладнання, програмних засобів, приміщень під час досліджень, місяців;

$T_{\text{с}}$ – строк корисного використання обладнання, програмних засобів, приміщень тощо, років.

$$A_{обл} = (24300,00 \cdot 1) / (2 \cdot 12) = 1012,50 \text{ грн.}$$

Проведені розрахунки зведемо до таблиці.

Таблиця 4.11 — Амортизаційні відрахування по кожному виду обладнання

Найменування обладнання	Балансова вартість, грн	Строк корисного використання, років	Термін використання обладнання, місяців	Амортизаційні відрахування, грн
Персональний комп'ютер розробки електронних компонентів	24300,00	2	1	1012,50
Робоче місце розробника	7632,00	5	1	127,20
Пристрої виводу інформації	6620,00	4	1	137,92
Оргтехніка	9560,00	4	1	199,17
Приміщення лабораторії	209000,00	20	1	870,83
ОС Windows 10	5420,00	2	1	225,83

4.3.7 Паливо та енергія для науково-виробничих цілей

Витрати на силову електроенергію (B_e) розраховуємо за формулою:

$$B_e = \sum_{i=1}^n \frac{W_{yi} \cdot t_i \cdot C_e \cdot K_{ені}}{\eta_i}, \quad (4.16)$$

де W_{yi} – встановлена потужність обладнання на визначеному етапі розробки, кВт;

t_i – тривалість роботи обладнання на етапі дослідження, год;

C_e – вартість 1 кВт-години електроенергії, грн; (вартість електроенергії визначається за даними енергопостачальної компанії), прийmemo $C_e = 4,50$ грн;

K_{eni} – коефіцієнт, що враховує використання потужності, $K_{eni} < 1$;

η_i – коефіцієнт корисної дії обладнання, $\eta_i < 1$.

$$B_e = 0,45 \cdot 160,0 \cdot 4,50 \cdot 0,95 / 0,97 = 324,00 \text{ грн.}$$

Проведені розрахунки зведемо до таблиці.

Таблиця 4.12 – Витрати на електроенергію

Найменування обладнання	Встановлена потужність, кВт	Тривалість роботи, год	Сума, грн
Персональний комп'ютер розробки електронних компонентів	0,45	160,0	324,00
Робоче місце розробника	0,10	160,0	72,00
Пристрої виводу інформації	0,03	25,0	3,38
Оргтехніка	0,65	5,0	14,63
Монтажна станція RADIOEFFECT-Z1200	0,10	5,0	2,25
Осцилограф цифровий ОЦ-1534	0,06	15,0	4,05
Всього			420,30

4.3.8 Службові відрядження

До статті «Службові відрядження» дослідної роботи на тему «Засоби апаратно-програмного опрацювання сигналів в аудіопроцесорах з дистанційним керуванням» належать витрати на відрядження штатних працівників, працівників організацій, які працюють за договорами цивільно-правового характеру, аспірантів, зайнятих розробленням досліджень, відрядження, пов'язані з проведенням випробувань машин та приладів, а також витрати на відрядження на наукові з'їзди, конференції, наради, пов'язані з виконанням конкретних досліджень.

Витрати за статтею «Службові відрядження» розраховуємо як 20...25% від суми основної заробітної плати дослідників та робітників за формулою:

$$B_{св} = (Z_o + Z_p) \cdot \frac{H_{св}}{100\%}, \quad (4.17)$$

де H_{cv} – норма нарахування за статтею «Службові відрядження», приймемо $H_{cv} = 20\%$.

$$B_{cv} = (32603,81 + 827,83) \cdot 20 / 100\% = 6686,33 \text{ грн.}$$

4.3.9 Витрати на роботи, які виконують сторонні підприємства, установи і організації

Витрати за статтею «Витрати на роботи, які виконують сторонні підприємства, установи і організації» розраховуємо як 30...45% від суми основної заробітної плати дослідників та робітників за формулою:

$$B_{cn} = (Z_o + Z_p) \cdot \frac{H_{cn}}{100\%}, \quad (4.18)$$

де H_{cn} – норма нарахування за статтею «Витрати на роботи, які виконують сторонні підприємства, установи і організації», приймемо $H_{cn} = 30\%$.

$$B_{cn} = (32603,81 + 827,83) \cdot 30 / 100\% = 10029,49 \text{ грн.}$$

4.3.10 Інші витрати

До статті «Інші витрати» належать витрати, які не знайшли відображення у зазначених статтях витрат і можуть бути віднесені безпосередньо на собівартість досліджень за прямими ознаками.

Витрати за статтею «Інші витрати» розраховуємо як 50...100% від суми основної заробітної плати дослідників та робітників за формулою:

$$I_s = (Z_o + Z_p) \cdot \frac{H_{is}}{100\%}, \quad (4.19)$$

де H_{is} – норма нарахування за статтею «Інші витрати», приймемо $H_{is} = 50\%$.

$$I_s = (32603,81 + 827,83) \cdot 50 / 100\% = 16715,82 \text{ грн.}$$

4.3.11 Загальновиробничі витрати

До статті «Накладні (загальновиробничі) витрати» належать: витрати, пов'язані з управлінням організацією; витрати на винахідництво та раціоналізацію; витрати на підготовку (перепідготовку) та навчання кадрів; витрати, пов'язані з набором робочої сили; витрати на оплату послуг банків; витрати, пов'язані з освоєнням виробництва продукції; витрати на науково-технічну інформацію та рекламу та ін.

Витрати за статтею «Накладні (загальновиробничі) витрати» розраховуємо як 100...150% від суми основної заробітної плати дослідників та робітників за формулою:

$$B_{нзв} = (Z_o + Z_p) \cdot \frac{H_{нзв}}{100\%}, \quad (4.20)$$

де $H_{нзв}$ – норма нарахування за статтею «Накладні (загальновиробничі) витрати», приймемо $H_{нзв} = 100\%$.

$$B_{нзв} = (32603,81 + 827,83) \cdot 100 / 100\% = 33431,64 \text{ грн.}$$

Витрати на проведення науково-дослідної роботи на тему «Засоби апаратно-програмного опрацювання сигналів в аудіопроцесорах з дистанційним керуванням» розраховуємо як суму всіх попередніх статей витрат за формулою:

$$B_{заг} = Z_o + Z_p + Z_{доп} + Z_n + M + K_v + B_{спец} + B_{прз} + A_{обл} + B_e + B_{св} + B_{сп} + I_v + B_{нзв}. \quad (4.21)$$

$$B_{заг} = 32603,81 + 827,83 + 4011,80 + 8237,555554 + 3256,66 + 990,61 + 13255,00 + 11275,00 + 2674,28 + 420,30 + 6686,33 + 10029,49 + 16715,82 + 33431,64 = 144416,11 \text{ грн.}$$

Загальні витрати $3B$ на завершення науково-дослідної (науково-технічної) роботи та оформлення її результатів розраховується за формулою:

$$3B = \frac{B_{заг}}{\eta}, \quad (4.22)$$

де η - коефіцієнт, який характеризує етап (стадію) виконання науково-дослідної роботи, прийmemo $\eta=0,95$.

$$3B = 144416,11 / 0,95 = 152016,96 \text{ грн.}$$

4.4 Розрахунок економічної ефективності науково-технічної розробки при її можливій комерціалізації потенційним інвестором

В ринкових умовах узагальнюючим позитивним результатом, що його може отримати потенційний інвестор від можливого впровадження результатів тієї чи іншої науково-технічної розробки, є збільшення у потенційного інвестора величини чистого прибутку.

Результати дослідження проведені за темою «Засоби апаратно-програмного опрацювання сигналів в аудіопроекторах з дистанційним керуванням» передбачають комерціалізацію протягом 4-х років реалізації на ринку.

Показник	1-й рік	2-й рік	3-й рік	4-й рік
Збільшення кількості споживачів, осіб	250	500	1000	800

В цьому випадку майбутній економічний ефект буде формуватися на основі таких даних:

— ΔN – збільшення кількості споживачів пристрою, у періоди часу, що аналізуються, від покращення його певних характеристик;

— N – кількість споживачів які використовували аналогічний пристрій у році до впровадження результатів нової науково-технічної розробки, прийmemo 7000 осіб;

— C_o – вартість пристрою у році до впровадження результатів розробки, прийmemo 2000,00 грн;

— $\pm\Delta C_o$ – зміна вартості пристрою від впровадження результатів науково-технічної розробки, прийmemo 150,00 грн.

Можливе збільшення чистого прибутку у потенційного інвестора $\Delta\Pi_i$ для кожного із 4-х років, протягом яких очікується отримання позитивних результатів від можливого впровадження та комерціалізації науково-технічної розробки, розраховуємо за формулою [13, 14]:

$$\Delta\Pi_i = (\pm\Delta C_o \cdot N + C_o \cdot \Delta N)_i \cdot \lambda \cdot \rho \cdot \left(1 - \frac{\vartheta}{100}\right), \quad (4.23)$$

де λ – коефіцієнт, який враховує сплату потенційним інвестором податку на додану вартість, у 2021 році ставка податку на додану вартість складає 20%, а коефіцієнт $\lambda = 0,8333$;

ρ – коефіцієнт, який враховує рентабельність інноваційного продукту).
Прийmemo $\rho = 35\%$;

ϑ – ставка податку на прибуток, який має сплачувати потенційний інвестор, у 2021 році $\vartheta = 18\%$;

Збільшення чистого прибутку 1-го року:

$$\Delta\Pi_1 = (150,00 \cdot 7000,00 + 2150,00 \cdot 250) \cdot 0,83 \cdot 0,35 \cdot (1 - 0,18/100\%) = 378158,38 \text{ грн.}$$

Збільшення чистого прибутку 2-го року:

$$\Delta\Pi_2 = (150,00 \cdot 7000,00 + 2150,00 \cdot 750) \cdot 0,83 \cdot 0,35 \cdot (1 - 0,18/100\%) = 634234,13 \text{ грн.}$$

Збільшення чистого прибутку 3-го року:

$$\Delta\Pi_3 = (150,00 \cdot 7000,00 + 2150,00 \cdot 1750) \cdot 0,83 \cdot 0,35 \cdot (1 - 0,18/100\%) = 1146385,63$$

грн.

Збільшення чистого прибутку 4-го року:

$$\Delta\Pi_4 = (150,00 \cdot 7000,00 + 2150,00 \cdot 2550) \cdot 0,83 \cdot 0,35 \cdot (1 - 0,18/100\%) = 1556106,83$$

грн.

Приведена вартість збільшення всіх чистих прибутків III , що їх може отримати потенційний інвестор від можливого впровадження та комерціалізації науково-технічної розробки:

$$III = \sum_{i=1}^T \frac{\Delta\Pi_i}{(1 + \tau)^t}, \quad (4.24)$$

де $\Delta\Pi_i$ – збільшення чистого прибутку у кожному з років, протягом яких виявляються результати впровадження науково-технічної розробки, грн;

T – період часу, протягом якого очікується отримання позитивних результатів від впровадження та комерціалізації науково-технічної розробки, роки;

τ – ставка дисконтування, за яку можна взяти щорічний прогнозований рівень інфляції в країні, $\tau = 0,14$;

t – період часу (в роках) від моменту початку впровадження науково-технічної розробки до моменту отримання потенційним інвестором додаткових чистих прибутків у цьому році.

$$\begin{aligned} III &= 378158,38/(1+0,14)^1 + 634234,13/(1+0,14)^2 + 1146385,63/(1+0,14)^3 + \\ &+ 1556106,83/(1+0,14)^4 = 331717,87 + 488022,56 + 773777,64 + 921340,16 = 2514858, \\ &24 \text{ грн.} \end{aligned}$$

Величина початкових інвестицій PV , які потенційний інвестор має вкласти для впровадження і комерціалізації науково-технічної розробки:

$$PV = k_{инв} \cdot 3B, \quad (4.25)$$

де $k_{инв}$ – коефіцієнт, що враховує витрати інвестора на впровадження науково-технічної розробки та її комерціалізацію, приймаємо $k_{инв} = 1,5$;

$3B$ – загальні витрати на проведення науково-технічної розробки та оформлення її результатів, приймаємо 152016,96 грн.

$$PV = k_{инв} \cdot 3B = 1,5 \cdot 152016,96 = 228025,44 \text{ грн.}$$

Абсолютний економічний ефект $E_{абс}$ для потенційного інвестора від можливого впровадження та комерціалізації науково-технічної розробки становитиме:

$$E_{абс} = III - PV, \quad (4.26)$$

де III – приведена вартість зростання всіх чистих прибутків від можливого впровадження та комерціалізації науково-технічної розробки, 2514858,24 грн;

PV – теперішня вартість початкових інвестицій, 228025,44 грн.

$$E_{абс} = III - PV = 2514858,24 - 228025,44 = 2286832,80 \text{ грн.}$$

Внутрішня економічна дохідність інвестицій E_6 , які можуть бути вкладені потенційним інвестором у впровадження та комерціалізацію науково-технічної розробки:

$$E_g = T_{ж} \sqrt[4]{1 + \frac{E_{абс}}{PV}} - 1, \quad (4.27)$$

де $E_{абс}$ – абсолютний економічний ефект вкладених інвестицій, 2286832,80 грн;

PV – теперішня вартість початкових інвестицій, 228025,44 грн;

$T_{ж}$ – життєвий цикл науково-технічної розробки, тобто час від початку її розробки до закінчення отримання позитивних результатів від її впровадження, 4 роки.

$$E_g = T_{ж} \sqrt[4]{1 + \frac{E_{абс}}{PV}} - 1 = (1 + 2286832,80/228025,44)^{1/4} - 1 = 0,82.$$

Мінімальна внутрішня економічна дохідність вкладених інвестицій $\tau_{мін}$

:

$$\tau_{мін} = d + f, \quad (4.28)$$

де d – середньозважена ставка за депозитними операціями в комерційних банках; в 2021 році в Україні $d = 0,11$;

f – показник, що характеризує ризикованість вкладення інвестицій, прийmemo 0,16.

$\tau_{мін} = 0,11 + 0,16 = 0,27 < 0,82$ свідчить про те, що внутрішня економічна дохідність інвестицій E_g , які можуть бути вкладені потенційним інвестором у впровадження та комерціалізацію науково-технічної розробки вища мінімальної внутрішньої дохідності. Тобто інвестувати в науково-дослідну роботу за темою «Засоби апаратно-програмного опрацювання сигналів в аудіопроекторів з дистанційним керуванням» доцільно.

Період окупності інвестицій $T_{ок}$ які можуть бути вкладені потенційним інвестором у впровадження та комерціалізацію науково-технічної розробки:

$$T_{ок} = \frac{1}{E_{\epsilon}}, \quad (4.29)$$

де E_{ϵ} – внутрішня економічна дохідність вкладених інвестицій.

$$T_{ок} = 1 / 0,82 = 1,22 \text{ р.}$$

$T_{ок} < 3$ -х років, що свідчить про комерційну привабливість науково-технічної розробки і може спонукати потенційного інвестора профінансувати впровадження даної розробки та виведення її на ринок.

Згідно проведених досліджень рівень комерційного потенціалу розробки за темою «Засоби апаратно-програмного опрацювання сигналів в аудіопроекторах з дистанційним керуванням» становить 34,3 бала, що, свідчить про комерційну важливість проведення даних досліджень (рівень комерційного потенціалу розробки вище середнього).

При оцінюванні рівня конкурентоспроможності, згідно узагальненого коефіцієнту конкурентоспроможності розробки, науково-технічна розробка переважає існуючі аналоги приблизно в 4,37 рази.

Також термін окупності становить 1,22 р., що менше 3-х років, що свідчить про комерційну привабливість науково-технічної розробки і може спонукати потенційного інвестора профінансувати впровадження даної розробки та виведення її на ринок.

Отже можна зробити висновок про доцільність проведення науково-дослідної роботи за темою «Засоби апаратно-програмного опрацювання сигналів в аудіопроекторах з дистанційним керуванням».

ВИСНОВКИ

В роботі вирішено такі задачі:

- проаналізовано сучасний стан і тенденції розвитку зарубіжних і вітчизняних розробок обробки звукових сигналів;
- обрано, вдосконалено й реалізовано методи опрацювання аудіосигналів;
- запропоновано програмні-апаратні засоби обробки та передачі звуку з використанням технологій бездротового з'єднання.

Отже, мету роботи — створення ефективних апаратно-програмних засобів для обробки звуку з використанням бездротової передачі сигналів керування, досягнуто.

Науковою новизною роботи є вдосконалення спеціалізованих методів опрацювання звукових сигналів з використанням бездротові технології керування їх параметрами. Базовані на цьому принципі структурні рішення дозволяють значно покращити функціональні можливості сучасних систем інструментального звукозапису та звуковідтворення.

Практичне значення роботи полягає в тому, що в апаратній частині запропоновано оригінальне рішення побудови цифрових пристроїв інструментального запису та відтворення звуку, що підвищить споживчі властивості останніх, не збільшуючи виробничі витрати у значній мірі. Використані науково-технічні рішення дозволяють реалізувати та адаптувати їх для концертної діяльності музичних колективів.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Alfred K. Susskind, Notes on Analog-Digital Conversion Techniques, John Wiley, 1957
2. Donald B. Bruck, Data Conversion Handbook, Hybrid Systems Corporation, 1974.
3. Michael J. Demler, High-Speed Analog-to-Digital Conversion, Academic Press, Inc., 1991, ISBN 0-12-209048-9.
4. Walt Kester, High Speed Design Techniques, Analog Devices, 1996, ISBN 0-916550-17-6.
5. Fundamentals of Music Processing: Audio, Analysis, Algorithms, Applications, 2015 edition, Meinard Müller
6. Digital Sound Processing for Music and Multimedia, 1999
7. Datasheet ATmega16U4/ATmega32U4 8-bit Microcontroller with 16/32K bytes of ISP Flash and USB Controller, Atmel 2015.
8. Блум Д. Изучаем Arduino: инструменты и методы технического волшебства. СПб: БВХ-Петербург, 2015. 336с.
9. Соммер У. Программирование микроконтроллерных плат Arduino/Freduino. СПб: БВХ-Петербург, 2012. 256с.
10. Ревич Ю. Занимательная электроника СПб: БВХ-Петербург, 2015. 322с.
11. Datasheet Atmega48a/PA/88A/PA/168A/PA/328/P Atmel 8-bit Atmel Microcontroller with 4/8/16/32KB In-System Programmable Flash, Atmel 2015
12. Гирін. Д., Програмні засоби для експериментальних досліджень діаграм направленості мікрофонних решіток. НТКП ВНТУ. Факультет інформаційних технологій та комп'ютерної інженерії, Ukraine, mar. 2018. Available at: <<https://conferences.vntu.edu.ua/index.php/all-fitki/all-fitki-2018/paper/view/4851/4037>>. Date accessed: 19 Dec. 2021
13. Методичні вказівки до виконання економічної частини магістерських кваліфікаційних робіт / Уклад. : В. О. Козловський, О. Й.

Лесько, В. В. Кавецький. – Вінниця : ВНТУ, 2021. – 42 с.

14. Кавецький В. В. Економічне обґрунтування інноваційних рішень: практикум / В. В. Кавецький, В. О. Козловський, І. В. Причепа – Вінниця : ВНТУ, 2016. – 113 с.

ДОДАТОК А

Міністерство освіти і науки України
Вінницький національний технічний університет
Факультет інформаційних технологій та комп'ютерної інженерії
Кафедра обчислювальної техніки

ЗАТВЕРДЖУЮ

Завідувач кафедри ОТ

Азаров О. Д.

« ____ » _____ 2021
року

ТЕХНІЧНЕ ЗАВДАННЯ

на виконання магістерської кваліфікаційної роботи на тему:
«Технологія формування та розпізнавання звукових фрагментів і команд в
комп'ютерних системах»
08-23.МКР.003.00.000 ТЗ

Науковий керівник: _____ Крупельницький

Л. В. (підпис)

студент групи _____ Гирін Д. Ю.

(підпис)

Вінниця 2021

1 Підстава для виконання магістерської кваліфікаційної роботи (МКР):

— актуальність досліджень;

— наказ про затвердження теми дипломної роботи.

2 Метою роботи є створення ефективних апаратно-програмних засобів для обробки звуку з використанням бездротової передачі сигналів керування.

3 Джерелами розробки МКР є:

— Соммер У. Программирование микроконтроллерных плат Arduino/Freduino. СПб: БВХ-Петербург, 2012. 256с.

— Ревич Ю. Занимательная электроника СПб: БВХ-Петербург, 2015. 322с.

— Fundamentals of Music Processing: Audio, Analysis, Algorithms, Applications, 2015 edition, Meinard Müller

4 Технічними вимогами до виконання МКР було врахувати можливі фактори, що впливають на якість і стабільність передачі звукового сигналу. Також пристрої повинні мати невеликі розміри, можливість модифікації. Програма повинна керувати мікроконтролером за допомогою Bluetooth.

5 Етапи МКР та очікувані результати наведено в таблиці А.1.

Таблиця А.1 — Етапи виконання роботи

№ етапу	Назва етапу	Термін виконання		Очікувані результати
		початок	кінець	
1	Огляд і аналіз джерел інформації	01.09.2021	01.10.2021	Аналітичний огляд літературних джерел
2	Розробка технічного завдання	01.10.2021	10.10.2021	Огляд задачі дослідження
3	Аналіз засобів перетворення й опрацювання звукових сигналів	11.10.2021	24.10.2021	Розділ 1
4	Аналіз методів і засоби обробки сигналів для аудіопроекторів з дистанційним керуванням	25.10.2021	7.11.2021	Розділ 2
5	Розробка методів цифрового опрацювання сигналів і схем аудіопроекторів з дистанційним керуванням	8.11.2021	21.11.2021	Розділ 3

Продовження таблиці А.1

6	Розрахунок економічної частини	22.11.2021	3.12.2021	Розділ 4
7	Підготовка до презентації і захист роботи	4.12.2021	20.12.2021	Презентація

6 Матеріали, що подаються до захисту МКР: пояснювальна записка МКР, графічні та ілюстративні матеріали, протокол попереднього захисту МКР на кафедрі, відгук наукового керівника, відгук рецензента, анотації до МКР українською та іноземною мовами, відповідність оформлення МКР діючим вимогам.

7 Виконання етапів графічної та розрахункової документації МКР контролюється науковим керівником згідно зі встановленими термінами. Захист МКР відбувається на засіданні Державної екзаменаційної комісії, затвердженою наказом ректора.

8 Вимоги до оформлення МКР викладені в ДСТУ 3008-95 та Положення про МКР ВНТУ 2021.

Технічне завдання до виконання прийняв _____ Гирін Д. Ю.

ДОДАТОК Б

Схема аудіопроецесора

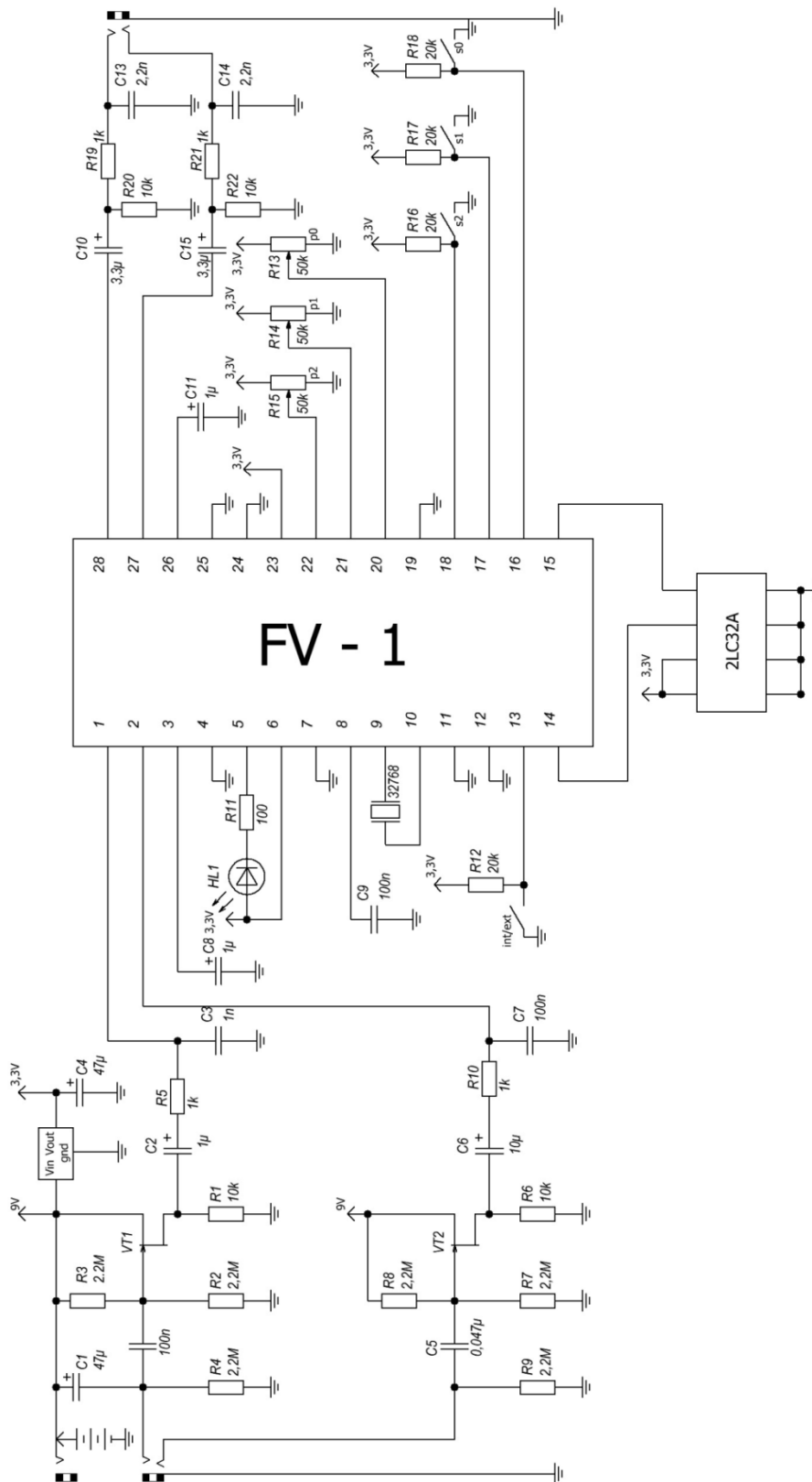


Рисунок Б.1 — Схема аудіопроецесора

ДОДАТОК В

Схема пристрою гітарного ефекту з дистанційним керуванням

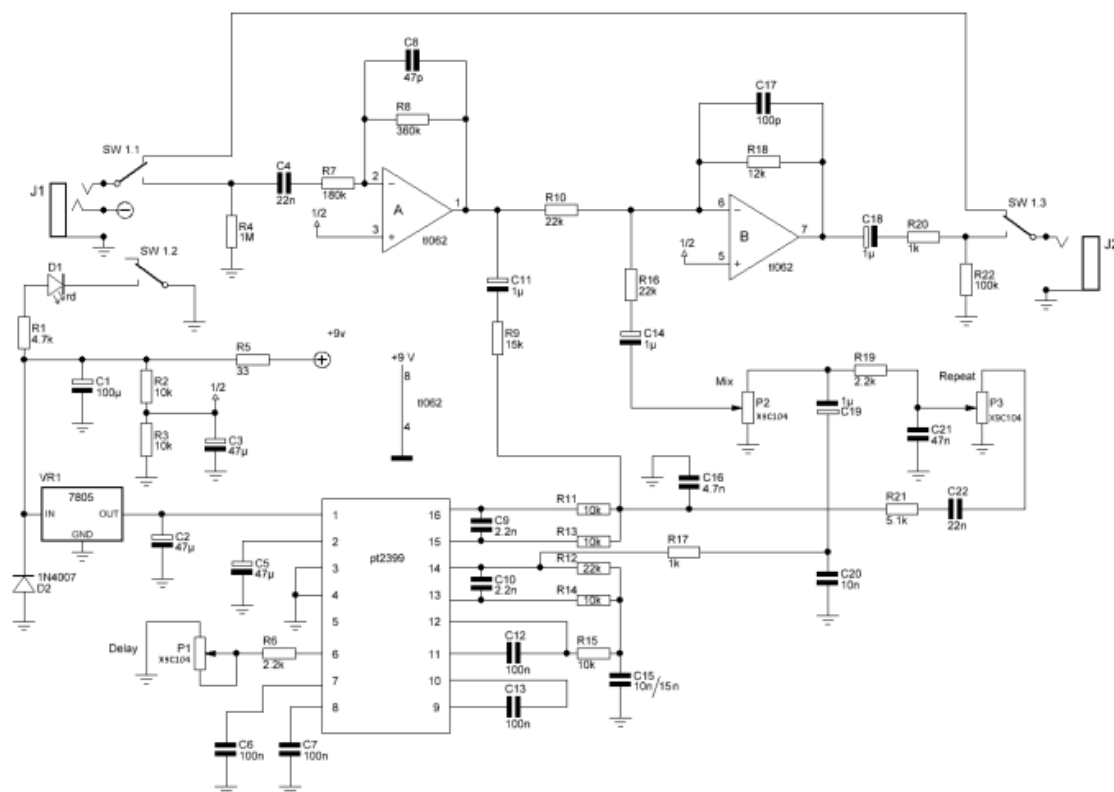


Рисунок В.1 — Схема пристрою гітарного ефекту з дистанційним керуванням

Додаток Г

Лістинг iOS програми для керування мікроконтролером

```
//
// BluetoothManager.swift
// BluetoothPedalBoard
//
//
// [CBUUID.init(string: "BT05")]
import UIKit
import CoreBluetooth

class BluetoothManager: NSObject, CBPeripheralDelegate, ObservableObject,
CBCentralManagerDelegate {

    private var centralManager: CBCentralManager!

    @Published private(set) var state: Int = CBManagerState.unknown.rawValue
    тд.
    @Published private(set) var ready: Bool = false
    @Published private(set) var deviceList: [CBPeripheral] = []
    @Published private(set) var connectedList: [CBPeripheral] = []

    override init() {
        super.init()
        self.centralManager = CBCentralManager(delegate: self, queue: nil, options: nil)
    }

    // сканирование доступных устройств
    func scan() -> Void {
        if(self.centralManager.isScanning) {
            self.stop()
        }
    }
}
```

```
}
```

```
deviceList.removeAll()
```

```
self.centralManager.scanForPeripherals(withServices: nil, options:
[CBCentralManagerScanOptionAllowDuplicatesKey: false])
```

```
}
```

```
// остановка сканирования
```

```
func stop() -> Void {
```

```
centralManager.stopScan()
```

```
}
```

```
func connect(peripheral: CBPeripheral, options: [String: Any]?) -> Void {
```

```
if(!self.connectedList.contains(peripheral)) {
```

```
print("Connecting to " + (peripheral.name ?? peripheral.identifier.uuidString))
```

```
centralManager.connect(peripheral, options: options)
```

```
}
```

```
}
```

```
// отправка данных на подключённое устройство
```

```
func write(peripheral: CBPeripheral, value: String) {
```

```
let data = value.data(using: String.Encoding.utf8)
```

```
65
```

```
if let services: [CBService] = peripheral.services {
```

```
for service: CBService in services {
```

```
if let characteristics: [CBCharacteristic] = service.characteristics {
```

```
for characteristic: CBCharacteristic in characteristics {
```

```
peripheral.writeValue(data!, for: characteristic, type: .withoutResponse) //
```

```
отправляем
```



```

}
}
}
}
}

```

```

func centralManager(_ central: CBCentralManager, didDiscover peripheral:
CBPeripheral, advertisementData: [String : Any], rssi

```

```

RSSI: NSNumber) {

```

```

    deviceList.append(peripheral)

```

```

}

```

```

// didConnect

```

```

func centralManager(_ central: CBCentralManager, didConnect peripheral:
CBPeripheral) {

```

```

    print("Connected to " + (peripheral.name ?? peripheral.identifier.uuidString))

```

```

    peripheral.delegate = self

```

```

    peripheral.discoverServices([CBUUID.init(string: "FFE0")]);

```

```

    self.connectedList.append(peripheral)

```

```

}

```

```

// didDiscoverSrvices

```

```

func peripheral(_ peripheral: CBPeripheral, didDiscoverServices error: Error?) {

```

```

    (error != nil) ? print(error as Any) : peripheral.services?.forEach { service in

```

```

        peripheral.discoverCharacteristics([CBUUID.init(string: "FFE1")], for: service)

```

```

    }

```

```

}

```

```

// didDiscoverCharacteristics

```

```

func peripheral(_ peripheral: CBPeripheral, didDiscoverCharacteristicsFor
service: CBService, error: Error?) {

    (error != nil) ? print(error as Any) : service.characteristics?.forEach {
characteristic in

    if(characteristic.uuid == CBUUID(string: "FFE1")) {

        //peripheral.setNotifyValue(true, for: characteristic) // Subscribe to retrieve from
device

        }

        }

        }

func disconnect(peripheral: CBPeripheral) -> Void {

    print("Disconnecting from " + (peripheral.name ??
peripheral.identifier.uuidString))

    centralManager.cancelPeripheralConnection(peripheral)

    }

// didDisconnect

func centralManager(_ central: CBCentralManager, didDisconnectPeripheral
peripheral: CBPeripheral, error: Error?) {

    print("Disconnected from " + (peripheral.name ??
peripheral.identifier.uuidString))

    if let index = connectedList.firstIndex(of: peripheral) {
connectedList.remove(at: index)

    }

    }

/*

// didUpdateCharacteristic

func peripheral(_ peripheral: CBPeripheral, didUpdateNotificationStateFor
characteristic: CBCharacteristic, error: Error?) {

```

66

```

//print(characteristic.value!)
if let characteristicValue = characteristic.value {
// Parse to Integer
print(characteristicValue)
let datastring = NSInteger(characteristicValue.base64EncodedString())
if let datastring = datastring {
print(datastring as NSInteger)
}
}
}
*/

// срабатывает когда состояние Bluetooth-модуля на устройстве изменено -
ВКЛ, ВЫКЛ И ТД.
func centralManagerDidUpdateState(_ central: CBCentralManager) {
switch central.state {
case .unknown:
print("Bluetooth manager state is Unknown")
self.ready = false
case .resetting:
print("Bluetooth manager state is Resetting")
self.ready = false
case .unsupported:
print("Bluetooth manager state is Unsupported")
self.ready = false
case .unauthorized:
print("Bluetooth manager state is Unauthorized")
self.ready = false
}
}
}

```

```

case .poweredOff:
print("Bluetooth manager state is Powered Off")
self.ready = false
case .poweredOn:
print("Bluetooth manager state is Powered On")
self.ready = true
@unknown default:
print("Bluetooth manager state is Unknown Default")
self.ready = false
}

self.state = central.state.rawValue
}
//
// MainView.swift
// BluetoothPedalBoard
//
//
import SwiftUI
import CoreBluetooth
struct MainView: View {

@EnvironmentObject var bluetoothManager: BluetoothManager

var body: some View {
Loader {
NavigationView {
VStack(alignment: .center, spacing: 10) {

```

```

HStack(alignment: .center) {
  Button(action: self.bluetoothManager.scan) { // кнопка для начала
сканирования
  Text("Scan")
  .fontWeight(.bold)
  .padding(10)
  .border(Color.blue)
  }
  .disabled(!self.bluetoothManager.ready)
  .padding(.top, 10)
  .padding(15)
  NavLink(destination: PedalView()) {
  Text("Control")
68
  .fontWeight(.bold)
  .padding(10)
  .border(Color.blue)
  }.padding(10)
  .disabled(self.bluetoothManager.connectedList.isEmpty)
  .padding(.top, 10)
  .padding(15)
  }.navigationBarTitle(Text("Bluetooth Pedal Board"), displayMode:
.inline)
  List(self.bluetoothManager.deviceList, id: \.identifier) { // список
найденных устройств
  device in Button(action: {
  self.bluetoothManager.connect(peripheral: device, options: nil)
  }) {

```

```

Text((device as CBPeripheral).name ?? (device as
CBPeripheral).identifier.uuidString)
}
.padding(10)
.disabled(self.bluetoothManager.connectedList.contains(device))
}
.disabled(!self.bluetoothManager.ready)
}
.onAppear {
}.onDisappear {
}
}
}
}
}
}
//
// PedalView.swift
// BluetoothPedalBoard
//
//
import SwiftUI
import CoreBluetooth
// класс для работы со слайдером
class Parameter {
var id: String
var name: String
var value: Double = 0.0
var oldValue: Double = 0.0

```

```

init(id: String, name: String) {
    self.id = id
    self.name = name
}
}

struct ParameterView: View {
    var device: CBPeripheral // устройство для взаимодействия
    var bluetoothManager: BluetoothManager // Bluetooth-менеджер

    var parameter: Parameter
    var body: some View {
70
        HStack {
            Text(parameter.name).frame(width: 70, height: 50, alignment: .leading)
            Slider(value: Binding(
                get: { // получение значения слайдера
                    self.parameter.value
                },
                set: {(newValue) in // установка значения слайдера
                    if(self.parameter.value != newValue) {

                        self.parameter.value = newValue
                    }
                }
            ), in: 0...100, step: 1.0, onEditingChanged: { editing in
                if(!editing) {

```

```

let action = self.parameter.value > self.parameter.oldValue ? "+" : "-"
// если изменили на больше то отсылаем + если на меньше то -
let value = self.parameter.value - self.parameter.oldValue //
высчитываем на сколько изменилось значение

print(self.parameter.oldValue)
print(self.parameter.value)
print(value)

self.bluetoothManager.write(peripheral: self.device, value:
"\(self.parameter.id)\(action)\(abs(Int(value)))")

self.parameter.oldValue = self.parameter.value
}
}).padding(10)
}

}
71
}
struct PedalView: View {

@EnvironmentObject var bluetoothManager: BluetoothManager

var mix: Parameter = Parameter(id: "m", name: "Mix")
var delay: Parameter = Parameter(id: "d", name: "Delay")
var repeats: Parameter = Parameter(id: "r", name: "Repeats")

```


ДОДАТОК Д

ПРОТОКОЛ ПЕРЕВІРКИ НАВЧАЛЬНОЇ (КВАЛІФІКАЦІЙНОЇ) РОБОТИ

Назва роботи: Засоби апаратно-програмного опрацювання сигналів в аудіопроекторах з дистанційним керуванням

Тип роботи: магістерська кваліфікаційна робота

(кваліфікаційна роботи, курсовий проект (робота), реферат, аналітичний огляд, інше (вказати))

Підрозділ кафедра обчислювальної техніки, ІКІ-20м

(кафедра, факультет (інститут), навчальна група)

Науковий керівник Крупельницький Л.В., доц. кафедри ОТ

(прізвище, ініціали, посада)

Показники звіту подібності

Plagiat.pl (StrikePlagiarism)		Unicheck	
КП1		Оригінальність	87,5
КП2			
ривого/Білі знаки	/	Схожість	12,5

Аналіз звіту подібності (відмінити подібне)

- Запозичення, виявлені у роботі, оформлені коректно і не містять ознак плагіату.
- Виявлені у роботі запозичення не мають ознак плагіату, але їх надмірна кількість викликає сумніви щодо цінності і відсутності самостійності її автора. Робот направити на доопрацювання.
- Виявлені у роботі запозичення є недобросовісними і мають ознаки плагіату та/або в ній містяться навмисні спотворення тексту, що вказують на спроби приховування недобросовісних запозичень. Заявляю, що ознайомлений(-на) з повним звітом подібності, який був згенерований Системою щодо роботи (додається)

Автор _____

(підпис)

Гирін Д.Ю.

(прізвище, ініціали)

Опис прийнятого рішення

Ступінь оригінальності роботи відповідає вимогам, що висуваються до МКР

Особа, відповідальна за перевірку Захарченко С.М.

(підпис)

(прізвище, ініціали)

Експерт _____

(за потреби) (підпис)

(прізвище, ініціали)