

Власник документу:  
Бевза Олег Миколайович

ID перевірки:  
1004031653

Дата перевірки:  
14.06.2020 20:41:26 EEST

Тип перевірки:  
Doc vs Internet + Library

Дата звіту:  
14.06.2020 20:56:00 EEST

ID користувача:  
90740

Назва документу: 2020-bachelor-EDD\_Natalych\_muz\_syntezator\_fch

ID файлу: 1004044647 Кількість сторінок: 32 Кількість слів: 11786 Кількість символів: 83427 Розмір файлу: 111.21 KB

## 2.49% Схожість

Найбільша схожість: 0.59% з джерело [https://en.m.wikipedia.org/wiki/Synthesizer\\_demo](https://en.m.wikipedia.org/wiki/Synthesizer_demo)

2.1% Схожість з Інтернет джерелами

25

Page 34

0.76% Текстові збіги по Бібліотеці акаунту

22

Page 34

## 0% Цитат

Не знайдено жодних цитат

## 0% Вилучень

Вилучений текст відсутній

## Підміна символів

Заміна символів

5

Наталичу Миколі Ярославовичу

## Система керування музичним синтезатором

### АНОТАЦІЯ

В дипломному проєкті розроблено систему керування музичним інструментом на базі роботизованої руки. Розглянуто історію створення синтезаторів та принципи синтезу звуку. Описано досягнення в сфері робототехніки та застосування роботів в сфері музики. Описано принцип роботи ROS та подібних середовищ. Розглянуто модульний та інтеграційний методи розробки роботів. Досліджено основні музичні аспекти та принцип роботи лінійного двигуна. Обґрунтовано використання модульного методу створення роботів. Показано всі його переваги та недоліки. Підібрано відповідні комплектуючі для створення музичного роботу та перевірено їх на сумісність для застосування модульного методу. Для розробки програмного забезпечення було розглянуто платформу для програмування Arduino ID та особливості використання портів Arduino Uno R3. Описано основні функції, які використовуються для програмування сервоприводів. Розглянуто принцип підключення периферійних пристроїв та їх налаштування за допомогою коду. Розраховано час затримки проходження сигналу від головної плати Arduino Uno R3, тобто головного мікроконтролера, до сервоприводів. Досліджено, яку потужність має вся система керування та визначено основні техніко-економічні параметри пристрою.

**Ключові слова:** СЕКВЕНСЕРИ, АВТОМАТИЗОВАНІ ПРОГРАМНІ ІНСТРУМЕНТИ, ROBOT OPERATING SYSTEM, ЧАСТОТНА МОДУЛЯЦІЯ, ШИРОТНО-ІМПУЛЬСНА МОДУЛЯЦІЯ, ІНТЕРФЕЙСИ I2C , USB .

### ANNOTATION

In the diploma project the control system of a musical instrument on the basis of the robotic hand is developed. The history of creation of synthesizers and principles of sound synthesis are considered. Achievements in the field of robotics and the use of robots in the field of music are described. The principle of operation of ROS and similar environments is described. Modular and integration methods of robot development are considered. The main musical aspects and the principle of operation of a linear motor are studied. The use of a modular method of creating robots is substantiated. All its advantages and disadvantages are shown. Appropriate components for creating a musical work were selected and tested for compatibility for the application of the modular method.

For software development, the Arduino ID programming platform and the features of using Arduino Uno R3 ports were considered. Describes the main functions used to program servos. The principle of connecting peripherals and configuring them using a code is considered.

The delay time of the signal from the main board of the Arduino Uno R3, ie the main microcontroller, to the servos is calculated. The power of the whole control system is investigated and the main technical and economic parameters of the device are determined.

**Key words:** SEQUENCERS, AUTOMATED SOFTWARE INSTRUMENTS, ROBOT OPERATING SYSTEM, FREQUENCY MODULATION, WIDTH-PULSE MODULATION, USB INTERFACES.

#### ВСТУП

Великим технологічним проривом людства стало створення роботів. Завдяки своїм інноваціям, робототехніка перетворилась в самостійну наукову сферу. Інвестиції, які вкладено в цю сферу, посприяли швидкому її розвитку.

В теперішній час пристрої робототехніки отримали суттєве розповсюдження головним чином завдяки успішному партнерству між теорією та практикою, що відрізняє сучасну робототехніку від ранньої робототехніки. Цей прогрес є результатом взаємодії між інженерними та науковими спільнотами різних дисципліни.

Перші роботи з'явилися в 40-х роках XX століття. Вони мали великі габарити і були малофункціональні [1-3]. Спочатку їх головними функціями були автоматизація виробництва і робота в небезпечних для людини умовах. З розвитком робототехніки їхні розміри зменшувалися, а функціональність збільшувалась.

Тенденції досліджень в сфері робототехніки стають все більш популярними. Роботів використовують не тільки для виконання складних операцій, які не може виконати людина, а й для розваг. Одним із яскравих прикладів є робот від компанії Sony під назвою AIBO. [4].

В музичному мистецтві також використовують роботів [5-6]. Їх «навчають» грати на різних інструментах, наприклад, скрипка, фортепіано, барабани і багато інших інструментів.

Розробка і програмування роботів є складними задачами. Для спрощення створення роботів прийняли систему ROS (Robot Operating System) – стандартна система для роботів [7]. В системі прийнято наступні методи для створення роботів: інтеграційний і модульний.

Інтеграційний метод містить наступні етапи:

- Складання переліку елементів і їх покупка.
- Інтеграція. Об'єднання компонентів в єдину систему.
- Будівництво робота (цей етап може проходити паралельно з наступним).

- Програмування робота.
- Тестування і корекція виявлених збоїв і помилок.
- Застосування робота до цілей для яких він створений.

Суть модульного методу полягає у об'єднанні сумісних модулів і виключення етапу інтеграції. Важливим етапом методу є перевірка модулів на сумісність перед їх об'єднанням.

Прикладом таких методів розробки є роботизована рука, яку розробили британські вчені. Така рука має обмежений набір рухів, але незважаючи на це здатна на багато. Вона виконує музичний твір на фортепіано переміщуючи тільки кисть руки. Деталі корпусу такої руки були виготовлені за допомогою 3D-принтера.

Дослідники запрограмували роботизовану руку виконувати декілька музичних фраз з використанням таких прийомів як стакато і легато. На виконання складних музичних творів така рука неспроможна, але дослідження в цьому напрямку може надати таким роботам природності в їхніх рухах[8].

Задача наслідування рухам людських рук при грі на клавішних інструментах для роботів є дуже складною. Для максимально точного наслідування, розробники оснащують таких роботів аналогами людських зв'язок, м'язів, нервів, кісток і суглобів. Но, для мінімально наслідування вистачить просто рухомої кисті. Така конструкція не зможе виконувати складні музичні твори, але її основною перевагою є простота в реалізації і в програмуванні, також незначне енергоспоживання і можливість вдосконалення [9].

**Мета дослідження:** розробка системи керування музичним синтезатором на базі роботизованої руки, що характеризується економічною вигідністю та покращенням якості відтворення музики.

Для досягнення мети розглядались наступні задачі:

- Огляд досягнень в сфері робототехніки;
- Аналіз музичних аспектів.
- Огляд систем керування музичними інструментів;
- Дослідження модульного методу для створення роботів
- Вибір елементної бази. Підбір можливих комплектуючих.
- Перевірка на сумісність комплектуючих модульним методом;
- Вибір платформи для створення програмного забезпечення;

**Об'єктом дослідження** є принципи керування роботизованої руки для гри на музичному синтезаторі.

**Предметом дослідження** є система керування роботизованим музичним інструментом на базі Arduino Uno R3.

## РОЗДІЛ 1. ОГЛЯД ІСНУЮЧИХ СПОСОБІВ ПОБУДОВИ МУЗИЧНИХ СИНТЕЗАТОРІВ

### 1.1. Системи керування музичним синтезатором

#### 1.1.1. Історія створення синтезаторів

Синтезатор - електронний музичний інструмент, який генерує звукові сигнали за допомогою таких компонентів як фільтри та низькочастотні осцилятори.

Аналогом першого синтезатора був пристрій RCA Mark II, що з'явився в США в середині 20 століття. Він контролювався перфокартами і складався з сотень вакуумних трубок. На рис. 1.1 зображено діючу модель синтезатора.

Рис. 1.1 Перший синтезатор RCA Mark II

Синтезатор Moog, розроблений Робертом Мугом і вперше проданий у 1964 році, відноситься до новаторських концепцій аналогового синтезу, таких як генератори, керовані напругою, генератори шуму, фільтри та секвенсери [10]. Секвенсер (Sequencer) - це пристрій запису та відтворення послідовності контрольної інформації для електронного музичного інструменту; згодом —

функція записуючого програмного забезпечення, що дозволяє користувачеві зберігати, програвати та редагувати MIDI-інформацію [11].

У 1970 році з'явилися більш дешеві і менші порівняно з модульними синтезаторами стандартизовані синтезатори MiniMoog. Вони являють собою автономні інструменти із вбудованими клавіатурами. Перший серійний синтезатор - Yamaha DX7, був запущений у 1983 році, популяризував цифровий синтез. Тепер програмні синтезатори вбудовуються в окремі мікросхеми на будь-якому електронному пристрої.

Синтезатор Moog був сприйнятий, як інструмент, великою кількістю відомих на той час музичних груп, наприклад, Doors, Grateful Dead, Rolling Stones, Beatles, і Emerson, Lake & Palmer. Найбільшої популярності Moog набув завдяки вокалісту гурту Lake & Palmer, він був першим великим рок-музикантом, який виступав з Moog. Це стало візитною картою його виступів, що привело його гурт до світового визнання.

Раніше, на синтезатор можна було натискати лише одну клавішу за раз, і це можна було використовувати тільки для басових ліній і соло, тобто вони були монофонічними. Прогрес електротехніки, на той час, посприяв створенню поліфонічних синтезаторів. Поява таких синтезаторів в 70-х і 80-х роках внесли значний вплив на всі музичні жанри і задали темп розвитку електронної і хіп-хоп музики.

Синтезатори часто керуються за допомогою електронних або цифрових клавіатур або клавіатур MIDI-контролера [12], які можуть бути вбудовані в блок синтезатора або приєднані через з'єднання, такі як, USB або MIDI. З клавіатури можуть бути задано такі види виразності виконання музичного твору, як чутливість до швидкості та сили натискання клавіш, що забезпечує більш високий контроль над звуком. Інші контролери включають стрічкові контролери, які відстежують рух пальця по чутливій до дотику поверхні; вітроконтролери, що грають аналогічно інструментам з духових інструментів; регулятори, чутливі до руху, подібні до контролерів руху відеоігор; електронні барабанні колодки, що грають аналогічно головкам барабанного набору; сенсорні панелі, які посилають сигнали залежно від положення та зусилля пальців; контролери, призначені для мікротонального налаштування; пристрої з сенсорним екраном, такі як планшети та смартфони [12].

Стрімкий розвиток синтезаторів призвів до серйозних змін в роботі музичної індустрії, це можна порівняти з більш раннім явищем таким як поява звуку в фільмі, що лишило живих музикантів, які супроводжували німі фільми, роботи. Здатність синтезаторів імітувати звуки різних інструментів дозволила замінювати групи музикантів. І тому, на протязі деякого часу Moog був заборонений для використання в комерційній роботі. Ця заборона була прийнята Американською федерацією музикантів.

### 1.1.2. Принципи синтезу звуку

Для синтезу звуку використовують наступні методи: адитивний, субтрактивний, частотної модуляції, фазових викривлень, фізичного моделювання [13].

Адитивний (підсумовувальний) метод створює звуки за рахунок поєднання декількох форм хвиль, як правило, синусоїд, у складений звук [13]. Принцип адитивного синтезу пояснений на рис. 1.2.

Рис. 1.2. Схематичне зображення принципу роботи адитивного синтезу

Субтрактивний (віднімальний) метод використовує осцилятори для генерування форм хвиль, а потім формує їх за допомогою фільтрів для видалення або підвищення конкретних частот. Принцип субтрактивного синтезу пояснений на рис. 1.3. [13]. Субтрактивний метод все ще використовується в різних синтезах, включаючи віртуальний аналоговий синтезатор.

Рис. 1.3. Схематичне зображення принципу роботи субтрактивного синтезу

Пристрій за методом частотної модуляції (FM) (рис.1.4) створює звуки, модулюючи одну форму хвилі з частотою іншої; отримана складна форма хвилі може, в свою чергу, використовуватися для модуляції іншого. Синтез FM може імітувати акустичні звуки, такі як фортепіано, струнні та органи. Синтез FM був надзвичайно успішним у найбільш ранніх цифрових синтезаторах. [13].

Рис. 1.4. Схематичне зображення принципу роботи частотної модуляції

Метод фазових викривлень - це метод, який в основному використовується на синтезаторах Casio CZ. Він замінює традиційну аналогову форму хвилі декількома цифровими формами хвиль, які є складнішими за стандартні хвилі квадратної, синусоїдної та пилоподібної форми. Сигнал цієї форми подається на цифровий фільтр та цифровий підсилювач. Потім звук може бути додатково модифікований за допомогою кільцевої модуляції або шумової модуляції. [13].

Метод фізичного моделювання - це метод синтезу звуку за допомогою моделювання кожної звукової характеристики інструменту, починаючи з гармонік, що складають сам тон, потім додаючи звук резонатора, корпусу інструменту тощо, поки звук не наблизиться до потрібного інструменту [13]. Коли початковий набір параметрів запускається за допомогою фізичного моделювання, генерується імітоване звучання. (рис. 1.5). Таке явище як фізичне моделювання не можна назвати новим в області акустики і синтезу, однак лише з розробкою алгоритму Карплуса-Стронга та збільшенням потужності DSP (Digital signal processing) в кінці 1980-х років комерційні впровадження стали можливими.

Рис. 1.5. Схематичне зображення алгоритму Карплуса-Стронга

Якість та швидкість фізичного моделювання на комп'ютерах покращується із збільшенням потужності обробки.

### 1.1.3 Переваги і недоліки синтезаторів музики

Перевага аналогових синтезаторів полягає у тому, що всі зміни в часі відбуваються плавно. А основним недоліком є високий рівень шуму. В переважній більшості, в аналогових синтезаторах використовуються адитивний і субтрактивний типи синтезу. Звук в таких синтезаторах обробляється і синтезується за допомогою електричних ланцюгів. Найвідоміші на даний час представники аналогових синтезаторів є Mimimoog Voyager LittlePhatty, Prophet '08 [14].

Розвиток у сфері музичних синтезаторів призвів до появи цифрових синтезаторів. В 1971 році було розроблено перший такий синтезатор, він мав назву Con Brio ADS (Advanced Digital Synthesizer). Такий синтезатор був дорогим, оскільки це було новинкою на той час. Пізніше, в 1973 році було розроблено гібридний аналогово-цифровий музичний синтезатор Marlin Synthesiser.

Більш дешевим, а відповідно і більш доступним цифровим синтезатором був Synclavier. Його принцип дії спирався на роботу мікропроцесорів. Мав жорсткий диск, дві клавіатури, комп'ютерного та фортепіанного типів.

Із цих двох видів музичних синтезаторів не можливо виділити якийсь один. Зустрічаються як і аналогові так і цифрові та їх комбінація, тобто, аналого-цифрові [14].

Однак, при грі на музичних інструментах, в тому числі і на синтезаторах, виконавці допускають помилки, навіть досвідчені. Щоб виключити такі ймовірності таких помилок під час гри на музичних інструментах доцільніше використовувати роботів-музикантів. Такі роботи програмуються на чітке виконання рухів і зводить до мінімуму ймовірність допущення помилок при грі на музичному інструменті, але не виключає повністю. Адже, при роботі таких роботів можуть статися збої програмного забезпечення. Але їх ймовірність набагато нижча ніж ймовірність допущення помилки виконавцем.

## **1.2. Створення музичних інструментів на основі робототехніки**

### **1.2.1. Огляд досягнень в сфері робототехніки**

Робототехніка базується на двох суміжних технологіях: цифровому (або числовому управлінні) та дистанційному керуванні.

Технологія числового управління була розроблена наприкінці 40-х - початку 1950-х років для керування осями верстатів за допомогою чисел, кодованих на перфорованій паперовій стрічці або інших носіях.

Перший верстат з цифровим керуванням був продемонстрований у 1952 році в США в Массачусетському технологічному інституті. Подальші дослідження призвели до розробки мови АПТ (автоматизованих програмних інструментів) для програмування верстатів.

Дистанційне керування лежить в основі механічного маніпулятора. Початкові роботи з проектування маніпуляторів можна простежити під час перших досліджень радіоактивних матеріалів, які проводились на початку 1940-х років.

Промислову робототехніку можна вважати поєднанням числової та дистанційної технологій. Числове управління забезпечує концепцію програмованої промислової машини, а дистанційне - механічної руки для виконання корисної роботи. Перший промисловий робот був встановлений у 1961 році для вивантаження деталей під час штампування. Його розробка значною мірою зумовлена зусиллями американців Джорджа К. Девола, винахідника, та Джозефа Ф. Енгельбергера, бізнесмена. Девол створив прототип програмованого маніпулятора, який був запатентований в США в 1961 році. Енгельбергер спільно з Деволом сприяли застосуванню роботів у промисловості та створенню першої корпорації з робототехніки - Unimation, Inc. [15].

На сьогодні більшість роботів, що використовують у виробництві можна поділити на 3 категорії:

- вантажно-розвантажувальні роботи;
- обробка матеріалів;
- складання і перевірка.

Вантажно-розвантажувальні роботи. Область застосування вантажно розвантажувальних роботів включає в себе передачу, завантаження та вивантаження матеріалу. Завданням для передачі є переміщення матеріалів або деталей із одного місця в інше. Більшість із цих задач прості: роботи повинні збирати деталі з одного конвеєру і розміщувати їх на другому, інші операції переміщення є більш складними, наприклад, розміщення деталей в певному порядку, як повинен визначатися роботом [15].

Обробка матеріалів. Промислові роботи цього типу маніпулюють інструментами для виконання завдання технологічного процесу на робочій частині. Приклади таких застосувань можуть бути: точкове зварювання, суцільне зварювання дугою та фарбування розпиленням. Точкове зварювання автомобільних кузовів - одне з найпоширеніших застосувань промислових роботів у США. Для завершення складання основного кузова автомобіля робот розміщує точковий зварювальник проти автомобільних панелей та рамок. Дугове зварювання - це безперервний процес, при якому робот переміщує зварювальний стрижень уздовж шва, який повинен бути зварений. Розпилювальний спрей передбачає маніпуляцію пістолетом для напилення фарбою над поверхнею предмета, який потрібно покрити. Інші операції в цій категорії включають шліфування, полірування та маршрутизацію, при яких обертовий шпindel служить інструментом робота [15].

Складання і перевірка. Третя область застосування промислових роботів - це складання та перевірка. Очікується, що використання роботів в складанні збільшиться через велику вартість ручної праці. Оскільки роботи є програмованими, однією із стратегій при складальних роботах є створення декількох стилів продукції партіями, перепрограмуючи роботів між партіями. Альтернативна стратегія полягає у створенні суміші різних стилів продукту в одній збірній комірці, вимагаючи від кожного робота в комірці визначити



стиль продукту під час його надходження, а потім виконати відповідне завдання для цього блоку [15].

Дизайн виробу є важливим аспектом роботизованої збірки. Задовільні для людини методи складання не обов'язково підходять для роботів. Наприклад, використання гвинта і гайки в якості способу кріплення, легко виконується людиною, але така ж операція вкрай складна для однорукого робота. Конструкції, в яких компоненти слід додавати з одного напрямку за допомогою фіксаторів та інших одноетапних процедур кріплення, дозволяють зробити роботу набагато простіше за допомогою автоматизованих та роботизованих методів складання [15].

### 1.2.2. Роботи в музиці

У музичній індустрії було досягнуто значного прогресу, завдяки синергічній взаємодії між роботами та людьми. Одним із яскравих прикладів цього є музична група роботів The Trons, (рис.1.6). Вона була створена в 2008 році Грегом Локком.

Рис. 1.6. Музичний гурт роботів The Trons

Ще одним прикладом є рок-група роботів Compressorhead, (рис. 1.7). Вона створена в 2013 році британським артистом Френком Барнсом. Група складається із 6 роботів:

- Stickboy – грає на барабанній установці. Має 4 руки і 2 ноги. В руках закріплені барабанні палички. Побудований в 2007 році;
- Stickboy молодший – грає на хай-хет, тобто на тарілках зі стійкою, які використовуються в барабанній установці;
- Hellga Tarr – робот гітарист, розроблений як жіночий персонаж;
- Fingers – робот гітарист, має дві руки на яких розміщено 78 пальців;
- Bones – робот бас-гітарист, має дві руки на яких розміщено по 4 пальці. Також, має гусеничну платформу, за допомогою якої переміщається по сцені;
- Mega-Wattson – робот вокаліст, має зуби, які рухаються і гусеничну платформу, за допомогою якої переміщається по сцені.

Їхній перший альбом Party Machine вийшов в 2017 році. Ще однієї їх особливістю є те, що вони грають на звичайних інструментах, тобто не налаштованих спеціально для гри роботів [16].

Рис. 1.7. Музичний гурт роботів Compressorhead

## 1.3. Підтримка розробки програмного забезпечення музичних роботів

### 1.3.1. Опис системи ROS (Robot Operating System)

Операційна система для роботів (ROS) - гнучка основа для написання програмного забезпечення робота [17]. Це колекція інструментів та бібліотек, які мають на меті спростити завдання створення робота на широкому спектрі робототехнічних платформ. ROS залежить від базової операційної системи. ROS вимагає багато функціональності від операційної системи. Велика популярність ROS обумовлена відносною простотою в користуванні і легкодоступністю.

ROS розпочав своє існування як особистий проєкт Кінана Вайрбека та Еріка Бергера, в Стенфорді, як спроба усунути найпоширеніші проблеми робототехніки.

Основними проблемами на той час було:

- занадто багато часу витрачається повторному впровадженню програмної інфраструктури, необхідної для побудови складних алгоритмів робототехніки (в основному, драйверів до датчиків і виконавчих механізмів, а також зв'язку між різними програмами всередині одного робота);

- багато часу, витрачалось для створення досконалих програм з робототехніки, які базувалися на цій інфраструктурі.

Навіть всередині однієї і тієї ж організації для кожного нового проєкту було здійснено повторне написання драйверів та систем зв'язку.

Для того, щоб вирішити цю проблему, Ерік і Кінан створили в 2006 році програму в Стенфорді під назвою Stanford Personal Robotics Program, з метою створити середовище, яке дозволило б процесам спілкуватися один з одним, а також деякі інструменти. Все це середовище використалось для розробки коду для робота "Персональний робот", як тестовий блок і приклад іншим. Вони побудували 10 таких роботів і надали їх університетам, щоб вони могли розробляти програмне забезпечення на основі своїх програм [17].

У 2009 році був випущений перший дистрибутив ROS: ROS Mango Tango, який також називається ROS 0.4. Випуск 1.0 цієї дистрибуції був запущений майже через рік у 2010 році. З цього моменту команда ROS вирішила назвати дистрибуції за типами тварин, а саме черепах [10]. Отже, було здійснено наступні дистрибуції та дати випуску:

- Vox Turtle в 2010;
- ROS C-Turtle в 2010;
- Diamond Back в 2011;
- ROS Electric Emys в 2011;
- ROS Fuerte Turtle в 2012;
- ROS Groovy Galapagos в 2012.

Основна причина розробки цієї системи полягає у тому, що створити справді надійне програмне забезпечення роботів загального призначення важко. З точки зору робота, проблеми, які здаються людьми тривіальними, часто сильно різняться між окремими завданнями та умовами. ROS розбито на більш ніж 2000 пакетів, кожен пакет забезпечує спеціалізовану функціональність [17].

ROS забезпечує функціональність апаратної абстракції, драйверів пристроїв, зв'язок між процесами на кількох машинах, інструменти для тестування і візуалізації та багато іншого. Також, потрібна операційна система з відкритим кодом, щоб операційну систему та ROS можна було змінити відповідно до вимог програми.

Операційні системи, такі як Windows 10 та Mac OS X можуть поставити певні обмеження щодо користування. Це може призвести до створення певних рамок в процесі розробки. Для нормальної роботи, без цих

рамок, можна запускати ROS на Linux, зокрема Debian та Ubuntu, оскільки ROS має дуже гарну підтримку з цими операційними системами.

Ключовою особливістю ROS є спосіб запуску програмного забезпечення та спосіб його передачі даних, що дозволяє розробляти складне програмне забезпечення, не знаючи, як працює певне обладнання. ROS забезпечує спосіб з'єднання мережі процесів (вузлів) з центральним вузлом. Вузли можна запускати на декількох пристроях, і вони підключаються до цього центрального вузла різними способами [17].

Розробники можуть зібрати складну систему, підключивши існуючі рішення для невеликих проблем. Спосіб реалізації системи дозволяє:

1. Мультиплексування виходів декількох компонентів на один вхід для іншого компонента, що дозволяє паралельно вирішувати різні задачі.

2. Підключення компонентів, виготовлені на різних мовах програмування, просто впроваджуючи належні роз'єми до системи обміну повідомленнями, полегшуючи розробку програмного забезпечення, підключаючи існуючі модулі від різних розробників.

Створення вузлів через мережу пристроїв, не турбуючись про те, де виконується код, та впровадження системи міжпроцесорного зв'язку (IPC) та віддаленого виклику процедур (RPC) [17].

ROS має підтримку декількох платформ і дозволяє з'єднання між процесами на декількох пристроях через однорангові з'єднання. Конструкція дозволяє підтримувати будь-які мови програмування.

### 1.3.2 Подібні середовища

Ідея ROS не була новою. Для спільноти робототехніки вже були доступні інші подібні проекти. Наприклад, Player / Stage - одна із найвідоміших у лінійці з відкритим кодом, та URBI - у лінійці закритих систем. Навіть Open-R, система, розроблена Sony, яка працювала на ранніх роботах Aibo 1999 року [17].

Одним з лідерів дослідницького проекту «Player / Stage» був Брайан Геркі, який пізніше почав працювати у Willow Garage, щоб розробити ROS, а зараз є генеральним директором компанії Open Robotics, компанії, яка стоїть за розробкою ROS. Зі свого боку, URBI була професійною системою, яку очолював Жан-Крістоф Бейлі, яка дуже добре працювала, але не могла конкурувати з вільністю ROS.

URBI була подібна до ROS. URBI мав стільки ж інструментів для налагодження та стільки ж документації, скільки і ROS. Але URBI не вдалося конкурувати з ROS тому, що за його використання потрібно було платити великі гроші [17].

### 1.4. Інтеграційний і модульний методи розробки музичних роботів

Для розробки роботів використовують два методи: інтеграційний і модульний. Інтеграційний метод складається з шести, а модульний з п'яти етапів.

В інтеграційному методі перший етап являє собою складання переліку елементів, для побудови робота, та їх покупка. Далі іде етап інтеграції. Цей

етап являє собою складання елементів в окремі модулі. Наступний етап – це будівництво робота, поєднання окремих модулів в єдину систему. Цей етап може проходити паралельно з наступним. А наступним етапом є програмування робота. На цьому етапі робота програмують на виконання певних задач, для яких він створюється. Передостанній етап – це етап тестування і корекція виявлених збоїв і помилок. Останній етап являє собою вже готового робота, який коректно виконує поставлені перед ним задачі.

Модульний з складається з п'яти етапів. Він включає такі ж етапи, що і інтеграційний крім одного, етапу інтеграції, тобто використовувати готові модулі. Из-за цього, на використання цього методу витрачається менше часу. Але, перед початком роботи, всі ці модулі потрібно перевірити на сумісність.

#### **Постановка задачі дослідження. Висновки до розділу 1**

1. На базі порівняння підходів до створення музичних інструментів у вигляді синтезаторов та роботизованою системою керування музичним синтезатором обрано останній підхід оскільки він має значно більші функціональні можливості відтворення музичних творів.

2. На базі аналізу платформ підтримки розробки програмного забезпечення музичних роботів обрано система ROS, оскільки вона характеризується відносною простотою в користуванні і легкодоступністю.

3. Огляд методів конструювання роботів показав, що модульний метод є має переваги для конструювання музичних роботів.

4. На теперішній час вартість музичних роботів залишається досить високою, тому в цій роботі основною задачею є поставлена зменшення вартості при збереженні якості відтворення музики.

## **РОЗДІЛ 2. ПРИНЦИП РОБОТИ ТА РОЗРОБКА СТРУКТУРНОЇ СХЕМИ СИСТЕМИ КЕРУВАННЯ МУЗИЧНИМ СИНТЕЗАТОРОМ**

### **2.1 Музичні аспекти**

Для реалізації системи керування музичним синтезатором, потрібно брати до уваги деякі музичні аспекти. Основну увагу потрібно звернути на тривалість нот. На рис. 2.1 зображено часто використовувані тривалості.

Рис. 2.1. Приклад цілої і до восьмої ноти

Другим важливим аспектом є аплікатура для гри на фортепіано. Який палець використовувати для відтворення певної ноти у музичному творі, часто залежить від особистих уподобань. Тим не менш, є деякі вимоги і правила аплікатури. На рис. 2.2 зображено нумерацію пальців правої руки, для гри на фортепіано [18].

Рис. 2.2. Нумерація пальців правої руки

Для новачків аплікатуру записують так, як зображено на рис. 2.3, тобто над відповідними нотами. Також, на цьому рисунку зображено темп, який вимірюється в кількості ударів за хвилину (в даному прикладі темп 60 ударів на хвилину) та музичний розмір.

Рис. 2.3. Приклад запису з ключем, аплікатурою та музичним розміром

Ще один цікавий аспект - це відмінність стакато від легато. Стакато – це уривчасте звучання нот, тобто коротке і різке натискання на клавішу.

Легато, з іншого боку, описує більш плавний стиль гри, де клавіші натискаються спокійніше, і перехід до наступної клавіші відбувається більш ПЛАВНО.

## 2.2 Принцип руху «руки». Система керування лінійним двигуном

Опис руху руки використовується для створення траєкторії переміщення робота із початкової конфігурації з'єднань в цільову конфігурацію з'єднань.

Найбільшою проблемою руху є уникнення перешкод. Без жодних перешкод можна було б інтерполювати від однієї спільної конфігурації до конфігурації цілі. Маючи перешкоди, необхідно перевірити наявність будь-яких зіткнень та знайти траєкторії, які їх уникають.

Перешкоди визначаються в декартовому просторі, а сам рух відбувається в спільному просторі. Внаслідок цього необхідно використовувати пряму кінематику для визначення декартових координат для конфігурації з'єднання, щоб перевірити наявність будь-яких перешкод.

З цієї причини сучасні методи планування руху зазвичай покладаються на певну форму випадковості потенційного поля. Це був одним з перших методів популяризації такого підходу. Пізніші алгоритми, засновані на алгоритмі швидкого вивчення випадкового дерева (RRT), стали популярним методом планування руху. RRT-Connect - приклад варіанту цього алгоритму.

RRT-Connect покращує оригінальний алгоритм, будуючи два дерева з метою їх з'єднання. Одне дерево будується з конфігурації запуску, а одне дерево будується з цільової конфігурації [19]. Деревя будуються з наступними кроками:

1. Нарисувати випадкову вибірку  $q$  із простору пошуку (рис. 2.4.).

Рис. 2.4. Розширення дерева в RRT

2. Пошук найближчого вузла  $q_{near}$  в дереві
3. Створення нового вузла  $q_{new}$ , пройшовши невеликий крок від  $q_{near}$  ДО  $q$ , якщо зіткнення не виявлено.

Ця операція з розширенням дерев використовується в повному алгоритмі RRT-Connect наступним чином:

1. Розширити дерево.
2. Розгорнути інше дерево, використовуючи новостворений вузол  $q_{near}$  як ціль.
3. Повторення кроку 2, поки не буде знайдено зіткнення або підключення дерев. Якщо зіткнення було знайдено, повернутися до кроку 1

RRT-Connect - це ефективний метод пошуку планів руху через його ймовірнісний характер. Потрібно перевіряти лише підмножину простору пошуку, поки обидва дерева не з'єднані і не знайдеться план руху від початкової до цільової точки.

Результатом планування руху є траєкторія. Траєкторія складається з декількох точок маршруту. Кожна маршрутна точка повинна, принаймні, містити інформацію про час, в який повинна бути досягнута точка, а також

положення кожного стику. Крім того, шляхові точки також можуть містити інформацію про швидкість і прискорення кожного з'єднання.

### 2.2.1 Кінематика руки

Рух руки під час гри на фортепіано є важливим фактором. В загальному, рука і її пальці повинні бути розміщені над клавішами, тому при переході від одної клавіші до іншої потрібно лише незначні рухи.

Можна також натискати клавіші великим пальцем. Однак великий палець розташований інакше, ніж решта пальців. Він коротший, так що коли інші пальці витягнуті і нависають над клавішами, великий палець все ще знаходиться далеко від них. Потрібна інша поза пальця, щоб усі пальці, включаючи великий палець, могли розташовуватися над клавішами одночасно. Для того, щоб це виконати потрібно просто зробити всі пальці, крім великого, напівзігнутими, тобто кутова поза. Тоді відстань великого пальця до фортепіано зменшиться і він буде вже нависати над клавішами [20].

Чорні клавіші на фортепіано розташовані інакше, ніж білі. Як і у випадку з великим пальцем, для перестановки пальців над чорними клавішами знадобиться більше часу, якщо всі пальці витягнуті та розміщені над рядом білих клавіш. Для натискання на чорні клавіші потрібно просто вирівняти той палець, яким потрібно натискати. Витягнуті пальці можна легко розмістити над чорним рядом клавіш, а розміщені під кутом пальці – над рядом білих клавіш. Відповідно, перестановка не потрібна і послідовне натискання на клавіші стає швидшим.

### 2.2.2 Лінійний двигун

В автоматичному виробничому процесі лінійний механізм переміщення є найбільш розповсюдженим, він має ступінь свободи від одного до кількох вимірів. Існує кілька видів конструкцій механізму – на базі непрямої і прямої передачі.

Непряма передача є найбільш широко використовуваною формою, при цьому обертовий рух перетворюється в лінійний за допомогою відповідного механізму. Найпоширенішими прикладами непрямої передачі є шестерні механізми та конвеєрна стрічка.

Прикладами прямої передачі є пневматичний пристрій, гідравлічний пристрій та лінійний двигун. У пневматичних та гідравлічних пристроях використовується рідина як середовище для передачі енергії. Це не тільки ускладнює складність систему, але й призведе до складності у обслуговування.

Лінійний двигун - **електродвигун**, у якого один з елементів магнітної системи розімкнений і має розгорнуту обмотку, створює магнітне поле, а інший взаємодіє з ним і виконаний у вигляді направляючої, що забезпечує лінійне переміщення рухомої частини двигуна. Лінійний двигун з електромагнітним полем як носієм здатний значно зменшити складність системи та питання технічного обслуговування. Для промисловості завжди було важливим висока точність, швидке виробництво та спрощення

технічного обслуговування, і лінійний двигун відповідає всім цим критеріям. [21].

Тому для забезпечення руху механічної руки обираємо лінійний двигуном. Для розрахунку швидкості, прискорення та часу, необхідного для такого руху, буде встановлена певна відстань механічного руху руки. Це налаштування призведе до більш точної швидкості лінійного двигуна під час руху «руки».

Цей розрахунок заснований на залежності між швидкістю та прискоренням. Рух лінійного двигуна відрізняється від робочого підходу кругового обертового двигуна таким чином, що розрахунок швидкості обертання та прискорення або уповільнення базується на принципі лінійного руху.

В момент лінійного руху час, необхідний для прискорення нерухомого об'єкта до максимальної швидкості, називається часом прискорення. Коли об'єкт рухається з максимальною швидкістю, це називають постійною швидкістю. Час, необхідний для сповільнення об'єкта від цієї постійної швидкості до нерухомого стану, називається часом уповільнення.

Лінійна структура лінійного двигуна призведе до фіксованому ходу руху так, що будуть існувати криві швидкості прискорення, постійної швидкості та швидкості уповільнення руху.

### 2.3 Вибір сервоприводу

Для того, щоб пальці роботизованої руки мали можливість рухатись, їх потрібно під'єднати до сервоприводу. Сервопривод – це привід з керуванням через від'ємний зворотній зв'язок, він має датчик (швидкості, положення) і блок керування приводом.

Принцип роботи сервопривода полягає в тому, що значення керуючого параметру подається на вхід. Наприклад, положення, в яке потрібно встановитися. Далі, блок керування порівнює значення, яке подали, зі значенням на своєму датчику і потім на основі цього порівняння робить певну дію. На рис.2.5 наведено будову сервопривода [22]. В загальному ціна варіюється від 40 і до 4000 гривень.

Рис. 2.5. Будова сервопривода

#### 2.3.1 Основні параметри сервоприводу

Для вибору сервопривода, потрібно спочатку визначитись з його типом:

- мікро – призначені для встановлення в мікромоделі;
- міні – створені для невеликих моделей, в основному використовуються в автомобілях;
- стандартні – найбільш розповсюджений тип, мають хорошу потужність.
- великі – найменш використовувані, зустрічаються використовуються у великих моделях

Великих відмінностей в схемах різних типів сервоприводів, немає. У всіх них однаковий принцип дії [23].

Наступне, на що потрібно звернути увагу, це тип двигуна. Найпоширенішим є стандартний тип мотору – це колекторний двигун постійного струму. Він має металевий сердечник і розділений на секції, на які накладена обмотка, що створить якір, по сторонам якого розміщені магніти. Якір потрібен для того, щоб на дати двигуну різко збільшувати обороти і так сам їх різко зменшувати [23]. В такому типі двигуна використовуються два типи сердечника триполюсні і п'ятиполюсні. Найкраще використовувати другий тип сердечників, адже вони забезпечують більш високий момент і плавність ходу [23].

Другий тип двигуна це двигуни Coreless. Це також двигун колекторного типу, але його відмінність від попереднього типу полягає в тому, що в своїй будові він не має сердечника. Замість нього встановлений центральний магніт навколо якого обертається обмотка. В такого двигуна відсутні секції, а отже він має меншу вагу, більшу динаміку і працює без ривків [23].

Останній тип двигунів – Brushless. Цей тип двигунів я відносно новим. Основною перевагою його є відсутність, так званих, щіток для електричних пристроїв. Це сприяє тому, що в таких моторах велика швидкість обертання при малому споживанні струму, також такі двигуни мають довгий строк служби [23].

При виборі сервоприводу, також необхідно звертати увагу на матеріал, з якого вироблені деталі (шестерні редуктора, вихідний вал). Не дивлячись на короткий строк служби, найпоширенішим матеріалом є пластик. Але найнадійнішим варіантом є металеві або карбонові шестерні. Вони є більш міцнішими ніж пластик і тому будуть слугувати довше. Але потрібно враховувати і те, що вони важчі ніж пластик [23].

Важливими параметра є і швидкість повороту і зусилля на валу. Швидкість характеризує за який час вихідний вал виконає оберт на 60 градусів. Запис швидкості сервоприводу наступний, 0.35с/60, це означає, що вихідний вал, за 0.35 секунд зробить оберт на 60 градусів. Зусилля на валу показує яку вагу може втримувати сервопривод на важелю довжиною в 1см. Приклад того, як розраховується зусилля на валу наведено на рис. 2.6.

Рис. 2.6. Розрахунок зусилля на валу

### 2.3.2 Принцип роботи

Щоб вказати сервоприводу в яке положення встановитися потрібно надіслати керуючі сигнали. Такі сигнали мають форму прямокутних імпульсів з однаковою частотою і різною тривалістю (рис.2.7). Частота сигналів складає 20мс, а тривалість змінюється від 0.8 мс до 2.2 мс.

Рис. 2.7 Імпульси керування

На рис. 2.8 зображено структурна схема роботи сервоприводу. Її принцип полягає в тому, що від ресивера надходить імпульсний сигнал на компаратор К. Це призводить до активації генератора опорного імпульсу ГОП. Тривалість опорного вхідного імпульсу залежить від положення потенціометра. Коли вихідний вал знаходиться у середньому положенню, то довжина сигналу становить 1.5 мс, а якщо положення крайнє ліве, або крайнє



праве – 0.8 мс і 2.2 мс відповідно. Керуючий сигнал і опорний імпульс порівнюються компаратором, який розраховує їх різницю імпульсів. Саме ця величина визначає чи збігається очікуване і фактичне положення вихідного валу. Цей показник зберігається в якості потенціалу в ПВЗ [24].

Рис. 2.8. Структурна схема роботи сервоприводу, де ГОП – генератор опорного імпульсу, К- компаратор, ПВЗ – пристрій вибірки-зберігання, ЕД - електродвигун

#### **2.4 Структурна схема системи керування музичним синтезатором**

Система керування музичним синтезатором складається з семи модулів: мікроконтролер, контролер сервоприводів і 5 сервоприводів. На рисунку 2.9 наведена структурна схема системи керування музичним синтезатором.

Принцип роботи такої схеми полягає в тому, що від мікроконтролера, по шині I2C передачі даних, передається сигнал на контролер сервоприводів. Далі Контролер сервоприводів подає керуючий імпульс на відповідний сервопривід. Далі сервопривід стає в положення, які їм здає запрограмований контролер сервоприводів. При подачі наступних імпульсів процес повторюється.

Рис. 2.9. Структурна схема системи керування музичним синтезатором

Шина I2C використовує лише 4 сигнали - 5В Vcc, GND, тактовий вивід SCL і вивід даних SDA. По лінії SDA передається сигнали керування всіма сервоприводами.

#### **Висновки до розділу 2**

1. На базі розгляду музичних аспектів показано, що основну увагу потрібно звернути на правильну реалізацію тривалості нот а також можливості реалізації режимів стакато та легато.

2. Показано, що створення траєкторії переміщення роботизованої руки із початкової конфігурації з'єднань в цільову конфігурацію з уникненням перешкод доцільно на вести на базі алгоритму швидкого вивчення випадкового дерева (RRT-Connect). Кожна маршрутна точка траєкторії повинна містити інформацію про час, в який повинна бути досягнута точка, а також положення кожного стику. Крім того, шляхові точки також можуть містити інформацію про швидкість і прискорення кожного з'єднання

3. На базі вивчення кінематики руки, оптимальна позиція пальців - витягнуті пальці розміщенні над чорним рядом клавіш, а коли потрібно натискати на білі, руку потрібно перевести у кутову позу. В цьому разі перестановка не потрібна і послідовне натискання на клавіші стає швидшим

4. Досліджено основні параметри сервопривода, за якими його потрібно вибирати для певних задач. В загальному ціна сервоприводів варіюється від 40 і до 4000 гривень. Також розглянуто принцип його дії та структурна схема.

### **РОЗДІЛ 3. РОЗРОБКА СХЕМИ ЕЛЕКТРИЧНОЇ ПРИНЦИПОВОЇ СИСТЕМИ КЕРУВАННЯ МУЗИЧНИМ СИНТЕЗАТОРОМ**

#### **3.1 Дослідження модульного методу для створення роботів**

Модульний метод являє собою набір різних модулів на рівні ланок і з'єднань, які можна роз'єднати і зібрати в різні конфігурації через

стандартизовані з'єднувальні інтерфейси. Змінюючи конфігурацію модулів, виникає можливість створювати різних роботів, щоб задовольнити вимоги до поставлених задач [25].

Переваги цього методу полягають у швидкому будівництві роботів, які відповідають вимогам певної технологічної задачі. Також, завдяки цьому методу ціна таких роботів стає набагато нижчою. Всі модулі для побудови вибираються з тих, які вже розроблені. І це прискорює процес їх виготовлення. Після розробки таких роботів їх можна і далі доповнювати іншими модулями, що робить їх універсальними [25].

До недоліків можна віднести необхідність розробки механічної системи конкретних модулів із-за обмеженої кількості деталей, які можна використати в певному модулі. Іноді це може призвести до зниження функціональних можливостей роботи.

Нехай розміщення і складня модулів в модульному методі називається конфігурацією складання. Набір модулів фіксується під час планування і таким чином єдиними діями для оптимізації конструкції є повторна комбінація і перестановка модулів. Для простих розроблених наборів модулів, загальна кількість можливих складальних конфігурацій може бути порівняно вручну і цей набір може бути легко протестований у відповідності до поставлених задач.

Оскільки конструкція модуля стає все більш складнішою і універсальною, можлива кількість складальних конфігурацій роботи сильно зростає, то потрібно розробити ефективний і систематичний метод для пошуку оптимальної конфігурації [26].

Одним із таких методів є генетичні алгоритми. Особливістю його є те, що в нього дискретна природа набору складальної конфігурації. Паредіс і Хосла [27] розглянули конструювання роботів на основі задач для роботів з фіксованою конфігурацією. В своєму формулюванні всі конструкторські параметри, такі як довжина ланок, кути скручування, приймають неперервні значення і відповідно можуть використовуватися традиційні методи оптимізації.

Модульна робототехніка забезпечує унікальну перевагу в плані можливості переконфігурації, над традиційними технологіями по виготовленню роботів.

Для ремонту та обслуговування таких конструкцій зазвичай потрібен окремий підготовлений персонал для кожної моделі, а отже, збільшення середнього споживання ресурсів у галузях. Наступна фаза роботизованих конструкцій розробляється в перспективі складання модульних блоків для підвищення простоти ремонту, заміни, управління тощо.

Дослідники на пізніших етапах розвитку ввели концепцію автоматизації, самовідновлення, преналаштовування тощо, створюючи модульних самореконфігуровуючих роботів (MSRR). Багато застосувань, таких як управління великими спорудами [28], космічна розвідка [29], спостереження у військових зонах, управління катастрофами часто

вимагають адаптації та здібності до самовідновлення, і MSRR створений саме для цього. Основна відмінність конструкцій MSRR від модульних робототехнічних конструкцій є здатність конструкцій приєднувати або від'єднуватись відповідно до вимог застосування з мінімальним втручанням людини.

Зростаючий попит на багаторазові, обмежені в просторі та багатоцільові рішення для реальних програм є чудовим стимулом для досліджень у галузі MSRR. Дослідники в області MSRR запропонували численні рішення за допомогою різних конструкцій прототипів, алгоритмів зв'язку, методів координації та дисперсії, використовуючи вибрані тестові сценарії [30].

Більшість розроблених до цього часу робототехнічних модулів використовували обмежені ресурси, наявні на момент розробки, і мали обмежені можливості завдяки повільному технологічному просуванню в області датчиків, прототипування обладнання, інтерфейсів зв'язку тощо. Різні підходи, застосовані дослідниками для затвердження конструкцій та прототипів, роблять відносно порівняння робототехнічних модулів досить складним процесом і представляють проблеми при кількісній оцінці та розумінні переваг і недоліків різних конструкцій [30].

Детальний огляд широкого спектру рішень для модульних конструкцій зовнішніх структур, фізичних інтерфейсів між модулями, протоколів зв'язку, технологій для стикування та вирівнювання, алгоритмів руху координат, характеристик середовища тощо, забезпечує краще розуміння новим дослідникам щодо переваг та недоліків попередніх розробок, щоб можна було забезпечити кращі рішення з використанням новітніх технологій.

Апаратні архітектури в MSRR розвиваються разом з технологіями. Перший прототип, розроблений в дослідженнях MSRR, - це SEBOT, що складається з неоднорідних окремих одиниць, здатних зв'язуватися між собою і з того часу дослідження спрямовувалося на розробку систем, здатних утворювати різні структури, що імітують біологічні організми.

Юм та ін. [29, 31] запропонував дві класифікації модульних робототехнічних систем: класифікація на основі структур, сформованих MSRR, і класифікація на основі стратегій реконфігурації. Гілпін і Рус [32] додали ще кілька підкласів за категоріями структур, включаючи дослідження з мікроелектронних механічних систем та інші останні розробки в MSRR до моменту публікації.

### **3.2. Вибір елементної бази. Підбір можливих комплектуючих**

Як було сказано вище, рух забезпечується сервоприводами, сервоприводи керує контролер сервоприводів, а на контролер сервоприводів подається сигнал з мікроконтролера.

В цьому проєкті використовується мікроконтролер ATmega328P, який вбудований в плату Arduino Uno R3. Ця плата має 14 цифрових портів, 6 аналогових входів, USB порт [33]. В табл. 3.1 наведено деякі основні параметри цього контролера.

Таблиця 3.1

## Основні характеристики Arduino Uno R3

Щоб запрограмувати цю плату використовують USB-порт, а вбудований стабілізатор напруги дозволяє використовувати різні елементи живлення з великим діапазоном від 6 і до 14 В. На рис. 3.1 зображено плату Arduino Uno R3 з назвою усіх пінів. Дану плату обрано за такі переваги: простота в програмуванні та використанні і відносно низька ціна. Коштують такі плати від 125 до 140 гривень.

Щоб вибрати потрібний сервопривід, потрібно враховувати максимальну допустиме зусилля на валу, швидкість повороту і матеріал з якого зроблені деталі. Звертаючи увагу на ці параметри було вибрано сервопривід Tower Pro MG90S Servo 14G.

Рис. 3.1 Назва пінів Arduino Uno R3

Нижче, в табл. 3.2 наведено основні параметри. Такий сервопривід має металічні шестерні, пластиковий корпус і кут повороту на 180 градусів. Це повністю задовольняє всі вимоги до сервоприводу.

Таблиця 3.2

## Основні параметри сервоприводу Tower Pro MG90S Servo 14G

Також, ще одним параметром, за яким вибирається сервопривід, є вартість. Ціна цього сервоприводу складає від 70 до 150 грн/шт. На рис. 3.2 зображено сервопривід Tower Pro MG90S Servo 14G.

Рис. 3.2 Сервопривід Tower Pro MG90S Servo 14G

Оскільки в даному проєкті використовується 5 сервоприводів, то під час передачі даних від плати Arduino Uno R3 до сервоприводів виникає ймовірність утворення значних затримок. Тому, для виконання даної задачі, краще використовувати ШІМ драйвер.

Для даного проєкту вибрано PCA9685. Ця плата - це 16-канальний 12-бітний ШІМ драйвер. Тобто, завдяки ній можна використовувати одразу 16 сервоприводів.

В табл. 3.3, наведено деякі параметри контролера PCA9685. На рис.3.3 зображено всі виводи PCA9685.

Таблиця 3.3

## Параметри контролера PCA9685

Рис. 3.3. Виводи контролера PCA9685

Контролер має відносно не складну будову. На рис. 3.4 наведено блок-схему цього контролера. Для виконання цього проєкту було використано цей контролер із-за його особливостей таких, як шина передачі даних I2C з підтримкою режиму Fast-mode Plus, програмований стан виходів, шумоподавляючі фільтри на входах і також його вартість. Ціна такого контролера коливається від 70 до 90 гривень, що робить його дуже доступним.

Рис. 3.4. Блок-схема контролера

Живлення буде подаватися від п'яти акумуляторів Ni-MH AA які будуть розміщуватися в батарейному відсіку на п'ять батарейок типорозміру AA. Ємність таких акумуляторів складає 2000 мА·год, напруга 1.2В, 1000 циклів

заряджання і має низький ступінь саморозрядження [34]. Ціна за п'ять таких акумуляторів складає 230-250 гривень.

З вище сказаного зрозуміло, що в даному проєкті використовується модульний метод розробки роботів, тобто використовується готові модулі. Для з'єднання цих модулів потрібно використати провідники-джампери типу «папа-папа».

При виборі елементної бази основна увага зверталась на ціну та якість. Ціна всіх елементів становить від 800 до 1250 гривень, але при цьому вони мають хорошу якість. Наприклад, використовуються сервоприводи з металевими елементами (шестерні редуктора, вихідний вал), а подібний за параметрами, але з пластиковими елементами, сервопривід коштує на 2-3% дешевший. Але дешевший не завжди кращий, адже, сервопривід з пластиковими елементами вийде з ладу в 1.5-2 рази швидше, що робить його економічно не вигідним.

Що стосується плати Arduino Uno R3, вибрано саме її тому, що вона є дуже проста у використанні має низьку ціну. Якщо порівнювати з аналогами, наприклад, RobotDun UNO, ціна така ж але менший об'єм оперативної пам'яті, що призведе до виникнення затримок. Тобто, самим оптимальним варіантом є Arduino Uno R3.

При виборі контролера сервоприводів важливим було звернути увагу на діапазон частоти в якому працює ШІМ. Оскільки потрібно враховувати економічну вигідність цього елемента, то було вибрано PCA9685. За такі характеристики, які він має, його вибір є економічно вигідним.

### **3.3.Перевірка на сумісність комплектуючих модульним методом**

Для перевірки на сумісність комплектуючих потрібно звертати увагу на те при якій напрузі встановлюються логічні рівні «0» та «1», напругу живлення всіх модулів, інтерфейси та протоколи обміну.

Оптимальна напруга живлення всіх модулів складає від 5.5 до 6, з цього впливає, що встановлення джерела живлення від п'яти акумуляторів Ni-MH AA є правильним рішенням. Навіть якщо врахувати допуск в 5%, то ці акумулятори ідеально підходять.

Для стабільної і передачі даних потрібно, щоб мікроконтролери використовували однакову напругу логічних рівнів. На платі Arduino Uno R3 логічний рівень «0» становить 0 В, алогічний рівень «1» становить 5 В. В контролері сервоприводами PCA9685 значення напруги логічних рівнів співпадає з значеннями на платі Arduino Uno R3.

Також, для передачі даними потрібно використовувати деякий інтерфейс передачі даних. В цьому проєкті використовується інтерфейс I2C. Він підтримується як і на Arduino Uno R3 так і на PCA9685.

Щоб сервоконтролери працювали правильно, на них потрібно подати плюс джерела живлення, землю, і керуючі імпульси. Керуючі імпульси являють собою ШІМ. Оскільки в цьому проєкті використовується 5 сервоприводів, то потрібно, відповідно, 5 портів з ШІМ. Для цього можна використати плату Arduino Uno R3, але така кількість одночасно підключених

сервоприводів може призвести до утворення затримок, тому раціональніше використати контролер сервоприводів PCA9685. В цьому контролері є 16 каналів ШІМ з частотою, яка налаштовується від 24 до 1526 Гц.

Отже, перевірка модулів показала, що всі вони є сумісними. Адже, в усіх модулях використовується однакова напруга живлення, яка задається акумуляторами Ni-MH AA, однакова напруга логічних рівнів. Для керування сервоприводами застосовується контролер з регульованою частотою ШІМ у великому діапазоні.

#### **Висновки до розділу 3**

1. Модульна робототехніка забезпечує унікальну перевагу в плані можливості переконфігурації, над традиційними технологіями по виготовленню роботів.

2. Обґрунтовано вибір елементної бази, досліджено параметри кожного з елементів і проаналізовано їх цінову політику. Аналіз виявив, що ціна і параметри є кращими ніж у аналогічних пристроїв.

3. Проведена перевірка на сумісність. Вона виявила, що всі модулі є сумісними. Адже, напруга живлення всіх модулів є однаковою, як і напруга логічних рівнів, використовується інтерфейс передачі I2C, який підтримується на всіх модулях, а яких він використовується. Контролер сервоприводів PCA9685 має 16 каналів ШІМ частотою, яка налаштовується від 24 до 1526 Гц. Це забезпечує надійну роботу сервоприводів.

### **РОЗДІЛ 4. РОЗРОБКА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ**

В даному розділі буде розглядатись розробка програмного забезпечення для системи керування музичним синтезатором. Головною задачею цього розділу є налаштування програмного зв'язку всіх модулів, а також забезпечення їх коректного функціонування.

#### **4.1 Середовище програмування Arduino IDE**

Всі плати Arduino програмуються в середовищі Arduino IDE. Для того, щоб програмувати саме Arduino Uno потрібно вибрати в налаштуваннях потрібну платформу. Це можна зробити так, як вказано на рис. 4.1.

Рис. 4.1. Вибір плати в Arduino IDE

При підключенні, плата прошивається відповідним завантажувачем і визначається системою в автоматичному режимі. Зв'язок мікроконтролера з комп'ютером створюється стандартним протоколом STK500. Також, на платі є роз'єм ICSP для внутрішньосхемного програмування.

Для програмування мікроконтролера використовуються скетчі. Скетч являє собою нескінчений цикл. В цьому циклі з контактів знімається значення і за допомогою команд формується керування зовнішніми приладами, які під'єднанні до плати. Для роботи з певною периферією потрібно створювати спеціальні функції, але є можливість підключити зразу готові бібліотеки. Вже готовий скетч завантажується на плату за допомогою USB-з'єднання.

#### **4.2 Особливості використання портів Arduino Uno**

Для зв'язку з комп'ютером або іншими мікроконтролерами на платі є інтерфейс передачі даних, На контактах «0» і «1» в контролері є UART послідовний інтерфейс. Він виконує роль програматора і транслюється через USB. Це дозволяє передавати дані до комп'ютера через COM-порт. Налаштування конфігурації контролера зразу має стандартні драйвери USB-COM, тому для підключення не потрібно ніяких додаткових драйверів.

За допомогою моніторингу послідовної шини **Serial Monitor**, середовище Arduino IDE отримує і відправляє дані від самої плати. При обміні даними можна спостерігати мигання світлодіоду «0» і «1» виводів. А при використанні UART-інтерфейсу вони не моргають.

Також плата може взаємодіяти через UART-інтерфейс не тільки апаратним но і програмним методом. Для цього в середовищі Arduino IDE використовується бібліотека **SoftwareSerial**. На даній платі, також передбачені контакти основних інтерфейсів взаємодії з периферією SPI і I2C. Оскільки в даному проекті використовується послідовна шина передачі I2C, то для її використання потрібно використовувати контакти «4» (SDA) і «5» (SCL).

Під час налаштування конфігурації мікроконтролера функція передає значення на заданий контакт. Всі контакти мають п'ятивольтову логіку, тобто, коли на контакті встановлюється логічна одиниця, то напруга на цьому контакті становить 5 В. Кожен контакт плати має резистори навантаження номіналом від 20 до 50 кОм і можуть пропускати до 40 мА. За замовчуванням вони всі відключенні.

Також, на платі є два додаткових контакти: AREF і RESET. AREF – видає опорну напругу для вбудованого АЦП. Для використання цього контакту під час програмування мікроконтролера, використовується функція `analogReference()`. RESET – являє собою кнопку, яка перезапускає мікроконтролер. При натисканні на кнопку, подається низький рівень сигналу, що призводить до перезапуску мікроконтролера.

#### **4.3 Програмування в середовищі Arduino IDE**

Для підключення сервоприводів до Arduino створена бібліотека «Servo». Для плат Arduino завдяки цій бібліотеці можна підтримувати до 12 сервоприводів.

За допомогою цієї бібліотеки здійснюється програмне керування сервоприводами. Вони керуються такими функціями:

Приклад програми для підключення і роботи з сервоприводами:

##### **4.3.1. Підключення PCA9685**

Для початку роботи з PCA9685 потрібно підключити бібліотеки і визначити його об'єкт. Потім, визначити тривалість імпульсів або ШІМ-вихід драйверу. Оскільки в цьому контролері сервоприводів виходи 12-бітні, тобто роздільна здатність 4096 кроків. Таким чином, мінімальна тривалість імпульсу в 0,5 мс або 0 градусів положення відповідала б 102 крокам, а максимальна тривалість імпульсу – 2,5 мс або 180 градусів – 512 кроків.

В розділі налаштування потрібно визначити тактову частоту I2C, встановити адресу драйвера і встановити частоту 50 Гц. В тілі головної

функції `void loop()` використовуються функції `setChannelPWM ()` і `rwmForAngle ()`, які керують сервоприводами, тобто встановлюють в певне положення. Для налаштування сервоприводів через контролер сервоприводів потрібно використовувати прошивку `Servo_Evaluator`.

Приклад програми для підключення і роботи з сервоприводами через контролер сервоприводів PCA9685:

#### 4.4 Апаратна частина

Кожен сервопривід з'єднувався з фалангою пальців за допомогою волосіння. Її діаметр не повинен бути більшим ніж 0,17 мм. На сервопривід встановлювалися пластичне кріплення, в крайні положення, тобто в  $0^\circ$ , вкручувалися по одному гвинту. Приклад таких кріплень наведено на рис. 4.2

Рис. 4.2. Пластикові кріплення

Волосінь фіксувалась на гвинтах, тобто при рухові сервоприводу за годинниковою стрілкою, волосінь натягується в одну сторону, при обертанні проти годинникової стрілки, волосінь натягується - в іншу.

Далі сервоприводи кріпляться на визначених місцях, при цьому волосінь натягується таким чином, щоб крайні положення вихідного валу сервоприводу відповідали крайнім положенням пальця, тобто зігнутий або прямиий.

Потім, сервоприводи підключалися до драйверу PCA9685, драйвер в свою чергу підключається до плати Arduino Uno R3 за допомогою чотирьох з'єднувальних проводів. Для нормального функціонування, на контролер сервоприводів подавалось додаткове живлення з батарейного відсіку, в якому знаходяться 5 акумуляторів.

#### 4.5 Розрахунки

##### 4.5.1 Швидкодія

Підчас визначення швидкодії системи керування музичним синтезатором можна не враховувати час, який потрібно на завантаження коду програми до мікроконтролера. Щоб розрахувати затримку передачі сигналу між Arduino Uno R3 і PCA 9685, потрібно враховувати довжину провідника яким вони з'єднані і матеріал цього провідника. В даному випадку довжина провідника складає 30см і виготовлений він із міді. Розрахунок можна провести за формулою:

де  $l$  - це довжина провідника [м],  $\epsilon$  - діелектрична проникність [ $\text{Om}\times\text{m}$ ],  $v$  – швидкість струму у провіднику [м/с].

Провівши розрахунки за формулою (4.1) було визначено час затримки між Arduino Uno R3 і PCA 9685 і він дорівнює приблизно 0,13 секунди. Далі, використовуючи (4.1), потрібно порахувати затримку між контролером сервоприводів і сервоприводами. Оскільки провідники використовуються однакові, то затримка також дорівнює 0,13 секунди. Для визначення значення повної затримки потрібно скористатися формулою

де  $t_1$  – час затримки між Arduino Uno R3 і PCA 9685,  $t_2$  – час затримки між PCA 9685 і сервоприводами.



Підставивши всі значення, з (4.2) випливає, що загальна затримка дорівнює 0,26 секунди. Цю затримку потрібно враховувати при написанні програмного забезпечення.

Приклад коду з врахуванням затримки:

#### 4.5.2 Потужність

Для розрахунку потужності потрібно знати деякі параметри модулів, які використовуються, а саме струм і напруга. Розрахувати потужність можна за наступною формулою

де  $U$  – напруга,  $I$  – струм.

Проаналізувавши характеристики плати Arduino Uno R3 визначено, що при 5,5 вольтах напруги сила напруги на кожному із пінів дорівнює 40 мА. Підставивши ці значення у формулу (4.3) результат виходить 0,22 Вт. В контролері сервоприводами PCA 9685 при напрузі 5,5 В, струм – 25мА. Підставляючи ці значення у (4.3), потужність контролера сервоприводами складає 0,13 Вт. В сервоприводі Tower Pro MG90S Servo 14G при напрузі 5,5 В, струм дорівнює 80 мА при рухові і 800 мА при блокуванні (фіксація положення). Відповідно при рухові потужність складає 0,44 Вт, а при блокуванні 4,4 Вт.

Для того, щоб порахувати загальну потужність системи керування музичним синтезатором потрібно скористатися формулою

де  $P_{Arduino}$  – потужність Arduino,  $P_{PCA9685}$  – потужність контролера сервоприводів,  $P_{Сервопривод}$  – потужність сервоприводу.

Підставляючи всі значення у формулу (4.4), значення загальної потужності дорівнює 0,8 Вт, при роботі сервопривода, і 4,75 Вт, при фіксуванні сервопривода.

#### 4.5.3 Техніко-економічні параметри

Для визначення техніко-економічних параметрів, взято для порівняння подібний пристрій Piano-Playing Robotic Arm [35]. В цьому пристрої також використовується Arduino але іншої серії, а саме Leonardo. Основна їх відмінність - це об'єм оперативної пам'яті, в Leonardo вона більша. Але для системи керування музичним синтезатором, щоб забезпечити достатню швидкість для обробки програмного коду, вистачить і Arduino Uno R3. Ціна Arduino Leonardo коливається від 200 до 250 гривень, що є дорожче від Arduino Uno R3 на 65-90 гривень. Це робить вибір плати серії Leonardo економічно не вигідним.

Контролер сервоприводів який використовується в цьому проєкті співпадає з тим який використовується в Piano-Playing Robotic Arm. Адже цей контролер має широкий діапазон частоти, яку можна налаштувати.

В пристрої Piano-Playing Robotic Arm використовуються сервоприводи FT5519M. Він є більш габаритним і потужнішим ніж Tower Pro MG90S Servo 14G, але швидкість повороту вихідного валу менша. В FT5519M вона складає 0,18 с/60°, а в Tower Pro MG90S Servo 14G - 0.11 с/60°. Також, для їхньої роботи потрібні імпульси різної тривалості, для FT5519M 900-1200 мкс, для іншого – 500 – 2400 мкс. Тобто, Tower Pro MG90S Servo має більший

діапазон, що більш підходить для кращого виконання музичних творів. Наприклад, з таким сервоприводом легато буде виходити більш плавніше, тобто забезпечує більш якісне виконання прийомів.

#### **Висновки до розділу 4**

1. Проаналізовано середовище Arduino IDE для програмування Arduino Uno R3. Для програмування мікроконтролера використовуються скетчі. Скетч являє собою нескінченний цикл. В цьому циклі з контактів знімається значення і за допомогою команд формується керування зовнішніми приладами, які під'єднанні до плати.

2. Досліджено роботу додаткових контактів: AREF і RESET. AREF – видає опорну напругу для вбудованого АЦП. Для використання цього контакту під час програмування мікроконтролера, використовується функція `analogReference().RESET` – являє собою кнопку, яка перезапускає мікроконтролер. При натисканні на кнопку, подається низький рівень сигналу, що призводить до перезапуску мікроконтролера.

3. Проаналізовано особливості підключення контролера сервоприводів. В тілі головної функції `void loop()` використовуються функції `setChannelPWM ()` і `pwmForAngle ()`, які керують сервоприводами, тобто встановлюють в певне положення.

4. Розраховано затримки. Загальна затримка дорівнює 0,26 секунди, вона має великий вплив на створення програмного забезпечення. Якщо її не врахувати, то постійно буде збиватися ритм мелодії.

5. Знайдена загальна потужність системи керування музичним синтезатором, яка дорівнює 0,8 Вт при рухові сервопривода і 4,75 Вт при його фіксації.

6. Проведено техніко-економічне дослідження, яке показало, що система керування синтезатором, яка розглядається в даній роботі, є економічно вигідною та забезпечує якісніше застосування музичних прийомів в порівнянні з аналогом.

#### **ВИСНОВКИ**

При написанні дипломного проекту було розглянуто та опрацьовано багато електронних джерел та технічної наукової літератури, досліджено систему керування музичним синтезатором.

Історія створення синтезаторів розпочинається ще з 50-х років 20 століття, але дана тема актуальна і на даний час, адже такий інструмент як синтезатор, дуже популярний у відомих музикантів і часто використовується на концертах.

Описані в проєкті методи синтезу звуку, такі як адитивний метод, субтрактивний метод, метод частотної модуляції, метод фазових викривлень, метод фізичного моделювання, враховують при створенні різних моделей сучасних синтезаторів.

Детально описано історію створення перших роботизованих механізмів, та їхні методи керування: чисельні (цифрові) та дистанційні. Чисельне керування відбувається за допомогою чисел, кодованих на

перфорованій паперовій стрічці або інших носіях. Дистанційне керування лежить в основі механічного маніпулятора. Поєднання цих двох методів лежить в основі промислової робототехніки.

Робототехніка широко використовується в музичній індустрії. Це дозволило створити ряд музичних гуртів, які створюють та відтворюють музику без музикантів. Наприклад, рок-група роботів Compressorhead або група роботів The Trons.

Для забезпечення функціонування робота створюється відповідне програмне забезпечення. Операційна система для роботів (ROS) - гнучка основа для написання програмного забезпечення робота. Це колекція інструментів та бібліотек, які мають на меті спростити завдання створення робота на широкому спектрі робототехнічних платформ.

Велика популярність ROS обумовлена відносною простотою в користуванні і легкодоступністю, забезпечує функціональність апаратної абстракції, драйверів пристроїв, зв'язок між процесами на кількох машинах, інструменти для тестування і візуалізації та багато іншого.

В дипломному проєкті розкриті музичні аспекти керування синтезатором. Основна увага звертається на тривалість нот, аплікатуру та музичні прийоми стакато і легато.

Для розробки роботів використовують два методи: інтеграційний і модульний. Інтеграційний метод має такі етапи: складання переліку елементів та їх придбання; інтеграція або об'єднання компонентів в єдину систему; будівництво робота (цей етап може проходити паралельно з наступним); програмування робота; тестування і корекція виявлених збоїв і помилок; застосування робота до цілей для яких він створений.

Модульний метод складається з таких самих етапів, що і інтеграційний, але не містить етапу інтеграції. Етап інтеграції займає досить багато часу при створенні роботів. Тому, його виключення призводить до прискорення виконання роботи.

В якості модулів, в даному проєкті, використовувались плата Arduino Uno R3, плата для збільшення ШІМ-виходів PCA9685 та 5 сервоприводів Tower Pro MG90S Servo 14G. Провівши перевірку модулів, було встановлено, що вони є сумісними.

Розглянуто середовище для програмування плати Arduino Uno R3, Arduino IDE, описано функції для керування сервоприводами, досліджено особливості підключення плати PCA9685, та написано програмний код.

Проведено розрахунки: затримки сигналу від Arduino Uno R3 і до сервоприводів; загальної потужності системи керування музичним синтезатором.

Техніко-економічні дослідження, проведені та проаналізовані, підтверджують, що система керування синтезатором, яка розглядається в даному проєкті, є економічно вигідною та забезпечує якісніше застосування музичних прийомів в порівнянні з аналогом.

## SUMMARY

## Control system for music synthesizer

Diplowma project of educational and qualification level "Bachelor" specialty 171 - Electronics, specialization - Electronic devices and devices. Natalich Mykola Yaroslavovich. KPI them. Igor Sikorsky. Faculty of Electronics, Department of Electronic Devices and Devices. DS-61 group. - K .: KPI them. Igor Sikorsky, 2020. - 55p., Ill. 22, table. 3.

**Theme, purpose and nature of the work**

Thesis is devoted to the study of the control system of a music synthesizer.

Purpose: to develop a control system for a music synthesizer based on a robotic arm, which is characterized by cost-effectiveness and improved quality of music playback.

The work is applied.

**Used methods and software products:** modular method of creating robots; calculated to calculate the parameters of the device, software to create software for a musical instrument. The following software products are used: ROS platforms and Arduino ID

**Summary of work:** The first section describes the history of music synthesizers. After analyzing these data, it was found that the analogue of the first synthesizer was the device RCA Mark II, which appeared in the United States in the mid-20th century. It was controlled by punch cards and consisted of hundreds of vacuum tubes.

The Moog Synthesizer, developed by Robert Mugg and first sold in 1964, is one of the innovative concepts of analog synthesis, such as voltage-controlled generators, noise generators, filters and sequencers.

The following methods are used for sound synthesis: additive, subtractive, frequency modulation, phase distortions, physical modeling.

The additive (summing) method creates sounds by combining several waveforms, usually a sine wave, into a compound sound. The subtractive (subtraction) method uses oscillators to generate waveforms and then generates them using filters to remove or increase specific frequencies. The subtractive method is still used in various synthesizers, including the virtual analog synthesizer.

The frequency modulation (FM) device generates sounds by modulating one waveform with the frequency of another; the resulting complex waveform can, in turn, be used to modulate another. FM synthesis can mimic acoustic sounds such as piano, strings and organs. FM synthesis was extremely successful in the earliest digital synthesizers.

The phase distortion method is a method mainly used on Casio CZ synthesizers. It replaces the traditional analog waveform with several digital waveforms that are more complex than standard square, sine, and dust waveforms. The signal of this form is fed to a digital filter and a digital amplifier. The sound can then be further modified by ring modulation or noise modulation.

The physical modeling method is a method of synthesizing sound by simulating each sound characteristic of the instrument, starting with the harmonics

that make up the tone itself, then adding the sound of the resonator, instrument body, etc., until the sound is closer to the desired instrument.

The main advantages and disadvantages of music synthesizers are described. The advantage of analog synthesizers is that all changes over time occur smoothly. And the main disadvantage is the high noise level. In the vast majority, analog synthesizers use additive and subtractive types of synthesis. The sound in such synthesizers is processed and synthesized using electrical circuits.

Advances in the field of robotics are studied. It is based on two related technologies: digital (or numerical control) and remote control.

Numerical control technology was developed in the late 1940s and early 1950s to control machine axes using numbers encoded on perforated paper tape or other media.

The first digitally controlled machine was demonstrated in 1952 in the United States at the Massachusetts Institute of Technology. Further research led to the development of the APT language (automated software tools) for machine programming.

Remote control is the basis of a mechanical manipulator. Initial work on the design of manipulators can be traced during the first studies of radioactive materials, which were conducted in the early 1940s.

Industrial robotics can be considered a combination of numerical and remote technologies. Numerical control provides the concept of a programmable industrial machine, and remote - a mechanical hand to perform useful work.

Robotics has developed not only in industry but also in entertainment. One striking example of this is the music group of robots The Trons. It was created in 2008 by Greg Locke. Also, another example is the rock band Robots Compressorhead. It was created in 2013 by British artist Frank Barnes. The group consists of 6 robots. Their first album Party Machine was released in 2017. Another feature is that they play the usual instruments, ie not configured specifically for the game robots

To simplify software writing use the ROS platform. This platform is a collection of tools and libraries that aim to simplify the task of creating robots on a wide range of robotic platforms. ROS depends on the base operating system. ROS requires a lot of functionality from the operating system. The great popularity of ROS is due to the relative ease of use and ease of access.

ROS provides hardware abstraction functionality, device drivers, multi-machine process communication, testing and visualization tools, and more. Also, you need an open source operating system so that the operating system and ROS can be modified according to the requirements of the program.

A key feature of ROS is the way software is run and the way it is transmitted, which allows you to develop complex software without knowing how certain hardware works. ROS provides a way to connect a network of processes (nodes) to a central node. Nodes can be run on multiple devices, and they connect to this central node in different ways.

ROS supports multiple platforms and allows connections between processes on multiple devices via peer-to-peer connections. The design allows to support any programming languages.

The idea of the ROS was not new. Other similar projects were already available to the robotics community. For example, Player/Stage is one of the most famous in the line of open source, and URBI - in the line of closed systems. Even Open-R, a system developed by Sony that worked on the early works of the 1999 Aibo.

Two methods are used to develop robots: integration and modular. The integration method consists of six and the modular method of five stages.

In the integration method, the first stage is to compile a list of elements to build a robot, and purchase them. Next is the stage of integration. This stage is the assembly of elements into separate modules. The next stage is the construction of the robot, the combination of individual modules into a single system. This stage can take place in parallel with the next. And the next step is programming the robot. At this stage, the work is programmed to perform certain tasks for which it is created. The penultimate stage is the stage of testing and correction of detected failures and errors. The last stage is a ready-made robot that correctly performs the tasks set before it.

The modular method consists of the same stages as the integration method, but does not contain the integration stage. The integration phase takes a long time to create robots. Therefore, its exclusion leads to faster performance.

Thesis reveals the musical aspects of synthesizer control. The main attention is paid to the duration of notes, fingering and musical techniques of staccato and legato.

When selecting possible components, much attention is paid to the servo controller. A servo drive is a drive with negative feedback control, it has a sensor (speed, position) and a drive control unit.

The principle of operation of the servo drive is that the value of the control parameter is fed to the input. For example, the position in which you want to stand. Next, the control unit compares the value provided with the value on its sensor and then on the basis of this comparison performs a certain action.

The type of motor plays an important role in the choice of servo. The most common is the standard type of motor - a DC collector motor. It has a metal core and is divided into sections, which are superimposed on the winding, which will create an anchor, on the sides of which are placed magnets. The anchor is needed in order to dramatically increase the speed of the engine and thus reduce them sharply. This type of motor uses two types of core, three-pole and five-pole. It is best to use the second type of cores, because they provide a higher torque and smoothness.

The second type of engine is Coreless engines. It is also a collector type engine, but its difference from the previous type is that in its structure it does not have a core. Instead, a central magnet is installed around which the winding rotates.

The latest type of engine is Brushless. This type of engine is relatively new. Its main advantage is the absence of so-called brushes for electrical devices. This contributes to the fact that such motors have a high speed with low current consumption, and such motors have a long service life.

The ATmega328P used in the Arduino Uno R3 board will be used as the main microcontroller. This board has 14 digital ports, 6 analog inputs, USB port. A USB port is used to program this board, and the built-in voltage stabilizer allows the use of various batteries with a wide range from 6 to 14 V. Figure 3.1 shows the Arduino Uno R3 board with the name of all pins. This board was chosen for the following advantages: ease of programming and use and relatively low price.

PCA9685 is used to control the servos. This board is a 16-channel 12-bit PWM driver. That is, thanks to it it is possible to use 16 servodrives at once.

The controller has a relatively simple structure. This controller was used for this work due to its features such as I2C data bus with support for Fast-mode Plus mode, programmable output status, noise suppression filters at the inputs and also its cost.

Power will be supplied from five Ni-MH AA batteries, which will be housed in a battery compartment for five AA-size batteries. The capacity of such batteries is 2000 mAh, voltage 1.2V, 1000 charge cycles and has a low degree of self-discharge.

To check the compatibility of components, you need to pay attention to the voltage at which the logic levels "0" and "1", the supply voltage of all modules, interfaces and exchange protocols.

The optimum supply voltage for all modules is 5.5 to 6, which means that installing a power supply from five Ni-MH AA batteries is the right decision. Even with a tolerance of 5%, these batteries are ideal.

For stable and data transmission, it is necessary that the microcontrollers use the same voltage of the logic levels. On the Arduino Uno R3 board, the logic level "0" is 0 V, the logic level "1" is 5 V. In the PCA9685 servo controller, the voltage level of the logic levels coincides with the values on the Arduino Uno R3 board.

You also need to use some data interface to transfer data. This work uses the I2C interface. It is supported on both the Arduino Uno R3 and the PCA9685.

All Arduino boards are programmed in the Arduino IDE. And this work also uses this software environment. When connected, the board is stitched by the appropriate bootloader and is determined by the system automatically. The connection of the microcontroller to the computer is created by the standard STK500 protocol.

Sketches are used to program the microcontroller. A sketch is an endless cycle. In this cycle, the value is removed from the contacts and the commands are used to control the external devices that are connected to the board. To work with certain peripherals you need to create special functions, but it is possible to connect ready-made libraries at once. The finished sketch is downloaded to the board via a USB connection.

The thesis uses a modular method of robot development, ie ready-made modules are used. You need to use papa-papa jumper wires to connect these modules.

When choosing the element base, the main attention was paid to price and quality. The price of all elements is from 800 to 1250 hryvnias, but at the same time they have good quality. For example, this work uses servos with metal elements (gears, output shaft), and similar in parameters, but with plastic elements, the servo is 2-3% cheaper. But cheaper is not always better, because the sulfur drive with plastic elements will fail 1.5-2 times faster, which makes it economically unprofitable.

As for the Arduino Uno R3 board, it was chosen because it is very easy to use and has a low price. Compared to counterparts such as RobotDyn UNO, the price is the same but less RAM, which will cause delays. That is, the most optimal option is the Arduino Uno R3.

When choosing a servo controller, it was important to pay attention to the frequency range in which the PWM operates. Because the cost-effectiveness of this element must be taken into account, PCA9685 was chosen. For the characteristics it has, its choice is cost-effective.

#### **The main technical and operational characteristics and indicators of a musical instrument are determined**

Calculations of the total delay of the music synthesizer control system, which is 0.26 seconds, and the total power, which is 0.8 W when moving the servo and 4.75 W when fixing the position of the servo.

That is, it follows that the device is low power, which provides low power consumption, but all operations are performed with little delay.

To determine the technical and economic parameters, a similar device Piano-Playing Robotic Arm is taken for comparison. This device also uses an Arduino but another series, namely Leonardo. The main difference is the amount of RAM, Leonardo has more. But for a music synthesizer control system, the Arduino Uno R3 is enough to provide enough speed for processing code. The price of the Arduino Leonardo ranges from 200 to 250 hryvnias, which is 65-90 hryvnias more expensive than the Arduino Uno R3. This makes the choice of the Leonardo series board not economically viable.

The servo controller used in this project is the same as that used in Piano-Playing Robotic Arm. After all, this controller has a wide range of frequencies that can be adjusted.

As for the servos, the servos used in this work are more suitable for the use of musical techniques.

#### **Results of work, their novelty and economic efficiency;**

The control system of the music synthesizer is developed, its parameters are investigated, its economic profitability is substantiated and technical and economic parameters are compared with a similar device.



**Recommendations for the use of project results.** Such devices are used mainly for home use, but more powerful and "smarter" types of such devices can be used as musicians on stage.

## Схожість

Схожість із джерелами з Інтернету

25

1	<a href="https://en.m.wikipedia.org/wiki/Synthesizer_demo">https://en.m.wikipedia.org/wiki/Synthesizer_demo</a>	6 Джерело	0.59%
2	<a href="https://rykovodstvo.ru/exspl/59516/index.html?page=10">https://rykovodstvo.ru/exspl/59516/index.html?page=10</a>	3 Джерело	0.35%
3	<a href="https://znaimo.com.ua/%D0%9B%D1%96%D0%BD%D1%96%D0%B9%D0%BD%D0%B8%D0%B9_%D0%B4%D0%B2%D0%B8%...">https://znaimo.com.ua/%D0%9B%D1%96%D0%BD%D1%96%D0%B9%D0%BD%D0%B8%D0%B9_%D0%B4%D0%B2%D0%B8%...</a>		0.3%
5	<a href="https://medium.com/@shubhamsipah00/ros-robot-operating-system-b73df0519a27">https://medium.com/@shubhamsipah00/ros-robot-operating-system-b73df0519a27</a>	2 Джерело	0.29%
6	<a href="https://www.wikizero.com/uk/%D0%A1%D0%B5%D0%BA%D0%B2%D0%B5%D0%BD%D1%81%D0%BE%D1%80">https://www.wikizero.com/uk/%D0%A1%D0%B5%D0%BA%D0%B2%D0%B5%D0%BD%D1%81%D0%BE%D1%80</a>	2 Джерело	0.22%
8	<a href="https://beegreen.com.ua/mikrokontroler-chip-mikroshema-atmega-168pa-mu-qfn-10299">https://beegreen.com.ua/mikrokontroler-chip-mikroshema-atmega-168pa-mu-qfn-10299</a>	4 Джерело	0.14%
9	<a href="https://www.theconstructsim.com/ja/history-ros">https://www.theconstructsim.com/ja/history-ros</a>	4 Джерело	0.14%
11	<a href="https://b-ok.cc/book/5066158/a17e61">https://b-ok.cc/book/5066158/a17e61</a>	2 Джерело	0.07%
12	<a href="http://ifreestore.net/836">http://ifreestore.net/836</a>		0.07%

Схожість по Бібліотеці акаунту

22

4	Студентська робота	ID файлу: 8158620	Institution: Lviv Polytechnic National University	4 Джерело	0.3%
7	Студентська робота	ID файлу: 5946109	Institution: Lviv Polytechnic National University	3 Джерело	0.16%
10	2020-bachelor-EDD_Salamatin_elektronna_harmata_fch	ID файлу: 1004040160	Institution: National Technic	12 Джерело	0.09%
13	Студентська робота	ID файлу: 1003995056	Institution: National Aviation University		0.07%
14	Студентська робота	ID файлу: 1003537241	Institution: National University of Life and Environmental Sciences of...		0.07%
15	Студентська робота	ID файлу: 1003928404	Institution: Ukrainian Catholic University		0.07%