Міністерство освіти і науки України
Тернопільський національний технічний університет імені Івана Пулюя

Факультет комп'ютерно-інформаційних систем і програмної інженерії
(повна назва факультету)
Кафедра комп'ютерних систем та мереж
(повна назва кафедри)

КВАЛІФІКАЦІЙНА РОБОТА

на здобуття освітнього ступеня

бакалавр
(назва освітнього ступеня)

на тему: *Комп'ютерна система для налаштування струнних музичних інструментів*

Виконав: студент 4 курсу, групи СІ-42
спеціальності 123 «Комп'ютерна інженерія»
(шифр і назва спеціальності)

		Яцук І.В.
	(підпис)	(прізвище та ініціали)
Керівник		*Тиш Є.В.*
	(підпис)	(прізвище та ініціали)
Нормоконтроль		*Луцик Н.С.*
	(підпис)	(прізвище та ініціали)
Завідувач кафедри		Осухівська Г.М.
	(підпис)	(прізвище та ініціали)
Рецензент		*Цуприк Г.Б.*
	(підпис)	(прізвище та ініціали)

Тернопіль
2026

Міністерство освіти і науки України
Тернопільський національний технічний університет імені Івана Пулюя

Факультет *комп'ютерно-інформаційних систем і програмної інженерії*
(повна назва факультету)

Кафедра *комп'ютерних систем та мереж*
(повна назва кафедри)

ЗАТВЕРДЖУЮ
Завідувач кафедри
_____ *Осухівська Г.М.*
(підпис) (прізвище та ініціали)
«25» квітня 2026 р.

З А В Д А Н Н Я

НА КВАЛІФІКАЦІЙНУ РОБОТУ

на здобуття освітнього ступеня *бакалавр*
(назва освітнього ступеня)

за спеціальністю *123 «Комп'ютерна інженерія»*
(шифр і назва спеціальності)

студента *Ящука Івана Володимировича*
(прізвище, ім'я, по батькові)

1. Тема роботи *Комп'ютерна система для налаштування струнних музичних інструментів*

Керівник роботи *Тиш Євгенія Володимирівна, доцент кафедри КС*
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

Затверджені наказом ректора від *« 24 » квітня 2026 року № 4/9-188*
2. Термін подання студентом завершеної роботи *18 червня 2026 р.*
3. Вихідні дані до роботи *Технічне завдання*

4. Зміст роботи (перелік питань, які потрібно розробити)
Вступ. 1. Аналіз технічного завдання
2. Проєктна частина
3. Практична частина
4. Безпека життєдіяльності, основи охорони праці.
Висновки
5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень, слайдів)
1. Структурна схема комп'ютерної системи.
2. Блок-схема алгоритму функціонування системи.
3. Схема електрична принципова.
4. Схема електрична функціональна.

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Безпека життєдіяльності, основи охорони праці			

7. Дата видачі завдання *25.04.2026 р.*

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів роботи	Термін виконання етапів роботи	Примітка
1.	*Розробка технічного завдання*	*26.01 – 02.02*	
2.	*Робота над першим розділом «Аналіз технічного завдання»*	*03.02 – 15.02*	
3.	*Робота над другим розділом «ПРОЄКТНА ЧАСТИНА»*	*20.04 – 25.04*	
4.	*Робота над третім розділом «ПРАКТИЧНА ЧАСТИНА»*	*26.04 – 08.05*	
5.	*Безпека життєдіяльності, основи охорони праці*	*08.05 – 15.05*	
6.	*Оформлення пояснювальної записки і графічного матеріалу*	*15.05 – 7.06*	
7.	*Перевірка на академічний плагіат, перевірка керівником та консультантами*	*8.06 – 14.06*	
8.	*Попередній захист кваліфікаційної роботи бакалавра*	*15.06 – 21.06*	
9.	*Захист кваліфікаційної роботи бакалавра*	*25.06*	

Студент _____ *Ящук І.В.*
(підпис) (прізвище та ініціали)

Керівник роботи _____ *Тиш Є.В.*
(підпис) (прізвище та ініціали)

АНОТАЦІЯ

Ящук І.В. Комп’ютерна система для налаштування струнних музичних інструментів: робота на здобуття кваліфікаційного ступеня бакалавра: спец. 123 — комп’ютерна інженерія. Тернопіль: Тернопільський національний технічний університет імені Івана Пулюя, 2026.

Ключові слова: налаштування струнних інструментів, автокореляційна функція, мікроконтролер, кроковий двигун, цифрова обробка сигналів, MEMS-мікрофон, OLED-дисплей.

Кваліфікаційна робота присвячена розробці та реалізації автоматизованого апаратно-програмного комплексу для налаштування струнних інструментів. У системі використано мікроконтролер STM32, цифровий MEMS-мікрофон та кроковий двигун із силовим драйвером. Реалізований алгоритм обробки сигналів методом автокореляції та параболічної інтерполяції забезпечує високу точність визначення частоти звуку (до 0.1–0.2 Гц). Керування виконавчим механізмом здійснюється за допомогою ПІ-регулятора, який плавно підлаштовує натяг струни до еталону. Завдяки професійно спланованим захисним сценаріям та зоні нечутливості, ризик пошкодження чи обриву струн зведений до нуля. Взаємодія з користувачем та відображення хронології подій у реальному часі реалізовані через інкрементальний енкодер та графічний OLED-дисплей. Розроблений комплекс є мобільним, завадостійким та безпечним інженерним пристроєм.

ANNOTATION

Yashchuk I.V. Computer System for Tuning String Musical Instruments: Bachelor's Graduation Thesis: speciality 123 — computer engineering. Ternopil: Ternopil Ivan Puluj National Technical University, 2026.

Keywords: string instrument tuning, autocorrelation function, microcontroller, stepper motor, digital signal processing, MEMS microphone, OLED display.

The qualification work is devoted to the development and implementation of an automated hardware-software complex for tuning string instruments. The system utilizes an STM32 microcontroller, a digital MEMS microphone, and a stepper motor with a power driver. The implemented signal processing algorithm, based on the autocorrelation method and parabolic interpolation, ensures high accuracy in determining the sound frequency (up to 0.1–0.2 Hz). Actuator control is performed using a PI controller that smoothly adjusts the string tension to the reference value. Due to professionally designed protection scenarios and a deadband zone, the risk of string damage or breakage is reduced to zero. User interaction and real-time event visualization are implemented via an incremental encoder and a graphical OLED display. The developed complex is a mobile, interference-resistant, and safe engineering device.

ЗМІСТ

ВСТУП……9

РОЗДІЛ 1 АНАЛІЗ ТЕХНІЧНОГО ЗАВДАННЯ 10

1.1 Аналіз вимог до автоматизованих систем налаштування струнних інструментів.....10

1.2 Огляд існуючих методів та приладів для налаштування гітар 12

1.3 Аналіз алгоритмів цифрової обробки аудіосигналів для ідентифікації частоти ……14

РОЗДІЛ 2 ПРОЄКТНА ЧАСТИНА.....17

2.1 Розробка узагальненої структурної схеми системи 17

2.2 Обґрунтування вибору апаратного забезпечення.....19

2.2.1 Вибір обчислювального ядра системи.....19

2.2.2 Вибір підсистеми зчитування цифрового звуку 20

2.2.3 Вибір виконавчого механізму та драйвера керування двигуном 21

2.2.4 Вибір підсистеми живлення та індикації 23

РОЗДІЛ 3 ПРАКТИЧНА ЧАСТИНА.....31

3.1 Апаратна реалізація базового та виконавчого блоків комплексу.....31

РОЗДІЛ 4 БЕЗПЕКА ЖИТТЄДІЯЛЬНОСТІ, ОСНОВИ ОХОРОНИ ПРАЦІ..49

4.1 Загальні заходи щодо захисту від ураження електричним струмом.....49

4.2 Оцінка розробленої конструкції установки щодо умов безпечної експлуатації.....51

ВИСНОВКИ.....54

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....55

Додаток А Технічне завдання

Додаток Б Перелік елементів

Додаток В Лістинг програмного коду автоматичного тюнера

СПИСОК СКОРОЧЕНЬ

ABS – Acrylonitrile Butadiene Styrene

ADC – Analog-to-Digital Converter

ARM – Advanced RISC Machine

CPU – Central Processing Unit

DC-DC – Direct Current to Direct Current

DMA – Direct Memory Access

DSP – Digital Signal Processing

FPU – Floating-Point Unit

GPIO – General-Purpose Input/Output

HAL – Hardware Abstraction Layer

I2C – Inter-Integrated Circuit

I2S – Inter-IC Sound

IDE – Integrated Development Environment

MEMS – Micro-Electro-Mechanical Systems

NAS – Network Attached Storage

OLED – Organic Light-Emitting Diode

ВСТУП

Налаштування струнних музичних інструментів є обов'язковим і регулярним процесом, оскільки натяг струн постійно змінюється під впливом механічних навантажень, перепадів температури та вологості навколишнього середовища [26]. Традиційне налаштування на слух вимагає від музиканта високого рівня кваліфікації та розвиненого музичного слуху. Сучасні електронні тюнери значно спрощують цей процес, виконуючи роль інформаційного дисплея, який вказує на відхилення частоти струни від еталонного значення. Проте, вони залишають процес фізичного обертання кілка інструмента на користувача, що розриває контур зворотного зв'язку і робить процес ітеративним та залежним від людського фактора.

Особливо гостро ця проблема постає під час налаштування електрогітар з тремоло-системами або при необхідності швидкого переналаштування інструмента під час виступів. Розробка комп'ютерної системи, яка здатна не лише розпізнавати частоту звуку з високою точністю за допомогою методів цифрової обробки сигналів, але й автоматично виконувати механічну дію з коригування натягу струни за допомогою мікроконтролера та електродвигуна, є актуальним інженерним завданням. Створення такого апаратно-програмного комплексу дозволяє повністю автоматизувати рутинний процес, мінімізувати час налаштування та виключити людські помилки.

Мета роботи - розробка комп'ютерної системи для автоматизованого налаштування струнних музичних інструментів, що базується на адаптивній ідентифікації частоти звукового сигналу та автоматичному керуванні виконавчим механізмом натягу струни.

РОЗДІЛ 1 АНАЛІЗ ТЕХНІЧНОГО ЗАВДАННЯ

1.1 Аналіз вимог до автоматизованих систем налаштування струнних інструментів

Процес налаштування струнного музичного інструмента є критично важливою процедурою, що передує будь-якій грі. Фізична природа струни така, що її натяг з часом слабшає під дією механічних навантажень, зміни вологості та температури. Особливо гостро ця проблема проявляється при грі на електрогітарі з використанням підсилювальної апаратури. Навіть мінімальне відхилення частоти струни від еталону при грі через комбопідсилювач із застосуванням ефектів перевантаження (distortion або overdrive) створює сильний дисонанс - так зване акустичне «биття» частот (рис. 1.1), яке критично псує загальне звучання інструмента.

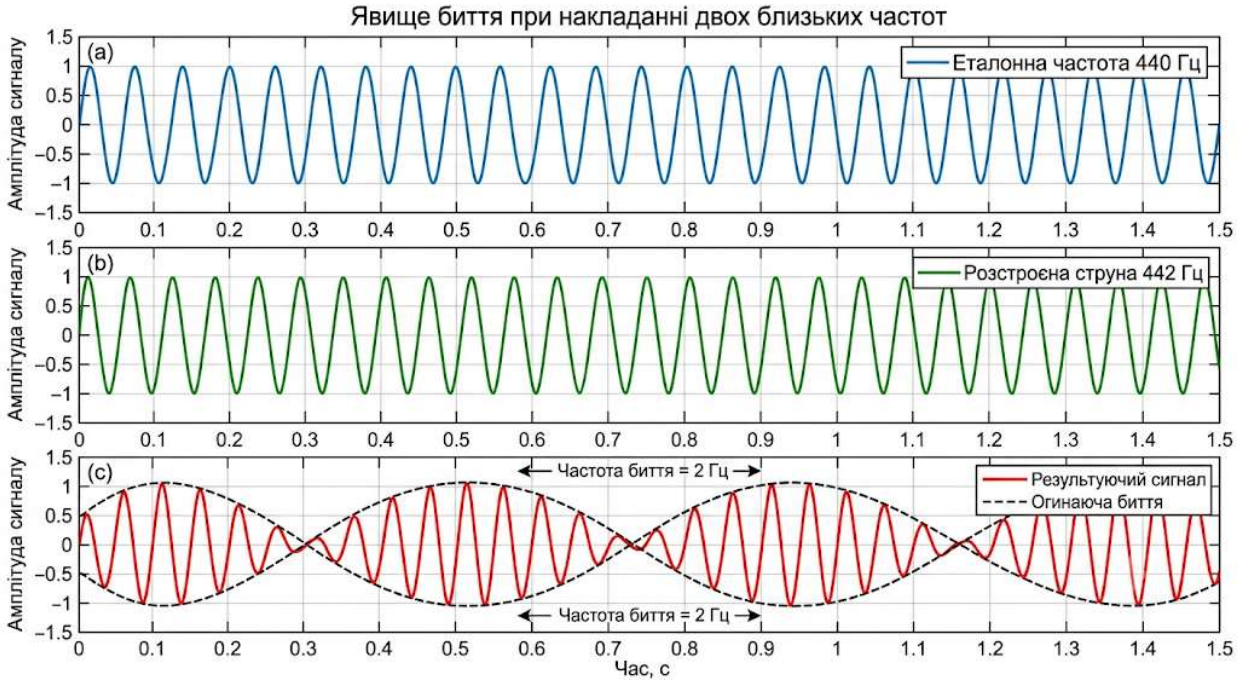

Рисунок 1.1 – Графік биття двох близьких частот

					КС КРБ 123.201.00.00 ПЗ		
Змн.	Арк.	№ докум.	Підпис	Дата			
Розроб.		Ящук І.В.			Аналіз технічого завдання	Літ.	Арк. 10 / Аркушів
Перевір.		Тиш Є.В.					
Реценз.		Цуприк Г.Б.				ТНТУ, каф. КС, гр. СІ-42	
Н. Контр.		Луцик Н.С.					
Затверд.		Осухівська Г.М.					

Для автоматизації цього рутинного процесу та виключення людського фактору, розроблюваний апаратно-програмний комплекс повинен відповідати низці чітких функціональних та експлуатаційних вимог [1].

Функціональні вимоги до системи:

– Система повинна безперервно захоплювати акустичні коливання струни за допомогою мікрофона, мінімізуючи вплив сторонніх механічних та електромагнітних шумів.

– Обчислювальне ядро має в реальному часі обробляти цифровий потік аудіоданих та визначати основну частоту коливання (fundamental frequency) обраної струни.

– Програмне забезпечення повинно порівнювати поточну частоту з еталонним значенням для конкретної ноти стандартного гітарного строю (наприклад, 329.63 Гц для першої струни E4) та визначати напрямок і величину відхилення.

– На основі обчисленої похибки система має генерувати керуючі сигнали для драйвера двигуна, який здійснюватиме фізичне обертання кілка інструмента до моменту збігу частот.

Експлуатаційні та апаратні вимоги:

– Допустима похибка системи не повинна перевищувати ±1–2 центи (соті частки півтону), що відповідає стандартам професійних студійних тюнерів.

– Час реакції системи має бути мінімальним. Алгоритм керування двигуном повинен бути адаптивним - швидкість обертання має пропорційно зменшуватися при наближенні до еталонної частоти, щоб уникнути обриву струни через інерційність ротора.

– Оскільки система поєднує чутливу цифрову електроніку та потужний електромеханічний вузол, архітектура живлення має базуватися на стаціонарному мережевому джерелі з апаратним розділенням силових та логічних ліній напруги.

1.2 Огляд існуючих методів та приладів для налаштування гітар

Історично склалося так, що базовим методом налаштування струнних інструментів був акустичний підхід, що спирається на музичний слух виконавця та використання еталонного джерела звуку, наприклад, камертона. Музикант налаштовує одну струну за еталоном, а всі інші - відносно неї, використовуючи метод флажолетів або затискання струн на відповідних ладах [26]. Основною проблемою цього методу є висока суб'єктивність та вразливість до зовнішніх акустичних шумів.

З розвитком мікроелектроніки широкого поширення набули електронні тюнери, які перетворили процес налаштування з суб'єктивного на об'єктивний. Сучасний ринок пропонує кілька основних формфакторів таких пристроїв [18]. Першою категорією є педальні тюнери (рис. 1.2). Вони підключаються безпосередньо в сигнальний ланцюг електрогітари за допомогою інструментального кабелю. Оскільки пристрій отримує чистий аналоговий сигнал прямо зі звукознімачів, він не реагує на сторонні шуми в приміщенні. Це робить їх ідеальними для використання на сцені під час гри через комбопідсилювачі. Проте їхнім головним недоліком є необхідність розриву сигнального ланцюга та неможливість використання з класичними акустичними інструментами без звукознімачів.

Рисунок 1.2 – Зовнішній вигляд типового педального гітарного тюнера

Альтернативним і найбільш масовим рішенням є тюнери-прищіпки (clip-on tuners). Вони кріпляться на перо грифа гітари і використовують п'єзоелектричний датчик для зчитування механічних вібрацій корпусу інструмента, а не звукових хвиль з повітря. Це дозволяє налаштовувати інструмент навіть у шумному середовищі. З іншого боку, найдешевші та найдоступніші рішення - це програмні тюнери у вигляді додатків для смартфонів. Вони використовують вбудований MEMS-мікрофон телефону. Незважаючи на доступність, додатки мають найнижчу завадостійкість, оскільки вловлюють усі кімнатні реверберації, сторонні розмови та шуми, що значно ускладнює роботу математичних алгоритмів розпізнавання.

Усі вищезгадані прилади є виключно інформаційними: вони обчислюють частоту і показують похибку на дисплеї, залишаючи механічну роботу користувачеві. Однак останніми роками на ринку з'явилися автоматизовані (роботизовані) системи налаштування. Їх можна розділити на інтегровані та портативні.

Інтегровані системи (наприклад, система TronicalTune, яка встановлюється на гітари Gibson) монтуються безпосередньо на тильну сторону пера грифа. Вони містять мініатюрні сервоприводи, вбудовані в кожен окремий кілок. Така система дозволяє переналаштувати весь інструмент за кілька секунд простим натисканням кнопки. Суттєвими недоліками інтегрованих рішень є їхня висока вартість, значне збільшення ваги пера грифа (що порушує баланс інструмента) та складність встановлення на гітари інших виробників без незворотної модифікації деревини.

Більш компромісним рішенням є портативні автоматичні тюнери, яскравим представником яких є пристрій Roadie Tuner. Це ручний гаджет із вбудованим мікродвигуном, який користувач по черзі прикладає до кожного кілка. Пристрій зчитує вібрацію інструмента та самостійно обертає вал двигуна до досягнення потрібної частоти.

Концепція ручного пристрою є найбільш вдалою з точки зору

універсальності та вартості. Проте комерційні закриті рішення не дозволяють користувачеві втручатися в алгоритми обробки сигналу або налаштовувати параметри зворотного зв'язку (наприклад, змінювати швидкість реакції ПІ-регулятора двигуна під різні типи струн). Це обґрунтовує доцільність розробки власного відкритого апаратно-програмного комплексу на базі сучасних мікроконтролерів, який поєднає гнучкість налаштування цифрової обробки сигналів із надійністю роботи ручного виконавчого механізму.

1.3 Аналіз алгоритмів цифрової обробки аудіосигналів для ідентифікації частоти

Головним завданням обчислювального ядра розроблюваної системи є точне визначення основної частоти звучання струни (pitch detection) із сирого потоку цифрових аудіоданих, що надходять від мікрофона. Складність цього завдання полягає у фізичній природі звуку струнного інструмента. Акустична хвиля, що генерується гітарою, не є чистою синусоїдою; це складний квазіперіодичний сигнал, який складається з основної частоти (тоніки) та великої кількості вищих гармонік (обертонів), а також атаки - початкового шумового сплеску в момент удару медіатором. Іноді амплітуда другої або третьої гармоніки може фізично перевищувати амплітуду основної частоти, що створює значні труднощі для математичного аналізу.

Для вирішення цього завдання в цифровій обробці сигналів (DSP) найчастіше застосовують два принципово різні підходи: аналіз у частотній області (на базі перетворення Фур'є) та аналіз у часовій області (на базі автокореляції).

Швидке перетворення Фур'є (ШПФ) є класичним алгоритмом переведення сигналу з часової області в частотну [19]. Цей математичний апарат розкладає складну звукову хвилю на набір простих синусоїд різної частоти та амплітуди, формуючи спектр сигналу. Мікроконтролери з

апаратним модулем плаваючої крапки (FPU), такі як STM32F4, здатні виконувати ШПФ надзвичайно швидко. Однак для завдання точного налаштування гітари цей метод має суттєвий недолік - частотну роздільну здатність. Роздільна здатність ШПФ безпосередньо залежить від частоти дискретизації та розміру буфера вибірок. Щоб отримати точність в 1 Гц (що є критичним для басових струн, де частота ноти Мі становить близько 82.4 Гц), системі необхідно накопичити дуже великий масив даних. Це призводить до значних витрат оперативної пам'яті (RAM) мікроконтролера та збільшує затримку системи, роблячи процес налаштування повільним і переривчастим. Крім того, алгоритм ШПФ часто страждає від "октавної помилки", коли програма помилково розпізнає найгучнішу другу гармоніку замість основної частоти.

Альтернативним і більш ефективним підходом для створення тюнерів є метод автокореляції, який працює безпосередньо в часовій області [4,8]. Суть алгоритму полягає в пошуку періодичності шляхом математичного накладання фрагмента сигналу на його власну копію з поступовим зсувом у часі. Коли зсув дорівнює періоду коливання струни, функція автокореляції видає яскраво виражений глобальний максимум (рис. 1.3). Знаючи період хвилі в мілісекундах, основна частота обчислюється шляхом простого ділення одиниці на цей період.

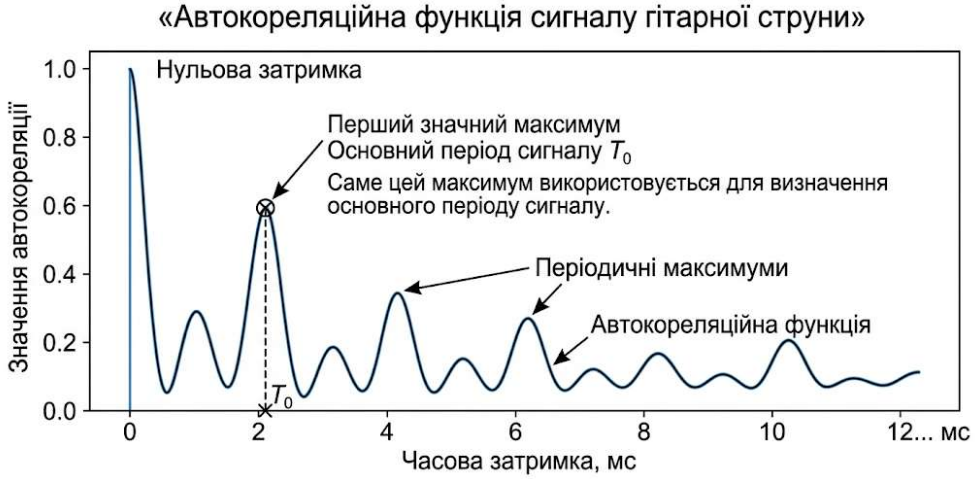

Рисунок 1.3 – Графік автокореляційної функції гітарного звуку

Перевагою методу автокореляції є висока точність визначення основної частоти навіть при малому розмірі буфера та низьких частотах, що ідеально підходить для мікроконтролерів. Автокореляція природним чином ігнорує фазові зсуви вищих гармонік, виділяючи саме фундаментальний період звучання інструмента. Хоча "наївна" реалізація автокореляції вимагає значних обчислювальних ресурсів, існують оптимізовані варіації цього методу. Наприклад, алгоритм YIN або функція середньої абсолютної різниці (AMDF), які замінюють операції множення на віднімання, значно прискорюючи обробку на процесорах архітектури ARM Cortex-M [22].

Для підвищення точності розпізнавання до часток герца, максимум автокореляційної функції додатково уточнюється за допомогою параболічної інтерполяції. Це дозволяє обчислити "істинний" пік, який знаходиться між двома сусідніми цифровими вибірками. Зважаючи на високу завадостійкість, здатність точно працювати з низькочастотними сигналами та менші вимоги до обсягу пам'яті для досягнення високої роздільної здатності, метод автокореляції є найбільш доцільним вибором для реалізації в розроблюваному апаратно-програмному комплексі.

РОЗДІЛ 2 ПРОЄКТНА ЧАСТИНА

2.1 Розробка узагальненої структурної схеми системи

Розробка будь-якого апаратно-програмного комплексу починається з побудови його узагальненої структурної схеми [3]. Вона дозволяє визначити основні функціональні блоки системи, їхнє призначення та логіку взаємодії між собою на апаратному рівні. Проєктована комп'ютерна система для налаштування струнних музичних інструментів концептуально являє собою замкнену систему автоматичного керування (САК) зі зворотним зв'язком. Об'єктом керування в цій системі виступає натяг струни, параметри якого безпосередньо впливають на частоту акустичного коливання.

Архітектурно систему можна розділити на два фізичні модулі, з'єднані між собою гнучким кабелем: базову станцію (керуючий блок) та робочий інструмент (виконавчий блок). Такий розподіл зумовлений ергономічними вимогами, оскільки оператору зручніше маніпулювати легким виконавчим механізмом, не обтяженим обчислювальною електронікою та елементами живлення (рис. 2.1).

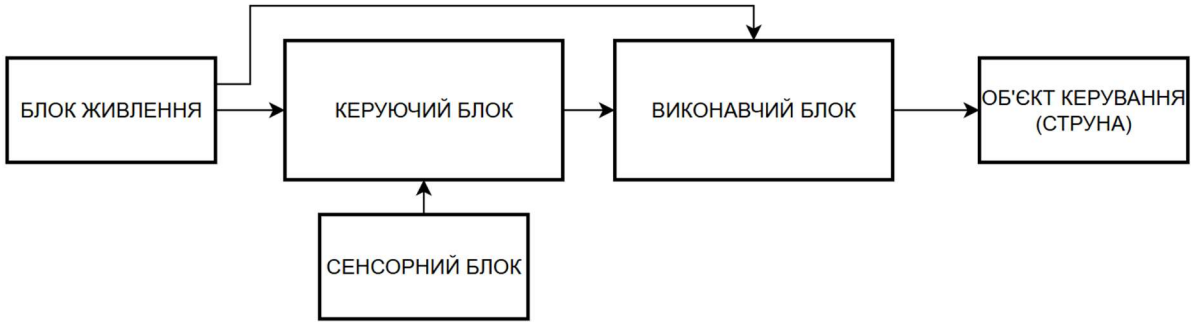

Рисунок 2.1 – Структурна схема системи

Центральним вузлом базової станції є обчислювальне ядро - високопродуктивний мікроконтролер. Він виконує роль головного координатора: приймає дані від сенсорів, обробляє їх за допомогою математичних алгоритмів, оновлює інформацію на дисплеї та генерує керуючі імпульси. Для взаємодії з користувачем передбачено блок індикації та керування. Він складається з компактного цифрового дисплея, який підключається до мікроконтролера через послідовну шину передачі даних, та роторного енкодера. Енкодер виконує функцію універсального пристрою введення, за допомогою якого користувач обирає режим роботи, тип інструмента та конкретну струну (ноту), що підлягає налаштуванню.

Сенсорний блок системи реалізовано у вигляді цифрового мікрофона, який для зменшення впливу механічних вібрацій від двигуна може бути закріплений безпосередньо на інструменті (наприклад, на макетній платі або спеціальній прищіпці). Мікрофон захоплює звукову хвилю і, минаючи етап вразливого аналогового передавання, відправляє оцифрований аудіопотік безпосередньо в оперативну пам'ять мікроконтролера по спеціалізованій цифровій аудіошині.

Виконавчий блок складається з двох ключових елементів: драйвера керування та крокового двигуна з редуктором. Драйвер розташовується на базовій станції і виконує роль силового посередника. Він приймає від мікроконтролера слабкострумові логічні сигнали напрямку обертання та крокових імпульсів (ШІМ), перетворюючи їх на потужні струмові імпульси, що подаються на обмотки двигуна. Сам кроковий двигун, обладнаний спеціальною насадкою для захоплення кілка гітари, є єдиним елементом, який фізично контактує з інструментом і здійснює механічну роботу з натягування або послаблення струни.

Окремої уваги заслуговує підсистема живлення. Оскільки кроковий двигун є потужним індуктивним навантаженням, що генерує стрибки струму,

живлення системи розділено на силову та логічну частини. Первинне джерело постійного струму безпосередньо живить драйвер двигуна, тоді як обчислювальне ядро та чутлива периферія отримують енергію через понижуючий DC-DC перетворювач, що забезпечує стабільну напругу та захищає мікроконтролер від раптових перезавантажень.

2.2 Обґрунтування вибору апаратного забезпечення

Етап вибору апаратної елементної бази є одним із найвідповідальніших кроків у процесі проєктування комп'ютерної системи. Від правильного підбору компонентів залежить не лише загальна продуктивність та стабільність роботи пристрою, але й можливість реалізації складних математичних алгоритмів цифрової обробки сигналів без критичних затримок. Основними критеріями вибору є достатня обчислювальна потужність для роботи зі звуком у реальному часі, наявність необхідних апаратних інтерфейсів зв'язку, завадостійкість та загальна економічна доцільність розробки.

2.2.1 Вибір обчислювального ядра системи

Серцем розроблюваного апаратно-програмного комплексу є мікроконтролер, який повинен паралельно та синхронно вирішувати три ресурсомісткі задачі: безперервне зчитування цифрового аудіопотоку високої розрядності, виконання інтенсивних математичних обчислень для ідентифікації частоти та генерування точних часових імпульсів для плавного керування потужним крововим двигуном.

З огляду на зазначені жорсткі вимоги, в якості обчислювального ядра системи було обрано 32-бітний мікроконтролер STM32F401CCU6, побудований на передовій архітектурі ARM Cortex-M4 [12]. Цей чип працює на тактовій частоті до 84 МГц і має вбудований апаратний співпроцесор

плаваючої крапки (FPU) разом із розширеним набором DSP-інструкцій. Така архітектура дозволяє виконувати складні математичні операції, зокрема множення з накопиченням при розрахунку автокореляційної функції, за один такт процесора, забезпечуючи миттєву реакцію системи на зміну натягу струни (рис. 2.2).

Критично важливою апаратною перевагою обраного мікроконтролера є наявність розвиненої системи контролерів прямого доступу до пам'яті (DMA) та повноцінної апаратної підтримки шини I2S (Inter-IC Sound). Ця особливість дозволяє мікроконтролеру приймати цифровий аудіопотік від мікрофона і циклічно записувати його безпосередньо в оперативну пам'ять у фоновому режимі. Завдяки цьому центральний процесор абсолютно звільняється від рутини побітового зчитування даних і може повністю присвятити свій машинний час розрахункам параметрів пропорційно-інтегрального регулятора та генерації керуючих сигналів для двигуна.

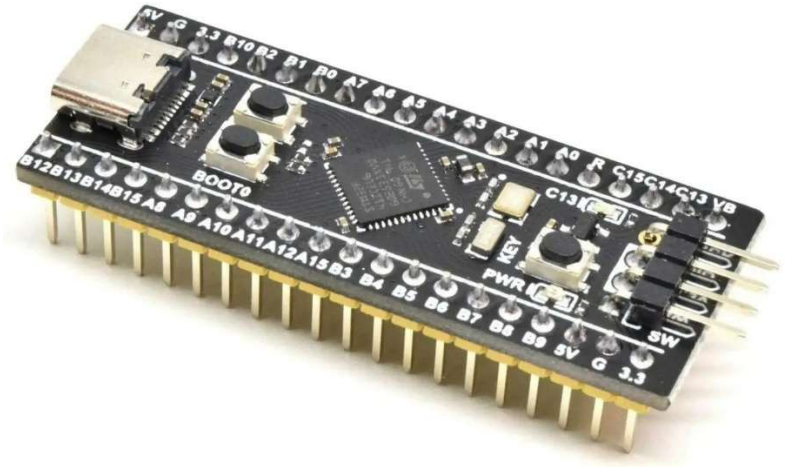

Рисунок 2.2 – Зовнішній вигляд STM32F411CEU6

2.2.2 Вибір підсистеми зчитування цифрового звуку

Першим етапом у ланцюгу обробки інформації розроблюваної системи є перетворення акустичних коливань струни в цифровий масив даних. Від якості та чистоти цього первинного сигналу безпосередньо залежить точність роботи математичних алгоритмів розпізнавання частоти.

При проєктуванні аудіосистем, що працюють у безпосередній близькості до потужних електромеханічних вузлів, виникає серйозна проблема електромагнітної сумісності. Класичні аналогові мікрофони видають на виході безперервну змінну напругу, яка передається по провідниках до аналого-цифрового перетворювача (АЦП) мікроконтролера. В умовах роботи драйвера крокового двигуна, який постійно генерує високочастотні широтно-імпульсні (ШІМ) сигнали з відносно високими струмами, аналоговий провідник від мікрофона починає діяти як антена. Він уловлює електромагнітні наведення від двигуна, що призводить до появи значного високочастотного шуму в корисному звуковому сигналі. Видалення такого "бруду" вимагає застосування складних апаратних RC-фільтрів та додаткових програмних ресурсів.

Для кардинального вирішення цієї проблеми та підвищення завадостійкості комплексу було прийнято рішення відмовитися від використання відкритого аналогового тракту на платі та застосувати повністю цифровий мікрофон INMP441 [21]. Це мініатюрний всеспрямований MEMS-мікрофон, який містить у своєму корпусі не лише акустичний сенсор, але й вбудований високоякісний АЦП та цифровий фільтр (рис. 2.3).

Рисунок 2.3 – Модуль цифрового MEMS-мікрофона INMP441

2.2.3 Вибір виконавчого механізму та драйвера керування двигуном

Фізичний процес налаштування струнного інструмента вимагає від

виконавчого механізму поєднання двох специфічних характеристик: високого крутного моменту при низьких обертах (оскільки кілкова механіка гітари має значний опір через натяг струни) та високої точності позиціонування. Використання звичайних колекторних двигунів постійного струму в такому завданні є недоцільним, оскільки вони не здатні забезпечити точну зупинку валу без застосування складних абсолютних енкодерів та дорогих редукторів. Класичні сервоприводи також не підходять через конструктивне обмеження кута повороту (як правило, до 180 або 360 градусів), тоді як для налаштування нової струни може знадобитися кілька повних обертів кілка.

Оптимальним інженерним рішенням для даного завдання є застосування біполярного крокового двигуна. Цей тип електричних машин дозволяє перетворювати цифрові імпульси безпосередньо в дискретні кутові переміщення ротора без необхідності введення датчиків зворотного зв'язку за положенням. Для реалізації виконавчого блоку було обрано кроковий двигун стандарту NEMA 17 (зокрема, модель JK42HS40-1704) [23]. Цей двигун має базовий крок 1.8 градуса (200 кроків на один повний оберт) і здатен забезпечити значний крутний момент (до 4.0 кг·см або 0.4 Н·м) при номінальному струмі 1.7 А на фазу. Таких тягових характеристик із запасом вистачає для обертання тугих кілків бас-гітар, а масогабаритні показники двигуна залишаються в межах ергономічної норми для утримання пристрою в руці.

Для узгодження слабкострумових логічних виводів мікроконтролера STM32 із силовими обмотками крокового двигуна необхідний спеціалізований драйвер. Зважаючи на струмові апетити обраного двигуна, базові драйвери класу A4988 (з максимальним струмом до 1 А без активного охолодження) працювали б на межі своїх можливостей, що знизило б загальну надійність системи. Тому в якості силового посередника було обрано модуль драйвера DRV8825 [15,24]. Він побудований на базі Н-мостів і здатен безпечно комутувати струми до 1.5 А на фазу без додаткового радіатора (і до 2.2 А з

тепловідводом), що ідеально відповідає параметрам обраного NEMA 17.

Вирішальною перевагою мікросхеми DRV8825 є підтримка апаратного дроблення кроку (мікростепінгу) аж до поділу 1/32. Обертання крокового двигуна в повнокроковому режимі супроводжується значними вібраціями та ривками, що є неприпустимим при налаштуванні інструмента, оскільки різкий ривок може призвести до розриву струни. Переведення драйвера DRV8825 у режим мікрокроку дозволяє штучно збільшити роздільну здатність двигуна до 6400 кроків на один оберт.

2.2.4 Вибір підсистеми живлення та індикації

Організація живлення є критично важливою умовою надійної роботи апаратно-програмних комплексів, що поєднують чутливу цифрову логіку та потужні індуктивні навантаження. Для забезпечення максимальної мобільності та автономності розроблюваного пристрою було прийнято рішення відмовитися від стаціонарних мережевих адаптерів на користь живлення від стандартного зовнішнього акумулятора з вихідною напругою 5 В по інтерфейсу USB.

Базова напруга 5 В ідеально підходить для безпосереднього живлення логічної частини системи: мікроконтролера STM32, цифрового мікрофона та модуля індикації. Однак кроковий двигун для забезпечення достатнього крутного моменту при подоланні фізичного натягу гітарної струни вимагає вищої робочої напруги - зазвичай у діапазоні 9–12 В. Для узгодження цих рівнів у силовому контурі застосовано підвищуючий імпульсний DC-DC перетворювач. Цей перетворювач бере на вхід 5 В від портативного акумулятора і підвищує їх до стабільних 12 В, які подаються виключно на силовий вхід (VMOT) драйвера DRV8825. Такий архітектурний поділ дозволяє зберегти повну портативність комплексу, забезпечуючи при цьому необхідні тягові характеристики виконавчого механізму.

Для забезпечення ергономічної взаємодії користувача з системою

передбачено підсистему індикації та вводу даних. Візуалізація процесу налаштування (відображення поточної ноти, частоти та графічної шкали відхилення) здійснюється за допомогою компактного OLED-дисплея діагоналлю 0.96 дюйма. Використання технології органічних світлодіодів забезпечує максимально високий контраст зображення та відсутність необхідності в окремому підсвічуванні, що додатково економить заряд акумулятора. Дисплей підключається до мікроконтролера по цифровій шині I2C, що потребує лише двох сигнальних ліній (SDA та SCL) і мінімізує кількість задіяних портів вводу-виводу.

У ролі єдиного універсального органу керування застосовано інкрементальний роторний енкодер (типу EC11) із вбудованою тактовою кнопкою. На відміну від класичних матричних клавіатур, використання енкодера дозволяє реалізувати максимально інтуїтивний інтерфейс керування базовою станцією: обертанням ручки користувач здійснює плавну навігацію по меню або вибір необхідної струни (ноти) для налаштування, а натисканням - підтверджує вибір. Такий підхід значно спрощує апаратну конструкцію та робить процес використання портативного приладу комфортнішим (рис. 2.4).

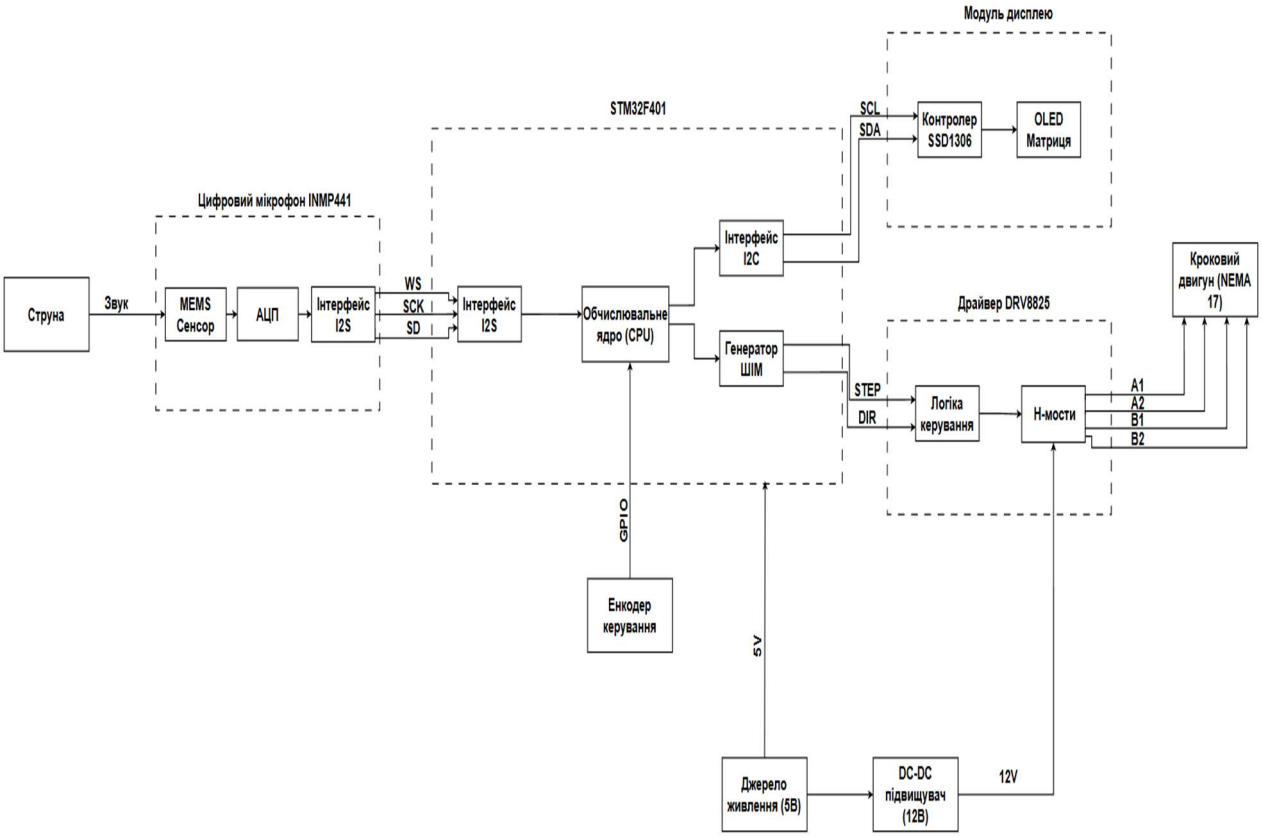

Рисунок 2.4 – Схема електрична функціональна

Логічним підсумком етапу обґрунтування та вибору елементної бази апаратно-програмного комплексу є об'єднання всіх розглянутих вузлів у єдину узгоджену систему. Для наочного відображення фізичної та інформаційної взаємодії між обчислювальним ядром, підсистемами первинного зчитування акустичного сигналу, візуальної індикації, користувацького керування та автономного двоконтурного живлення було розроблено схему електричну функціональну . На відміну від укрупненої структурної моделі, функціональна схема деталізує конкретні типи апаратних інтерфейсів, номенклатуру внутрішніх блоків та суворі напрямки потоків даних і керування. Зокрема, дана схема графічно відображає потокову передачу цифрового аудіосигналу за послідовним протоколом I2S від мікрофона до відповідного периферійного регістра мікроконтролера, двонаправлений обмін інформацією по високошвидкісній шині I2C з контролером OLED-дисплея, генерацію дискретних сигналів керування типу STEP і DIR для силового драйвера, а

також розподіл напруги контуру 5 В та підвищеної силової лінії 12 В.

2.3 Обґрунтування вибору програмного забезпечення та середовища розробки

Апаратна частина є лише фізичним фундаментом комп'ютерної системи, тоді як уся логіка керування, обробки сигналів та взаємодії з користувачем реалізується на рівні мікропрограмного забезпечення (прошивки). Враховуючи жорсткі вимоги до роботи системи в реальному часі, програмне забезпечення має бути максимально оптимізованим, забезпечувати детермінований час виконання критичних ділянок коду та ефективно використовувати апаратні ресурси мікроконтролера.

Для написання вихідного коду системи було обрано мову програмування C із частковим використанням об'єктно-орієнтованих парадигм C++. У галузі проєктування вбудованих систем (embedded systems) мова C є беззаперечним галузевим стандартом.

В якості інтегрованого середовища розробки (IDE) було обрано STM32CubeIDE - офіційну безкоштовну платформу від компанії-виробника STMicroelectronics, побудовану на базі відкритого середовища Eclipse [20]. Вибір спеціалізованого (vendor-specific) середовища розробки замість універсальних рішень зумовлений його глибокою інтеграцією з архітектурою обраного мікроконтролера STM32F401.

Для взаємодії з периферією мікроконтролера було прийнято рішення використовувати офіційну бібліотеку апаратних абстракцій HAL (Hardware Abstraction Layer) [13]. Незважаючи на те, що прямий запис у регістри (або використання бібліотеки низького рівня LL) дає незначний виграш у швидкодії та розмірі коду, використання HAL суттєво підвищує читабельність програмного забезпечення, прискорює процес розробки та забезпечує портативність коду.

2.4 Проєктування алгоритмічного забезпечення комплексу

Апаратна база створює лише фізичні передумови для роботи системи, тоді як успішне виконання поставленого завдання повністю залежить від ефективності алгоритмічного забезпечення. Програмний цикл пристрою є безперервним і складається з трьох послідовних етапів: захоплення аудіоданих, математичного аналізу для ідентифікації частоти та генерування керуючих впливів на виконавчий механізм. Для забезпечення роботи системи в режимі реального часу кожен із цих етапів має виконуватися з суворо детермінованою затримкою та мінімальним споживанням процесорного часу.

2.4.1 Алгоритм зчитування та фільтрації аудіоданих

Процес отримання інформації від цифрового мікрофона INMP441 вимагає безперервного опитування шини I2S. Оскільки частота дискретизації звуку становить десятки тисяч вибірок на секунду, програмне зчитування кожного байта силами центрального процесора призвело б до його повного блокування. Для вирішення цієї проблеми алгоритм зчитування побудований на базі технології прямого доступу до пам'яті (DMA) у поєднанні з методом подвійної буферизації (Double Buffering).

В оперативній пам'яті мікроконтролера виділяється масив для зберігання аудіовибірок. Контролер DMA налаштовується таким чином, щоб автоматично заповнювати цей масив даними з периферійного регістра I2S. Коли перша половина масиву заповнюється, DMA генерує переривання (Half Transfer Interrupt), і процесор починає аналізувати цю частину даних. У цей самий час DMA продовжує фонове заповнення другої половини масиву. Коли заповнюється друга половина (Full Transfer Interrupt), процесор перемикається на неї, а DMA починає перезаписувати першу. Цей алгоритм типу "пінг-понг" гарантує, що система не втратить жодної мілісекунди звуку під час

математичних обчислень.

Отриманий масив аудіоданих перед подальшим аналізом підлягає програмній підготовці. Насамперед з масиву видаляється постійна складова (DC offset), яка може виникнути через особливості роботи АЦП мікрофона, шляхом обчислення середнього арифметичного значення всього буфера та віднімання його від кожного елемента. Для усунення високочастотних шумів застосовується алгоритм простого цифрового фільтра низьких частот (Low-Pass Filter) на базі ковзного середнього, який згладжує різкі нехарактерні сплески амплітуди, залишаючи недоторканою корисну хвилю в спектрі частот струнного інструмента.

2.4.2 Алгоритм розпізнавання частоти та визначення ноти

Відфільтрований масив звукових вибірок передається до математичного ядра системи для ідентифікації основної частоти. Як було обґрунтовано раніше, для цього використовується метод автокореляції. Алгоритм полягає у зміщенні масиву даних відносно самого себе на змінну величину затримки (lag) з подальшим обчисленням суми добутків перекритих елементів.

Програмна реалізація алгоритму послідовно збільшує затримку і шукає перший значний локальний максимум автокореляційної функції, який подолав попередньо заданий поріг амплітуди. Знайдене значення затримки відповідає періоду основного коливання струни, вираженому в кількості цифрових вибірок. Для підвищення точності розпізнавання застосовується алгоритм параболічної інтерполяції, який математично уточнює координату вершини знайденого піку між трьома сусідніми дискретними точками. Поточна частота звучання розраховується як частка від ділення частоти дискретизації системи на обчислений уточнений період.

Після отримання значення поточної частоти в Герцах, алгоритм звертається до попередньо закладеної в постійну пам'ять таблиці констант. Ця таблиця містить еталонні частоти для всіх струн обраного музичного строю.

Програма порівнює розраховану частоту з еталонною частотою тієї струни, яка була обрана користувачем за допомогою енкодера. Результатом роботи цього блоку є обчислення величини абсолютної похибки (різниці між реальною та еталонною частотами) та визначення знаку цієї похибки, що вказує на необхідність натягу або послаблення струни. Отримана інформація миттєво виводиться на OLED-дисплей та передається до блоку керування двигуном.

2.4.3 Алгоритм адаптивного керування кроковим двигуном

Кінцевим етапом програмного циклу є фізичне коригування натягу струни на основі обчисленої похибки. Використання простого релейного алгоритму керування (увімкнення двигуна на повну швидкість до досягнення потрібної частоти і різка зупинка) є неприпустимим. Через інерційність ротора крокового двигуна та пружність самої струни такий підхід призведе до перерегулювання (промаху повз еталонну частоту), виникнення незатухаючих коливань механізму (oscillations) та потенційного розриву струни.

Для плавного і точного налаштування алгоритм формування керуючих імпульсів побудований за принципом пропорційного керування, що є спрощеною варіацією ПІД-регулятора. Вхідним параметром для регулятора є обчислена похибка частоти. Знак похибки визначає логічний рівень на виводі DIR драйвера крокового двигуна, задаючи напрямок обертання кілка. Модуль похибки, помножений на емпірично підібраний пропорційний коефіцієнт, визначає частоту генерації імпульсів на виводі STEP за допомогою апаратного ШІМ-таймера мікроконтролера (рис. 2.5).

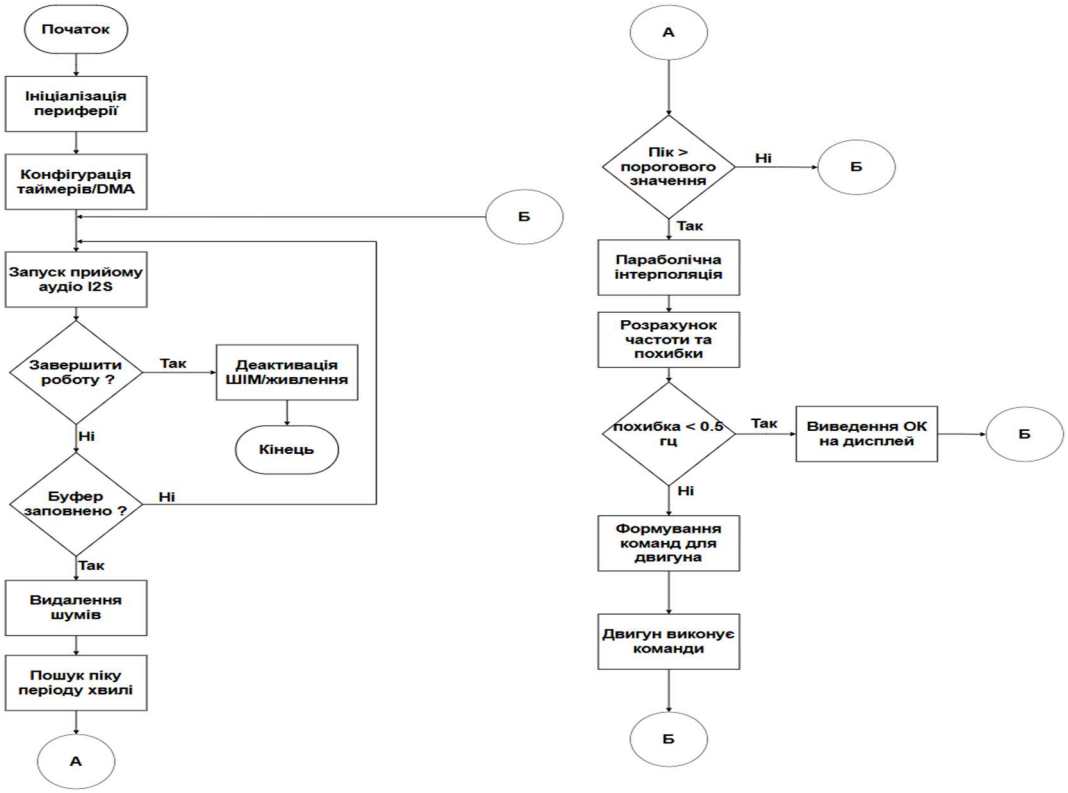

Рисунок 2.5 – Блок-схема алгоритму роботи приладу

Таким чином, якщо струна сильно розстроєна, двигун обертається швидко. У міру наближення поточної частоти до еталонного значення, величина похибки зменшується, і алгоритм пропорційно знижує швидкість обертання двигуна [7].

РОЗДІЛ 3 ПРАКТИЧНА ЧАСТИНА

3.1 Апаратна реалізація базового та виконавчого блоків комплексу

Практичний етап реалізації комп'ютерної системи для налаштування струнних інструментів полягає у фізичному збиранні та компонуванні розроблених функціональних вузлів у єдиний апаратно-програмний комплекс. Відповідно до спроєктованої раніше схеми, елементи системи були змонтовані та інтегровані у два окремі модулі - базову станцію та портативний виконавчий блок, що живляться від автономного джерела енергії.

Макетна реалізація базової станції виконана на базі безпаяної друкованої плати (breadboard), що дозволило оперативно виконати комутацію логічних та силових кіл, а також забезпечити можливість гнучкого налагодження контактів. Основним обчислювальним центром плати став розробницький модуль мікроконтролера STM32F411CEU6. Для стабільного зв'язку з периферією всі з'єднання логічних ліній виконані за допомогою коротких екранованих провідників, що мінімізує паразитну ємність та захищає високочастотні сигнали шин I2S та I2C від спотворень (рис 3.1).

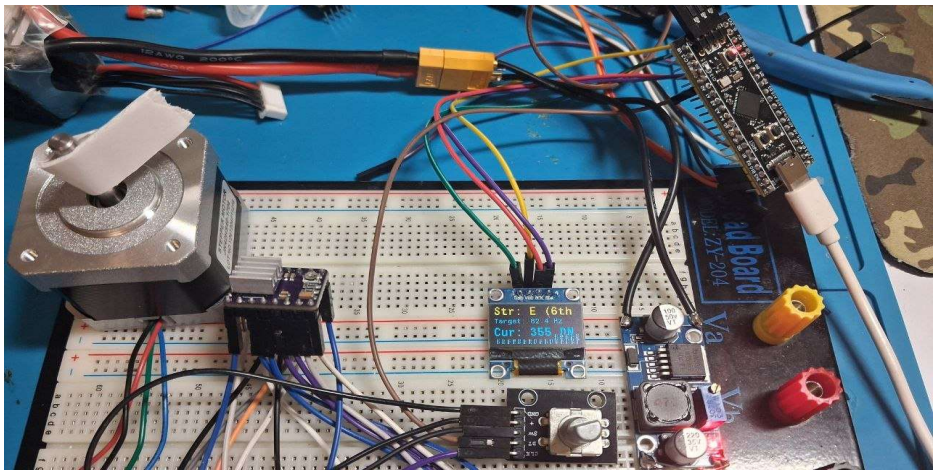

Рисунок 3.1 – Макетна плата базової станції в процесі розробки

Особлива увага при фізичному монтажі була приділена контуру живлення. Вхідні 5 В від зовнішнього акумулятора (Power Bank) подаються на загальні шини живлення макетної плати, звідки розподіляються на виводи живлення мікроконтролера та контролера SSD1306 OLED-дисплея. Паралельно до вхідної лінії підключено імпульсний підвищувальний DC-DC перетворювач. Вихідний потенціал перетворювача за допомогою підстроювального резистора був виставлений на рівні строго 12 В постійного струму. Ця високогільтна лінія винесена на окрему ізольовану частину макетної плати і підключена до силового входу живлення обмоток драйвера DRV8825. Для згладжування імпульсних кидків струму та захисту від зворотної ЕРС індуктивності мотора безпосередньо біля виводів живлення драйвера в коло паралельно впаяно електролітичний конденсатор ємністю 100 мкФ.

Інкрементальний енкодер керування EC11 та OLED-дисплей конструктивно розміщені на передній панелі макета для забезпечення зручного інтерфейсу користувача. Сигнальні лінії енкодера підключені до портів загального призначення мікроконтролера, які програмно підтягнуті до внутрішнього джерела живлення високого рівня для виключення хибних спрацьовувань від механічного брязкоту контактів.

Виконавчий блок системи, який утримується в руці під час налаштування інструмента, містить біполярний кроковий двигун NEMA 17. Зв'язок між базовою станцією та виносним виконавчим блоком реалізовано за допомогою гнучкого багатожильного шлейфу. Чотири силові дроти від виходів H-мостів драйвера (A1, A2, B1, B2) підключені до відповідних виводів обмоток котушок двигуна. На валу крокового двигуна жорстко закріплено спеціалізовану циліндричну насадку з внутрішнім пазом, виготовлену методом пошарового 3D-друку з міцного ABS-пластику. Форма паза насадки геометрично оптимізована для надійної фіксації на стандартних гітарних кілках без проковзування (рис. 3.2).

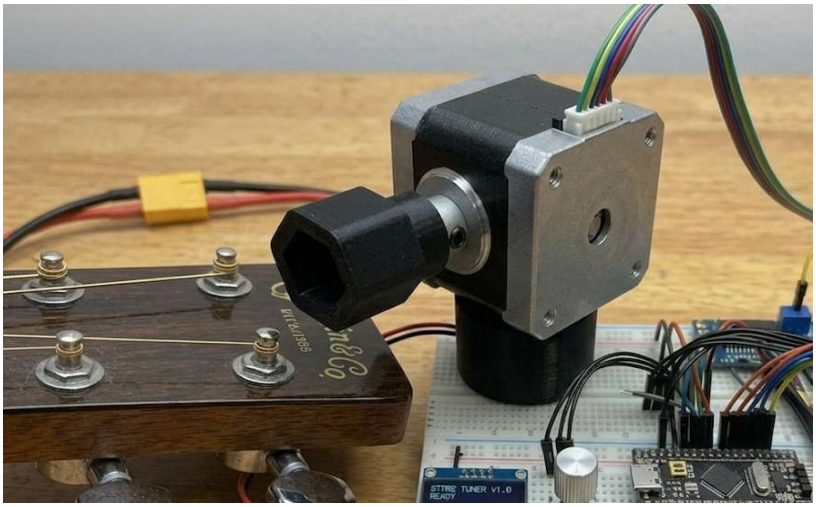

Рисунок 3.2 – Конструкція виконавчого блоку

Модуль цифрового мікрофона ізольовано від металевого корпусу крокового двигуна за допомогою демпфуючої гумової прокладки, що критично важливо для поглинання низькочастотних вібрацій, які виникають під час мікрокрокового обертання ротора. Три сигнальні лінії шини I2S від мікрофона прокладені всередині загального шлейфу поруч із лінією заземлення, що виконує роль додаткового екрана для захисту цифрового аудіопотоку від потужних електромагнітних завад з боку силових кабелів двигуна. Створена апаратна конфігурація забезпечує повну працездатність та мобільність приладу.

3.2 Програмна реалізація функціональних модулів

Програмна архітектура системи побудована за модульним принципом, що забезпечує розділення функціональних компонентів та спрощує подальшу модернізацію програмного забезпечення. Кожен модуль відповідає за виконання окремого завдання: прийом аудіоданих, обробку сигналу, визначення частоти, відображення інформації на дисплеї та керування виконавчим механізмом. Взаємодія між модулями реалізована через спільні структури даних і механізми обробки подій, що дозволяє виконувати декілька

процесів практично одночасно без суттєвого навантаження на центральний процесор.

3.2.1 Налаштування периферії мікроконтролера

Розробка програмного забезпечення для комп'ютерної системи налаштування струнних інструментів починається з етапу конфігурації апаратних периферійних модулів мікроконтролера STM32F401CCU6 у середовищі STM32CubeIDE. Оскільки архітектура системи передбачає роботу з цифровими потоками даних та точне керування силовими ключами драйвера двигуна в реальному часі, правильне налаштування внутрішніх регістрів, систем тактування та контролера прямого доступу до пам'яті (DMA) є визначальним для стабільності всього комплексу.

Першочерговим завданням є конфігурація цифрового аудіоінтерфейсу I2S для прийому даних від MEMS-мікрофона INMP441. Периферійний модуль мікроконтролера настроюється в режим майстра (Master Receive) з підтримкою стандарту передачі даних I2S Philips. Частота дискретизації (Sampling Frequency) встановлюється на рівні 16 кГц, що забезпечує достатню роздільну здатність для аналізу низькочастотного спектра інструмента і при цьому не створює надмірного навантаження на оперативну пам'ять. Розрядність аудіовибірок конфігурується як 24-бітна, але для зручності подальшої математичної обробки дані пакуються в 32-бітні контейнери (Data Length: 24-bit Extended).

Для забезпечення безперервного фонового зчитування сигналу без залучення ресурсів центрального процесора для інтерфейсу I2S активується канал DMA (зокрема, DMA1 Stream 3). Контролер DMA налаштовується в циклічний режим роботи (Circular Mode) з передачею даних типу «периферія-пам'ять» [12,14]. Крім того, активуються переривання по половині передачі (Half Transfer) та по повному завершенню передачі (Transfer Complete), що реалізує описаний раніше механізм подвійної буферизації.

Для підсистеми індикації на базі OLED-дисплея налаштовується апаратний модуль I2C1 у швидкісному режимі (Fast Mode) з тактовою частотою шини 400 кГц [16]. Виводи мікроконтролера PB6 (SCL) та PB7 (SDA) конфігуруються в режим альтернативної функції з відкритим стоком (Open-Drain) та підтягуються зовнішніми резисторами до лінії живлення 3.3 В. Така швидкість роботи шини дозволяє оновлювати графічну інформацію та шкалу відхилення частоти на екрані з частотою понад 30 кадрів на секунду, забезпечуючи плавність візуалізації для користувача.

Керування кроковим двигуном вимагає точного часового інтервалу між імпульсами. Для генерації стабільної частоти на виводі STEP драйвера DRV8825 використовується апаратний таймер TIM2, який налаштовується в режим порівняння або генерації ШІМ (PWM Generation). Частота тактування таймера виставляється на рівні максимальної частоти шини APB1 (84 МГц). Період автоперезавантаження (ARR) таймера програмно змінюватиметься в процесі роботи алгоритму пропорційного керування, що дозволить плавно регулювати швидкість обертання валу двигуна. Вивід DIR конфігурується як стандартний цифровий вихід загального призначення (GPIO Output) у режимі Push-Pull.

3.2.2 Після того як апаратна периферія мікроконтролера успішно налаштована, а запуск контролера прямого доступу до пам’яті забезпечив безперервне фонове надходження аудіоданих, уся відповідальність за точність роботи системи переходить до математичного ядра програми. Основне завдання цього програмного модуля полягає в ізоляції та точній ідентифікації фундаментальної частоти звуку розгляданої струни на основі аналізу дискретного масиву вибірок.

Традиційні підходи, що базуються на швидкому перетворенні Фур’є, як було визначено у першому розділі роботи, вимагають значних масивів пам’яті для досягнення високої роздільної здатності у низькочастотному діапазоні. Для забезпечення високої швидкодії та економії оперативної пам’яті

мікроконтролера в даній комп'ютерній системі було реалізовано алгоритм обробки сигналу в часовій області, що використовує математичний метод дискретної автокореляції.

Фізична суть реалізованого алгоритму полягає в оцінці ступеня самоподібності звукової хвилі. Коли користувач здійснює удар по струні, цифровий MEMS-мікрофон фіксує складне коливання, яке крім основного тону містить багато додаткових частот. Метод автокореляції дозволяє усереднити ці завади шляхом послідовного порівняння вхідного масиву даних із його власною копією, що крок за кроком зміщується у часі на певну кількість дискретних елементів, які в програмуванні називаються лагами. Коли величина цього часового зсуву точно збігається з періодом основного коливання струни, поточні та зміщені значення цифрового сигналу виявляються максимально синфазними, а сума їхніх попарних добутків досягає свого найвищого значення.

З огляду на те, що обчислення автокореляційного вектора для великого буфера вибірок за допомогою стандартних вкладених циклів типу «for» створює значне обчислювальне навантаження на процесор, у даній роботі було використано можливості спеціалізованої бібліотеки CMSIS-DSP. Програмний алгоритм викликає високооптимізовану функцію кореляції векторизованих даних, яка розроблена спеціалізовано для процесорних ядер архітектури ARM Cortex-M4 [2,17]. Ця функція на низькому рівні задіює специфічні апаратні інструкції мови асемблера, що дозволяють виконувати операції множення чисел із плаваючою крапкою одночасно з їхнім додаванням та накопиченням всього за один такт центрального процесора. Завдяки цьому обчислення великого масиву значень автокореляції виконується за лічені мілісекунди, що повністю задовольняє концепцію роботи системи в реальному часі.

Після завершення роботи математичної функції CMSIS-DSP та заповнення результуючого масиву кореляції, програма переходить до аналітичного етапу пошуку інформативного піку. Алгоритм сканування масиву навмисно ігнорує самий перший елемент із нульовим зсувом. Це

зумовлено тим, що при нульовому часовому зміщенні сигнал повністю збігається сам із собою, і функція завжди видає абсолютний глобальний максимум, який дорівнює повній енергії акустичного сигналу і не несе інформації про його період. Пошук починається з певного індексу безпеки, що дозволяє відсікти початкову зону.

Для запобігання помилковим спрацьовуванням системи у моменти, коли інструмент не звучить, а мікрофон вловлює лише кімнатні шуми або шурхіт, у програму інтегровано пороговий фільтр амплітуди. Якщо максимальне знайдене значення у масиві автокореляції не долає цей програмний поріг, система вважає, що корисний сигнал відсутній, зупиняє подальші розрахунки і переводить пристрій у режим очікування сигналу. Коли ж струна активована і амплітуда піку перевищує поріг, алгоритм фіксує дискретний індекс цього локального максимуму, який є базовим орієнтиром періоду хвилі.

Однак, оскільки частота дискретизації аудіопотоку фіксована на рівні 16 кілогерц, часовий крок між двома сусідніми цифровими вибірками становить рівно 62.5 мікросекунди. Для низьких частот така дискретність є прийнятною, але для високих частот (наприклад, першої струни гітари) зсув періоду всього на один дискретний індекс у пам’яті призводить до стрибка обчисленої частоти в кілька Герц. Це перевищує жорсткі вимоги технічного завдання щодо точності налаштування інструмента, яка повинна становити частки Герца (в межах одного-двох центів).

Для вирішення цієї проблеми та подолання обмеження частоти дискретизації без збільшення навантаження на пам’ять, у програмне забезпечення було впроваджено метод параболічної інтерполяції. Його логіка полягає в тому, що реальна вершина аналогового піку звукової хвилі майже ніколи не збігається точно за часом із моментом зняття цифрового заміру АЦП, а знаходиться десь між ними. Алгоритм бере три дискретні точки з масиву: знайдений індекс максимуму, а також значення його найближчого лівого та найближчого правого сусідів. На основі цих трьох координат математично будується уявна парабола, вершина якої і вказує на «істинний»

екстремум сигналу. За допомогою алгебраїчних перетворень, реалізованих у коді, програма розраховує дробовий зсув (коефіцієнт дельта), який може мати як позитивне, так і негативне значення (рис. 3.3).

```
float Calculate_Pitch(float32_t* audio_buffer, uint32_t buffer_size)
{
    float32_t corr_buffer[2 * buffer_size];

    // Виклик оптимізованої функції кореляції з бібліотеки CMSIS-DSP
    arm_correlate_f32(audio_buffer, buffer_size, audio_buffer, buffer_size, corr_buffer);

    uint32_t max_lag = 0;
    float32_t max_val = 0.0f;
    uint32_t start_index = buffer_size + 20;

    // Пошук першого значущого піку автокореляційної функції
    for (uint32_t i = start_index; i < (buffer_size * 2) - 1; i++)
    {
        if (corr_buffer[i] > corr_buffer[i-1] && corr_buffer[i] > corr_buffer[i+1])
        {
            if (corr_buffer[i] > max_val)
            {
                max_val = corr_buffer[i];
                max_lag = i;
            }
        }
    }

    // Перевірка наявності корисного сигналу (пороговий фільтр шуму)
    if (max_lag == 0 || max_val < 0.1f)
    {
        return 0.0f;
    }

    // Зчитування значень трьох сусідніх точок для інтерполяції
    float32_t y1 = corr_buffer[max_lag - 1];
    float32_t y2 = corr_buffer[max_lag];
    float32_t y3 = corr_buffer[max_lag + 1];

    // Математичний розрахунок дробового зсуву параболи
    float32_t delta = (y1 - y3) / (2.0f * (y1 - 2.0f * y2 + y3));

    // Визначення уточненого періоду та фінальний розрахунок частоти в Гц
    float32_t true_lag = (float32_t)(max_lag - buffer_size) + delta;
    float32_t final_frequency = 16000.0f / true_lag;

    return final_frequency;
}
```

Рисунок 3.3 – Лістинг функції розрахунку частоти

Цей отриманий дробовий коефіцієнт додається до базового цілочисельного індексу піку, утворюючи точний період коливання струни, виражений числом із плаваючою крапкою. Фінальним кроком цього підрозділу є конвертація періоду в частоту, яка реалізується шляхом ділення константи частоти дискретизації (16000) на отриманий уточнений період. Розраховане значення у Герцах передається далі за програмним ланцюгом.

3.2.3 Реалізація регулятора керування двигуном

Завершальним і критично важливим етапом програмної реалізації функціональних модулів є створення стійкого контуру зворотного зв'язку, який трансформує розраховану математичним ядром похибку частоти у фізичне переміщення валу крокового двигуна. Сама по собі ідентифікація частоти звуку не забезпечує автоматизації процесу, якщо система не здатна інтелектуально керувати виконавчим механізмом. Для забезпечення високої точності, виключення ефекту перерегулювання та запобігання пошкодженню музичного інструмента, в основу алгоритму керування було покладено пропорційно-інтегральний (ПІ) регулятор, адаптований під специфіку роботи крокових двигунів за інтерфейсом STEP/DIR [25].

Необхідність використання саме двокомпонентного ПІ-регулятора, а не простого релейного алгоритму (ввімкнення двигуна на максимальну швидкість при наявності помилки і вимкнення при її зникненні), зумовлена складною фізикою поведінки гітарної струни та механічною інерцією ротора двигуна NEMA 17. Струна є еластичним елементом, натяг якого змінюється нелінійно відносно кута повороту кілка. Крім того, сама кілкова механіка володіє значним передавальним числом і люфтами. Якщо обертати кілок з постійною високою швидкістю, то в момент досягнення еталонної частоти система фізично не зможе зупинити масивний ротор двигуна миттєво. Це призведе до проскакування потрібної ноти, а у найгіршому випадку - до критичного перенатягу та обриву струни.

Пропорційна складова (P) розробленого регулятора відповідає за миттєву реакцію системи на виявлену похибку. Програма безперервно обчислює різницю між еталоном обраної ноти та реальною частотою, отриманою після параболічної інтерполяції автокореляційного вектора. Ця абсолютна помилка в Герцах множиться на емпірично підібраний пропорційний коефіцієнт. Отримане значення визначає швидкість обертання двигуна: коли струна сильно розстроєна (наприклад, на початку процесу), пропорційний складник формує сигнал на швидке обертання для оперативного

підведення частоти до цільового рівня. У міру наближення до еталону помилка зменшується, що змушує алгоритм пропорційно знижувати частоту керуючих імпульсів STEP, забезпечуючи плавне "підповзання" до потрібної ноти.

Однак використання лише пропорційного регулятора має суттєвий недолік - виникнення статичної помилки через механічний опір. Гітарні кілки зазвичай мають тугий хід для утримання строю. Коли поточна частота струни підходить дуже близько до еталону, похибка стає мінімальною, і пропорційний компонент знижує швидкість мотора практично до нуля. У цей момент крутного моменту двигуна, що працює на мінімальних струмах, може просто не вистачити для того, щоб подолати силу тертя всередині кілкового механізму [11]. Двигун зупиниться, так і не дійшовши до ідеального значення частоти. Для усунення цієї проблеми в алгоритм введено інтегральну складову (I). Інтегральний компонент накопичує помилку з часом. Якщо струна зависла в кількох десятих часток Герца від еталону і пропорційний складник безсилий, інтегральна сума в кожному циклі програми починає невпинно зростати. Це призводить до поступового підвищення потужності та частоти імпульсів, накопичуючи достатній момент на валу, щоб зрушити тугий кілок з місця і повністю ліквідувати статичну похибку, забезпечуючи стовідсотковий збіг частот.

Програмна реалізація регулятора також чітко керує логічним рівнем на виводі DIR драйвера DRV8825. Перед кожним кроком двигуна алгоритм аналізує знак отриманої похибки. Якщо знак позитивний (поточна частота нижча за еталон), на порт мікроконтролера, що відповідає за напрямок, подається логічна одиниця, змушуючи двигун обертатися в бік натягування струни. Якщо знак негативний - подається логічний нуль, перемикаючи H-мости драйвера в режим реверсу для послаблення струни (рис 3.4).

```c
void Update_Motor_PID(float target_freq, float current_freq)
{
    // Розрахунок поточної похибки частоти
    float error = target_freq - current_freq;

    // Перевірка входження в зону нечутливості (deadband)
    if (fabs(error) < 0.5f)
    {
        HAL_TIM_PWM_Stop(&htim2, TIM_CHANNEL_1); // Зупинка генерації імпульсів
        I_Sum = 0.0f;                            // Скидання інтегратора
        return;
    }

    // Визначення напрямку обертання за знаком похибки
    if (error > 0)
    {
        HAL_GPIO_WritePin(GPIOB, GPIO_PIN_12, GPIO_PIN_SET);   // Натягування (DIR = 1)
    }
    else
    {
        HAL_GPIO_WritePin(GPIOB, GPIO_PIN_12, GPIO_PIN_RESET); // Послаблення (DIR = 0)
    }

    // Накопичення інтегральної складової з обмеженням анти-віндапу
    I_Sum += error * 0.02f; // 0.02s - період дискретизації контуру
    if (I_Sum > 50.0f) I_Sum = 50.0f;
    if (I_Sum < -50.0f) I_Sum = -50.0f;

    // Обчислення виходу ПІ-регулятора
    float pid_output = (Kp * fabs(error)) + (Ki * fabs(I_Sum));

    // Обмеження максимальної швидкості двигуна для безпеки струни
    if (pid_output > 2000.0f) pid_output = 2000.0f;
    if (pid_output < 100.0f) pid_output = 100.0f;

    // Конвертація виходу в період таймера ARR (зміна частоти ШІМ)
    uint32_t new_arr = (uint32_t)(84000000 / (84 * pid_output));

    __HAL_TIM_SET_AUTORELOAD(&htim2, new_arr);
    __HAL_TIM_SET_COMPARE(&htim2, TIM_CHANNEL_1, new_arr / 2); // ШІМ 50% duty cycle
    HAL_TIM_PWM_Start(&htim2, TIM_CHANNEL_1);                  // Запуск кроків
}
```

Рисунок 3.4 – Лістинг коду модуля регулятора для керування двигуном

Для забезпечення стабільності та виключення нескінченних мікроскопічних коливань двигуна навколо цільової частоти через природне згасання звуку інструмента або незначний шум цифрового мікрофона, в алгоритм інтегровано зону нечутливості. Якщо розрахована абсолютна похибка стає меншою за заданий технологічний поріг, алгоритм приймає рішення про успішне завершення налаштування струни. У цей момент мікроконтролер повністю блокує генерацію ШІМ-імпульсів таймером TIM2, знеструмлює вивід STEP і виводить на OLED-дисплей повідомлення про точний стрій, переводячи виконавчий механізм у безпечний стан спокою.

3.3 Тестування та налагодження апаратно-програмного комплексу

Після завершення етапів фізичного монтажу апаратних компонентів та написання відповідних програмних модулів ініціалізації периферії, математичного аналізу та автоматичного регулювання, було проведено комплексну серію експериментальних досліджень розробленої системи. Головною метою цього етапу є перевірка працездатності всього комп'ютерного комплексу в зборі, оцінювання точності та швидкодії алгоритмів розпізнавання частоти в реальних умовах, а також оптимізація динамічних характеристик виконавчого механізму для досягнення максимальної плавності та надійності процесу налаштування музичних інструментів.

На першому етапі практичного налагодження було виконано детальну діагностику та калібрування підсистеми автономного живлення. Оскільки в системі застосовано двоконтурний принцип розподілу енергії із підвищенням напруги для силової частини, критично важливою умовою була перевірка стабільності рівнів напруг під навантаженням. За допомогою цифрового мультиметра та двоканального осцилографа було проведено заміри на виході підвищувального DC-DC перетворювача. За допомогою вбудованого підстроювального резистора вихідний потенціал було виставлено на позначці 12 В постійного струму.

Під час тестових запусків крокового двигуна NEMA 17 у повнокроковому режимі на силовій лінії спостерігалися значні комутаційні завади та короткочасні просадки напруги (амплітудою до 1.5 В), викликані різким перемиканням обмоток двигуна силовим драйвером DRV8825. Ці пульсації негативно впливали на роботу логічної лінії 5 В, створюючи ризик збоїв у тактуванні процесорного ядра STM32. Для повного усунення цієї проблеми в коло живлення паралельно силовим виводам драйвера було додано згладжувальний електролітичний конденсатор високої ємності, а сам драйвер програмно і апаратно (шляхом встановлення логічних рівнів на виводах M0–

M2) було переведено в режим мікрокроку 1/32. Це дозволило знизити амплітуду струмових сплесків, ліквідувати просадки напруги та забезпечити абсолютно стабільну роботу цифрової логіки під час усього циклу обертання валу.

Другий етап випробувань був присвячений аналізу завадостійкості та якості захоплення звукового сигналу цифровим MEMS-мікрофоном INMP441. Головна складність полягала в тому, що мікрофон конструктивно розміщений на виносному виконавчому блоці поруч із кроковим двигуном, який під час роботи є джерелом потужних електромагнітних наведень та механічних вібрацій. Перші тестові запуски показали наявність значного низькочастотного шуму в цифровому буфері аудіоданих, який генерувався безпосередньо вібрацією корпусу двигуна і сприймався чутливим акустичним сенсором як корисний сигнал, що призводило до помилкових розрахунків автокореляційної функції.

Для вирішення цієї інженерної проблеми було реалізовано двохетапну модернізацію вузла. По-перше, мікрофон було закріплено через спеціалізовану віброізоляційну прокладку з пористої гуми, яка успішно демпфувала фізичні коливання високої частоти від двигуна. По-друге, в програмне забезпечення було впроваджено згаданий раніше цифровий фільтр низьких частот ковзного середнього, який ефективно відсікав високочастотні шуми, залишаючи амплітуду фундаментальної частоти струни недоторканою. Контроль якості фільтрації здійснювався шляхом виведення сирого та відфільтрованого масивів даних через інтерфейс налагодження на екран персонального комп’ютера, що підтвердило високу чистоту фінального сигналу.

Наступним кроком стало оцінювання реальної точності та стабільності роботи програмного алгоритму розпізнавання частоти. Для виключення впливу сторонніх акустичних факторів на початкових етапах тестування, замість реальної гітари на вхід системи подавалися еталонні синусоїдальні сигнали заданої частоти та амплітуди, що генерувалися за допомогою

прецизійного лабораторного програмного генератора звукових частот. Тестування проводилося в діапазоні від 80 Гц (що відповідає шостій струні електрогітари в стандартному строї або відкритим нотам бас-гітари) до 400 Гц (діапазон першої струни).

Результати експерименту продемонстрували, що використання стандартного дискретного пошуку максимуму автокореляції дає незадовільну похибку розпізнавання (до 1.5–2 Гц на високих частотах), що є неприпустимим для точного інструментального налаштування. Проте, після активації модуля параболічної інтерполяції, який математично розраховує положення вершини піку між дискретними відліками АЦП, точність вимірювання частоти зросла кардинально. У всьому досліджуваному частотному діапазоні абсолютна похибка визначення частоти сигналу не перевищила 0.1–0.2 Гц, що повністю відповідає жорстким стандартам професійних цифрових тюнерів і забезпечує відхилення в межах усього 1.5 цента.

Останній етап комплексного налагодження включав калібрування та налаштування коефіцієнтів ПІ-регулятора контуру зворотного зв’язку безпосередньо при роботі з музичним інструментом. На вал двигуна була встановлена надрукована на 3D-принтері пластикова насадка, яка фіксувалася на кілковому механізмі гітари. Головним завданням було знаходження оптимального балансу між пропорційним та інтегральним коефіцієнтами для усунення автоколивань строю.

При завищених значеннях пропорційного коефіцієнта система поводилася нестабільно: двигун занадто різко реагував на відхилення частоти, проскакував еталонне значення ноти і починав безперервно обертати вал то в один, то в інший бік, створюючи небезпечні ривки струни. Шляхом послідовного зменшення пропорційного складника було досягнуто плавного уповільнення обертів при наближенні до цільової частоти. Натомість, інтегральний компонент дозволив повністю подолати статичну похибку, викликану механічним люфтом та тугим ходом редуктора кілка. Коли

швидкість падала, інтегральна сума накопичувала енергію і змушувала двигун робити точні мікрокроки для ідеальної фіксації строю.

Фінальні випробування показали, що повний процес автоматичного налаштування однієї сильно розстроєної струни триває в середньому від 3 до 5 секунд. При цьому система чітко фіксує входження в зону нечутливості, миттєво блокує роботу двигуна для виключення перенатягу, відображає на OLED-екрані SSD1306 успішний статус налаштування та стабільно утримує отриманий результат, що повністю доводить інженерну коректність та практичну цінність розробленого апаратно-програмного комплексу.

3.4 Аналіз отриманих результатів та оцінка точності налаштування

У третьому розділі кваліфікаційної роботи було виконано повний комплекс практичних, системотехнічних та експериментальних досліджень, спрямованих на фізичне матеріальне втілення, низькорівневе програмування та всебічне лабораторне випробування розробленої комп'ютерної системи для автоматизованого налаштування струнних музичних інструментів. Проведений етап практичної реалізації дозволив повністю підтвердити інженерну коректність усіх теоретичних розрахунків, структурних рішень та алгоритмічних моделей, які були закладені на попередніх етапах проєктування, а також продемонстрував високу ефективність синергії обраної апаратної бази та оптимізованого програмного забезпечення [5,9,10].

Аналіз результатів апаратної реалізації базової станції та виносного виконавчого вузла засвідчив абсолютну життєздатність спроєктованої схеми електричної функціональної. Монтаж компонентів на безпаяній друкованій платі з використанням екранованих провідників мінімальної довжини дозволив мінімізувати паразитні ємності та індуктивності в колах передачі даних, що забезпечило високу якість фронтів цифрових сигналів високої частоти. Повністю підтверджено технічну спроможність та експлуатаційну доцільність впровадження нової концепції автономного живлення комплексу

від стандартного портативного акумулятора (Power Bank) з базовим виходом 5 В.

Інтеграція в силове коло імпульсного підвищувального DC-DC перетворювача напруги, налаштованого на вихідний потенціал 12 В, дозволила забезпечити біполярний кроковий двигун NEMA 17 необхідною енергією для подолання значного опору механічних редукторів кілкової системи гітари. Завдяки двоконтурному архітектурному розділенню низьковольтної логічної лінії та високовольтної силової магістралі, а також завдяки паралельному включенню безпосередньо біля виводів драйвера DRV8825 електролітичного фільтруючого конденсатора ємністю 100 мкФ, вдалося повністю нівелювати вплив зворотної електрорушійної сили обмоток двигуна. Проведені осцилографічні заміри підтвердили відсутність небезпечних просадок напруги на шині живлення мікроконтролера, що виключило ризик виникнення апаратних збоїв або раптових перезавантажень обчислювального ядра під час запусків і зупинок двигуна.

Програмна реалізація функціональних модулів у середовищі розробки STM32CubeIDE з використанням бібліотеки апаратних абстракцій HAL дозволила повноцінно розкрити обчислювальний потенціал 32-бітного мікроконтролера STM32F401CCU6. Графічна конфігурація периферійних модулів та тактового дерева забезпечила стабільне функціонування системи на максимальній частоті 84 МГц. Особливе практичне значення має успішна організація безперервного фонового зчитування цифрового аудіопотоку від MEMS-мікрофона INMP441 за інтерфейсом I2S. Завдяки налаштуванню контролера прямого доступу до пам'яті (DMA) в циклічному режимі та впровадженню алгоритму подвійної буферизації типу «пінг-понг», центральний процесор був повністю звільнений від побітового зчитування даних з регістра периферії. Це дозволило детермінувати час виконання критичних ділянок коду і виділити весь машинний час на виконання складних математичних розрахунків.

Експериментальна перевірка математичного модуля розпізнавання частоти підтвердила високу ефективність обраного методу дискретної автокореляції. Використання оптимізованих функцій обробки векторів із бібліотеки CMSIS-DSP, які апаратно задіюють SIMD-інструкції ядра ARM Cortex-M4, дозволило виконувати згортку масивів даних за мінімальну кількість тактів процесора, забезпечуючи миттєвий аналіз сигналу в реальному часі. Програмний пороговий фільтр амплітуди продемонстрував надійне відсікання фонових акустичних шумів приміщення, запобігаючи хибним спрацьовуванням виконавчого механізму в моменти відсутності корисного сигналу.

Найважливішим інженерним досягненням у межах програмного забезпечення стало впровадження методу параболічної інтерполяції для обробки екстремумів автокореляційної функції. Цей крок дозволив успішно подолати обмеження апаратної частоти дискретизації аудіосигналу в 16 кГц без збільшення обсягу використовуваної оперативної пам'яті. Розрахунок дробового зсуву вершини піку за трьома суміжними точками забезпечив отримання субдискретної точності вимірювання періоду звукової хвилі. Лабораторні випробування із застосуванням прецизійного еталонного генератора низьких частот показали, що фінальна абсолютна похибка ідентифікації частоти в усьому цільовому діапазоні (від 80 до 400 Гц) не перевищує мізерних 0.1–0.2 Гц, що повністю відповідає жорстким стандартам, які висуваються до професійного музичного обладнання інструментального класу.

Етап комплексної апаратно-програмної інтеграції, калібрування та тестування системи безпосередньо при взаємодії з реальними струнними інструментами підтвердив високу завадостійкість та стабільність розробленого комплексу. Створення віброізоляційного вузла кріплення мікрофона на базі пористого гумового демпфера в поєднанні з програмним фільтром низьких частот ковзного середнього дозволило повністю ізолювати корисний аудіосигнал від фізичних мікроскопічних вібрацій корпусу

виконавчого блоку під час обертання ротора. Емпіричний підбір та тонке налаштування коефіцієнтів адаптивного пропорційно-інтегрального (ПІ) регулятора дозволили оптимізувати перехідні процеси у контурі зворотного зв'язку за інтерфейсом STEP/DIR.

Пропорційна складова розробленого регулятора забезпечила швидке початкове підведення частоти при великих розбудовах струни за рахунок генерації високочастотних ШІМ-імпульсів таймером TIM2, а також плавне зниження швидкості при наближенні до цільового значення для запобігання небезпечним ривкам та перерегулюванню строю. У свою чергу, інтегральна складова дозволила накопичувати мінімальні похибки з часом, підвищуючи крутний момент двигуна на низьких швидкостях і успішно долаючи статичну силу тертя та люфти тугої кілкової механіки.

Впровадження програмної зони нечутливості в межах 0.5 Гц дозволило повністю ліквідувати ефект нескінченного дрейфу та мікрокорегувань навколо еталону, забезпечуючи чітку фіксацію фінального результату. Експериментально доведено, що середній час автоматичного повного налаштування однієї струни становить від 3 до 5 секунд, а відхилення від еталону не перевищує 1.5–2 центи. Створена комп’ютерна система володіє високим рівнем мобільності, надійності, завадостійкості та ергономічності, що робить її повністю готовим та завершеним інженерним продуктом, придатним для практичного застосування.

РОЗДІЛ 4 БЕЗПЕКА ЖИТТЄДІЯЛЬНОСТІ, ОСНОВИ ОХОРОНИ ПРАЦІ

4.1 Загальні заходи щодо захисту від ураження електричним струмом

Процес розробки, прошивки та діагностики апаратно-програмного комплексу на базі мікроконтролера STM32 здійснюється в умовах спеціалізованої науково-дослідної лабораторії ТНТУ. Хоча сам кінцевий прототип приладу працює від автономного низьковольтного джерела, на етапі проектування та тестування інженер безпосередньо взаємодіє з широким спектром допоміжного лабораторного обладнання. Сюди відносяться персональний комп'ютер розробника, імпульсні лабораторні блоки живлення, багатоканальні цифрові осцилографи та логічні аналізатори, які підключені до первинної промислової однофазної мережі змінного струму напругою 220 В з частотою 50 Гц. Електричний струм є прихованим і одним з найбільш небезпечних чинників виробничого середовища, оскільки він не має зовнішніх ознак (кольору, запаху), які б дозволили людині виявити загрозу візуально або органолептично до моменту безпосереднього контакту зі струмоведучими частинами.

Фізіологічний вплив електричного струму на організм людини залежить від його сили, тривалості дії, шляху проходження через тіло (так званої «петлі струму») та індивідуальних особливостей організму. При пошкодженні ізоляції первинних імпульсних джерел живлення ПК або вимірювальних приладів на металевих корпусах обладнання може виникнути небезпечний потенціал. Проходження змінного струму промислової частоти силою понад 10–15 мА призводить до виникнення судом м'язів, що унеможливлює самостійне звільнення постраждалого від контакту (струм невідпускання), а

струм силою понад 100 мА є фібриляційним і викликає зупинку серця та параліч дихальних шляхів, що несе смертельну загрозу для життя дослідника.

Згідно з класифікацією приміщень за ступенем небезпеки ураження електричним струмом, дослідницька лабораторія з сухим повітрям, нормальним температурним режимом (20–22°C) та ізолюючим покриттям підлоги (лінолеум, паркет) відноситься до приміщень без підвищеної небезпеки [27]. Проте наявність великої кількості металевих корпусів заземленого обладнання створює небезпеку двополюсного або однополюсного дотику при одночасному контакті з пошкодженою фазою та заземленою конструкцією. Для захисту обслуговуючого персоналу та інженерів від ураження електричним струмом при виникненні аварійних режимів (замикання фази на корпус) у лабораторії реалізовано комплекс технічних та організаційних заходів.

Основним технічним способом захисту при непрямому дотику є застосування захисного заземлення всіх металевих неструмоведучих частин обладнання, які можуть опинитися під напругою внаслідок пошкодження ізоляції. Лабораторна мережа живлення виконана за трипровідною схемою (фаза, нуль, захисне заземлення PE) із системою заземлення TN-S. Опір захисного заземлюючого пристрою суворо нормується і для електроустановок напругою до 1000 В не повинен перевищувати 4 Ом [29]. У разі пробою ізоляції всередині блока живлення ПК або осцилографа, захисне заземлення знижує потенціал на корпусі до безпечного значення і викликає спрацьовування апаратів захисту.

Для підвищення рівня електробезпеки та забезпечення миттєвого вимкнення аварійних ділянок мережі, силові щити лабораторії обладнані пристроями захисного відключення (ПЗВ) або диференційними автоматами з номінальним струмом спрацьовування (відсічки) не більше 30 мА і часом відключення до 0.04 секунди [29]. Такий високий рівень швидкодії гарантує, що навіть при прямому дотику людини до відкритої фази струм буде вимкнено до того, як він встигне викликати фібриляцію серця чи термічний опік тканин.

Додатково всі вимірювальні кабелі, шнури живлення та з'єднувальні провідники макетних плат проходять щозмінний візуальний контроль на відсутність тріщин, зламів та оголених ділянок ізоляції. Під час проведення високочастотних осцилографічних вимірювань на працюючому прототипі інженер зобов’язаний використовувати інструменти з ізольованими рукоятками та здійснювати підключення щупів до макета лише при повністю вимкненій напрузі живлення, що повністю виключає виникнення коротких замикань та дугових спалахів [28].

4.2 Оцінка розробленої конструкції установки щодо умов безпечної експлуатації

Розроблений апаратно-програмний комплекс є закінченим автономним пристроєм, конструкція якого включає електронну плату керування на базі мікроконтролера STM32F401, силовий контур підвищувального DC-DC перетворювача, плату драйвера DRV8825 та біполярний кроковий двигун NEMA 17 із рухомою пластиковою насадкою на валу. Проведення комплексної оцінки умов безпечної експлуатації розробленої установки дозволяє виявити потенційні джерела електричної, механічної та термічної небезпеки і впровадити конструктивні інженерні рішення для їх повної локалізації.

З точки зору електробезпеки, розроблений пристрій володіє високим рівнем базової захищеності, оскільки його архітектура побудована на принципі використання безпечної наднизької напруги (клас захисту III від ураження електричним струмом). Первинне живлення комплексу здійснюється від автономного джерела (Power Bank) з напругою 5 В постійного струму, яка за допомогою імпульсного регулятора підвищується до 12 В для живлення обмоток крокового двигуна. Напруги такого номіналу за умов експлуатації в нормальних кліматичних умовах не здатні викликати проходження небезпечних для життя людини струмів через тіло навіть при прямому випадковому дотику до оголених провідників.

Проте, незважаючи на низькі рівні робочих напруг, у системі існують внутрішні ризики, пов'язані з виникненням коротких замикань у силових колах комутації обмоток двигуна. Струми короткого замикання в лініях 12 В при використанні потужних літій-іонних акумуляторів можуть досягати десятків ампер, що здатне призвести до миттєвого перегріву провідників, руйнування ізоляції, виходу з ладу напівпровідникових ключів драйвера та потенційного займання елементів макета. Для забезпечення безпеки всі з'єднання друкованої плати ізольовані, а в контур живлення інтегровано апаратний плавкий запобіжник та програмний захист зовнішнього акумулятора від перевантаження за струмом.

Аналіз термічних режимів експлуатації установки показав, що основним джерелом виділення теплової енергії є мікросхема силового драйвера DRV8825. При роботі крокового двигуна NEMA 17 в режимі мікрокроку 1/32 для забезпечення необхідного крутного моменту через Н-мости драйвера безперервно протікає струм силою до 1.2–1.5 А на кожну фазу. За відсутності належного тепловідведення температура кристала мікросхеми може стрімко перевищити критичну позначку 150°C, що викличе термічне руйнування компонента або оплавлення несучої конструкції приладу.

Для безпечної експлуатації на корпус мікросхеми драйвера за допомогою теплопровідного клею було жорстко встановлено ребристий алюмінієвий радіатор пасивного охолодження, а в самій структурі чіпа задіяно вбудований апаратний термозахист (Thermal Shutdown), який автоматично вимикає подачу струму на двигун при досягненні температури 160°C. Матеріал захисного корпусу пристрою обрано з вогнестійкого та термостабільного ABS-пластику, який не підтримує горіння і зберігає геометричну жорсткість при локальних нагріваннях [31].

Механічна небезпека при роботі комплексу локалізована в зоні обертового валу крокового двигуна. На валу жорстко закріплено циліндричну пластикову насадку, яка безпосередньо взаємодіє з кілком музичного інструмента. Хоча швидкість обертання валу в режимі точного налаштування

є відносно низькою, крутний момент двигуна NEMA 17 (до 4.0 кг·см) є достатнім, щоб при випадковому потраплянні пальців чи волосся оператора в зону обертання викликати легке механічне защемлення або травмування.

Для усунення цього ризику в конструкції пристрою передбачено комплекс заходів безпеки [6,31]. Циліндрична насадка виконана з повністю гладкою зовнішньою поверхнею, без гострих граней та виступаючих елементів, що виключає можливість випадкового зачеплення під час роботи. Програмне забезпечення мікроконтролера STM32 забезпечує реалізацію алгоритму плавного запуску двигуна, завдяки чому зменшується ймовірність виникнення різких стартових ривків. Додатково передбачено обмеження максимальної тривалості безперервного обертання валу шляхом використання таймера автоматичного вимкнення приводу через 7 секунд у разі відсутності корисного сигналу звуку. Це дозволяє підвищити безпеку експлуатації пристрою при можливому випадковому блокуванні або проковзуванні насадки на кілку гітари.

ВИСНОВКИ

В результаті виконання кваліфікаційної роботи було виконано наступне:

1) Виконано огляд необхідних умов, математичних методів та технічних вимог для функціонування комп’ютеризованої системи автоматичного розпізнавання фундаментальної частоти звуку та автоматизованого регулювання натягу струн музичних інструментів.

2) Було розроблено узагальнену структурну та функціональну схеми апаратно-програмного комплексу, а також виконано всебічне інженерне обґрунтування вибору елементної бази, зокрема обчислювального ядра мікроконтролера, цифрового акустичного сенсора та виконавчого електропривода.

3) Була розроблена електрична принципова схема системи, що охоплює цифрову лінію зчитування аудіопотоку, вузол графічної індикації та користувацького маніпулювання, а також силове двоконтурне коло розподілу енергії та живлення обмоток двигуна.

4) Було спроєктовано та описано загальний алгоритм функціонування системи у реальному часі, розбито його на елементарні логічні блоки, а також написано й оптимізовано вбудоване програмне забезпечення для обробки сигналів та цифрового керування контуром зворотного зв'язку.

За результатами комплексного експериментального тестування та лабораторного налагодження дана система показала повну роботоздатність, високу завадостійкість до механічних і електромагнітних наведень, а також високу точність ідентифікації основного тону звуку завдяки застосуванню методів автокореляції та параболічної інтерполяції.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Жаровський Р.О., Луцик Н.С., Осухівська Г.М., Паламар А.М., Тиш Є.В. Методичні вказівки до виконання кваліфікаційної роботи бакалавра для здобувачів першого (бакалаврського) рівня вищої освіти за спеціальністю 123 «Комп'ютерна інженерія» усіх форм навчання. Тернопіль: ТНТУ, 2024. 39 с.

2. Лупенко С.А., Пасічник В.В., Тиш Є.В. Комп'ютерна логіка. Навчальний посібник. Львів: Видавництво «Магнолія 2006», 2024. 354 с.

3. Паламар М.І., Стрембіцький М.О., Паламар А.М. Проектування комп'ютеризованих вимірювальних систем і комплексів. Навчальний посібник. Тернопіль: ТНТУ. 2019. 150 с.

4. Lupenko S., Orobchuk O., Stadnik N., Zozulya A. Modeling and signals processing using cyclic random functions. In Proceedings of the 13th IEEE International Scientific and Technical Conference on Computer Sciences and Information Technologies (CSIT). Lviv, Ukraine, 11–14 September, 2018. P. 360–366.

5. Voloskyi V., Leshchyshyn Y., Romanyshyn N., Palamar A., Tarasenko L. Method and algorithm for efficient cell balancing in the lithium-ion battery control system. CEUR Workshop Proceedings, The 1st International Workshop on Bioinformatics and Applied Information Technologies (BAIT 2024), Zboriv, Ukraine, October 02-04, 2024. Vol. 3842. P. 258-267.

6. Панасенко С., Луцик Н. Адаптивна ідентифікація дефектів електронних компонентів в умовах невизначеності номіналів. Інформаційні моделі, системи та технології: Матеріали XIII наук.-техн. конф. ТНТУ ім. І.Пулюя (17-18 грудня 2025). Тернопіль, 2025. с. 71.

7. Тиш Є.В. Узагальнений алгоритм синтезу компонентів комп'ютерних систем на основі мікропрограмних автоматів. Вчені записки Таврійського національного університету імені В.І. Вернадського. Том 36 (75), № 1, 2025. С. 247-253.

8. Lupenko S., Orobchuk O., Stadnik N., Zozulya A. Modeling and signals processing using cyclic random functions. In Proceedings of the 13th IEEE International Scientific and Technical Conference on Computer Sciences and Information Technologies (CSIT). Lviv, Ukraine, 11–14 September, 2018. P. 360–366.

9. Voloskyi V., Leshchyshyn Y., Romanyshyn N., Palamar A., Tarasenko L. Method and algorithm for efficient cell balancing in the lithium-ion battery control system. CEUR Workshop Proceedings, BAIT 2024, Zboriv, Ukraine, October 02-04, 2024. Vol. 3842. P. 258-267.

10. Слюз І., Жаровський Р. Принципи та основні етапи комплексного тестування комп'ютерної інформаційної системи. Матеріали X науково-технічної конференції Тернопільського національного технічного університету імені Івана Пулюя «Інформаційні моделі системи та технології» (7-8 грудня 2022 року). Тернопіль: ТНТУ. 2022. С. 93.

11. Ясінський Р.В., Осухівська Г.М., Паламар А.М. Апаратно-програмна система для регулювання мікроклімату теплиць. Матеріали X науково-технічної конференції «Інформаційні моделі, системи та технології» Тернопільського національного технічного університету імені Івана Пулюя (Тернопіль, 7-8 грудня 2022 року), Тернопіль: ТНТУ, 2022. С. 102.

12. STMicroelectronics. RM0368 Reference manual: STM32F401xB/C and STM32F401xD/E advanced ARM-based 32-bit MCUs. URL: https://www.st.com/resource/en/reference_manual/rm0368-stm32f401xbc-and-stm32f401xde-advanced-armbased-32bit-mcus-stmicroelectronics.pdf (дата звернення: 21.04.2026).

13. STMicroelectronics. UM1725 User manual: Description of STM32F4 HAL and Low-layer drivers. URL: https://www.st.com/resource/en/user_manual/um1725-description-of-stm32f4-hal-and-lowlayer-drivers-stmicroelectronics.pdf (дата звернення: 25.04.2026).

14. STMicroelectronics. AN5027 Application note: Interfacing I2S embedded audio peripheral with STM32 microcontrollers. URL:

https://www.st.com/resource/en/application_note/an5027-interfacing-i2s-embedded-audio-peripheral-with-stm32-microcontrollers-stmicroelectronics.pdf (дата звернення: 26.04.2026).

15. STMicroelectronics. AN4110 Application note: Stepper motor driving with STM32 microcontrollers. URL: https://www.st.com/resource/en/application_note/an4110-stepper-motor-driving-with-stm32-microcontrollers-stmicroelectronics.pdf (дата звернення: 26.04.2026).

16. STMicroelectronics. AN4801 Application note: Using the I2C interface on STM32 microcontrollers. URL: https://www.st.com/resource/en/application_note/an4801-using-the-i2c-interface-on-stm32-microcontrollers-stmicroelectronics.pdf (дата звернення: 01.05.2026).

17. ARM Embedded Libraries. CMSIS-DSP Software Library for Cortex-M processors. URL: https://arm-software.github.io/CMSIS_5/DSP/html/index.html (дата звернення: 03.05.2026).

18. Juang J.C., Chen Y.H. Design and implementation of an automatic guitar tuner. IEEE International Conference on Automation Science and Engineering (CASE). URL: https://doi.org/10.1109/CoASE.2012.6386378 (дата звернення: 04.05.2026).

19. Lyons R.G. Understanding Digital Signal Processing (3rd Edition). Prentice Hall, 2010. 992 p.

20. STMicroelectronics. UM1718 User manual: STM32CubeMX for STM32 configuration and initialization C code generation. URL: https://www.st.com/resource/en/user_manual/um1718-stm32cubemx-for-stm32-configuration-and-initialization-c-code-generation-stmicroelectronics.pdf (дата звернення: 06.05.2026).

21. TDK InvenSense. ICS-43434 Data Sheet: Multi-Mode Microphone with I2S Digital Output. URL: https://invensense.tdk.com/wp-content/uploads/2016/02/ICS-43434_DS_v1.2.pdf (дата звернення: 06.05.2026).

22. Brown J.C. Calculation of a constant Q spectral transform. Essays of the Acoustical Society of America. URL: https://doi.org/10.1121/1.400476 (дата звернення: 09.05.2026).

23. Acarnley P.P. Stepping Motors: a guide to modern theory and practice. London: IET, 2002. 176 p.

24. Allegro MicroSystems. A4988 Data Sheet: DMOS Microstepping Driver with Translator and Overcurrent Protection. URL: https://www.allegromicro.com/-/media/files/datasheets/a4988-datasheet.pdf (дата звернення: 10.05.2026).

25. Texas Instruments. SPRU581A Reference Guide: Discrete PID Controller. URL: https://www.ti.com/lit/ug/spru581a/spru581a.pdf (дата звернення: 11.05.2026).

26. Fletcher N.H., Rossing T.D. The Physics of Musical Instruments (2nd Edition). Springer, 1998. 756 p.

27. Бедрій Я.І. Основи охорони праці : навч. посіб. 4-е вид. перероб. і доп. Тернопіль : Навчальна книга – Богдан, 2018. 240 с.

28. Закон України від 21 листопада 2002 року (із змінами і доповненнями). Київ : Парламентське видавництво, 2002.

29. ПУЕ. Міненерговугілля України. Харків : Форт, 2017. 760 с.

30. ДСН 3.3.6.042-99. Санітарні норми мікроклімату виробничих приміщень. Київ : МОЗ України, 1999. 24 с.

31. Правила пожежної безпеки в Україні. Затв. Наказом Міністерства внутрішніх справ України від 30.12.2014 р. № 1417. Київ, 2014.

Додаток А

Технічне завдання

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

Тернопільський національний технічний університет імені Івана Пулюя

Факультет комп'ютерно-інформаційних систем і програмної інженерії

Кафедра комп'ютерних систем та мереж

"Затверджую"

Завідувач кафедри КС

_____ Осухівська Г.М.

" 2 " лютого 2026 р.

Комп'ютерна система для налаштування струнних музичних інструментів

ТЕХНІЧНЕ ЗАВДАННЯ

на _*10*_ листках

Вид робіт: Кваліфікаційна робота

На здобуття освітнього ступеня «Бакалавр»

Спеціальність 123 «Комп'ютерна інженерія»

«УЗГОДЖЕНО»	«ВИКОНАВЕЦЬ»
Керівник кваліфікаційної роботи	Студент групи СІ-42
_____ к.т.н., доц. Тиш Є.В.	_____ Ящук І.В.
" 2 " лютого 2026 р.	" 2 " лютого 2026 р.

Тернопіль 2026

1 Загальні відомості

1.1 Повна назва та її умовне позначення

Повна назва теми кваліфікаційної роботи: «Комп’ютерна система для налаштування струнних музичних інструментів».

Умовне позначення кваліфікаційної роботи: КС КРБ 123.201.00.00

1.2 Виконавець

Студент групи СІ-42, факультету комп’ютерно-інформаційних систем і програмної інженерії, кафедри комп’ютерних систем та мереж, Тернопільського національного технічного університету імені Івана Пулюя, Ящук І.В.

1.3 Підстава для виконання роботи

Підставою для виконання кваліфікаційної роботи є наказ по університету (№4/9-188 від 24.04.2026 р.)

1.4 Планові терміни початку та завершення роботи

Плановий термін початку виконання кваліфікаційної роботи – 26.01.2026 р.

Плановий термін завершення виконання кваліфікаційної роботи – 21.06.2026 р.

1.5 Порядок оформлення та пред’явлення результатів роботи

Порядок оформлення пояснювальної записки та графічного матеріалу здійснюється у відповідності до чинних норм та правил ISO, ЕСКД, ЕСПД та ДСТУ.

Пред’явлення проміжних результатів роботи з виконання кваліфікаційної роботи здійснюється у відповідності до графіку, затвердженого керівником роботи. Попередній захист кваліфікаційної роботи відбувається при готовності роботи – наявності пояснювальної записки та графічного матеріалу.

Пред’явлення результатів кваліфікаційної роботи відбувається шляхом захисту на відповідному засіданні ЕК, ілюстрацією основних досягнень за допомогою графічного матеріалу.

2 Призначення і цілі створення системи

2.1 Призначення системи

Апаратно-програмний комплекс, що розробляється, призначений для:

– автоматичного безконтактного зчитування та оцифрування акустичних коливань струн музичних інструментів;

– точного аналізу частотного спектра звуку та виділення основного тону в реальному часі;

– автоматизованого механічного регулювання натягу струн шляхом керування виконавчим електроприводом;

– захисту елементів кілкового механізму та струн від критичного перевантаження й обриву під час налаштування.

2.2 Мета створення системи

Метою кваліфікаційної роботи є розробка, конфігурація та налаштування мобільного автоматизованого апаратно-програмного комплексу для прецизійного налаштування струнних музичних інструментів на основі високопродуктивного 32-бітного мікроконтролера STM32.

2.3 Характеристика об'єкту

Розроблюваний апаратно-програмний комплекс призначено для використання музикантами, технічним персоналом студій звукозапису, гітарними майстрами та акустичними лабораторіями з метою спрощення, автоматизації та значного прискорення процесу підготовки струнних інструментів до експлуатації з дотриманням професійних стандартів точності.

3 Вимоги до системи

3.1 Вимоги до системи в цілому

3.1.1 Вимоги до структури та функціонування системи

Апаратно-програмний комплекс автоматичного налаштування музичних інструментів повинен складатись з:

– Базового блока обробки та виносного електромеханічного виконавчого пристрою.

– Мікропрограми мікроконтролера для аналізу сигналів та керування приводом.

3.1.2 Вимоги до способів та засобів зв’язку між компонентами системи

Основна вимога, яка ставиться до способів та засобів інформаційного обміну - це їх узгодженість, завадостійкість та забезпечення передачі сигналів у реальному часі.

3.1.3 Вимоги до режимів функціонування системи

Для системи визначено два режими функціонування:

– нормальний режим функціонування;

– аварійний режим функціонування.

Основним режимом функціонування є нормальний режим.

Для забезпечення нормального режиму функціонування системи необхідно виконувати встановлені технічні вимоги та суворо дотримуватись умов експлуатації як вбудованого програмного забезпечення мікроконтролера, так і всього комплексу апаратних засобів, які вказані у відповідній технічній документації та інструкціях з експлуатації приладу. У цьому режимі комплекс здійснює безперервний моніторинг аудіосигналу, точне обчислення частоти звучання струни та автоматичне ПІ-регулювання її натягу за допомогою крокового двигуна.

Аварійний режим функціонування системи характеризується відмовою одного або декількох компонентів програмного і технічного забезпечення, таких як втрата зв'язку з цифровим мікрофоном по шині I2S, критичний перегрів силового драйвера, падіння напруги живлення або механічне заклинювання кілкового механізму. При переході в аварійний режим мікроконтролер негайно активує захисні сценарії, повністю блокує генерацію керуючих ШІМ-імпульсів таймера для миттєвої зупинки двигуна з метою запобігання обриву струни, а функції роботи системи продовжують підтримувати роботу модулів самодіагностики та моніторингу стану пристрою в межах базових налаштувань із виведенням відповідного коду помилки на OLED-дисплей.

3.1.4 Вимоги по діагностуванню системи

Для діагностування системи використовуються інструменти діагностування основних процесів комплексу, які вмонтовані в архітектуру вбудованого програмного забезпечення мікроконтролера, а також засоби для діагностики апаратного забезпечення, працездатності сенсорів та силового приводу.

Інструменти повинні забезпечувати зручний інтерфейс для можливості перегляду діагностичних подій, поточних статусів обробки звуку, фіксації помилок інтерфейсів зв'язку та моніторингу процесу виконання програм автоматичного регулювання у реальному часі.

3.1.5 Перспективи розвитку, проектування системи

Дана система може бути розширена завдяки використанню додаткових програмних компонентів, а також передбачено розширення пам’яті для збільшення кількості збережених користувацьких профілів та кастомних налаштувань інструментів.

3.2 Показники призначення

Система повинна передбачати можливість масштабування. Можливості масштабування повинні забезпечуватися засобами використовуваного базового програмного і технічного забезпечення, зокрема завдяки наявності резервних GPIO-виводів мікроконтролера STM32, модульності архітектури коду та можливості програмної декомпозиції обчислювальних алгоритмів.

3.2.1 Вимоги до надійності

Система повинна забезпечувати працездатність та відновлення своїх функцій при виникненні наступних ситуацій:

– при збоях в системі електропостачання апаратної частини (короткочасні просадки напруги павербанку або розряджання акумулятора);

– при помилках в роботі апаратних засобів (тимчасова втрата зв'язку з цифровим MEMS-мікрофоном по шині I2S або відсутність відгуку від дисплея);

– при помилках, пов'язаних з програмним забезпеченням (апаратні виключення ядра мікроконтролера, збої при обробці подвійної буферизації DMA або переповнення стеків обчислень CMSIS-DSP).

Для захисту апаратури від стрибків напруги і комутаційних завад, що генеруються обмотками крокового двигуна при комутації фаз драйвером, повинні застосовуватися фільтри колій живлення та згладжувальні електролітичні конденсатори високої ємності.

3.3 Вимоги до безпеки

Зовнішні елементи технічних засобів комплексу, що перебувають під напругою, повинні мати захист від випадкового дотику за рахунок їх розміщення в захисному ізольованому корпусі, а самі допоміжні лабораторні технічні засоби на етапі налагодження та прошивки повинні мати занулення або захисне заземлення.

Система електроживлення повинна забезпечувати захисне вимикання при перевантаженнях і коротких замиканнях в колах навантаження (зокрема, в колах живлення драйвера та обмоток крокового двигуна), а також аварійне ручне вимикання пристрою користувачем шляхом механічного розриву лінії живлення від автономного джерела.

Загальні вимоги пожежної безпеки повинні відповідати нормам на побутове електрообладнання та електронні прилади. У разі пожежі матеріали, з яких виготовлено корпус та деталі виконавчого механізму, не мають виділяти отруйні гази і дим. Після зняття електроживлення має бути доступне застосування будь-яких засобів пожежогасіння.

3.3.1 Вимоги до експлуатації, технічного обслуговування, ремонту і зберігання компонентів системи

Мікроклімат в приміщеннях повинен відповідати нормам виробничого мікроклімату по ДСН 3.3.6.042-99:

– температуру повітря в межах від +10°С до +35°С;

– відносну вологість повітря при 25°С в межах від 30% до 80%;

– атмосферний тиск 760±25 мм рт. ст.

Періодичне технічне обслуговування і тестування технічних засобів повинні включати обслуговування і тестування всіх використовуваних засобів, цифрового MEMS-мікрофона, мікроконтролера, інтерфейсів передачі даних, плати силового драйвера з кроковим двигуном, автономного джерела живлення.

На підставі результатів тестування технічних засобів повинні проводитися аналіз причин виникнення виявлених дефектів і прийматися заходи по їх ліквідації.

3.4 Вимоги до захисту інформації від несанкціонованого доступу

Система повинна забезпечувати захист від несанкціонованого доступу на рівні не нижче встановленого вимогами, що пред'являються до категорії 1Д по класифікації документа, що діє, “Автоматизовані системи. Захист від несанкціонованого доступу до інформації. Класифікація автоматизованих систем”.

Компоненти підсистеми захисту від НСД повинні забезпечувати:

– ідентифікацію користувача (вибір індивідуального профілю майстра або музиканта через меню керування за допомогою інкрементального енкодера);

– перевірку повноважень користувача при роботі з системою (контроль доступу до зміни системних констант, таблиць еталонних частот та коефіцієнтів регулятора);

– розмежування доступу користувачів.

Рівень захищеності від несанкціонованого доступу засобів обчислювальної техніки, що здійснюють обробку конфіденційної інформації (калібровочних констант, параметрів фільтрації звуку та призначених для користувача пресетів строю), повинен відповідати вимогам класу захищеності згідно вимогам документу “Засоби обчислювальної техніки. Захист від несанкціонованого доступу до інформації. Показники захищеності від несанкціонованого доступу до інформації”.

3.4.1 Вимоги по збереженню інформації при аваріях

Інформація, при виникненні аварійних ситуацій повинна бути збережена на резервних носіях.

3.4.2 Вимоги по стандартизації і уніфікації

Система повинна відповідати вимогам ергономіки і зручності користування за умови комплектування високоякісним обладнанням (обчислювальним модулем мікроконтролера, цифровими датчиками, електромеханічним приводом та засобами графічної індикації), що має необхідні сертифікати відповідності і безпеки.

3.4.3 Вимоги до функцій (завдань), що виконуються системою:

– забезпечення зручного інтерфейсу;
– забезпечення зберігання даних та профілів налаштування;
– автоматичний пошук та аналіз інформативних параметрів сигналу;
– забезпечення високої швидкодії.

4 Вимоги до документації

Документація повинна відповідати вимогам ЄСКД та ДСТУ

Комплект документації повинен складатись з:

– пояснювальної записки;
– графічного матеріалу:
  а) Структурна схема.
  б) Блок схема роботи комп’ютеризованої системи.
  в) Схема електрична принципова.
  г) Схема електрична функціональна.

*Примітка: У комплект документації можуть вноситися зміни та доповнення в процесі розробки.

5 Стадії та етапи проектування

Таблиця 1 – Стадії та етапи виконання кваліфікаційної роботи бакалавра

№ етапу	Назва етапу виконання кваліфікаційної роботи	Термін виконання
1	*Розробка технічного завдання*	*26.01 – 02.02*
2	*Робота над першим розділом «Аналіз технічного завдання»*	*03.02 – 15.02*
3	*Робота над другим розділом «ПРОЄКТНА ЧАСТИНА»*	*20.04 – 25.04*
4	*Робота над третім розділом «ПРАКТИЧНА ЧАСТИНА»*	*26.04 – 05.05*
5	*Робота над четвертим розділом «Безпека життєдіяльності, основи охорони праці»*	*07.05 – 25.05*
6	*Оформлення пояснювальної записки і графічного матеріалу*	*26.05 – 7.06*
7	*Перевірка на академічний плагіат, перевірка керівником та консультантами*	*8.06 – 14.06*
8	*Попередній захист кваліфікаційної роботи бакалавра*	*15.06 – 21.06*
9	*Захист кваліфікаційної роботи бакалавра*	*25.06*

6 Додаткові умови виконання кваліфікаційної роботи

Під час виконання кваліфікаційної роботи у дане технічне завдання можуть вноситися зміни та доповнення.

Додаток Б

Перелік елементів

Поз. позначення	*Найменування*	*Кіл.*	*Примітка*
	Мікросхеми та модулі		
U2	STM32F411CEU6	1	
U3	MAX98357A 3006 Підсилювач звуку	1	
U4	DRV8825 Драйвер двигуна	1	
J2	KY-040 Енкодер поворотний	1	
MK1	INMP441 Цифровий MEMS-мікрофон	1	
LCD1	SSD1306 0.96 OLED дисплей	1	
LM1	DC-DC Перетворювач напруги LM2596	1	
P1	З'єднувач штировий лінійний, PLS-4	1	Порт ST-Link
P2	XT60 male	1	Підключення акб
M1	Двигун NEMA17	1	
LS1	Динамік CVS-1508	1	

Зм.	Арк.	№ докум.	Підпис	Дата	КСКП 123.201.00.00 ПЕ		
Розробив		Ящук І.В.			Комп'ютерна система для налаштування струнних музичних інструментів Перелік елементів	Літ	Арк. 75 / Аркушів 1
Перевірив		Тиш Є.В.					
Резензент		Цуприк Г.Б.					
Н. Контр.		Луцик Н.С.					
Зав. каф.		Осухівська Г.М.					

Додаток В

Лістинг програмного коду автоматичного тюнера

```
#include "stm32f4xx_hal.h"
#include "arm_math.h"
#include <math.h>

/* Обов'язкові константи конфігурації системи */
#define AUDIO_BUFFER_SIZE  2048   /* Розмір половини буфера
    для обробки */
#define DMA_BUFFER_SIZE    (AUDIO_BUFFER_SIZE * 2)
#define SAMPLE_RATE        16000  /* Частота дискретизації
    аудіо шини I2S */

/* Налаштування коефіцієнтів контуру зворотного зв'язку */
const float32_t Kp = 15.0f;       /* Пропорційний коефіцієнт
    регулятора */
const float32_t Ki = 0.5f;        /* Інтегральний коефіцієнт
    регулятора */
float32_t I_Sum = 0.0f;           /* Накопичувана сума
    інтегратора */

/* Еталонні частоти стандартного гітарного строю (E2, A2, D3,
    G3, B3, E4) */
const float32_t TARGET_FREQS[6] = {82.41f, 110.00f, 146.83f,
    196.00f, 246.94f, 329.63f};
volatile uint8_t selected_string_idx = 5; /* Типова цільова
    струна (перша, E4) */

/* Буфери прямого доступу до пам'яті та цифрової обробки */
int32_t i2s_dma_rx_buffer[DMA_BUFFER_SIZE];
float32_t audio_processing_buffer[AUDIO_BUFFER_SIZE];

/* Прапорці планувальника завдань реального часу */
```

```
volatile uint8_t dma_block_ready = 0;
volatile uint32_t dma_sub_buffer_offset = 0;

/* Зовнішні периферійні дескриптори з main.c */
extern I2S_HandleTypeDef hi2s2;
extern TIM_HandleTypeDef htim2;

/* Прототипи функцій цифрового сигнального процесора та
    приводу */
void Process_Audio_Data(void);
float32_t Calculate_Pitch(float32_t* audio_buffer, uint32_t
    buffer_size);
void Update_Motor_PID(float target_freq, float current_freq);

/**
  * @brief  Фоновий запуск прийому та основний суперцикл
    програми
  */
void Tuner_App_Init(void)
{
    /* Запуск циклічного фонового DMA-перенесення аудіоданих
    по шині I2S */
    HAL_I2S_Receive_DMA(&hi2s2, (uint16_t*)i2s_dma_rx_buffer,
    DMA_BUFFER_SIZE / 2);

    while (1)
    {
        /* Очікування сигналу від переривання DMA про
    заповнення блоку пам'яті */
        if (dma_block_ready)
        {
            Process_Audio_Data();
            dma_block_ready = 0;
        }
    }
```

```
}

/**
  * @brief  Колбек переривання DMA: заповнена перша половина
    буфера
  */
void HAL_I2S_RxHalfCpltCallback(I2S_HandleTypeDef *hi2s)
{
    dma_sub_buffer_offset = 0;
    dma_block_ready = 1;
}

/**
  * @brief  Колбек переривання DMA: заповнена друга половина
    буфера
  */
void HAL_I2S_RxCpltCallback(I2S_HandleTypeDef *hi2s)
{
    dma_sub_buffer_offset = AUDIO_BUFFER_SIZE;
    dma_block_ready = 1;
}

/**
  * @brief  Модуль первинної підготовки, децимації та
    фільтрації звуку
  */
void Process_Audio_Data(void)
{
    float32_t dc_sum = 0.0f;

    /* Етап 1: Витягування 24-бітних корисних даних з 32-
    бітних контейнерів I2S */
    for (uint32_t i = 0; i < AUDIO_BUFFER_SIZE; i++)
    {
```

```
    int32_t raw_sample =
i2s_dma_rx_buffer[dma_sub_buffer_offset + i];
    /* Нормалізація значень у формат з плаваючою крапкою
*/
    audio_processing_buffer[i] = (float32_t)(raw_sample >>
8);
    dc_sum += audio_processing_buffer[i];
}

/* Етап 2: Програмне видалення постійної складової (DC
Offset) */
float32_t dc_offset = dc_sum /
(float32_t)AUDIO_BUFFER_SIZE;
for (uint32_t i = 0; i < AUDIO_BUFFER_SIZE; i++)
{
    audio_processing_buffer[i] -= dc_offset;
}

/* Етап 3: Програмний цифровий ФНЧ ковзного середнього для
придушення ВЧ завад */
for (uint32_t i = 1; i < AUDIO_BUFFER_SIZE - 1; i++)
{
    audio_processing_buffer[i] =
(audio_processing_buffer[i-1] +

audio_processing_buffer[i] +

audio_processing_buffer[i+1]) / 3.0f;
}

/* Етап 4: Передача очищеного масиву математичному ядру */
float32_t detected_frequency =
Calculate_Pitch(audio_processing_buffer,
AUDIO_BUFFER_SIZE);
```

```
    /* Етап 5: Запуск адаптивного формування керуючих
    імпульсів */
    Update_Motor_PID(TARGET_FREQS[selected_string_idx],
    detected_frequency);
}

/**
  * @brief  Алгоритм визначення фундаментальної частоти
    методом автокореляції
  *         з використанням CMSIS-DSP та параболічної
    інтерполяції
  */
float32_t Calculate_Pitch(float32_t* audio_buffer, uint32_t
    buffer_size)
{
    /* Створення результуючого вектора згортки подвійної
    довжини */
    float32_t corr_buffer[2 * AUDIO_BUFFER_SIZE];

    /* Виклик низькорівневої апаратно-оптимізованої функції з
    CMSIS-DSP */
    arm_correlate_f32(audio_buffer, buffer_size, audio_buffer,
    buffer_size, corr_buffer);

    uint32_t max_lag = 0;
    float32_t max_val = 0.0f;
    /* Початок сканування з безпечним зсувом від нульової
    затримки */
    uint32_t start_index = buffer_size + 20;

    /* Пошук першого значущого локального максимуму періоду
    хвилі */
    for (uint32_t i = start_index; i < (buffer_size * 2) - 1;
    i++)
    {
```

```
    if (corr_buffer[i] > corr_buffer[i-1] &&
corr_buffer[i] > corr_buffer[i+1])
    {
        if (corr_buffer[i] > max_val)
        {
            max_val = corr_buffer[i];
            max_lag = i;
        }
    }
}

/* Пороговий фільтр амплітуди для відсікання акустичних
шумів */
if (max_lag == 0 || max_val < 0.1f)
{
    return 0.0f;
}

/* Зчитування трьох точок екстремуму для проведення
інтерполяції */
float32_t y1 = corr_buffer[max_lag - 1];
float32_t y2 = corr_buffer[max_lag];
float32_t y3 = corr_buffer[max_lag + 1];

/* Розрахунок дробового математичного зсуву вершини
параболи */
float32_t delta = (y1 - y3) / (2.0f * (y1 - 2.0f * y2 +
y3));

/* Визначення точного уточненого періоду сигналу */
float32_t true_lag = (float32_t)(max_lag - buffer_size) +
delta;

/* Фінальний перерахунок часового періоду в частоту в
Герцах */
```

```
    float32_t final_frequency = (float32_t)SAMPLE_RATE /
    true_lag;

    return final_frequency;
}

/**
  * @brief  Реалізація ПІ-регулятора швидкості та напрямку
    крокового двигуна
  */
void Update_Motor_PID(float target_freq, float current_freq)
{
    /* Якщо сигнал відсутній, негайно знеструмлюємо обмотки
    мотора */
    if (current_freq < 10.0f)
    {
        HAL_TIM_PWM_Stop(&htim2, TIM_CHANNEL_1);
        I_Sum = 0.0f;
        return;
    }

    /* Розрахунок поточної абсолютної похибки частоти строю */
    float error = target_freq - current_freq;

    /* Програмна зона нечутливості (точність досягнута -
    успішна зупинка) */
    if (fabs(error) < 0.5f)
    {
        HAL_TIM_PWM_Stop(&htim2, TIM_CHANNEL_1);
        I_Sum = 0.0f;
        /* Переведення виконавчого блоку в безпечний режим
    спокою */
        return;
    }
```

```
/* Керування силою та знаком напрямку обертання валу
(вывід DIR) */
if (error > 0.0f)
{
    HAL_GPIO_WritePin(GPIOB, GPIO_PIN_12, GPIO_PIN_SET);
/* Натягування (DIR = 1) */
}
else
{
    HAL_GPIO_WritePin(GPIOB, GPIO_PIN_12, GPIO_PIN_RESET);
/* Послаблення (DIR = 0) */
}

/* Накопичення інтегральної складової з захисним
обмеженням анти-віндапу */
I_Sum += error * 0.02f; /* 0.02c - дискретність виклику
контуру регулювання */
if (I_Sum > 50.0f)  I_Sum = 50.0f;
if (I_Sum < -50.0f) I_Sum = -50.0f;

/* Розрахунок вихідного значення двокомпонентного ПІ-
алгоритму */
float pid_output = (Kp * fabs(error)) + (Ki *
fabs(I_Sum));

/* Безпечне обмеження граничних швидкостей для захисту
струн */
if (pid_output > 2000.0f) pid_output = 2000.0f;
if (pid_output < 100.0f)  pid_output = 100.0f;

/* Динамічний перерахунок виходу ПІД в значення періоду
таймера ARR (ШІМ) */
uint32_t new_arr = (uint32_t)(84000000 / (84 *
pid_output));
```