

КВАЛІФІКАЦІЙНА РОБОТА

на здобуття освітнього ступеня

бакалавр

(назва освітнього ступеня)

на тему: *Комп'ютерна система перетворення
голосових повідомлень у текст*

Виконав: студент IV курсу, групи СІс-43
спеціальності 123 «Комп'ютерна інженерія»

(шифр і назва спеціальності)

(підпис)

Козінець Е.С.

(прізвище та ініціали)

Керівник

(підпис)

Луцків А.М.

(прізвище та ініціали)

Нормоконтроль

(підпис)

Луцик Н.С.

(прізвище та ініціали)

Завідувач кафедри

(підпис)

Осухівська Г.М.

(прізвище та ініціали)

Рецензент

(підпис)

Марценко С.В.

(прізвище та ініціали)

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)
« » 2022 р.

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ

на здобуття освітнього ступеня бакалавр
(назва освітнього ступеня)

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

студенту Козінцю Ернесту Сергійовичу
(прізвище, ім'я, по батькові)

1. Тема роботи Комп'ютерна система перетворення голосових повідомлень у текст

Керівник роботи Луцків Андрій Мирославович, к.т.н., доцент
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

Затверджені наказом ректора від «23» березня 2022 року № 4.7-180

2. Термін подання студентом завершеної роботи 23.06.2022 р.

3. Вихідні дані до роботи Методи розпізнавання голосових повідомлень, плата макетування Arduino Uno, лінійна передача даних, тип мікрофону

4. Зміст роботи (перелік питань, які потрібно розробити)

Вступ. 1. Аналіз технічного завдання та систем перетворення і розпізнавання голосових повідомлень. 2. Проектування комп'ютерної системи перетворення голосових повідомлень у текст. 3. Програмна реалізація та налаштування комп'ютерної системи перетворення голосових повідомлень у текст. 4. Безпека життєдіяльності, основи охорони праці. Висновки

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень, слайдів)

1. Існуючі системи розпізнавання голосових повідомлень

2. Функціональна схема комп'ютерної системи

3. Структура Arduino Nano

4. Схема підсилювача звукового сигналу

5. Архітектура комп'ютерної системи

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
<i>Безпека життєдіяльності, основи охорони праці</i>	<i>Лазарюк В.В., к.т.н., доц. каф. МТ</i>		

7. Дата видачі завдання _____

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів роботи	Термін виконання етапів роботи	Примітка
1	<i>Розробка та аналіз технічного завдання</i>	<i>23.03-28.03.2022</i>	
2	<i>Аналіз принципів функціонування систем перетворення голосових повідомлень у текст</i>	<i>29.03-07.04.2022</i>	
3	<i>Проектування структури комп'ютерної системи перетворення голосових повідомлень у текст</i>	<i>08.04-20.04.2022</i>	
4	<i>Обґрунтування вибору та аналіз технічних характеристик апаратного забезпечення системи</i>	<i>20.04-26.04.2022</i>	
5	<i>Розробка алгоритмів та програмного забезпечення для управління процесом одержання і перетворення голосових повідомлень</i>	<i>27.04-10.05.2022</i>	
6	<i>Налаштування параметрів комп'ютерної системи</i>	<i>10.05-20.05.2022</i>	
7	<i>Розробка інструкцій з використання комп'ютерної системи</i>	<i>21.05-26.05.2022</i>	
8	<i>Безпека життєдіяльності, основи охорони праці</i>	<i>27.05-30.05.2022</i>	
9	<i>Оформлення кваліфікаційної роботи</i>	<i>30.05-10.06.2022</i>	
10	<i>Попередній захист кваліфікаційної роботи</i>	<i>10.06-20.06.2022</i>	
11	<i>Захист кваліфікаційної роботи</i>	<i>22.06-24.06.2022</i>	

Студент

(підпис)

Козінець Ернест Сергійович

(прізвище та ініціали)

Керівник роботи

(підпис)

Луцків Андрій Мирославович

(прізвище та ініціали)

АНОТАЦІЯ

Комп'ютерна система перетворення голосових повідомлень у текст // Кваліфікаційна робота на здобуття освітнього ступеня бакалавр // Козінець Ернест Сергійович // ТНТУ, спеціальність 123 «Комп'ютерна інженерія»// Тернопіль, 2022 // с.– 79, рис. – 37 , табл. – 8, аркушів А1 – 5, бібліогр. – 14.

Ключові слова: система, перетворення, голос, повідомлення, текст.

Основними результатами кваліфікаційної роботи є спроектована концептуальна схема комп'ютерної системи перетворення голосових повідомлень у текст, обґрунтований вибір мікроконтролера для управління та опрацювання голосових повідомлень, а також мікрофона з автоматичним регулюванням коефіцієнта підсилення. Розроблено програмне забезпечення для формування шаблонів голосових повідомлень, визначення амплітуди і спектру голосового повідомлення, а також проведено налаштування мікрофона для формування голосових повідомлень одним користувачем.

Застосування обґрунтованих у роботі апаратних пристроїв та відповідного програмного забезпечення дало змогу реалізувати повно функціональну та продуктивну систему перетворення простих голосових повідомлень у текст, що в подальшому може бути використано для управління та передачі даних у текстовому форматі для виконання команд периферійними пристроями та для вирішення задач розпізнавання мовлення.

ABSTRACT

Computer system for converting voice messages into text // Bachelor's thesis // Kozinets Ernest Serhiiovych // TNTU, speciality 123 «Computer engineering»// Ternopil, 2022 // p.– 79 , fig. – 37 , tab. – 8, posters A1 – 5, ref. – 14.

Keywords: system, converting, voice, message, text.

The main results of the qualification work are the designed conceptual scheme of the computer system for converting voice messages into text, reasonable choice of microcontroller for control and processing of voice messages, as well as a microphone with automatic gain control.

Software for the formation of voice message templates, determination of the amplitude and spectrum of the voice message, as well as the setting of the microphone for the formation of voice messages by one user. The use of sound hardware devices and related software has made it possible to implement a fully functional and productive system for converting simple voice messages into text, which can then be used to control and convert data in text format to execute commands by peripherals and speech recognition.

ЗМІСТ

ПЕРЕЛІК ОСНОВНИХ УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ І СКОРОЧЕНЬ	8
ВСТУП	9
РОЗДІЛ 1 АНАЛІЗ ТЕХНІЧНОГО ЗАВДАННЯ ТА СИСТЕМ ПЕРЕТВОРЕННЯ І РОЗПІЗНАВАННЯ ГОЛОСОВИХ ПОВІДОМЛЕНЬ	11
1.1 Аналіз вимог до комп'ютерної системи перетворення голосових повідомлень у текст	11
1.2 Аналіз принципів та існуючих систем перетворення голосових повідомлень у текст	17
РОЗДІЛ 2 ПРОЕКТУВАННЯ КОМП'ЮТЕРНОЇ СИСТЕМИ ПЕРЕТВОРЕННЯ ГОЛОСОВИХ ПОВІДОМЛЕНЬ У ТЕКСТ	22
2.1 Проектування схеми організації комп'ютерної системи	22
2.2 Налаштування АЦП	26
2.3 Особливості технічних характеристик Arduino Uno/Nano	28
2.4 Підсилювач голосових повідомлень	33
РОЗДІЛ 3 ПРОГРАМНА РЕАЛІЗАЦІЯ ТА НАЛАШТУВАННЯ КОМП'ЮТЕРНОЇ СИСТЕМИ ПЕРЕТВОРЕННЯ ГОЛОСОВИХ ПОВІДОМЛЕНЬ У ТЕКСТ	40
3.1 Проектування архітектури програмного забезпечення при перетворенні голосових повідомлень у текст	40
3.2 Обчислення коефіцієнтів цифрового частотного фільтру	42
3.3 Зразки (шаблони) голосових повідомлень	47
3.4 Навчання розпізнаванню голосових повідомлень	50

					КС КРБ 123.220.00.00 ПЗ			
Змн.	Арк.	№ докум.	Підпис	Дата	<i>Комп'ютерна система перетворення голосових повідомлень у текст</i>	Літ.	Арк.	Аркуші
Розроб.		Козінець Е.С.					6	
Перевір.		Луцків А.М.				ТНТУ, каф. КС, гр. СІс-43		
Реценз.								
Н. Контр.		Луцкич Н.С.						
Затверд.		Осухівська Г.М.						

3.5	Комплексне розпізнавання голосових повідомлень	53
РОЗДІЛ 4 БЕЗПЕКА ЖИТТЄДІЯЛЬНОСТІ, ОСНОВИ ОХОРОНИ ПРАЦІ		55
4.1	Способи проведення штучного дихання та масажу серця	55
4.2	Заходи щодо боротьби з шкідливою дією ультразвуку на організм людини	58
ВИСНОВКИ		61
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....		62
Додаток А. Технічне завдання		
Додаток Б. Програмний код для безперервного зчитування штрих-кодів		

					КС КРБ 123.220.00.00 ПЗ	Арк.
						7
Змн.	Арк.	№ докум.	Підпис	Дата		

ПЕРЕЛІК ОСНОВНИХ УМОВНИХ ПОЗНАЧЕНЬ,
СИМВОЛІВ І СКОРОЧЕНЬ

АРЗ	Автоматичний регулятор звуку
КС	Комп'ютерна система
ПЗ	Програмне забезпечення
ГРІО	General purpose input/output
ІоТ	Internet of Things
USB	Universal Serial Bus
GPU	Graphics Proccesing Unit

					<i>КС КРБ 123.220.00.00 ПЗ</i>	Арк.
						8
Змн.	Арк.	№ докум.	Підпис	Дата		

ВСТУП

Сьогоднішній темп розвитку інформаційних технологій стимулює розвиток та автоматизацію процесів у різних галузях і сферах виробничої діяльності та надання послуг. Враховуючи ефективність реалізації комп'ютерних систем із застосуванням технологій IoT, методів і технологій машинного навчання, набув широкої популярності напрям, який стосується опрацювання природної мови.

Одним з піднапрямків розвитку NLP є автоматизація розпізнавання природної мови і перетворення голосових повідомлень у текст, англійською мовою – «Speech to text».

Технологія перетворення голосових повідомлень у текст орієнтована на розпізнавання мовлення, яке дозволяє визначати й транслювати розмовну мову у текст за допомогою методів обчислювальної лінгвістики.

На сьогодні існують програми, інструменти та пристрої, які можуть перетворювати аудіопотоки в режимі реального часу для відображення тексту та виконання певних операцій над ним.

Функцію перетворення голосових повідомлень у текст забезпечує програмне забезпечення, яке працює шляхом прослуховування аудіо та надсилання дослівної стенограми, яку можна редагувати, на певному пристрої. Комп'ютерна програма використовує лінгвістичні алгоритми для сортування звукових сигналів від вимовлених слів і передачі цих сигналів у текст за допомогою Unicode символів

Перетворення мовлення в текст працює за допомогою складної моделі машинного навчання, яка включає кілька кроків. Коли з'являються звуки, то відбувається формування серії вібрацій. Технологія перетворення голосу в текст працює, вловлюючи ці вібрації та перекладаючи їх на цифрову мову за допомогою аналого-цифрового перетворювача.

Аналого-цифровий перетворювач приймає звуки з аудіофайлу, вимірює хвилі та фільтрує їх, щоб розрізнити відповідні звуки. Потім звуки

					КС КРБ 123.220.00.00 ПЗ	Арк.
						9
Змн.	Арк.	№ докум.	Підпис	Дата		

розбиваються на соті чи тисячні секунди, а потім узгоджуються з фонемами. Потім фонему проходять через мережу за допомогою математичної моделі, яка порівнює їх із добре відомими реченнями, словами та фразами.

Запровадження голосового пошуку дає змогу маркетологам отримувати інформацію про тенденції в даних і поведінку споживачів. Наприклад, розпізнавання мовлення надає інформацію про акценти та словниковий запас людей, інтерпретуючи вік, місце розташування та інші важливі демографічні показники.

Медіа-субтитри Amazon transcribe також може записувати зустрічі та розмови за допомогою функції цифрового переписувача, покращуючи продуктивність, доступність та впорядковуючи важливі нотатки.

Клінічна документація Amazon Transcribe Medical — це інструмент для медичних працівників, який дозволяє швидко та ефективно записувати клінічні розмови в електронні системи медичних записів для аналізу.

У банківській справі перетворення голосових повідомлень у текст використовується при голосовому обслуговуванні клієнтів. У сфері охорони здоров'я така технологія допомагає підвищити ефективність, забезпечуючи доступ до інформації та введення даних.

Тому актуальність розробки комп'ютерної системи перетворення голосових повідомлень у текст на сьогодні є доволі актуальною і потребує додаткового дослідження для підвищення ефективності та продуктивності трансляції звуку у текст.

					КС КРБ 123.220.00.00 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		10

РОЗДІЛ 1 АНАЛІЗ ТЕХНІЧНОГО ЗАВДАННЯ ТА СИСТЕМ ПЕРЕТВОРЕННЯ І РОЗПІЗНАВАННЯ ГОЛОСОВИХ ПОВІДОМЛЕНЬ

1.1 Аналіз вимог до комп'ютерної системи перетворення голосових повідомлень у текст

Комп'ютерна система перетворення голосових повідомлень у текст покликана розв'язати основну задачу трансляції звуку у текст, що в подальшому може бути використано для накопичення текстової інформації та її опрацювання.

Галузями, де можливе застосування такої системи, є збір та аналіз вимог при проектуванні окремих технічних та нетехнічних систем, при проведенні переговорів для забезпечення автоматизації процесу створення стенограм, при формуванні систем управління командами для зовнішніх пристроїв і т.п.

Перевагами і перспективами застосування комп'ютерної системи перетворення голосових повідомлень у текст є забезпечення можливості розширення джерел одержання інформації з фіксацією їх у текстовому вигляді.

Більшість систем сьогодні спроектовані і базуються на використанні методів і систем штучного інтелекту, що дає можливість доволі ефективно розпізнавати голосові повідомлення, однак це вимагає використання значних апаратних ресурсів, а відповідно й вартість таких систем є досить високою.

У даній роботі необхідно побудувати комп'ютерну систему перетворення голосових повідомлень у текст, яка б використовувала мінімальні характеристики апаратного забезпечення, забезпечувала мобільність системи та була зорієнтованою на одного користувача.

У комп'ютерній системі необхідно передбачити можливість навчання щодо розпізнавання голосових повідомлень, які в подальшому можуть бути трансльовані у текст або відповідні команди.

					КС КРБ 123.220.00.00 ПЗ			
<i>Змн.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>				
<i>Розроб.</i>		<i>Козінець Е.С.</i>			<i>Аналіз технічного завдання та систем перетворення і розпізнавання голосових повідомлень</i>	<i>Лім.</i>	<i>Арк.</i>	<i>Аркушів</i>
<i>Перевір.</i>		<i>Луцків А.М.</i>					11	
<i>Реценз.</i>						<i>ТНТУ, каф. КС, гр. СІс-43</i>		
<i>Н. Контр.</i>		<i>Луцкич Н.С.</i>						
<i>Затверд.</i>		<i>Осухівська Г.М.</i>						

Проектована комп'ютерна система перетворення голосових повідомлень у текст потребує обладнання для одержання голосових повідомлень, фільтрації сторонніх шумів, а також програмного забезпечення для навчання та розпізнавання голосових повідомлень. Окрім цього, необхідна наявність засобів відображення тексту, сформованого на основі одержаного голосового повідомлення та шаблонів для визначення відповідності звуків.

Мета проектування комп'ютерної системи перетворення голосових повідомлень у текст полягає в автоматизації процесу одержання і трансляції звукових сигналів у текст з подальшим його відображенням та зберіганням на зовнішніх пристроях. При цьому необхідно забезпечити ефективність використання апаратних ресурсів та продуктивність функціонування комп'ютерної системи.

Для того, щоб досягти поставленої мети роботи необхідно розв'язати ряд задач, найбільш важливими з яких є:

- визначення потенційних рішень проектування системи на основі аналізу існуючих рішень;
- побудова концептуальної моделі комп'ютерної системи перетворення голосових повідомлень у текст;
- обґрунтування і дослідження технічних характеристик апаратного забезпечення комп'ютерної системи;
- проектування функціональної та схеми електричної принципової підсистеми одержання та перетворення аналогового сигналу у цифровий;
- реалізація системного програмного забезпечення та програмування мікроконтролера для одержання і трансляції звуку у текст;
- обґрунтування прикладного програмного забезпечення для навчання системи перетворення голосових повідомлень у текст;
- налаштування параметрів комп'ютерної системи на використання одним користувачем – генератором звуку.

Найбільш важливими задачами і функціями, які має забезпечувати комп'ютерна система перетворення голосових повідомлень у текст полягає у:

					КС КРБ 123.220.00.00 ПЗ	Арк.
						12
Змн.	Арк.	№ докум.	Підпис	Дата		

- здатності одержання аналогового звукового сигналу;
- трансформації звукового сигналу у цифровий вигляд;
- фільтрації шумів;
- створенні навчальної вибірки для інтерпретації голосових повідомлень;
- формуванні шаблонів звуків;
- безпосередньому перетворення одержаного сигналу у текст;
- відображенні текстового повідомлення із голосового.

Програмне забезпечення комп'ютерної системи передбачає виконання задач на системному і прикладному рівнях. Системне програмне забезпечення використовується в процесі функціонування готової системи, а прикладне – в основному при проектуванні та налаштуванні параметрів апаратного забезпечення.

Передбачається, що основними складовими комп'ютерної системи на рівні апаратного забезпечення є:

- мікрофон;
- підсилювач мікрофона;
- аналогово-цифровий перетворювач;
- мікроконтролер управління процесом перетворення голосових повідомлень у текст;
- ПК або пристрій відображення та зберігання тексту.

Доступність налаштувань використовуваних апаратних і програмних ресурсів комп'ютерної системи перетворення голосових повідомлень у текст має бути захищеним на рівні фізичного доступу та процедури авторизації. Доступ до одержаного тексту, у випадку зберігання на ПК також передбачає наявність засобів аутентифікації.

В цілому, вимоги, які висуваються до проектованої комп'ютерної системи перетворення голосових повідомлень у текст полягають у забезпеченні визначеної функціональності та продуктивності в процесу опрацювання аналогового звукового сигналу.

					КС КРБ 123.220.00.00 ПЗ	Арк.
						13
Змн.	Арк.	№ докум.	Підпис	Дата		

При створенні прототипу комп'ютерної системи доцільним є використання плат макетування Arduino, які є недорогими по вартості і забезпечують високу продуктивність при залученні мінімуму апаратних ресурсів. Одержаний у результаті перетворення текст повинен бути надісланий до зовнішнього пристрою з дисплеєм.

У загальному випадку, вимоги до комп'ютерної системи перетворення голосових повідомлень у текст можна представити наступним чином:

- можливість отримання аналогового сигналу голосового повідомлення;
- здатність перетворення аналогового сигналу у цифровий вигляд;
- можливість розкладу сигналу за 4 смугами;
- можливість аналізу частотного та амплітудного спектрів;
- можливість формування шаблонів оцифрованих голосових повідомлень;
- здатність визначення коефіцієнтів підсилення в залежності від типу джерела звуку;
- забезпечення продуктивності перетворення звуку у часі. Близькому до реального;
- можливість передачі повідомлення у текстовому вигляді до зовнішнього пристрою (дисплею, ПК);
- здатність відображення текстового повідомлення на дисплеї периферійного пристрою.

Структурно компоненти комп'ютерної системи перетворення голосових повідомлень у текст визначаються її функціональною орієнтованістю і передбачають використання наступного апаратного забезпечення:

- плата прототипування Arduino Uno;
- мікрофон послідовного порта з автоматичним регулюванням коефіцієнта підсилення;
- підсилювач мікрофона;
- аналогово-цифровий перетворювач;
- середовище програмування для Arduino;

					КС КРБ 123.220.00.00 ПЗ	Арк.
						14
Змн.	Арк.	№ докум.	Підпис	Дата		

– прикладне програмне забезпечення аналізу звукових сигналів.

Комп'ютерна система перетворення голосових повідомлень у текст повинна забезпечувати стабільність роботи і дозволяти одержувати аналоговий звуковий сигнал, декодувати його у цифровий, виконувати відповідні трансформації для розбиття сигналу на смуги, порівнювати характеристики сигналу із шаблонними значенням, передавати повідомлення у текстовому вигляді у зовнішній пристрій.

Архітектура комп'ютерної системи передбачає виконання пристрою у вигляді модуля з мікрофоном, що під'єднується до ПК для обміну даними.

Спосіб зв'язку компонентів системи, зокрема, плати прототипування Arduino Nano та мікрофона з підсилювачем, передбачає безпосереднє під'єднання визначених контактів за допомогою провідників.

З'єднання та обмін даними Arduino з комп'ютером, або іншим зовнішнім пристроєм з дисплеєм, виконується за допомогою з'єднання з лінійним інтерфейсом через СОМ-порт.

Процедури щодо проведення діагностичних заходів модулів комп'ютерної системи перетворення голосових повідомлень у текст повинні бути регламентовані відповідним графіком та проводитися у відповідності до інструкцій з експлуатації компонентів системи.

Діагностика плати макетування Arduino Nano виконується кожен раз при подачі на неї живлення шляхом самоініціалізації окремих складових. Якщо проявляються помилки або збої у роботі системи необхідно вжити заходи для негайно їх усунення.

Модернізацію комп'ютерної системи доцільно проводити у випадку, коли є необхідність розширити її функціональні можливості, зокрема в контексті опрацювання і формування шаблонів голосових повідомлень, а також розпізнавання великої кількості голосових повідомлень, згенерованих різними джерелами звуку. Такий варіант модернізації перехід на більш потужні плати макетування або використання мінікомп'ютерів по типу Raspberry Pi.

					КС КРБ 123.220.00.00 ПЗ	Арк.
						15
Змн.	Арк.	№ докум.	Підпис	Дата		

До перспектив розвитку комп'ютерної системи перетворення голосових повідомлень у текст належить можливість створення моделей машинного навчання для розпізнавання голосових повідомлень та їх інтеграція й сумісність з апаратними модулями системи.

До важливих аспектів надійності комп'ютерної системи перетворення голосових повідомлень у текст належать:

- здатність забезпечувати визначений рівень працездатності у визначених умовах використання ;
- здатність до відновлення працездатності після збоїв і помилок;
- можливість зберігати налаштування апаратних компонентів системи;
- здатність працювати у визначеному режимі підсилення звукового сигналу;
- можливість адаптовуватися до нових джерел звукового сигналу без втрати точності розпізнавання звуку;
- стабільність роботи АЦП;
- наявність механізму авторизованого доступу до налаштувань комп'ютерної системи.

Функціональні вимоги, а відповідно й задачі, які повинна забезпечувати комп'ютерна система перетворення голосових повідомлень у текст:

- здатність отримання аналогового звукового сигналу;
- можливість оцифрування аналогового звукового сигналу;
- здатність виконувати розбиття сигналу на смуги;
- можливість використання цифрового рекурсивного фільтра;
- здатність налаштування параметрів підсилення звуку;
- можливість визначати частотний спектр, сегменти і амплітуду звукового сигналу;
- здатність формування навчальної вибірки та шаблонів голосових повідомлень;
- можливість обміну та забезпечення сумісності між компонентами системи і зовнішніми периферійними пристроями;

					КС КРБ 123.220.00.00 ПЗ	Арк.
						16
Змн.	Арк.	№ докум.	Підпис	Дата		

– здатність налаштування параметрів системи під конкретного користувача.

Основні вимоги і обмеження для апаратних пристроїв комп'ютерної системи перетворення голосових повідомлень у текст:

- мікроконтролер ATmega 368;
- мікрофон з підсилювач МАХ9814;
- персональний комп'ютер або ноутбук.

Характеристики ПК для налаштування комп'ютерної системи, проведення розрахунків і навчання розпізнавання голосових повідомлень повинні відповідати таким технічним характеристикам:

- частота процесора $\geq 2,0$ ГГц;
- розмір оперативної пам'яті 8 Гб;
- дисковий простір ≥ 120 Гб.

Для реалізації системного ПЗ Arduino Nano щодо керування та забезпечення сумісності з іншими структурними компонентами системи має використовувати середовище Arduino IDE, мова програмування – C/C++. Системне програмне забезпечення комп'ютера повинно забезпечувати підтримку необхідних сервісів і середовищ для налаштування комп'ютерної системи.

1.2 Аналіз принципів та існуючих систем перетворення голосових повідомлень у текст

Взаємодія людини з машиною на сьогодні стала набагато складнішою, ніж була до цього часу. Про це можна говорити, спостерігаючи навіть за переходом від великих механічних кнопок до сенсорних екранів.

Однак апаратне забезпечення – це не єдине, що змінюється. Протягом усієї історії комп'ютерів текст був основним методом введення. Але завдяки розвитку

					КС КРБ 123.220.00.00 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		17

NLP і ML (машинне навчання), Data Science, тепер існують засоби для використання мови як засобу взаємодії з електронними пристроями.

Віртуальні помічники є найпоширенішим використанням цих інструментів. Google, Siri, Alexa та безліч інших цифрових помічників підняли високу планку можливостей, коли справа доходить до спілкування з цифровим світом на побутовому рівні.

Уперше в історії сучасних технологій можливість перетворювати вимовлені слова в текст є вільно доступною для всіх, хто хоче з нею поекспериментувати. Коли справа доходить до створення програм перетворення мовлення в текст, Python, одна з найбільш широко використовуваних мов програмування, яка має багато варіантів використання.

Перш ніж зануритися в технологію Python щодо функцій роботи із звуком і текстом, потрібно проаналізувати існуючі системи трансляції звуку у текст.

Перша система розпізнавання мови, створена трьома інженерами Bell Labs у 1952 році. Вона вміла читати лише цифри.

IBM Shoebox, побудувала у 1962 році систему Coils, яка могла розрізняти 16 слів окрім чисел при розпізнаванні голосу та виконувати основні математичні операції і публікувати результати. На рисунку 1.1 показано зовнішній вигляд цієї системи.



Рисунок 1.1 – Система розпізнавання голосу від IBM

Агентство передових оборонних дослідницьких проєктів (DARPA) підтримало дослідження щодо розпізнавання мовлення, що призвело до реалізації можливості системи Narгу розпізнавати 1011 слів.

Проблеми, які потребують послідовного опрацювання інформації, можуть бути представлені за допомогою статистичної прихованої моделі Маркова. Ця модель була використана при розробці нових методів розпізнавання голосу.

У 2001 році Google запустив свій інструмент голосового пошуку, який дозволяв користувачам здійснювати пошук за допомогою мови. Це був перший широко використовуваний додаток із підтримкою голосу. На рисунку 1.2 показано стрічку голосового пошуку Google.



Рисунок 1.2 – Стрічка для пошуку за голосом

У 2011 році Siri надала зручний спосіб підключення до електронних пристроїв Apple в режимі реального часу за допомогою голосових повідомлень (рисунок 1.3).

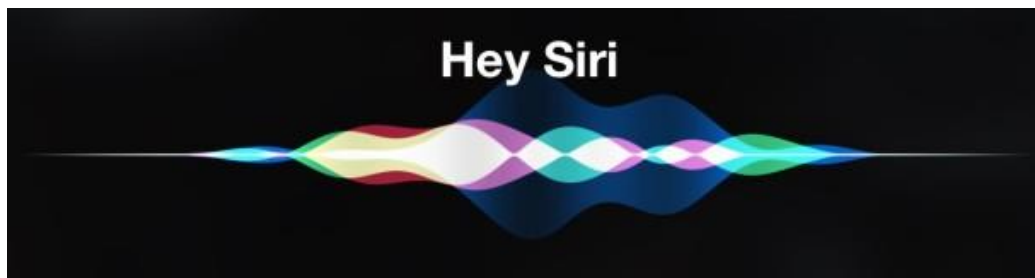


Рисунок 1.3 – Інтерфейс Siri для голосових повідомлень

Віртуальні помічники з голосовою активацією, такі як Alexa і Google Home, які разом продали понад 150 мільйонів одиниць, увійшли в популярність у 2014 і 2016 роках відповідно та представлені на рисунку 1.4.



Рисунок 1.3 – Віртуальні помічники з голосовою активацією Alexa і Google Home

Перетворення звуку у текст – складна тема, яка ще далека від вирішення. В загальному випадку, схему опрацювання голосових повідомлень і перетворення їх у текст показано на рисунку 1.4.

Змн.	Арк.	№ докум.	Підпис	Дата

КС КРБ 123.220.00.00 ПЗ

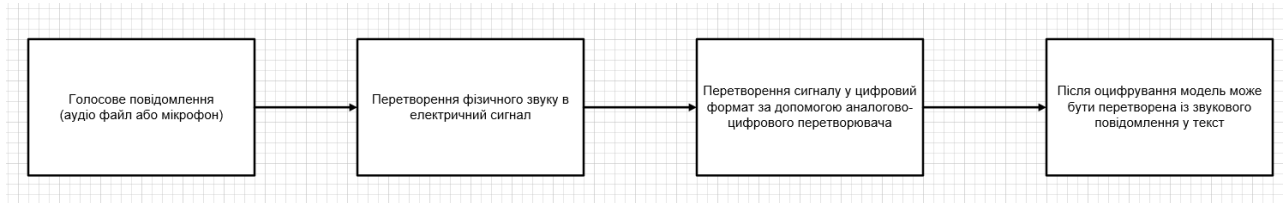


Рисунок 1.4 – Загальна схема перетворення голосових повідомлень у текст

Численні технічні обмеження роблять цей інструмент у кращому випадку неякісним. Нижче наведено деякі з труднощів, які найбільш часто трапляються при реалізації технології розпізнавання голосу.

Складність 1. Неточне розпізнавання мови

Системи розпізнавання мовлення не завжди точно розуміють вимовлені слова. VUI (voice user interface –голосові інтерфейси користувача) не настільки вміють розуміти контексти, які змінюють зв’язок між словами та фразами, як люди. Таким чином, машини можуть мати труднощі з розумінням семантики висловлювання.

Складність 2. Час розпізнавання звуків

Іноді системи розпізнавання голосових повідомлень потребують великої кількості часу для опрацювання. Це може бути пов’язано з тим, що люди володіють різноманітними вокальними моделями. Такі труднощі з розпізнаванням голосу можна подолати, якщо говорити повільніше або точніше, але це зменшує зручність інструменту.

3. Акценти мовлення

VUI можуть мати труднощі з розумінням діалектів, які не є стандартними. У межах однієї мови люди можуть вимовляти одні й ті ж слова у різні способи.

Складність 4. Фоновий шум і гучність голосових повідомлень

В ідеальному світі це не було б проблемою, але це не так, і, отже, VUI можуть не якісно працювати в шумному оточенні (громадські місця, великі офіси тощо).

РОЗДІЛ 2 ПРОЕКТУВАННЯ КОМП'ЮТЕРНОЇ СИСТЕМИ ПЕРЕТВОРЕННЯ ГОЛОСОВИХ ПОВІДОМЛЕНЬ У ТЕКСТ

2.1 Проектування схеми організації комп'ютерної системи

Arduino Uno/Nano містить 2 КБ оперативної пам'яті, 32 КБ ПЗУ і працює зі швидкістю близько 10 MIPS (залежно від комбінації інструкцій).

Мінікомп'ютер, яким раніше користувалися люди (1976 рік), працював зі швидкістю від 0,5 до 8 MIPS і мав, скажімо, від 2 до 32 КБ пам'яті, які розділені між програмним забезпеченням і даними. Більшість груп компанії IBM мали PDP-8 або PDP-11. Одна група мала величезний IBM-360 з 128 КБ, але менше 1 MIPS. Інша група переобладнала систему управління ракетним вогнем Univac, яка працює зі швидкістю 1MIPS.

Існують інші проекти розпізнавання голосу, але вони потребують підключення до Інтернету та надсилання всіх своїх приватних розмов на Amazon чи Google, або вони потребують значно потужніших характеристик комп'ютера, наприклад мінікомп'ютер Raspberry Pi.

Складною проблемою при розпізнаванні голосових повідомлень є безперервне мовлення будь-якої людини, яка використовує величезний словниковий запас. З іншої сторони, у комп'ютерній системі існує лише один «оратор», який озвучує окремі слова з невеликого словникового запасу.

У випадку, коли необхідне застосування мультиметра на голові або крихітного пристрою, що працює з мобільним телефоном і кріпиться за вухом без екрану чи клавіатури, прості голосові команди можуть бути відображені на будь-якому дисплеї. Необхідність реалізації системи перетворення голосових повідомлень у текст пов'язана також з дистанційним керуванням, або наприклад, при дистанційному управлінні MP3-плеєра.

					КС КРБ 123.220.00.00 ПЗ			
<i>Змн.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>				
<i>Розроб.</i>		<i>Козінець Е.С.</i>			<i>Проектування комп'ютерної системи перетворення голосових повідомлень у текст</i>	<i>Літ.</i>	<i>Арк.</i>	<i>Аркуші</i>
<i>Перевір.</i>		<i>Луцків А.М.</i>					22	
<i>Реценз.</i>						<i>ТНТУ, каф. КС, гр. СІс-43</i>		
<i>Н. Контр.</i>		<i>Луцки Н.С.</i>						
<i>Затверд.</i>		<i>Осухівська Г.М.</i>						

Існує багато застосувань, де десяток командних слів можуть бути корисними. Якщо використовувати такі системи як "Alexa" або "Siri", то можна знайти близько 200 проектів їх реалізації – багато з них можуть бути корисними, якщо не потребують підключення до Інтернету. У такому випадку можна додати мовлення, використовуючи, наприклад, бібліотеку Talkie.

При проектуванні і реалізації комп'ютерної системи пропонується використати Arduino Nano (або Uno, Mini, що використовує мікроконтролер 16 МГц ATmega328), мікрофон і підсилювач для нього.

Загальна схема підключення мікрофона на базі MAX9814 та з використанням Arduino Nano показано на рисунку 2.1.

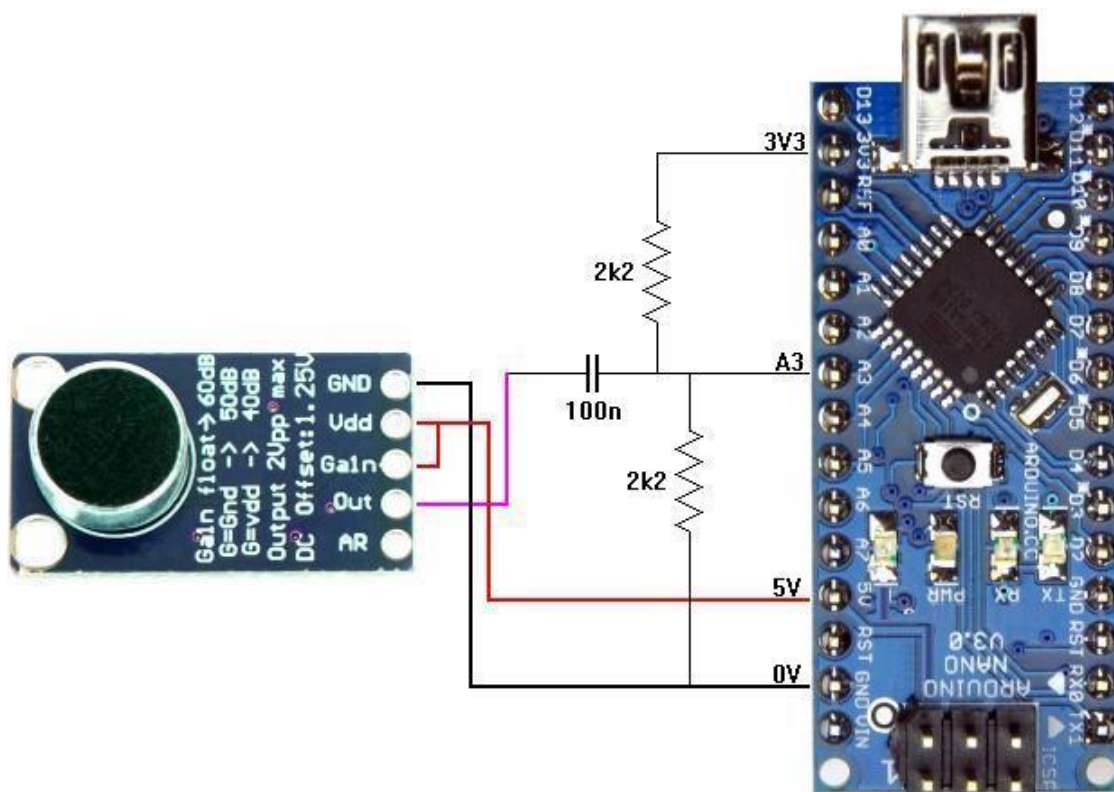


Рисунок 2.1 – Схема з'єднання плати макетування з мікрофоном

У схемі, показаній на рисунку 2.1, вивід A/R є не підключеним. Підсилювач під'єднаний до VDD, що відповідає мінімальному рівню підсилення голосового повідомлення. Можна було б підключити MAX9814 до цифрових контактів Arduino, щоб керувати ними за допомогою програмного забезпечення:

Змн.	Арк.	№ докум.	Підпис	Дата

для «непідключених» потрібно підключити контакт на вхід. На рисунку 2.2 показано експериментальне підключення мікрофона та плати макетування.

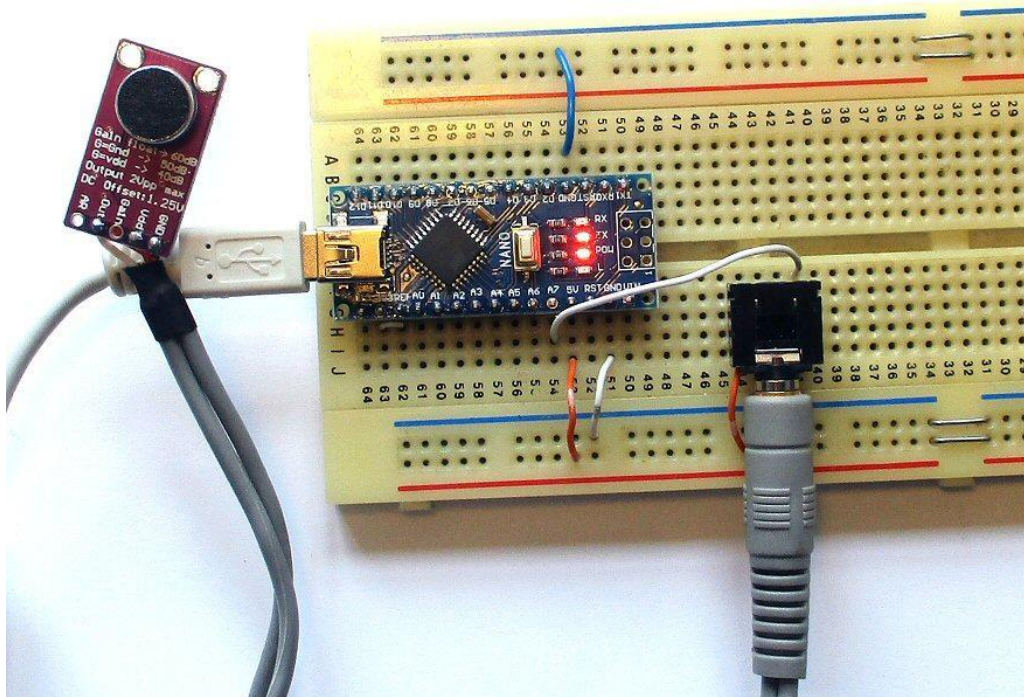


Рисунок 2.2 – Підключення пристроїв комп'ютерної системи розпізнавання голосових повідомлень

Варто прикріпити мікрофон і модуль MAX9814 до «мікрофонної штанги». Мікрофон повинен розташовуватися з одного боку з метою уникнення сторонніх шумів або інших звуків дихання.

Встановлено, що підсилення 40 дБ дає найкраще співвідношення сигнал/шум, коли мікрофон прикріплений до штанги. При більш високому підсиленні, фоновий шум посилюється занадто швидко; коли наявним був голосовий сигнал, регулятор підсилення зменшував його інтенсивність до «розумного» рівня, але коли сигнал був відсутнім, то шум повертався.

Звуковий сигнал від модуля підсилення відповідає номіналу напруги 1,25 В, а сам сигнал змінюється від 0 В до 2,5 В. АЦП Arduino є 10 бітовим, тому числове значення змінюється від 0 до 1023. У даному випадку вихід 3,3 В

використовується як аналогова опорна напруга, тому діапазон від 0 до 1023 відповідає діапазону напруг від 0 В до 3,3 В.

Модуль підсилення можна підключити безпосередньо до одного з вхідних контактів АЦП, але на схемі вище підключено простий RC-фільтр високих частот. Це означає, що нижчі частоти звуку (нижче 1,4 кГц) знижені. Спектр є більш плоским, і можна використовувати цілочисельну арифметику більш ефективно. Видаляючи низькі частоти, підсилювач і АЦП менш схильні до кліпування.

Існує багато дискусій щодо попереднього наголосу під час розпізнавання мовлення. Оскільки модуль підключений до змінного струму, два резистори використовуються для центрування входу АЦП навколо 1,65 В. Програмне оголошення та ініціалізація виводу аналогово-цифрового перетворювача, який фізично з'єднаний з операційним підсилювачем, передбачає використання 7 контакту на Arduino Nano, тобто:

```
const int AUDIO_IN = A7;
```

Організація комп'ютерної системи перетворення голосових повідомлень у текст можливе також і без застосування модуля MAX9814, а з використанням звичайного електретного мікрофона і операційного підсилювача. В якості операційного підсилювача можна використати LM358, який безумовно, не "малошумний", і його вихід може видавати номінали лише в межах 1,5 В від V_{cc} . Однак він може працювати на 5 В і достатньо хороший для цього проекту. Схема, операційного підсилювача, яка використана у проекті, показана на рисунку 2.4.

Загальний коефіцієнт підсилення операційного підсилювача становить близько 200. Це приводить вихідний сигнал у потрібний діапазон, якщо використовувати мікрофон, що кріпиться до штанги.

C1 і R4 діють як фільтр високих частот з плавним зниженням частоти нижче 1,5 кГц. Позитивний вхід операційних підсилювачів (і, отже, вихід) знаходиться на половині шляху між 0 В і 3,3 В. У даному випадку використовується Nano АЦП з 3,3 В як V_{ref} , тому вихід LM358 буде коливатися в потрібному діапазоні.

					КС КРБ 123.220.00.00 ПЗ	Арк.
						25
Змн.	Арк.	№ докум.	Підпис	Дата		

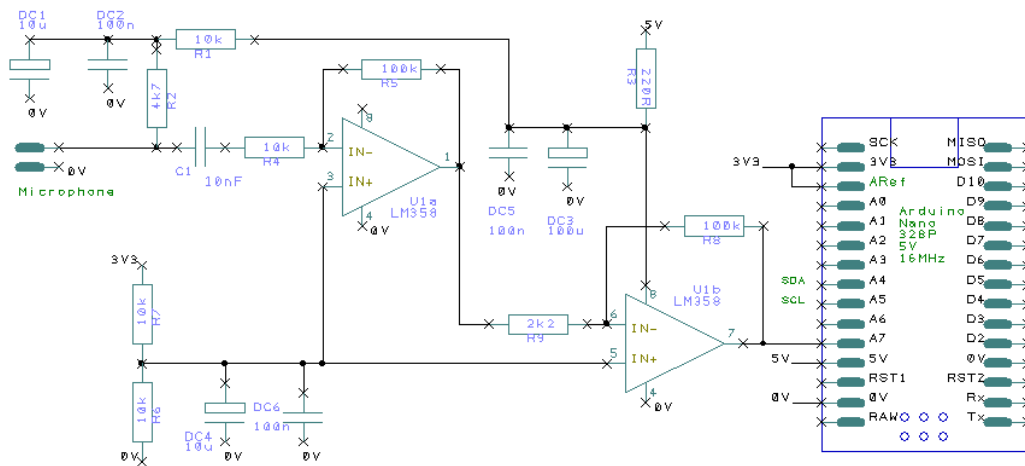


Рисунок 2.4 – Схема операційного підсилювача LM358

Вихід 3,3 В, який генерується Nano, досить шумний, тому потрібно використовувати DC4 і DC6 як роз'єднувальні конденсатори.

LM358 живиться від виходу 5 В Arduino Nano. Вивід 5V Nano має багато шуму, тому він згладжується R3, DC3, DC5. Ця згладжена напруга номіналом 5 В ще більше фільтрується R1, DC1, DC2 і діє як джерело зміщення для мікрофона через R2.

2.2 Налаштування АЦП

Стандартним способом використання АЦП Arduino Nano є виклик функції `analogRead()`, однак вона працює досить повільно. Функція ініціалізує АЦП і вибирає правильний вхідний контакт. Потім починається перетворення, що триває до того часу поки воно завершиться. Все це займає близько 100 мкс. Однак цей варіант не є ефективним, тому пропонується у `Setup()` використовувати стандартний код бібліотеки Arduino для ініціалізації АЦП, який показаний у лістингу 2.1.

Лістинг 2.1 – Ініціалізація АЦП

```
analogReference(EXTERNAL);
analogRead(AUDIO_IN);
```

Змн.	Арк.	№ докум.	Підпис	Дата

Опорна напруга для АЦП встановлюється на контакт ARef, а ARef підключається до контакту 3,3 В. Викликавши analogRead() один раз, отримуємо бібліотеку Arduino для налаштування АЦП.

У головному циклі для початку перетворення встановлюється біт ADSC (ADC Start Conversion). Це повідомляє АЦП розпочати перетворення. Бібліотека Arduino перевела АЦП в режим єдиного перетворення, тому потрібно налаштувати ADSC для початку кожного наступного.

Біт ADIF (прапорець переривання АЦП) встановлюється після завершення перетворення. Це означає, що можна виконувати інші операції поки АЦП зайнятий. Може здатися дивним, але варто очистити ADIF, встановлюючи його на 1.

Біт ADIE (ADC Interrupt Enable) був очищений бібліотекою Arduino, тому фактичного переривання не відбувається – просто використовується прапорець переривання, щоб перевірити, коли перетворення ADC закінчено. 10-бітовий результат перетворення АЦП зчитується шляхом зчитування 8-розрядного регістру ADCL, а потім регістру ADCH.

Коли виконується читання ADCL, значення в ADCH заморожується, поки не відбудеться його читання. Це зроблено таким чином, щоб гарантувати, що не переплутати молодші і старші байти з різних зразків. Прочитати ADCL та ADCH необхідно у правильному порядку. Програмний код повного налаштування АЦП показано у лістингу 2.2.

Лістинг 2.2 – Програмний код повного налаштування АЦП

```
while (true) {  
    while (!getBit(ADCSRA, ADIF)) ; // wait for ADC  
    byte val1 = ADCL;  
    byte val2 = ADCH;  
    bitSet(ADCSRA, ADIF); // clear the flag  
    bitSet(ADCSRA, ADSC); // start ADC conversion  
    int val = val1;  
    val += val2 << 8;  
    ... process the sample in val  
}
```

					КС КРБ 123.220.00.00 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		27

Код «опрацювати зразок» виконується під час наступного перетворення АЦП. Напряга від підсилювача буде зосереджена навколо 512. Для обробки сигналу потрібно, щоб він був центрований навколо 0. Тому виникає необхідність виконання операції віднімання середнього поточного значення вхідного значення від val, як показано у лістингу 2.3.

Лістинг 2.3 – Центрування відносно «0»

```
static int zero = 512;  
if (val < zero)  
    zero--; else  
    zero++;  
val = val - zero;
```

Скетч speakresog0.ino виконує перевірку АЦП. Він може збирати зразки зі швидкістю близько 9 kps. Скетч може надсилати значення на ПК через послідовну шину, але вона сповільнює швидкість приблизно до 1100 sps (при 57 600 бод).

Якщо обрати команду «Tools Serial Plotter» в Arduino IDE, то можна побачити «осцилограф» відображення голосових повідомлень. Це є хорошим тест на працездатність і продуктивність обладнання.

2.3 Особливості технічних характеристик Arduino Uno/Nano

В якості центрального вузла генерації та управління голосовими повідомленнями при проектуванні та реалізації комп'ютерної системи запропоновано скористатися платою макетування Arduino Uno. Зовнішній вигляд, а також особливості і призначення виводів цієї плати наведено на рисунку 2.2.

					КС КРБ 123.220.00.00 ПЗ	Арк.
						28
Змн.	Арк.	№ докум.	Підпис	Дата		

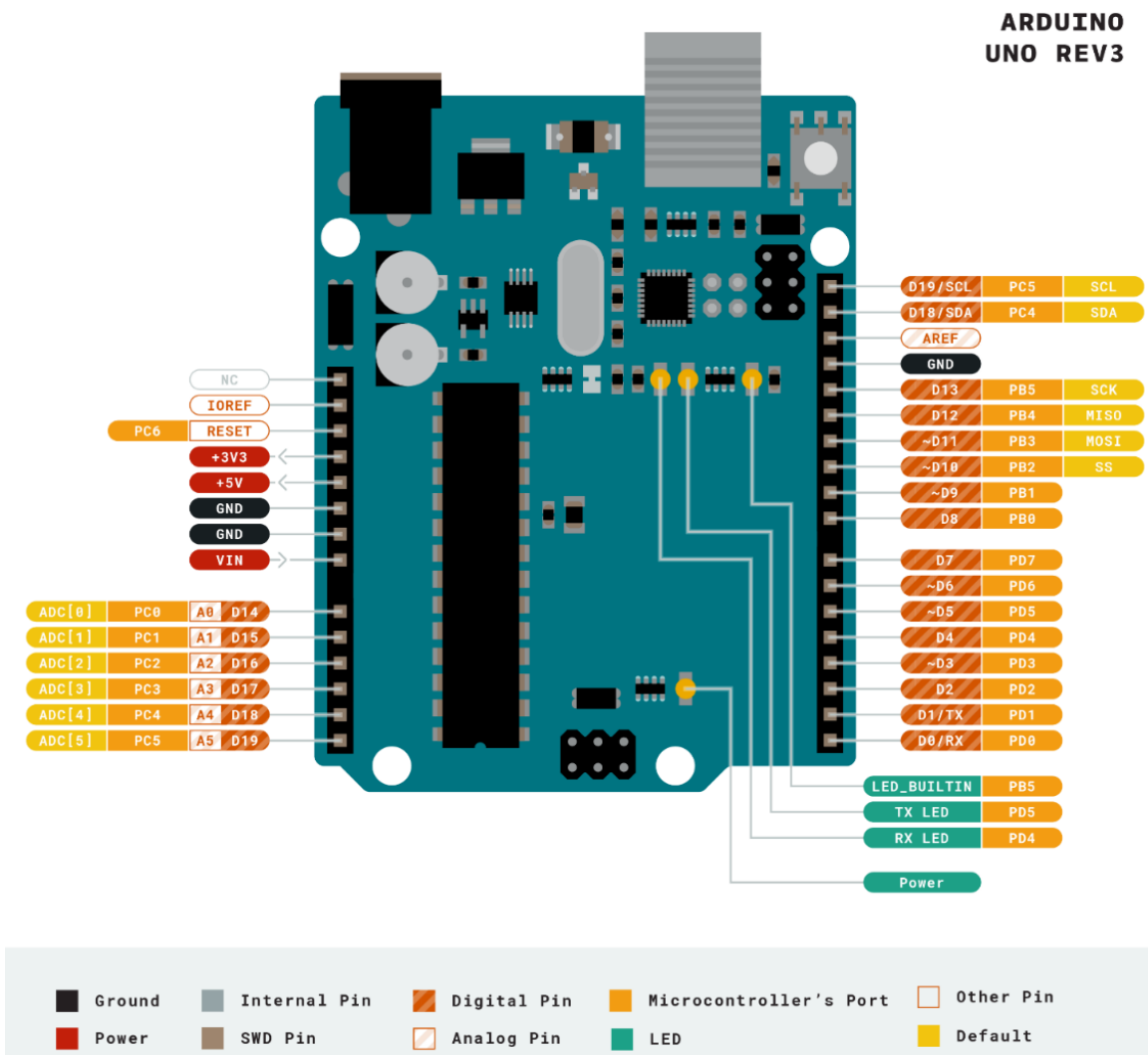


Рисунок 2.2 – Зовнішній вигляд та виводи Arduino UNO

Структурою організації Arduino Uno/Nano передбачено аналогові (6 одиниць) та цифрові (14 одиниць) виводи. Це дає змогу одночасно використовувати як аналогові, так і цифрові периферійні та зовнішні пристрої. Опис детальних характеристик Arduino Uno/Nano продемонстровано у табл. 2.1., а складові елементи – на рисунку 2.3.

Таблиця 2.1 – Технічні характеристики Arduino Uno/Nano

Характеристика	Значення
Тип мікроконтролера	АТmega328P
Операційна напруга	5В
Вхідна напруга (рекомендована)	7-12В
Вхідна напруга (крайні значення)	6-20В
Цифрові входи/виходи	14
Цифрові виходи	6
Аналогові входи	6
Живлення (струм) виводів входу/виходу 5В	20 мА
Живлення (струм) виводів входу/виходу 3,3 В	50 мА
Flash пам'ять	32 КБ (АТmega328P)
SRAM	2 КБ
EEPROM	1 КБ
Частота процесора	16 МГц
Довжина	68.6 мм
Ширина	53.4 мм
Вага	25 г

Для подачі напруги живлення до плати макетування передбачено використання VCC і GND. Важливо дотримуватися номіналу напруги, яка визначена в інструкції з експлуатації даного виду мікроконтролера.

Верхньою межею напруги, яка може подаватися на мікроконтролер АТmega328P, є 5,5В. При цьому, абсолютна межа становить – 6В. Однак при тривалій роботі в такому режимі, можливий вихід плати макетування з ладу.

					КС КРБ 123.220.00.00 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		30

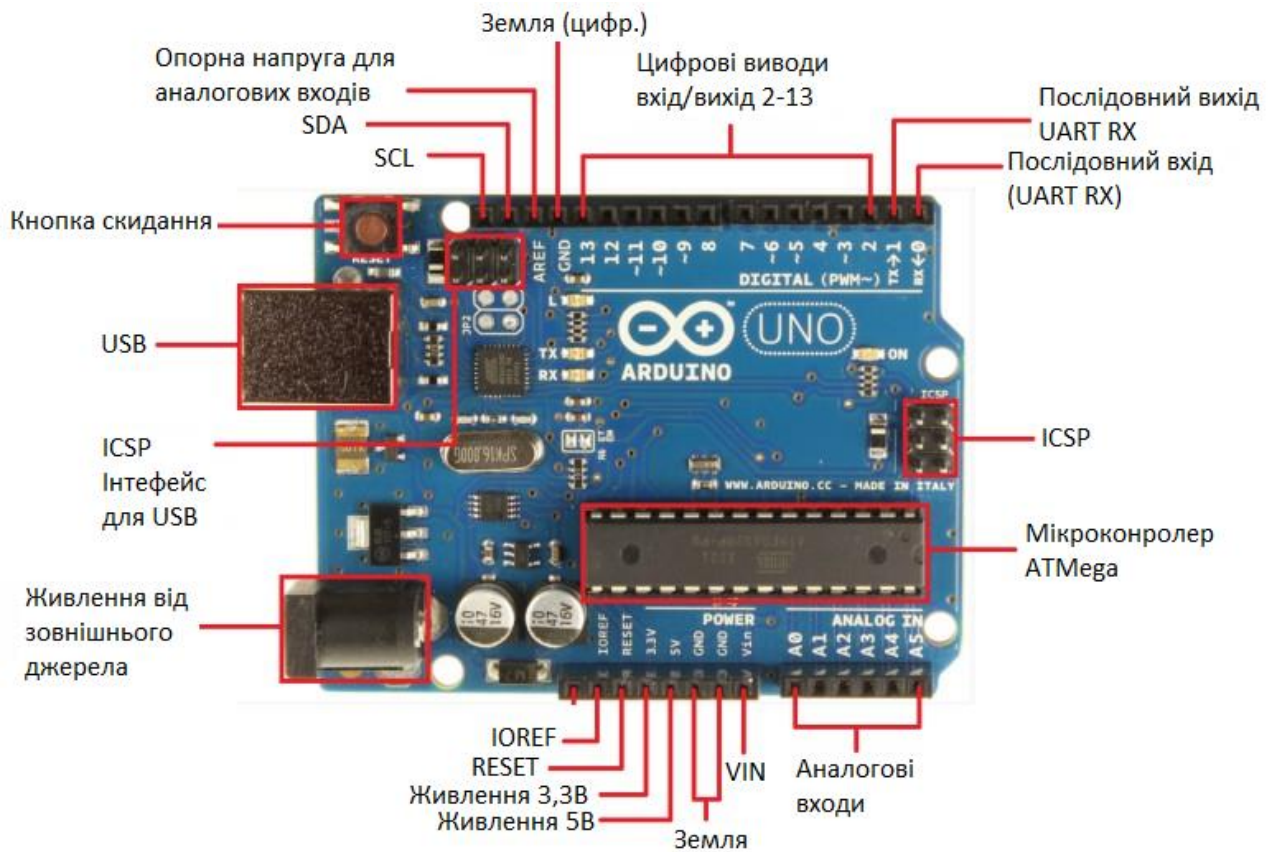


Рисунок 2.3 – Інтерфейси Arduino Uno/Nano

На рисунку 2.4 представлено позначення контактів, які можна використовувати при організації комп'ютерних систем із застосуванням плати макетування Arduino Uno/Nano в основі якого лежить функціонування мікроконтролера ATmega 328.

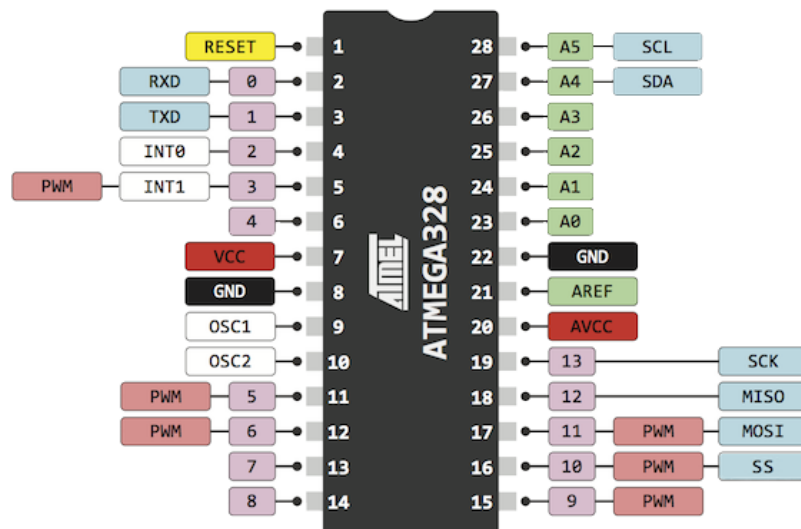


Рисунок 2.4 – Призначення виводів мікроконтролера ATmega 328

Змн.	Арк.	№ докум.	Підпис	Дата

КС КРБ 123.220.00.00 ПЗ

Арк.

31

З метою знешкодження перешкод високочастотного спектру при організації схеми живлення бажано підключати конденсатор (керамічний) з номіналом ємністю 0.1мкФ. Встановлюється він між виводами VCC та GND. Особливістю підключення такого конденсатора є те, що його розташовують якнайближче до контактів живлення. Це дозволить значно зменшити паразитну індуктивність та опір провідників.

Живлення мікроконтролера ATmega328P можна організувати двома способами:

- при реалізації живлення цифрових пристроїв і мікроконтролера задіюють виводи VCC (7 вивід) і GND (8 вивід);
- при підключенні аналого-цифрового перетворювача використовуються виводи 20 і 22, з відповідними мітками AVCC та GND.

В загальному випадку, представлення схеми електричної принципової для Arduino Uno/Nano наведено на рисунку 2.5.

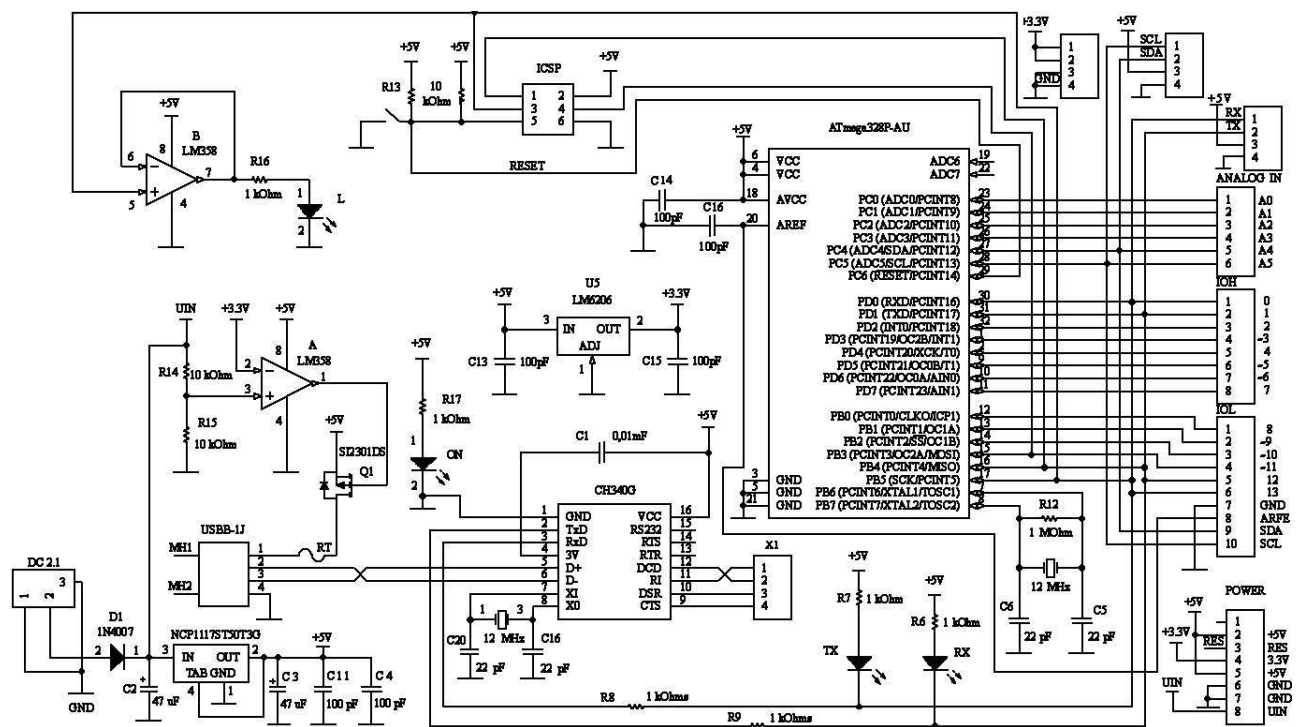


Рисунок 2.5 – Організація схеми Arduino Uno/Nano

Проаналізувавши основні технічні характеристики та функціональні можливості Arduino на базі мікроконтролера ATmega328P можна зробити

висновок, що його можна ефективно використовувати для розв'язання задач, передбачених при проектуванні комп'ютерної системи перетворення голосових повідомлень у текст.

2.4 Підсилювач голосових повідомлень

Оптимальним вибором при організації комп'ютерної системи перетворення голосових повідомлень у текст є мікрофонний підсилювач МАХ9814, оскільки до його складу входить автоматичне регулювання підсилення звуку.

МАХ9814 – це недорогий високоякісний мікрофонний підсилювач з автоматичним регулюванням коефіцієнту підсилення (AGC) і зміщенням мікрофона з низьким рівнем шуму. Пристрій має малошумний попередній підсилювач, підсилювач із змінним підсиленням (VGA), вихідний підсилювач, мікрофонний генератор напруги зміщення та АРЗ (автоматичний регулятор звуку).

Попередній підсилювач з низьким рівнем шуму має фіксований коефіцієнт підсилення 12 дБ, тоді як коефіцієнт підсилення VGA автоматично регулюється від 20 дБ до 0 дБ, залежно від вихідної напруги та порогу АРЗ.

Вихідний підсилювач пропонує вибір коефіцієнта підсилення 8 дБ, 18 дБ і 28 дБ. Без стиснення каскад підсилювачів призводить до загального підсилення 40 дБ, 50 дБ або 60 дБ. Трирівневий цифровий вхід програмує підсилення вихідного підсилювача. Зовнішній резистивний дільник контролює поріг AGC, а один конденсатор програмує час атаки/звільнення.

Трирівневий цифровий вхід програмує співвідношення часу атаки до звільнення. Час утримання AGC фіксується на рівні 30 мс. Генератор напруги зміщення мікрофона з низьким рівнем шуму може зміщувати більшість електретних мікрофонів.

МАХ9814 доступний у компактному 14-контактному корпусі TDFN. Цей пристрій призначений для розширеного діапазону температур від -40°C до $+85^{\circ}\text{C}$.

					КС КРБ 123.220.00.00 ПЗ	Арк.
						33
Змн.	Арк.	№ докум.	Підпис	Дата		

Основними прикладними застосування мікрофону МАХ9814:

- цифрові фотоапарати;
- цифрові відеокамери;
- КПК;
- гарнітури Bluetooth;
- розважальні системи (наприклад, караоке);
- двосторонні комунікатори;
- високоякісні портативні рекордери;
- ІР-телефони/телефонні конференції.

Основні технічні характеристики МАХ9814:

- автоматичне регулювання коефіцієнта підсилення (AGC);
- три налаштування діапазону підсилення (40 дБ, 50 дБ, 60 дБ);
- програмований час атаки;
- програмований коефіцієнт атаки/ звільнення;
- діапазон напруги живлення від 2,7 В до 5,5 В;
- низька щільність вхідного шуму 30 нВ/Гц;
- низький THD: 0,04%;
- режим вимкнення з низьким енергоспоживанням;
- зміщення внутрішнього мікрофона з низьким рівнем шуму, 2 В;
- доступний у компактному 14-контактному TDFN (3 мм x 3 мм);
- від -40°C до +85°C розширений діапазон температур

На рисунку 2.2 показано електрична-принципова схему реалізації МАХ9814.

					КС КРБ 123.220.00.00 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		34

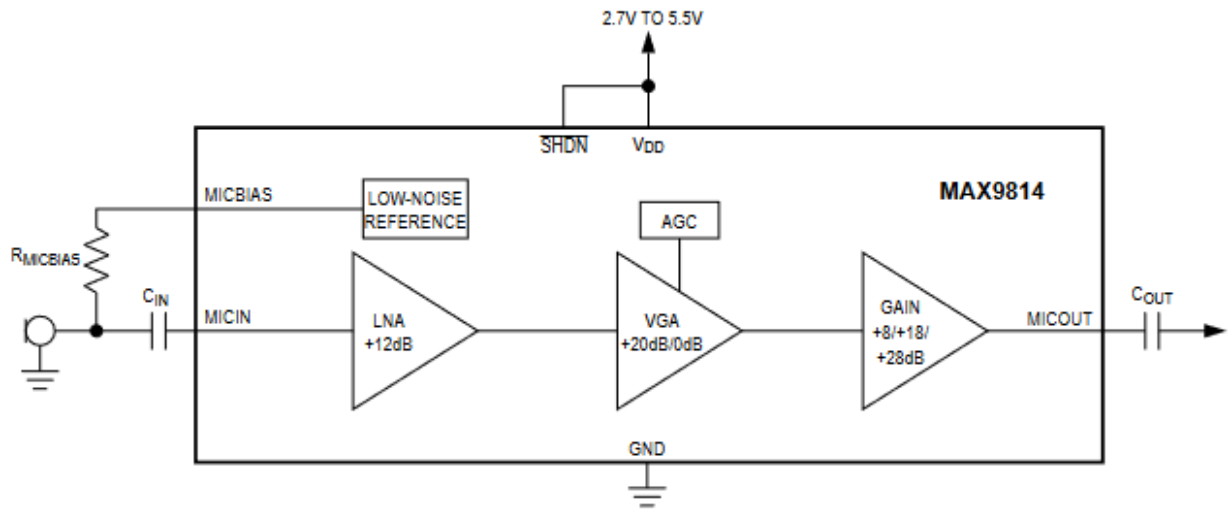


Рисунок 2.2 – Схема електрична-принципова МАХ9814

На рисунку 2.3 показано вигляд зверху та контакти модуля МАХ9814.

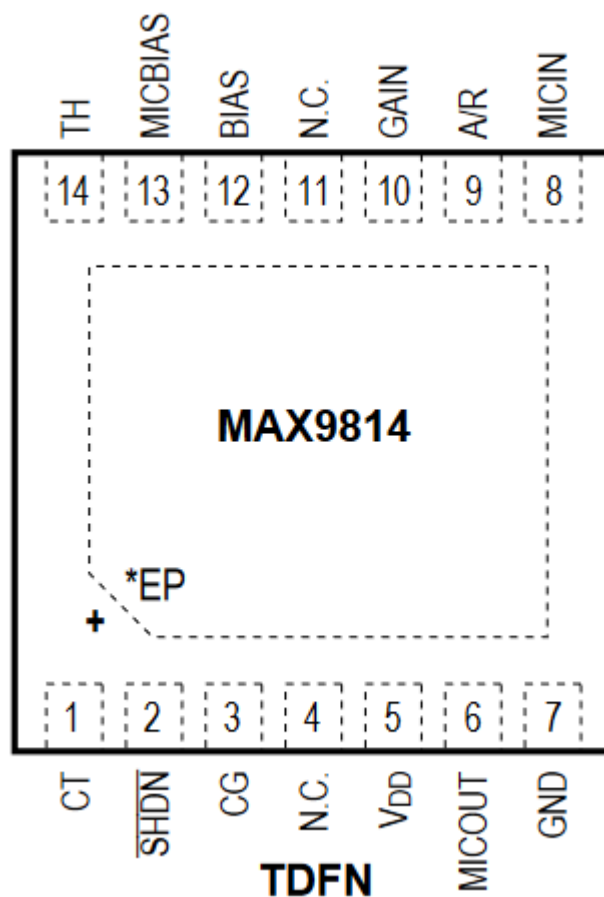


Рисунок 2.3 – Інтерфейс модуля МАХ9814

Змн.	Арк.	№ докум.	Підпис	Дата

У таблиці 2.1 показано виводи та призначення виводів підсилвача звуку з мікрофоном MAX9814

Таблиця 2.1 – Технічні характеристики MAX9814

Вивід	Мітка	Функції
1	CT	Підключення конденсатора контролю часу, використовується для контролювання часу атаки та спаду AGC.
2	SHDN	Активне низьке вимкнення підсилювача
3	CG	Регулятор зсуву підсилювача постійного струму. Підключається конденсатор 2,2 мкФ до виводу заземлення, щоб забезпечити нульове зміщення на виході.
4,11	N.C.	Немає зв'язку. Підключіться до GND.
5	V _{DD}	Байпас до GND за допомогою конденсатора 1 мкФ
6	MICOUT	Вихід підсилювача
7	GND	Заземлення
8	MICIN	Мікрофонний неінвертований вхід
9	A/R	Вибір коефіцієнта трирівневої атаки та спаду. Контролює співвідношення часу атаки до часу спаду для схема АРУ. A/R = GND: Співвідношення атаки/спаду становить 1:500 A/R = VDD: Співвідношення атаки/спаду становить 1:2000 A/R = Unconnected: Співвідношення атаки/спаду становить 1:4000

Змн.	Арк.	№ докум.	Підпис	Дата

КС КРБ 123.220.00.00 ПЗ

Арк.

36

Вивід	Мітка	Функції
10	GAIN	Регулювання коефіцієнта підсилення трирівневого підсилювача. GAIN = VDD, підсилення встановлено на 40 дБ. GAIN = GND, підсилення встановлено на 50 дБ. GAIN = не підключений, підсилення встановлено на 60 дБ.
12	BIAS	Зміщення підсилювача. Байпас до GND за допомогою конденсатора 0,47 мкФ.
13	MICBIAS	Вихід зміщення мікрофона
14	TH	Контроль порогового значення AGC підсилювача. Напруга TH встановлює поріг регулювання коефіцієнта підсилення. При підключенні TH до MICBIAS відбувається вимкнення підсилювача

MAX9814 представляє собою недорогий високоякісний мікрофонний підсилювач з автоматичним регулюванням коефіцієнту підсилення (AGC) і зміщенням мікрофона з низьким рівнем шуму.

MAX9814 складається з кількох окремих схем: попереднього підсилювача з низьким рівнем шуму, підсилювача зі змінним підсиленням (VGA), вихідного підсилювача, генератора напруги зміщення мікрофона та схеми керування коефіцієнта підсилення.

Генератор напруги внутрішнього зміщення мікрофона забезпечує зміщення 2 В, яке підходить для більшості електретних конденсаторних мікрофонів.

MAX9814 підсилює вхідний сигнал за трьома різними стадіями. На першому етапі вхід буферизується і посилюється через малозумний попередній підсилювач з коефіцієнтом підсилення 12 дБ.

Другий етап складається з VGA, керованого AGC. Комбінація VGA/AGC здатна змінювати коефіцієнт підсилення від 20 дБ до 0 дБ.

Вихідний підсилювач є кінцевим каскадом, в якому фіксований рівень підсилення 8 дБ, 18 дБ, 20 дБ програмується через один трирівневий логічний вхід. Без стиснення від AGC MAX9814 здатний забезпечувати підсилення 40 дБ, 50 дБ або 60 дБ.

Пристрій без автоматичного регулювання коефіцієнта підсилення зазнає відсікання на виході, коли до входу застосовується занадто велике значення підсилення.

AGC запобігає відсіканню на виході, коли на вхід застосовується занадто велике підсилення, усуваючи відсічення вихідного сигналу. На рисунку 2.4 показано порівняння мікрофонного входу з надмірним підсиленням з автоматичним регулюванням підсилення та без нього.

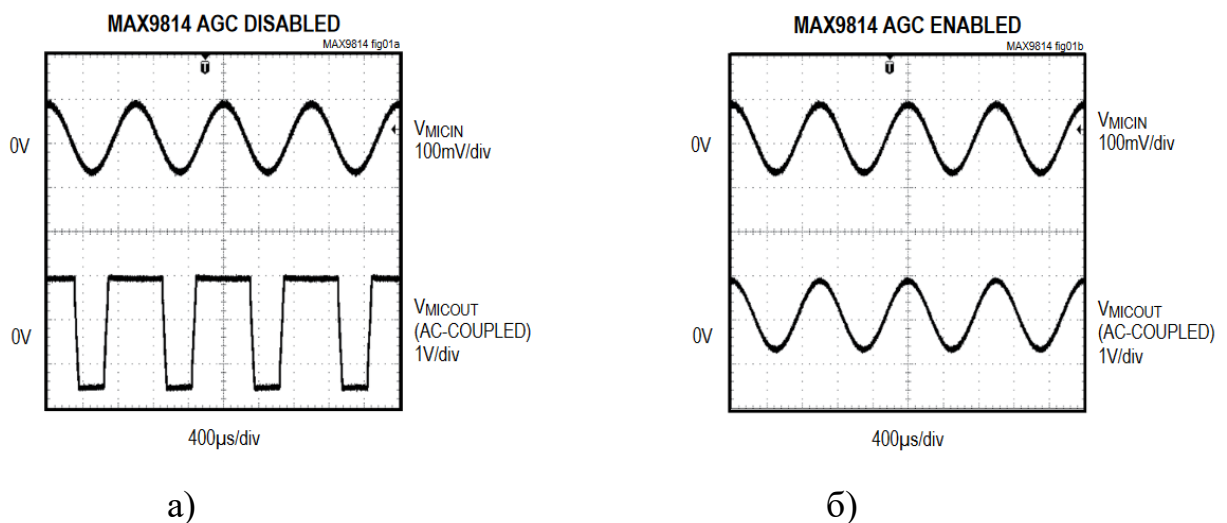


Рисунок 2.4 – Сигнал без автоматичного коефіцієнта регулювання (а) та з регулюванням підсилення (б)

Автоматичний регулятор коефіцієнту підсилення MAX9814 контролює відповідне значення, спочатку виявляючи, що вихідна напруга перевищила

задану межу. Потім коефіцієнт підсилення мікрофонного підсилювача зменшується за допомогою постійної часу, що вибирається, щоб виправити надмірну амплітуду вихідної напруги. Цей процес відомий як час атаки.

Коли вихідний сигнал згодом зменшується по амплітуді, посилення утримується у зниженому стані протягом короткого періоду, а потім повільно збільшується до нормального значення. Цей процес відомий як час утримання та звільнення – спаду.

Швидкість, з якою підсилювачі налаштовуються на зміну вхідних сигналів, встановлюється зовнішнім синхронізуючим конденсатором C_{ST} і напругою, що подається на A/R .

Поріг AGC можна встановити за допомогою регулювання напруги TN . Зменшення підсилення є функцією підсилення вхідного сигналу з максимальним загасанням до 20 дБ.

На рисунку 2.4 показано вплив вхідного пакету, що перевищує встановлену межу, час атаки на виході, час утримання та звільнення. Якщо час атаки та звільнення налаштовано на надто швидко відповідь, можуть виникнути відчутні артефакти, які часто описуються як «накачування» або «дихання», оскільки посилення швидко налаштовується відповідно до динаміки сигналу.

Для досягнення найкращих результатів потрібно відрегулювати постійну часу AGC відповідно до вихідного сигналу. Для програм, у яких музичні компакт-диски є основним джерелом звуку, рекомендований час атаки 160 мкс з часом відтворення 80 мс. Музичні додатки, зазвичай, потребують меншого часу відтворення, ніж голосовий або відеоконтент.

					КС КРБ 123.220.00.00 ПЗ	Арк.
						39
Змн.	Арк.	№ докум.	Підпис	Дата		

РОЗДІЛ 3 ПРОГРАМНА РЕАЛІЗАЦІЯ ТА НАЛАШТУВАННЯ КОМП'ЮТЕРНОЇ СИСТЕМИ ПЕРЕТВОРЕННЯ ГОЛОСОВИХ ПОВІДОМЛЕНЬ У ТЕКСТ

3.1 Проектування архітектури програмного забезпечення при перетворенні голосових повідомлень у текст

Одним з найбільш основних етапів розробки комп'ютерної системи перетворення голосових повідомлень у текст є проектування і реалізація програмної архітектури.

На концептуальному рівні архітектури передбачається використання звичайного комп'ютера, що дозволить реалізувати програмну модель, яка реалізує алгоритми машинного навчання для розпізнавання звуку. Наступний крок полягає у формуванні бібліотеки, яка буде збережена на Arduino Nano і забезпечить перетворення голосу у тексту. Однак, при цьому необхідно обчислити коефіцієнти цифрового фільтра, тобто реалізувати додаткову програму для їх обчислення.

Навчання щодо розпізнавання звуків проводиться на ПК, але навчена система повністю функціонує на Arduino. При цьому контролер надсилає зразки висловлювань на ПК, а ПК обчислює шаблони висловлювань.

Шаблони експортуються у вигляді файлу .h, який компілюється в скетч. Потім цей ескіз може розпізнавати голосові повідомлення без підключення до ПК.

Програма SpeechRecog1.exe для Windows обчислює коефіцієнти цифрового фільтра. Коефіцієнти цифрового фільтра експортуються як файл Coeffs.h. Скетч speakrecog1.ino (завантаження: крок 7) компілюється з використанням цих коефіцієнтів.

					КС КРБ 123.220.00.00 ПЗ			
Змн.	Арк.	№ докум.	Підпис	Дата				
Розроб.		Козінець Е.С.			, Програмна реалізація та налаштування комп'ютерної системи перетворення голосових повідомлень у текст	Літ.	Арк.	Аркуші
Перевірів		Луцкіє А.М.					40	
Консульт.		Лазарюк В.В.				ТНТУ, каф. КС, гр. СІс-43		
Н. Контр.		Луцик Н.С.						
Затверд.		Осухівська Г.М.						

Скетч speakrecog1.ino отримує зразки висловлювань і надсилає їх на ПК, а після цього SpeechRecog1.exe обчислює шаблони, які розпізнають ці висловлювання. При бажанні, SpeechRecog1.exe може збирати більше висловлювань для тестування. Він перевіряє шаблони за допомогою цих висловлювань.

Шаблони експортуються як файл Templates.h. Ескіз speakrecog2.ino (завантаження: крок 10) компілюється за допомогою файлів Templates.h і Coeffs.h. Після цього, використовуються шаблони для розпізнавання голосових повідомлень. На рисунку 3.1 показано загальний алгоритм і структуру компонентів при розпізнаванні голосових повідомлень.

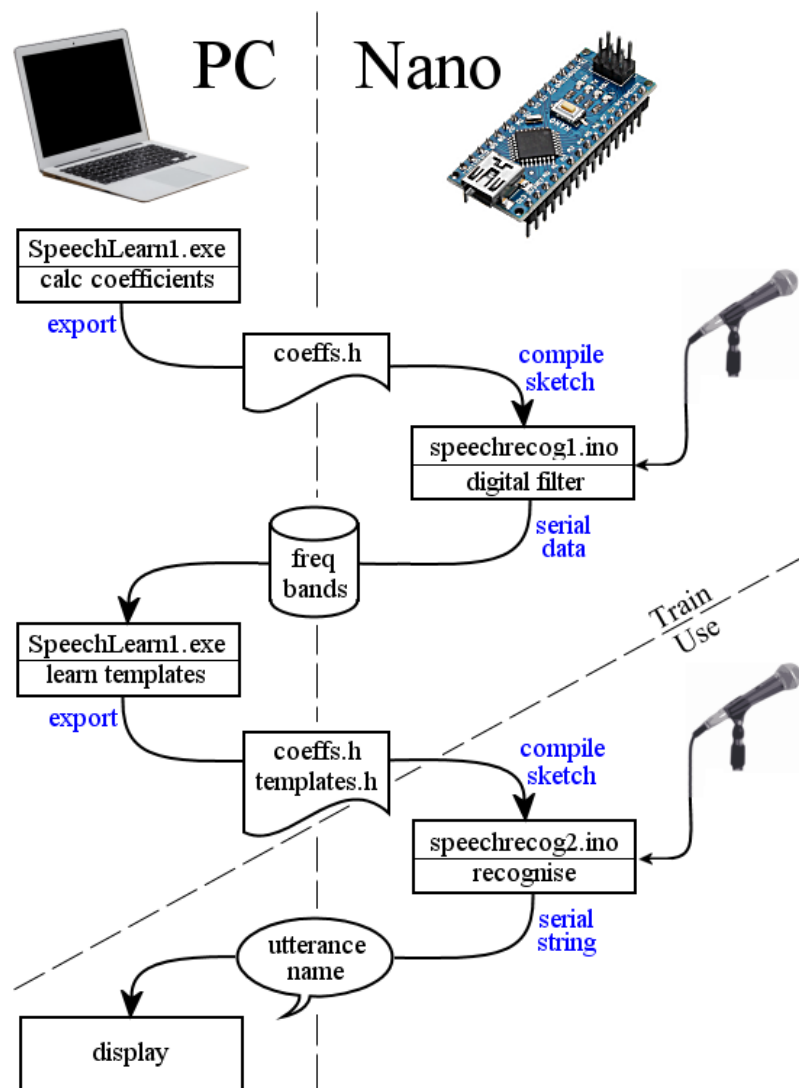


Рисунок 3.1 – Загальна архітектура програмного забезпечення перетворення голосових повідомлень у текст

Проаналізуємо більш детально кожен з програмних компонентів, які реалізуються при перетворенні голосових повідомлень у текст.

3.2 Обчислення коефіцієнтів цифрового частотного фільтру

Типове слово - "висловлювання" - може тривати секунду. Nano має лише 2 Кбайт оперативної пам'яті, тому не може зберігати всі зразки висловлювань, а їх аналіз відбувається доволі повільно. У зв'язку з цим, необхідно забезпечити аналіз значної частини повідомлень в режимі реального часу.

Розпізнавання мовлення, зазвичай, починається з вимірювання «енергії» у різних діапазонах частот, тобто амплітуди. Отже, перший етап полягає в тому, щоб пропустити вхідний сигнал через різні смугові фільтри. Найпопулярнішим способом фільтрації даних є виконання перетворення Фур'є на вході для отримання його спектру. Arduino Nano не має достатньої обчислювальної потужності для обчислення перетворення Фур'є, коли надходять зразки. (Nano може просто обчислити перетворення Фур'є, але недостатньо швидко.)

Більш дієвим способом поділу даних на діапазони є використання цифрових фільтрів. Цифровий фільтр виконує деякі нескладні обчислення для попередніх N вхідних вибірок i , можливо, попередніх N вибірок фільтра-виходу, щоб обчислити наступне вихідне значення фільтра. На схемі 3.2 показано типовий фільтр.

Програмно реалізація фільтра передбачає збереження двох попередніх вхідних значень та двох попередніх вихідних значень. Оскільки, у даному випадку зберігаються по 2 значення кожного із зразків входу і виходу, то такий фільтр є фільтром другого порядку. Якщо вихід залежить лише від попередніх вхідних значень, то він називається фільтром «скінченна імпульсна характеристика»: «FIR» – рекурсивний фільтр.

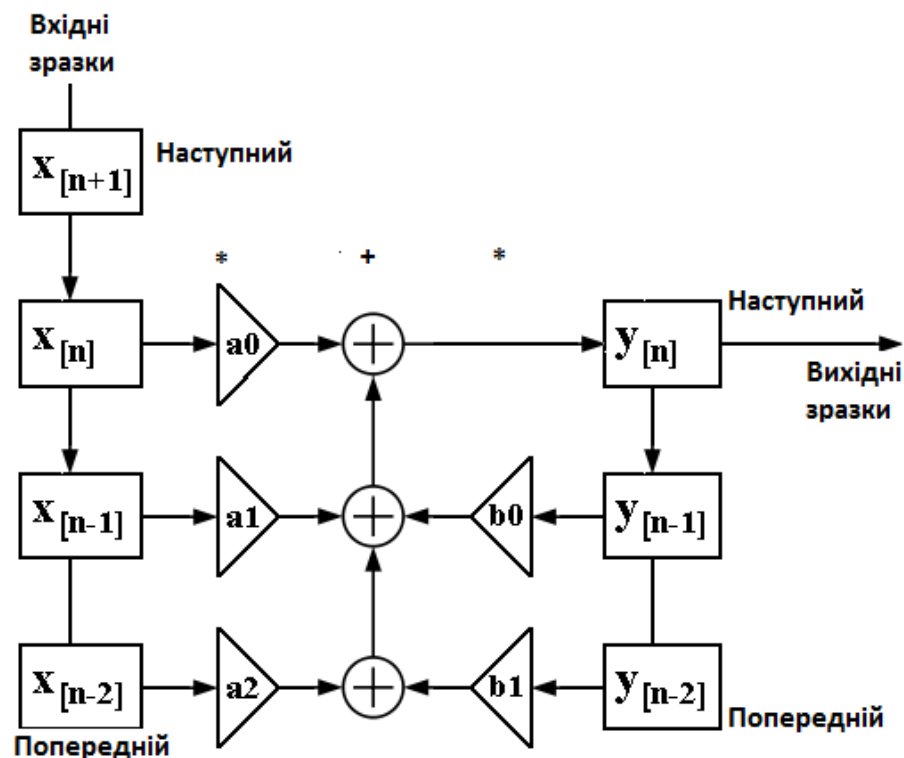


Рисунок 3.2 – Схема типового цифрового фільтра

Якщо вихід також залежить лише від попередніх вихідних значень, то це фільтр «нескінченна імпульсна характеристика»: «ІІР» – рекурсивний фільтр.

Для смугового фільтра порядок фільтра визначає, наскільки круто фільтр спадає вище і нижче частоти пропускання. Чим вище порядок, тим більше можливостей контролю над кривою відгуку фільтра.

Зрозуміло, що чим вище порядок, тим більше коефіцієнтів потрібно і тим більше обчислень потрібно виконати над вибіркою. FIR-фільтр вимагає більше коефіцієнтів, щоб отримати ту ж криву відгуку, що і фільтр ІІР. Але «ІІР»-фільтр менш стабільний.

При використанні рекурсивного фільтра легше помилитися. Це стосується виконання обчислень, оскільки складно знайти остаточне значення того, наскільки швидко $N_{\text{апо}}$ може виконувати додавання та множення. Це також залежить від того, як спроектована процедура вимірювання, зокрема, чи входить в алгоритм одержання та збереження значень вибірки. 8-бітове додавання займає від 0,4 до 0,9 мкс. Множення займає приблизно на 50% більше часу.

Змн.	Арк.	№ докум.	Підпис	Дата

16-бітове додавання або множення займає приблизно вдвічі більше, а 32-розрядне – займає приблизно в 5 разів більше часу. Операції ділення з плаваючою комою займає набагато більше часу, оскільки виконується у програмному забезпеченні. Нехай необхідно забезпечити частоту дискретизації 8000 sps, це 125 мкс на вибірку з 4 діапазонами частот, що становить 31 мкс на вибірку на діапазон.

Окрім цього, потрібно зібрати дані з АЦП, обчислити амплітуду смуг і зберегти результати в масиві. У результаті виникають обмеження, що супроводжуються десятком арифметичних операцій на вибірку, тому у даному випадку неможливо використовувати ПР-фільтр вищий, ніж фільтр другого порядку.

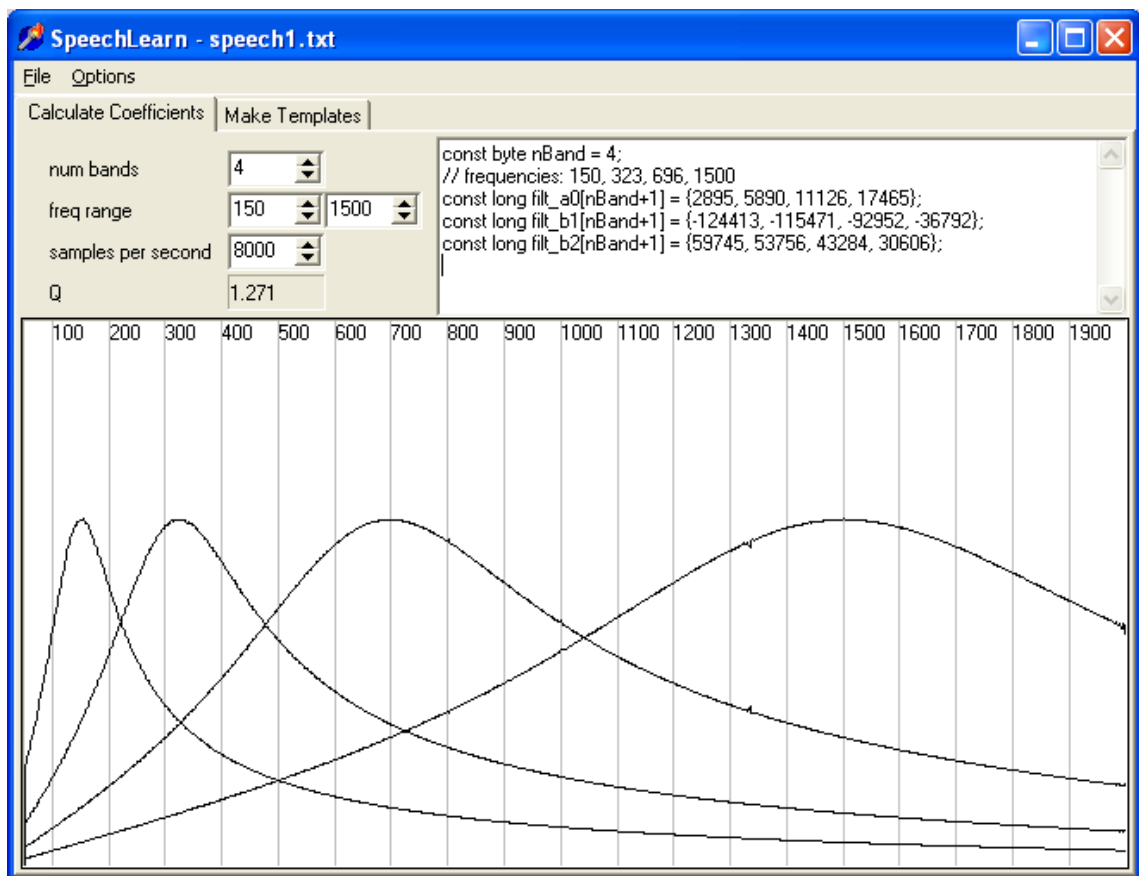


Рисунок 3.3 – Програма обчислення коефіцієнтів фільтра

Популярним цифровим фільтром є біквдратний – рекурсивний фільтр другого порядку. Очевидно, що хитрість будь-якого цифрового фільтра полягає

у пошуку правильних значень коефіцієнтів. Програмна реалізація обчислення коефіцієнтів для смугового біквдратного фільтра показана у лістингу 3.1.

Лістинг 3.1 – Фрагмент коду для обчислення коефіцієнтів біквдратного фільтра

```
K = tan(pi * f/sps);  
norm = 1 / (1 + K / Q + K * K);  
a0 = K / Q * norm;  
a1 = 0;  
a2 = -a0;  
b1 = 2 * (K * K - 1) * norm;  
b2 = (1 - K / Q + K * K) * norm;
```

У лістингу 3.1, f – частота смуги, sps – частота дискретизації, Q - ширина смуги. Значення $a1$ можна ігнорувати, оскільки воно дорівнює нулю. Значення $a2$ відповідає « $-a0$ », що дозволяє спростити обчислення.

Значення Q залежить від того, наскільки віддалені між собою смуги. Якщо є багато смуг близьких одна до одної, але вони не перекриваються, і потрібно збільшити значення Q або якщо смуги розташовані далеко одна від одної, а потрібно, щоб значення Q було настільки великим, щоб між ними були проміжки, то для смугового фільтра використовується формула « $Q = f_{center} / (f_{max} - f_{min})$ ».

У біквдратному фільтрі, якщо значення Q занадто велике, фільтр стає нестабільним. Переупорядкувавши рівняння, можемо обчислити значення коефіцієнтів фільтра як показано у лістингу 3.2.

Лістинг 3.2 – Обчислення значень коефіцієнтів фільтра

```
L1 = -b1*pd-b2*ppd+input;  
output = a0*(L1-ppd);  
ppd = pd;  
pd = L1;
```

Типові реалізації відгуків фільтрів показано вище на рисунку 3.3. Аналізуючи графіки можна помітити, що голосні звуки розрізняються за

					КС КРБ 123.220.00.00 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		45

формантами. Формант — це пік енергії спектра, а голосний звук розпізнається за відносними розмірами та частотами перших двох або трьох формантів.

Частота першої чоловічої форманти коливається від 250 Гц до 850 Гц, друга форманта – від 600 Гц до 2500 Гц. Форманти жінок вищі на 15%, а дітей приблизно на 35%. Звичайно, є великі індивідуальні відмінності.

Маючи лише 4 діапазони частот, не можна сподіватися на обчислення формантних частот, але вони впливатимуть на енергію в різних діапазонах. Смуги, які одержуються, залежатимуть від джерела звуку.

Оскільки ця система призначена для розпізнавання голосу з одного джерела звуку, її слід налаштувати на цей динамік.

При реалізації комп'ютерної системи перетворення голосових повідомлень у текст, коефіцієнти можна обчислити на ПК, але сам фактичний фільтр буде працювати на Arduino за допомогою цілочисельної арифметики.

Коефіцієнти є дійсними числами в діапазоні 0..2, тому їх множать на 0×10000 і перетворюють на цілі числа. Вихід АЦП знаходиться в діапазоні 0..1023, тому Arduino виконує відповідне зміщення, щоб повернути вихід фільтра в той самий діапазон.

Вхідні та вихідні значення фільтра зберігаються як 16-розрядні цілі числа, але деякі обчислення необхідно виконувати за допомогою 32-розрядних int, щоб уникнути проблем із переповненням.

Отже, тепер можна одержувати «енергію» (тобто амплітуду) 4 діапазонів частот. Додамо п'яту «смугу» для нульової швидкості переходу – «ZCR». ZCR – це наскільки часто сигнал перетинає значення 0 В. Це дуже хороший спосіб помітити фрикативні та неголосні губні звуки, такі як s, sh, th, t, k тощо.

У даному випадку потрібно використовувати невеликий гістерезис, який обчислює ZCR, щоб не вловлювати шум низького рівня. Відтепер можна ставитися однаково до 5 смуг.

Програма для Windows SpeechRecog1.exe, доступна на Github, обчислює коефіцієнти та експортує їх як файл Coeffs.h. При її використанні на вкладці «Розрахувати коефіцієнти» необхідно ввести частоти верхнього та нижнього діапазонів. Отримані коефіцієнти відображаються як оголошення цілих чисел у

					КС КРБ 123.220.00.00 ПЗ	Арк.
						46
Змн.	Арк.	№ докум.	Підпис	Дата		

пам'яті. Для експорту коефіцієнтів потрібно у меню File обрати ExportCoefficients, що дозволить зберегти константи у файл Coeffs.h. Далі одержаний файл можна підключити у програмному забезпеченні Arduino, шляхом копіювання Coeffs.h у той самий каталог, що й скетчі speakrecog1.ino та voicerecog2.ino. Наступний крок полягає у перекомпіляції скетчів, щоб вони могли ефективно виконувати смугову фільтрацію на Arduino.

«SpeechRecog1.exe» робить смугові фільтри «рівномірними» в логарифмічній шкалі. На рисунку 3.3 вісь частоти (вісь x) є лінійною. Параметр Q мав би бути однаковим для усіх смуг, що означає, що вони повинні бути рівномірно розташовані при використанні логарифмічної шкали.

Після того, як перекомпілюваний скетч speakrecog1.ino, він отримує зразки висловлювань і надсилає їх на ПК, щоб ПК міг визначити вибірки. Скетч speakrecog1.ino використовує шаблони для розпізнавання висловлювань, програмний код якого наведено у додатку Б.

3.3 Зразки (шаблони) голосових повідомлень

Arduino Nano розбиває все висловлювання на «сегменти» довжиною 50 мс (у деякій літературі вони називаються «кадри»). У кожному сегменті визначається амплітуда кожного з 5 діапазонів. Передбачається, що висловлювання має 13 сегментів, тож це загалом 65 16-бітових цілих чисел, що покривають час – 0,65 секунд.

Звичайно, деякі висловлювання можуть бути коротші, тому останні кілька сегментів будуть близькі до нуля, а деякі висловлювання довші, тому остання частина буде втрачена.

Вважається, що висловлювання починається, коли а загальна енергія в смугах перевищує поріг. Після того, як 13 сегментів буде збережено, потрібно вибрати, яке із шаблонних слів найбільше нагадує ці 65 чисел.

Припустимо, що звукове повідомлення, яке потрібно розпізнати, є цифрами від «нуля» до «дев'яти». Першим кроком є «нормалізація» вхідних даних. Тобто переконатися, що всі «три» висловлювання мають приблизно

					КС КРБ 123.220.00.00 ПЗ	Арк.
						47
Змн.	Арк.	№ докум.	Підпис	Дата		

однакову гучність. Отже, дані сегментної смуги помножаться на константу, щоб вони мали середню енергію (наприклад) 100. На рисунку 3.4 показано амплітудно-частотні спектр та його дискретизацію за сегментами при вимовлянні слова «Two».

Зазвичай, розробників будуть цікавити два найбільші піки в спектрі. Контроль формантів полягає у спостереженні за зміною частоти цих піків під час висловлювання. Легко запустити відстеження формантів, коли наявний повний спектр перетворення Фур'є або якщо використовується LPC, але воно не працює з 4 наявними діапазонами частот. Тому пропонується створити десять «шаблонів» для десяти різних цифр і обчислити різницю між вхідними даними та кожним із шаблонів. Шаблон — типовий приклад висловлювання. Кожен шаблон містить 65 цілочисельних значень, і кожне значення порівнюється з відповідним у вхідному висловлюванні.

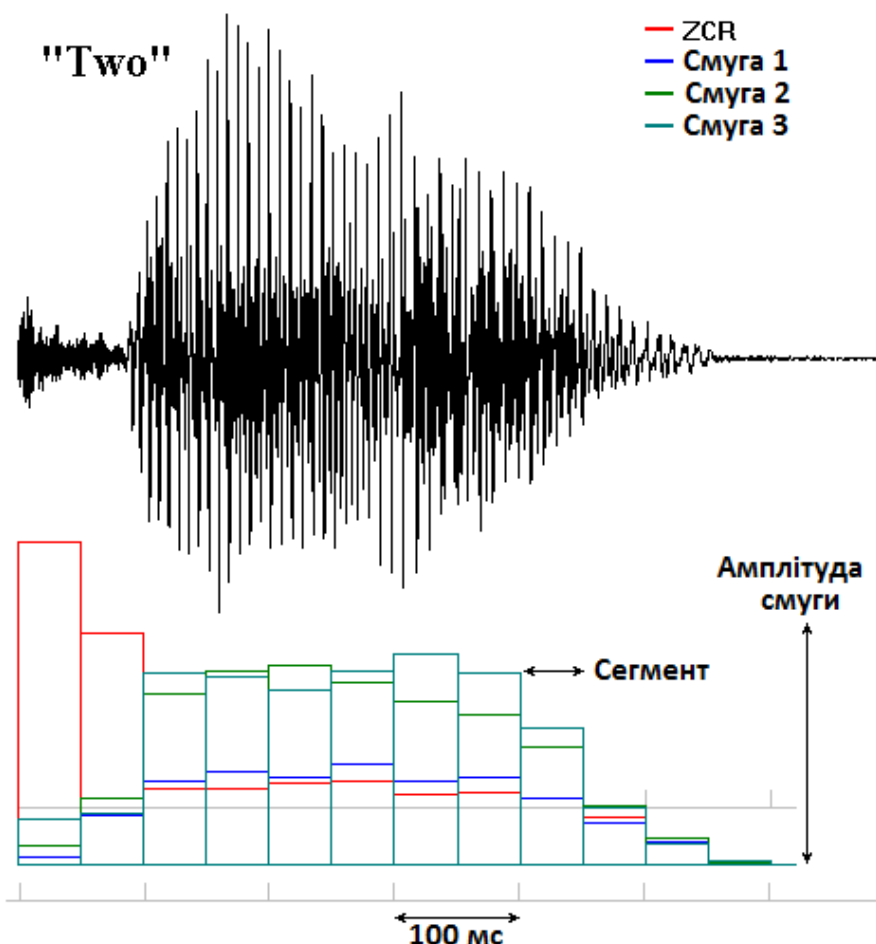


Рисунок 3.4 – Амплітуда смуг при вимовлянні слова «Two»

Змн.	Арк.	№ докум.	Підпис	Дата

Обчислення загальної різниці виконується так, як показано на рисунку 3.5.

```
for t = each template
  difference[t] = 0
  for seg = each segment
    for band = each band
      difference[t] = difference[t] + abs(template[t,seg,band] - incoming[seg,band])
```

Рисунок 3.5 – Фрагмент коду обчислення різниці між шаблонними значеннями і реальними

Але деякі смуги і сегменти є більш «важливими», ніж інші. Наприклад, "th" частина "three" досить відмінна у порівнянні з частиною "ee". Отже, до кожного числа в шаблоні потрібно додавати «важливість»:

$$\text{difference}[t] = \text{difference}[t] + \text{importance}[t, \text{seg}, \text{band}] * \text{abs}(\text{template}[t, \text{seg}, \text{band}] - \text{incoming}[\text{seg}, \text{band}])$$

Якщо значення для a (t,seg,band) сильно відрізняються для цього класу висловлювання, значення шаблону менш важливе, ніж якщо значення завжди майже однакові. Отже, «важливість» дорівнює 1.

При розпізнаванні голосових повідомлень до записаних висловлювань зазвичай застосовують «Динамічне викривлення часу». Візуалізація цього підходу показана на рисунку 3.5.

Dynamic Time Warping

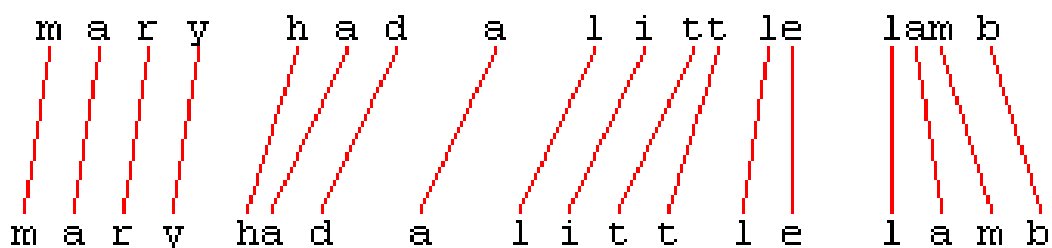


Рисунок 3.5 – Динамічне зміщення по часу

Ідея полягає в тому, що якщо порівнювати два приклади одного речення, можливо, перша половина в одному з прикладів була сказана трохи швидше.

Отже, можна розтягнути першу половину, або розтягнути все і трохи змістити вліво. При реалізації проекту застосовано динамічну деформацію часу до вхідного висловлювання і виконано його порівняння з шаблонами. Окрім цього, використано й інші підходи: переміщення та розтягнення всього висловлювання, а також переміщення, розтягнення та переміщення центральної частини.

Алгоритм полягає в тому, щоб знайти деформацію, яка найкраще відповідає шаблону вхідного висловлювання. Проблема в тому, що можна застосувати деформацію, щоб висловлювання краще відповідало правильному шаблону, але це також дозволить краще відповідати й неправильним шаблонам. Додаткові помилки, викликані поганими збігами, перевищують покращення, викликані хорошими збігами. Одне слово настільки коротке, що Dynamic Time Warping не є корисним. Розтягування всього висловлювання або його частини погіршує ситуацію. Однак, зміщення висловлювання вліво або вправо може дати більше хороших збігів, не створюючи неточностей. Зміщення всього висловлювання може становити 2 сегменти, проте він може зміщуватися і на частину відрізка. Тому зсув на 0,3 означає, що нове значення становить 70% поточного плюс 30% суміжного значення.

3.4 Навчання розпізнаванню голосових повідомлень

Програму «SpeechRecog1.exe», яку використовували для обчислення коефіцієнтів, також можна застосувати для формування шаблонів. Для цього потрібно підключити Arduino Nano до ПК, обравши правильний послідовний порт. Після цього необхідно завантажити скетч speakrecog1.ino на Arduino.

Скетч використовує АЦП для формування вибірки голосового повідомлення зі швидкістю приблизно 8000 секунд. Далі відбувається фільтрація одержаної вибірки на 4 діапазони частот та ZCR і зберігається 13 сегментів даних тривалістю 50 мс кожен.

Голосове повідомлення починається тоді, коли загальна енергія в смузі перевищує порогове значення. Після збереження 13 сегментів даних отримані 65 чисел передаються на ПК.

					КС КРБ 123.220.00.00 ПЗ	Арк.
						50
Змн.	Арк.	№ докум.	Підпис	Дата		

На рисунку 3.6 показано інтерфейс при формуванні шаблонів голосових повідомлень.

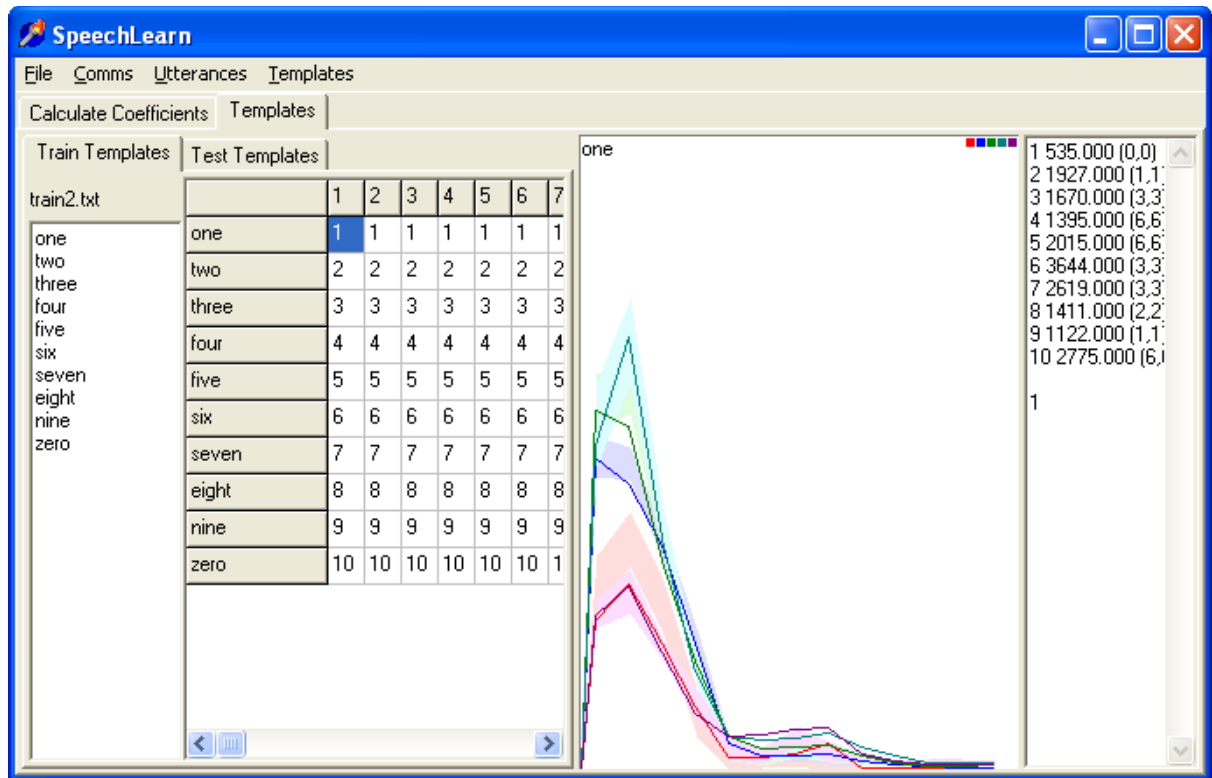


Рисунок 3.6 – Формування шаблонів голосових повідомлень

Обравши вкладку «Шаблони (Templates)», а потім «Навчання (Train Templates)» можна переглянути висловлювання і за допомогою них встановити шаблони. Натиснувши на пункт меню «Файл» і обравши «Відкрити», потрібно завантажити файл «Train2raw.txt».

Якщо клацнути на будь-яку клітинку в інтерфейсі програми, то можна побачити сегменти висловлювання. Якщо обрати кілька клітинок, то усі вони будуть відображені і їх можна порівняти між собою. У даному випадку горизонтальна вісь – це час, а вертикальна вісь – амплітуда кожної смуги. Червона смуга – це ZCR. Якщо клацнути лівий стовпець сітки, то відобразиться середнє значення та дисперсія. Програма обчислює середнє значення і дисперсію кожного сегменту смуги для кожного шаблону (рядка сітки). На рисунку 3.7 показано шаблони спектрів і сегментів для різних голосових повідомлень, які відповідають цифрам англійською мовою.

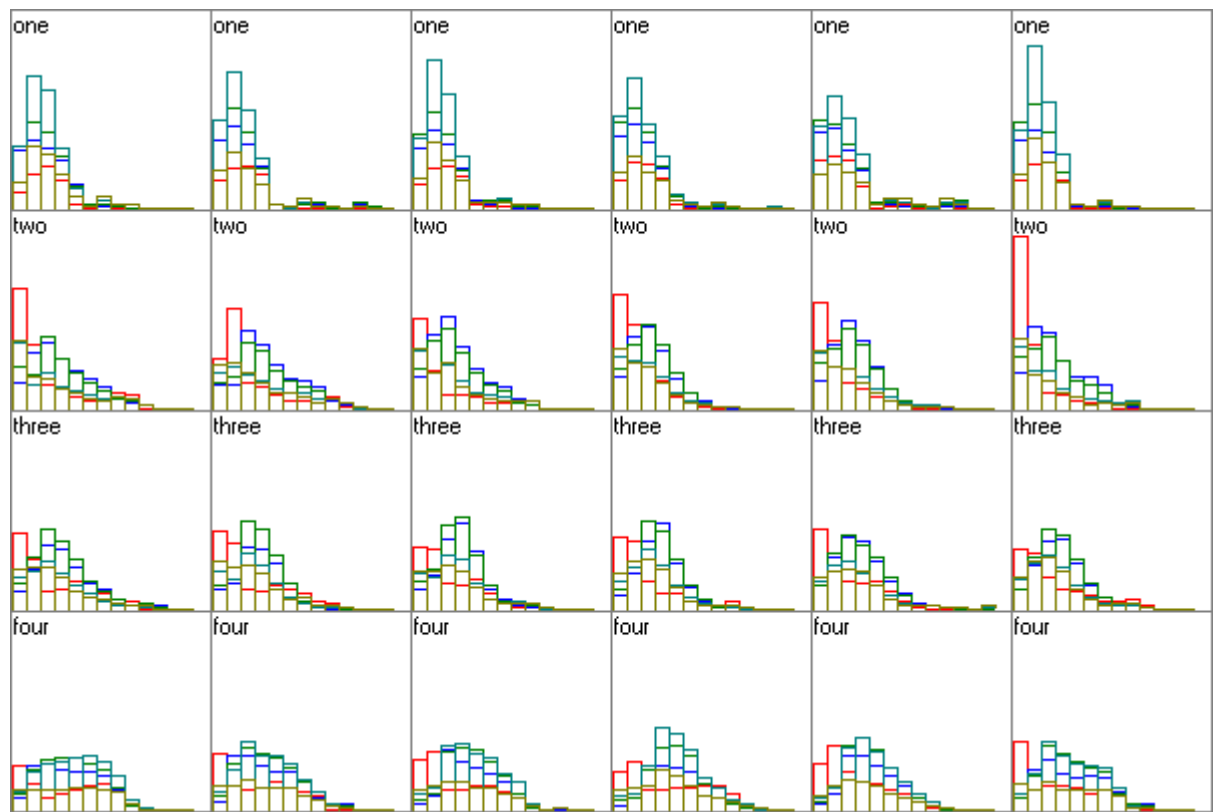


Рисунок 3.7 – Сегменти і спектр шаблонів

Іншими словами, 10 шаблонів тепер містять середнє значення даних. Можна порівняти кожен із реалізацій вибірки голосового повідомлення з будь-яким шаблоном. Дозволяється зміщувати вибірку ліворуч/праворуч, щоб отримати більш точну відповідність. Коли виконується натискання на клітинку для відображення голосового повідомлення, воно порівнюється з шаблоном для всіх рядків (тобто для всіх висловлювань). Результати відображаються в правій частині вікна. Числове значення інтерпретується як різниця («відстань») між голосовим повідомленням і поточним шаблоном. Найменша відстань є найбільш оптимальною, і вона відображається на сітці як найкраща.

Для розпізнавання усіх голосових повідомлень необхідно обрати відповідне меню – «Utterances|Recognise|RecogniseAll». У даному випадку отримано близько 30% поганих збігів. «Трійка» часто виглядала як «сімка», а «четвірка» — як «нуль». Тому шаблони варто трохи впорядкувати. Для цього потрібно обрати меню «Templates|OptimalShift». Далі кожна із вибірок зміщується вліво або вправо, поки не відбудеться найкраща відповідність

шаблону для цього висловлювання. Це означає, що потрібно кожну навчальну вибірку налаштувати так, щоб вона відповідала його шаблону. Середнє значення і дисперсія перераховується для кожного шаблону. У результаті проведених маніпуляцій побудовано шаблони висловлювань для конкретного спікера з хорошим набором навчальної вибірки, що відповідає точності майже на 100%.

Далі потрібно перейти на вкладку «Test Templates» і завантажити файл «Test2.txt» з голосовими повідомленнями, за допомогою яких можна перевірити шаблони.

Наступний крок полягає у порівняти кожного із тестових наборів з кожним шаблоном. Результати не такі хороші, однак їх точність вища за 90%. Тепер, коли відкрити COM-порт і сформувавши голосове повідомлення у мікрофон, відобразиться висловлювання.

Коли уже є сформований набір шаблонів, який відповідає очікуванням користувача, то його можна експортувати до Arduino як файл Templates.h. Для цього потрібно обрати пункт меню File|ExportTemplates, щоб зберегти шаблони як файл Templates.h, а після того програмно підключити у скетчі Arduino. Перед цим варто скопіювати файл Templates.h в той самий каталог, що й скетч speakrecog2.ino, ще раз перекомпілювати його. Це дозволить використовувати шаблони для розпізнавання мови на Arduino. Окрім цього, потрібно скопіювати файл «Coeffs.h» у цю ж директорію.

3.5 Комплексне розпізнавання голосових повідомлень

Скетч speakrecog2.ino виконує розпізнавання голосових повідомлень на Arduino Nano, Uno, Mini тощо. Після того, як виконано копіювання файлу Templates.h і файл Coeffs.h в той самий каталог, що й speakrecog2.ino, потрібно повторно скомпілювати його та завантажити в Arduino.

Загальна схема функціонування системи перетворення голосових повідомлень у текст показана на рисунку 3.8.

					КС КРБ 123.220.00.00 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		53

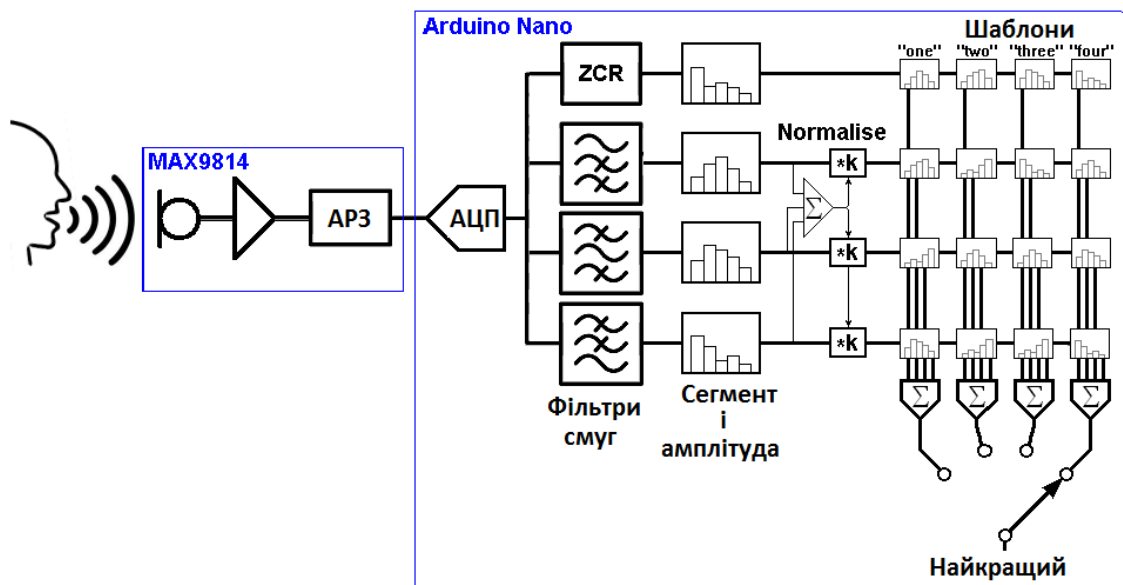


Рисунок 3.8 – Функціональна архітектура комп'ютерної системи перетворення голосових повідомлень у текст

Комп'ютерна система функціонує на Arduino і використовує АЦП для оцифрування вхідного аудіосигналу. Результатом є 16-бітове ціле число з центром на 0. Кілька смугових рекурсивних цифрових фільтрів розділяють сигнал на діапазони частот. Час поділено на сегменти по 50 мс. Вимірюється амплітуда кожної смуги в кожному сегменті. Фіксована кількість сегментів формують «висловлювання» (голосове повідомлення). Висловлювання починається, коли амплітуда перевищує поріг. Вимірюється середня амплітуда всього висловлювання, щоб дані можна було нормалізувати. Розраховується швидкість перетину нуля (ZCR) сигналу.

Значення амплітуди смуги порівнюються зі значеннями шаблону. Сегменти можна зміщувати вліво або вправо, щоб покращити їх відповідність. Виявляється найкраща відповідність. Скетч `speakrecog2.ino` надсилає текст розпізнаного слова на ПК або інший пристрій, здатний відтворити текст через послідовний порт. Таким чином, у роботі спроектовано та реалізовано програмне забезпечення для перетворення голосових повідомлень у текст з використанням не дорогих та ефективних для вирішення поставлених задач апаратних пристроїв.

Змн.	Арк.	№ докум.	Підпис	Дата

РОЗДІЛ 4 БЕЗПЕКА ЖИТТЄДІЯЛЬНОСТІ, ОСНОВИ ОХОРОНИ ПРАЦІ

4.1 Способи проведення штучного дихання та масажу серця

Процедуру штучного дихання необхідно проводити при закупорюванні трахеї та бронхів блювотними масами, згустками крові (при пораненні шиї, обличчя), водою при утопленні не надходить повітря до легень. Крім того, дихання може припинитися, коли людина непритомніє і через розслаблення м'язів у неї западає корінь язика, прикриваючи вхід у гортань; при порушенні функції центру дихання внаслідок тяжких черепно-мозкових травм, крововиливу в мозок, множинних переломів ребер; при ушкодженні легень і плеври. У потерпілого з'являється спочатку блідість, потім синюшність шкірних покривів обличчя. Дихальні рухи прискорюються, стають нерегулярними (іноді рідшають), а через деякий час, якщо не буде подано потрібної допомоги, припиняються зовсім.

У таких випадках, насамперед перевіряють порожнину рота і пробувають відновити прохідність верхніх дихальних шляхів. Для цього вказівним пальцем, обмотаним бинтом чи носовою хусточкою, звільняють рот від блювотних мас, слизу, згустків крові, виймають знімні зубні протези. Коли запав язик, голову потерпілого слід повернути набік і підтримати щелепу.

Коли ж потерпілий не дихає, треба перевірити, чи не припинилася в нього серцева діяльність, і негайно розпочати штучне дихання. Для цього його кладуть спиною на жорстку поверхню (на підлогу автомашини, на дорогу), підмостивши під лопатки зроблений з одягу валик. Потерпілого розгинають і закидають його голову назад так, щоб максимально відвести підборіддя від грудної клітки, розстібають комір і пояс, що утруднюють дихання.

Найефективнішим способом штучного дихання є вдування повітря в ніс

					КС КРБ 123.220.00.00 ПЗ			
<i>Змн.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>				
<i>Розроб.</i>		<i>Козінець Е.С.</i>			<i>Безпека життєдіяльності, основи охорони праці</i>	<i>Літ.</i>	<i>Арк.</i>	<i>Аркуші</i>
<i>Перевірів</i>		<i>Луцкіє А.М.</i>					55	
<i>Консульт.</i>		<i>Лазарюк В.В.</i>				<i>ТНТУ, каф. КС, гр. СІс-43</i>		
<i>Н. Контр.</i>		<i>Луцик Н.С.</i>						
<i>Затверд.</i>		<i>Осухівська Г.М.</i>						

або рот потерпілого. Такий спосіб називають «рот до рота» або «рот до носа». У ніс вдувають повітря тоді, коли у хворого ушкоджені губи, нижня або верхня щелепа.

Штучне дихання способом «рот до рота» роблять так. Той, хто надає допомогу, стає на коліна біля голови потерпілого, краще з правого боку, і кладе ліву руку на його лоб, затискуючи пальцями йому ніс; прикривши рот потерпілого марлею або носовою хусточкою, робить глибокий вдих, а потім, притиснувши рот до рота потерпілого, вдуває в нього повітря, роблячи енергійний видих. Внаслідок такого вдування, що замінює вдих, повітря надходить у легені потерпілого. Згодом завдяки еластичності легеневої тканини і грудної клітки настає пасивний видих. При цьому рот потерпілого повинен бути відкритий. Вдувати повітря слід ритмічно, з однаковим інтервалом, 15...20 разів на хвилину [13].

Так само роблять штучне дихання способом «рот до носа», тільки повітря вдувають через ніс, а рот потерпілого закривають. Ефективність штучного дихання підвищується, коли застосовувати спеціальні S-подібні повітроводи, що забезпечують прохідність верхніх дихальних шляхів.

Штучне дихання слід виконувати доти, доки у потерпілого не відновиться самостійне ритмічне дихання. Якщо штучне дихання розпочинають робити своєчасно і проводять його правильно, то самостійне дихання відновлюється через 1...2 хв [13].

Робити штучне дихання нелегко. Той, хто надає допомогу, особливо при надто енергійному вдуванні повітря, може навіть відчувати запаморочення, слабкість і знепритомніти. Тому бажано, щоб приблизно через кожні 60 с особи, які виконують штучне дихання, змінювали одна одну. Це підвищить ефективність допомоги потерпілому.

Часом при тяжких ушкодженнях припиняється дихання і серцева діяльність. Тоді у потерпілого настає так звана клінічна смерть. У нього не промацується пульсація сонної артерії на шиї, не прослуховується серцебиття, припиняється дихання, зіниці розширюються, шкірні покриви й слизові оболонки губ синюшні. Через 5...6 хв після настання клінічної смерті

					КС КРБ 123.220.00.00 ПЗ	Арк.
						56
Змн.	Арк.	№ докум.	Підпис	Дата		

відбуваються необоротні зміни в нервових клітинах центрів головного мозку, що керують життєво важливими функціями організму. Внаслідок цього настає біологічна смерть. З клінічної смерті потерпілого можуть вивести лише негайні (не пізніш як через 4...5 хв після її настання) й енергійні заходи, спрямовані на відновлення серцевої діяльності та дихання. З цією метою роблять штучне дихання і непрямий (закритий) масаж серця.

Роблячи непрямий масаж серця, потерпілого кладуть спиною на тверду поверхню (шосе, дорога). Той, хто на дає допомогу, стає на коліна збоку (краще справа) біля потерпілого і, поклавши кисті рук одну на другу в нижній частині груднини, робить енергійний поштовх, натискуючи на груднину так, щоб вона зміщувалася приблизно на 4...5 см у напрямі до хребта. При цьому стискається серце між грудниною і хребтом, кров із порожнини серця виштовхується в кровоносні судини. Оскільки стінка грудної клітки еластична, то вона повертається у вихідне положення, а порожнини серця наповнюються кров'ю. Поштовхи слід робити ритмічно, близько 60 разів на хвилину.

Дітям, особливо молодшого віку, непрямий масаж серця можна робити однією рукою або навіть двома пальцями, але частоту поштовхів треба збільшити до 100...120 разів на хвилину [14].

Іноді непрямий масаж серця поєднують із штучним диханням. Щоправда, це потребує чималих зусиль. Тому бажано, щоб потерпілим у стані клінічної смерті допомогу надавали двоє: один робить непрямий масаж серця, другий — штучне дихання. Після кожного вдуття повітря в легені потерпілого чотири рази натискують на груднину. Під час вдуття повітря не можна стискувати грудну клітку.

Непрямий масаж серця і штучне дихання звичайно може робити й одна людина. У такому разі після кожного вдуття повітря в рот або в ніс натискують чотири рази на груднину.

Показником ефективності непрямого масажу серця і штучного дихання є порожевіння шкірних покривів, звуження зіниць, поява на великих артеріях (стегновій, сонній) синхронно натискуванню на груднину пульсових поштовхів і, нарешті, відновлення самостійного дихання й серцебиття. Непрямий масаж

					КС КРБ 123.220.00.00 ПЗ	Арк.
						57
Змн.	Арк.	№ докум.	Підпис	Дата		

серця і штучне дихання слід проводити до відновлення серцевої діяльності й дихання. Коли вони не дають ефекту, то їх припиняють тільки після огляду потерпілого медичним працівником.

4.2 Заходи щодо боротьби з шкідливою дією ультразвуку на організм людини

ДСН 3.3.6.037 – 99 „Державні санітарні норми виробничого шуму, ультразвуку та інфразвуку” визначає допустимі рівні впливу різних видів шуму на організм людини.

Ультразвук представляє собою механічні коливання пружного середовища і відрізняються від звукових хвиль більш високою частотою, що перевищує верхній поріг чутності. Ультразвукові хвилі поширюються в будь-якому пружному середовищі (рідкому, твердому, газоподібному), краще в металах, воді, гірше в повітрі. При проходженні в різних середовищах ультразвукові хвилі по різному поглинаються ними. Абсорбційні властивості м'язової тканини вище жирової, в сірій мозковій речовині поглинання майже в два рази вище, ніж в білій. Найбільше поглинання спостерігається в кістковій тканині, найменше – в спинномозковій речовині.

Поглинання ультразвуку супроводжується нагріванням середовища, причому термічний ефект посилюється з підвищенням частоти коливань. Також при проходженні ультразвуку в рідині виникає ефект кавітації (пароутворення та наступного схлопування бульбашок пари з одночасною конденсацією пара в струмі рідини, що супроводжується шумом та гідравлічними ударами, утворення в рідині порожнин, які заповнюються паром самої рідини). З цим явищем пов'язана механічна дія ультразвуку. Утворення кавітаційних порожнин супроводжується появою на поверхнях електричних зарядів, що викликають люмінесцентне світіння, іонізацію молекул води. З цими явищами пов'язані хімічні ефекти — окислювальна дія ультразвуку, прискорення хімічних реакцій, руйнування органічних сполук.

					КС КРБ 123.220.00.00 ПЗ	Арк.
						58
Змн.	Арк.	№ докум.	Підпис	Дата		

Вищезазначені прояви дії ультразвуку широко використовується в багатьох галузях промисловості для інтенсифікації процесів хімічного травлення, нанесення металевого покриття, очищення, змивання та знежирення деталей і виробів, дефектоскопії (оцінка якості зварних швів, структури сплаву).

За способом передачі від джерела до людини ультразвук поділяють на: повітряний (передається через повітря) та контактний (передається на руки людини, що працює через тверде чи рідинне середовище).

За спектром ультразвук поділяють на: низькочастотний (коливання частотою від $1,2 \times 10^4$ до $1,0 \times 10^6$ Гц, що передаються людині повітряним чи контактним шляхом) та високочастотний (коливання частотою від $1,0 \times 10^5$ до $1,0 \times 10^9$ Гц, що передаються людині тільки контактним шляхом).

Параметрами повітряного ультразвуку, що нормуються у робочій зоні, є рівні звукового тиску в третинооктавних смугах з середньгеометричними частотами 12,5; 16,0; 20,0; 25,0; 31,5; 40,0; 63,0; 80,0; 100,0 кГц. Для контактного ультразвуку параметром, що нормується, є пікове значення віброшвидкості в частотному діапазоні від 0,1 МГц до 10,0 МГц або його логарифмічний рівень. Допускається, також, застосовувати як параметр інтенсивність ультразвуку.

Ультразвук, так само як і інфразвук, орган слуху людини не сприймає, однак він може спричиняти біль голови, загальну втому, розлади серцево-судинної та нервової систем. При клінічному обстеженні може бути виявлений астеничний синдром. У осіб, що тривалий час зайняті експериментальною роботою на ультразвукових установках, іноді спостерігаються діенцефальні порушення (зниження ваги, різкий підйом вмісту цукру в крові з повільним зниженням до вихідного рівня, підвищення механічного збудження м'язів тощо). Можливі порушення периферичної нервової системи (оніміння, зниження чутливості, гіпергідроз) порушення вестибулярного апарата. Периферичні порушення обумовлені переважно контактним впливом ультразвукових коливань.

Заходи щодо зниження шкідливої дії ультразвуку мають бути направлені на обмеження впливу шуму та ультразвуку, що передається через повітря, а також контактним засобом.

					КС КРБ 123.220.00.00 ПЗ	Арк.
						59
Змн.	Арк.	№ докум.	Підпис	Дата		

У випадку неможливості зниження інтенсивності шуму та ультразвуку в джерелі через повітря, найбільш ефективним є звукоізоляція обладнання (використання звукоізоляційних кожухів, захисних екранів, звукоізольованих кабін, розміщення ультразвукового устаткування в окремому звукоізольованому приміщенні).

Для унеможливлення впливу контактного ультразвуку роботи з коливними рідинними середовищами (завантаження, вивантаження) необхідно проводити при вимкненому джерелі ультразвуку або використовувати для цього спеціальні інструменти, що мають ручки з еластичним покриттям, наприклад, гумовим. Як засоби індивідуального захисту використовують протишумові навушники (дія через повітря) та двошарові рукавички із зовнішнім гумовим шаром (контактна дія).

Робітники, працюючі в умовах впливу ультразвуку, згідно з наказом МОЗ № 246 від 21.05.2007. «Про затвердження порядку проведення медичних оглядів працівників певних категорій» підлягають щорічному періодичному медичному огляду з обов'язковим залученням до складу лікарняної комісії невропатолога, офтальмолога, хірурга та проведенням досліджень вібраційної чутливості (за показанням). Обмеженням при прийомі на роботу в умовах впливу ультразвуку (окрім загальних) є хронічні захворювання периферичної нервової системи, а також захворювання артерій, периферичний ангіоспазм.

					КС КРБ 123.220.00.00 ПЗ	Арк.
						60
Змн.	Арк.	№ докум.	Підпис	Дата		

ВИСНОВКИ

Під час побудови проекту комп'ютерної системи перетворення голосових повідомлень у текст розв'язано усі поставлені задачі кваліфікаційної роботи і досягнуто її мети. Основними результатами кваліфікаційної роботи є наступні:

- спроектовано концептуальну схему комп'ютерної системи перетворення голосових повідомлень у текст;
- обгрунтовано вибір мікроконтролера для управління та опрацювання голосових повідомлень;
- обгрунтовано вибір мікрофона з автоматичним регулюванням коефіцієнта підсилення;
- розраховано коефіцієнти підсилення при використанні мікрофона, який прикріплений до штанги;
- розроблено програмне забезпечення для формування шаблонів голосових повідомлень;
- розроблено програмне забезпечення для визначення амплітуди і спектру голосового повідомлення;
- проведено налаштування мікрофона для формування голосових повідомлень одним користувачем;
- проведено навчання і формування шаблонів голосових повідомлень на основі мовлення одного користувача;
- забезпечено відправлення голосового повідомлення у вигляді тексту на зовнішній пристрій за допомогою COM-порта.

Застосування обгрунтованих у роботі апаратних пристроїв та відповідного програмного забезпечення дало змогу реалізувати повно функціональну та продуктивну систему перетворення простих голосових повідомлень у текст, що в подальшому може бути використано для управління та передачі даних у текстовому форматі для виконання команд периферійними пристроями та для вирішення задач розпізнавання мовлення.

					КС КРБ 123.220.00.00 ПЗ	Арк.
						61
Змн.	Арк.	№ докум.	Підпис	Дата		

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Arduino MKR1000 WIFI. URL: <https://store.arduino.cc/arduino-mkr1000-wifi> (дата звернення 20.04.2022 р).
2. Getting Started with Arduino products. URL: <https://www.arduino.cc/en/Guide> (дата звернення 19.04.2022 р.).
3. UART-USB перехідник на FT232RL. URL: <https://arduino.ua/prod2544-uart-usb-perehodnik-na-ft232rl> (дата звернення 08.05.2022 р.)
4. GM65 в 1D і 2D кодів, сканер штрих-коду зчитування QR-кодів модуль 5 В постійного струму. URL: <https://gudvin.com.ua/ua/p1321406325-gm65-kodiv-skaner.html> (дата звернення 14.05.2022 р.)
5. Петин В. Датчики для Arduino и Raspberry Pi в проектах Internet of Things. БХВ-Петербург. 2016. 320 с.
6. Shelgaonkar S.K. Creating a smart home environment with IOT driven home appliances. GRIN Verlag. 2016 p. 80 p.
7. ElNashar A., El-saidny M. Practical Guide to LTE-A, VoLTE and IoT: Paving the way towards 5G. John Wiley & Sons. 2018. 480 p.
8. Петин В. Arduino и Raspberry Pi в проектах Internet of Things. БХВ Петербург. Електроніка. 2016. 320 с.
9. IoT-шлюзы: автоматизация производства на уровне Индустрии 4.0 - Control Engineering Russia URL: http://www.controlengrussia.com/internet-veshhej/iot_gateways/ (дата звернення 15.10.2020 р.).
10. Waher P. Learning Internet of Things. Packt Publishing. 2015. 286 p.
11. Автоматизация бизнес-процесів підприємства. URL: http://stud.com.ua/37085/ekonomika/avtomatizatsiya_biznes_protsesiv_pidpriyemstva (дата звернення 20.05.2022 р.).
12. Политанский Р.Л. Система передачи данных с шифрованием хаотическими последовательностями. Технология и конструирование в электронной аппаратуре. 2014. № 2-3. С. 28–34.
13. НПАОП 0.00-7.15-18 «Вимоги щодо безпеки та захисту здоров'я працівників під час роботи з екранними пристроями». Київ. 2018.

					КС КРБ 123.220.00.00 ПЗ	Арк.
						62
Змн.	Арк.	№ докум.	Підпис	Дата		

14. Бедрій Я. Основи охорони праці користувачів персональних комп'ютерів: навчальний посібник для студентів ВНЗ та інженерів-практиків. Навчальна книга-Богдан. 2014. 144 с.

					<i>КС КРБ 123.220.00.00 ПЗ</i>	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		63

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.

“ ____ ” _____ 2022 р

КОМП'ЮТЕРНА СИСТЕМА ПЕРЕТВОРЕННЯ

ГОЛОСОВИХ ПОВІДОМЛЕНЬ У ТЕКСТ

ТЕХНІЧНЕ ЗАВДАННЯ

на 11 листках

Вид робіт:

Кваліфікаційна робота

На здобуття освітнього ступеня «Бакалавр»

Спеціальність 123 «Комп'ютерна інженерія»

«УЗГОДЖЕНО»

Керівник кваліфікаційної роботи

_____ к.т.н., доц. Луцків А.М.

« ____ » _____ 2022 р.

«ВИКОНАВЕЦЬ»

Студент групи СІс-43

_____ Козінець Е.С.

« ____ » _____ 2022 р.

Тернопіль 2022

1 Загальні відомості

1.1 Повна назва та її умовне позначення

Повна назва теми кваліфікаційної роботи: «Комп'ютерна система перетворення голосових повідомлень у текст».

Умовне позначення кваліфікаційної роботи: КС КРБ 123.220.00.00

1.2 Виконавець

Студент групи СІс-43, факультету комп'ютерно-інформаційних систем і програмної інженерії, кафедри комп'ютерних систем та мереж, Тернопільського національного технічного університету імені Івана Пулюя, Козінець Ернест Сергійович.

1.3 Підстава для виконання роботи

Підставою для виконання кваліфікаційної роботи є наказ по університету (№ 4.7-97 від 10.02.2022 р.)

1.4 Планові терміни початку та завершення роботи

Плановий термін початку виконання кваліфікаційної роботи – 23.03.2022 р.

Плановий термін завершення виконання кваліфікаційної роботи – 23.06.2022 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 ІСО, ГОСТ, 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 на 90% , 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

Комп'ютерна система перетворення голосових повідомлень у текст покликана розв'язати основну задачу трансляції звуку у текст, що в подальшому може бути використано для накопичення текстової інформації та її опрацювання.

Галузями, де можливе застосування такої системи, є збір та аналіз вимог при проектуванні окремих технічних та нетехнічних систем, при проведенні переговорів для забезпечення автоматизації процесу створення стенограм, при формуванні систем управління командами для зовнішніх пристроїв і т.п.

Перевагами і перспективами застосування комп'ютерної системи перетворення голосових повідомлень у текст є забезпечення можливості розширення джерел одержання інформації з фіксацією їх у текстовому вигляді.

Більшість систем сьогодні спроектовані і базуються на використанні методів і систем штучного інтелекту, що дає можливість доволі ефективно розпізнавати голосові повідомлення, однак це вимагає використання значних апаратних ресурсів, а відповідно й вартість таких систем є досить високою.

У даній роботі необхідно побудувати комп'ютерну систему перетворення голосових повідомлень у текст, яка б використовувала мінімальні характеристики

апаратного забезпечення, забезпечувала мобільність системи та була зорієнтованою на одного користувача.

У комп'ютерній системі необхідно передбачити можливість навчання щодо розпізнавання голосових повідомлень, які в подальшому можуть бути трансльовані у текст або відповідні команди.

Проектована комп'ютерна система перетворення голосових повідомлень у текст потребує обладнання для одержання голосових повідомлень, фільтрації сторонніх шумів, а також програмного забезпечення для навчання та розпізнавання голосових повідомлень. Окрім цього, необхідна наявність засобів відображення тексту, сформованого на основі одержаного голосового повідомлення та шаблонів для визначення відповідності звуків.

2.2 Мета створення системи

Мета проектування комп'ютерної системи перетворення голосових повідомлень у текст полягає в автоматизації процесу одержання і трансляції звукових сигналів у текст з подальшим його відображенням та зберіганням на зовнішніх пристроях. При цьому необхідно забезпечити ефективність використання апаратних ресурсів та продуктивність функціонування комп'ютерної системи.

Для того, щоб досягти поставленої мети роботи необхідно розв'язати ряд задач, найбільш важливими з яких є:

- визначення потенційних рішень проектування системи на основі аналізу існуючих рішень;
- побудова концептуальної моделі комп'ютерної системи перетворення голосових повідомлень у текст;
- обґрунтування і дослідження технічних характеристик апаратного забезпечення комп'ютерної системи;
- проектування функціональної та схеми електричної принципової підсистеми одержання та перетворення аналогового сигналу у цифровий;

- реалізація системного програмного забезпечення та програмування мікроконтролера для одержання і трансляції звуку у текст;
- обґрунтування прикладного програмного забезпечення для навчання системи перетворення голосових повідомлень у текст;
- налаштування параметрів комп'ютерної системи на використання одним користувачем – генератором звуку.

2.3 Характеристика об'єкту

2.3.1 Основні задачі та функції об'єкту

Найбільш важливими задачами і функціями, які має забезпечувати комп'ютерна система перетворення голосових повідомлень у текст полягає у:

- здатності одержання аналогового звукового сигналу;
- трансформації звукового сигналу у цифровий вигляд;
- фільтрації шумів;
- створенні навчальної вибірки для інтерпретації голосових повідомлень;
- формуванні шаблонів звуків;
- безпосередньому перетворення одержаного сигналу у текст;
- відображенні текстового повідомлення із голосового.

Програмне забезпечення комп'ютерної системи передбачає виконання задач на системному і прикладному рівнях. Системне програмне забезпечення використовується в процесі функціонування готової системи, а прикладне – в основному при проектуванні та налаштуванні параметрів апаратного забезпечення.

Передбачається, що основними складовими комп'ютерної системи на рівні апаратного забезпечення є:

- мікрофон;
- підсилювач мікрофона;
- аналогово-цифровий перетворювач;

- мікроконтролер управління процесом перетворення голосових повідомлень у текст;
- ПК або пристрій відображення та зберігання тексту.

Доступність налаштувань використовуваних апаратних і програмних ресурсів комп'ютерної системи перетворення голосових повідомлень у текст має бути захищеним на рівні фізичного доступу та процедури авторизації. Доступ до одержаного тексту, у випадку зберігання на ПК також передбачає наявність засобів аутентифікації.

3 Вимоги до системи

3.1 Вимоги до системи в цілому

В цілому, вимоги, які висуваються до проектованої комп'ютерної системи перетворення голосових повідомлень у текст полягають у забезпеченні визначеної функціональності та продуктивності в процесу опрацювання аналогового звукового сигналу.

При створенні прототипу комп'ютерної системи доцільним є використанням плат макетування Arduino, які є недорогими по вартості і забезпечують високу продуктивність при залученні мінімуму апаратних ресурсів. Одержаний у результаті перетворення текст повинен бути надісланий до зовнішнього пристрою з дисплеєм.

У загальному випадку, вимоги до комп'ютерної системи перетворення голосових повідомлень у текст можна представити наступним чином:

- можливість отримання аналогового сигналу голосового повідомлення;
- здатність перетворення аналогового сигналу у цифровий вигляд;
- можливість розкладу сигналу за 4 смугами;
- можливість аналізу частотного та амплітудного спектрів;
- можливість формування шаблонів оцифрованих голосових повідомлень;

- здатність визначення коефіцієнтів підсилення в залежності від типу джерела звуку;
- забезпечення продуктивності перетворення звуку у часі. Близькому до реального;
- можливість передачі повідомлення у текстовому вигляді до зовнішнього пристрою (дисплею, ПК);
- здатність відображення текстового повідомлення на дисплеї периферійного пристрою.

3.1.1 Вимоги до структури та функціонування системи

Структурно компоненти комп'ютерної системи перетворення голосових повідомлень у текст визначаються її функціональною орієнтованістю і передбачають використання наступного апаратного забезпечення:

- плата прототипування Arduino Uno;
- мікрофон послідовного порту з автоматичним регулюванням коефіцієнта підсилення;
- підсилювач мікрофона;
- аналогово-цифровий перетворювач;
- середовище програмування для Arduino;
- прикладне програмне забезпечення аналізу звукових сигналів.

Комп'ютерна система перетворення голосових повідомлень у текст повинна забезпечувати стабільність роботи і дозволяти одержувати аналоговий звуковий сигнал, декодувати його у цифровий, виконувати відповідні трансформації для розбиття сигналу на смуги, порівнювати характеристики сигналу із шаблонними значенням, передавати повідомлення у текстовому вигляді у зовнішній пристрій.

Архітектура комп'ютерної системи передбачає виконання пристрою у вигляді модуля з мікрофоном, що під'єднується до ПК для обміну даними.

3.2 Вимоги до способів та засобів зв'язку між компонентами системи

Спосіб зв'язку компонентів системи, зокрема, плати прототипування Arduino Nano та мікрофона з підсилювачем, передбачає безпосереднє під'єднання визначених контактів за допомогою провідників.

З'єднання та обмін даними Arduino з комп'ютером, або іншим зовнішнім пристроєм з дисплеєм, виконується за допомогою з'єднання з лінійним інтерфейсом через СОМ-порт.

3.2.1 Вимоги по діагностуванню системи

Процедури щодо проведення діагностичних заходів модулів комп'ютерної системи перетворення голосових повідомлень у текст повинні бути регламентовані відповідним графіком та проводитися у відповідності до інструкцій з експлуатації компонентів системи.

Діагностика плати макетування Arduino Nano виконується кожен раз при подачі на неї живлення шляхом самоініціалізації окремих складових. Якщо проявляються помилки або збої у роботі системи необхідно вжити заходи для негайно їх усунення.

3.2.2 Перспективи розвитку, модернізація системи

Модернізацію комп'ютерної системи доцільно проводити у випадку, коли є необхідність розширити її функціональні можливості, зокрема в контексті опрацювання і формування шаблонів голосових повідомлень, а також розпізнавання великої кількості голосових повідомлень, згенерованих різними джерелами звуку. Такий варіант модернізації перехід на більш потужні плати макетування або використання мінікомп'ютерів по типу Raspberry Pi.

До перспектив розвитку комп'ютерної системи перетворення голосових повідомлень у текст належить можливість створення моделей машинного навчання для розпізнавання голосових повідомлень та їх інтеграція й сумісність з апаратними модулями системи.

3.2.3 Вимоги до надійності системи

До важливих аспектів надійності комп'ютерної системи перетворення голосових повідомлень у текст належать:

- здатність забезпечувати визначений рівень працездатності у визначених умовах використання ;
- здатність до відновлення працездатності після збоїв і помилок;
- можливість зберігати налаштування апаратних компонентів системи;
- здатність працювати у визначеному режимі підсилення звукового сигналу;
- можливість адаптовуватися до нових джерел звукового сигналу без втрати точності розпізнавання звуку;
- стабільність роботи АЦП;
- наявність механізму авторизованого доступу до налаштувань комп'ютерної системи.

3.2.4 Вимоги до функцій та задач, які виконує система

Функціональні вимоги, а відповідно й задачі, які повинна забезпечувати комп'ютерна система перетворення голосових повідомлень у текст:

- здатність отримання аналогового звукового сигналу;
- можливість оцифрування аналогового звукового сигналу;
- здатність виконувати розбиття сигналу на смуги;
- можливість використання цифрового рекурсивного фільтра;
- здатність налаштування параметрів підсилення звуку;
- можливість визначати частотний спектр, сегменти і амплітуду звукового сигналу;
- здатність формування навчальної вибірки та шаблонів голосових повідомлень;
- можливість обміну та забезпечення сумісності між компонентами системи і зовнішніми периферійними пристроями;
- здатність налаштування параметрів системи під конкретного користувача.

3.2.5 Вимоги до апаратного забезпечення

Основні вимоги і обмеження для апаратних пристроїв комп'ютерної системи перетворення голосових повідомлень у текст:

- мікроконтролер ATmega 368;
- мікрофон з підсилювач MAX9814;
- персональний комп'ютер або ноутбук.

Характеристики ПК для налаштування комп'ютерної системи, проведення розрахунків і навчання розпізнавання голосових повідомлень повинні відповідати таким технічним характеристикам:

- частота процесора $\geq 2,0$ ГГц;
- розмір оперативної пам'яті 8 Гб;
- дисковий простір ≥ 120 Гб.

3.2.6 Вимоги до програмного забезпечення

Для реалізації системного ПЗ Arduino Nano щодо керування та забезпечення сумісності з іншими структурними компонентами системи має використовувати середовище Arduino IDE, мова програмування – C/C++. Системне програмне забезпечення комп'ютера повинно забезпечувати підтримку необхідних сервісів і середовищ для налаштування комп'ютерної системи.

4 Вимоги до документації

Документація повинна відповідати вимогам ЄСКД та ДСТУ

Комплект документації повинен складатись з:

- пояснювальної записки;
- графічного матеріалу:
 1. Існуючі системи розпізнавання голосових повідомлень
 2. Функціональна схема комп'ютерної системи
 3. Структура Arduino Nano
 4. Схема підсилювача звукового сигналу

5. Архітектура комп'ютерної системи

*Примітка: У комплект документації можуть вноситися міни 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 та аналіз технічного завдання	23.03-28.03.2022
2	Аналіз принципів функціонування систем перетворення голосових повідомлень у текст	29.03-07.04.2022
3	Проектування структури комп'ютерної системи перетворення голосових повідомлень у текст	08.04-20.04.2022
4	Обґрунтування вибору та аналіз технічних характеристик апаратного забезпечення системи	20.04-26.04.2022
5	Розробка алгоритмів та програмного забезпечення для управління процесом одержання і перетворення голосових повідомлень	27.04-10.05.2022
6	Налаштування параметрів комп'ютерної системи	10.05-20.05.2022
7	Розробка інструкцій з використання комп'ютерної системи	21.05-26.05.2022
8	Безпека життєдіяльності, основи охорони праці	27.05-30.05.2022
9	Оформлення кваліфікаційної роботи	30.05-10.06.2022
10	Попередній захист кваліфікаційної роботи	10.06-20.06.2022
11	Захист кваліфікаційної роботи	22.06-24.06.2022

6 Додаткові умови виконання кваліфікаційної роботи

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

Додаток Б.

Програмний код для безперервного зчитування штрих-кодів

```
#include <Arduino.h>
#include <SPI.h>
#include "Coeffs.h"

//-----
// Defines, constants and Typedefs
//-----

// pins
const int AUDIO_IN = A7;

// get register bit - faster: doesn't turn it into 0/1
#ifndef getBit
#define getBit(sfr, bit) (_SFR_BYTE(sfr) & _BV(bit))
#endif

//-----
// Global Constants
//-----

const byte NumSegments = 13;
const byte SegmentSize = 50; //in mS
const byte hyster = 2;
const byte thresh = 100;

//-----
// Global Variables
//-----

word CurBandData[NumSegments][nBand + 1]; //current band data

//-----
// GetSerial
//-----

byte GetSerial() {
    while ( Serial.available() == 0 ) ;
    return Serial.read();
}
```

```

//-----
// PollBands
//-----
bool PollBands(bool init)
{
    bool IsPos, Collecting;
    static unsigned long prevTime;
    byte band, seg, val1, val2;
    const byte hyster = 20;
    static int zero = 500;
    static byte curSegment = 255;
    long val;
    word zcr;
    static int valmax[10];
    static int pd[10] = {0, 0, 0, 0, 0, 0, 0, 0, 0, 0};
    static int ppd[10] = {0, 0, 0, 0, 0, 0, 0, 0, 0, 0};
    static int ppval = 0;
    static int pval = 0;

    if (init)
    {
        memset(pd, 0, sizeof(pd));
        memset(ppd, 0, sizeof(ppd));
        memset(CurBandData, 0, sizeof(CurBandData));
        pval = 0;
        ppval = 0;
        return false;
    }

    val = 0;
    IsPos = true;
    Collecting = false;

    for (curSegment = 0; curSegment < NumSegments; curSegment++) {
        zcr = 0;
        prevTime = 0;
        memset(valmax, 0, sizeof(valmax));

        prevTime = millis();
        while (millis() - prevTime < SegmentSize)
        {
            while (!getBit(ADCSRA, ADIF)) ; // wait for ADC
            val1 = ADCL;
            val2 = ADCH;
            bitSet(ADCSRA, ADIF); // clear the flag
            bitSet(ADCSRA, ADSC); // start ADC conversion
            val = val1;
            val += val2 << 8;

            if (val < zero)

```

```

    zero--; else
    zero++;
    val = val - zero;

    if (!Collecting) {
        if (abs(val) > thresh)
            Collecting = true; else
            return false;
    }

    if (IsPos)
    {
        if (val < -hyster)
        {
            IsPos = false;
            zcr++;
        }
    } else {
        if (val > +hyster)
        {
            IsPos = true;
            zcr++;
        }
    }
    ppval = pval;
    pval = val;

    for (band = 0; band < nBand; band++)
    {
        int L1, L2;
        L1 = ((-(filt_b1[band]) * pd[band] - filt_b2[band] *
ppd[band]) >> 16) + val;
        L2 = (filt_a0[band] * L1 - filt_a0[band] * ppd[band]) >> 16;
        ppd[band] = pd[band];
        pd[band] = L1;
        if (abs(L2) > valmax[band])
            valmax[band]++;
    }
}

for (band = 0; band < nBand; band++)
    CurBandData[curSegment][band + 1] = valmax[band];
CurBandData[curSegment][0] = zcr;
}

if (Collecting) {
    PrintCurBandData();
    delay(500);
}

return Collecting;
}

```

```

//-----
-----
// PrintCurBandData
//-----
-----
void PrintCurBandData(void)
{
    byte seg, i, band;

    Serial.println("a");
    for (seg = 0; seg < NumSegments; seg++) {
        for (band = 0; band <= nBand; band++) {
            Serial.print(CurBandData[seg][band]);
            Serial.print(" ");
        }
        Serial.println("");
    }
}

//-----
-----
// setup
//-----
-----
void setup(void)
{
    Serial.begin(57600);
    Serial.println("speechrecog");

    pinMode(AUDIO_IN, INPUT);
    analogReference(EXTERNAL);
    analogRead(AUDIO_IN); // initialise ADC to read audio input

    Serial.println("0 0 0 0 0 0 350");

    PollBands(true);
}

//-----
-----
// Main routines
// loop
//-----
-----
void loop(void)
{
    PollBands(false);
}

```