

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное автономное
образовательное учреждение высшего образования
«Южно-Уральский государственный университет
(национальный исследовательский университет)»

Высшая школа электроники и компьютерных наук
Кафедра «Электронные вычислительные машины»

РАБОТА ПРОВЕРЕНА

Рецензент

_____ 2023 г.
«__» _____

ДОПУСТИТЬ К ЗАЩИТЕ

Заведующий кафедрой ЭВМ

_____ Д.В. Топольский

«__» _____ 2023 г.

Прототип многорежимного слухового аппарата

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА
К ВЫПУСКНОЙ КВАЛИФИКАЦИОННОЙ РАБОТЕ
ЮУРГУ-090401.2023.222 ПЗ ВКР

Руководитель работы,
к.т.н., доцент каф. ЭВМ

_____ И.Л. Кафтаников

«__» _____ 2023 г.

Автор работы,
студент группы КЭ-222

_____ С.В. Буторин

«__» _____ 2023 г.

Нормоконтролёр,
ст. преподаватель каф. ЭВМ

_____ И.Г. Топольская

«__» _____ 2023 г.

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное автономное
образовательное учреждение высшего образования
«Южно-Уральский государственный университет
(национальный исследовательский университет)»
Высшая школа электроники и компьютерных наук
Кафедра «Электронные вычислительные машины»

УТВЕРЖДАЮ

Заведующий кафедрой ЭВМ

_____ Д.В. Топольский

«___» _____ 2023 г.

ЗАДАНИЕ

на выпускную квалификационную работу магистра
студенту группы КЭ-222
Буторину Сергею Вячеславовичу,
обучающемуся по направлению
09.04.01 «Информатика и вычислительная техника»

1. **Тема работы:** «Прототип многорежимного слухового аппарата»,
утверждена приказом по университету от 25 апреля 2023 года №753-
13/12.
2. **Срок сдачи студентом законченной работы:** 1 июня 2023 г.
3. **Исходные данные к работе:**
 - 3.1 Преобразователь должен принять аналоговый сигнал на микрофон;
 - 3.2 Преобразователь должен оценивать какой из звуковых отрезков является высокочастотным или низкочастотным, а какой нет;
 - 3.3 Преобразователь должен изменить высокочастотный сигнал таким образом, чтобы пользователь смог слышать тот же самый сигнал на другой частоте, более низкой или высокой;

3.4 Преобразователь должен вернуть аналоговый сигнал на моно-наушник.

4. Перечень подлежащих разработке вопросов:

4.1 Обзор литературы;

4.2 Проектирование модели слухового преобразователя;

4.3 Реализация слухового преобразователя;

4.4 Тестирование прототипа слухового преобразователя.

5. Дата выдачи задания: 1 декабря 2022 г.

Руководитель работы:

Доцент каф. ЭВМ, к.т.н. _____ /И.Л. Кафтанников /

Студент _____ /С.В. Буторин /

КАЛЕНДАРНЫЙ ПЛАН

Этап	Срок сдачи	Подпись руководителя
Обзор литературы	10.03.2023	
Проектирование модели слухового преобразователя	21.03.2023	
Реализация слухового преобразователя	04.04.2023	
Тестирование прототипа слухового преобразователя	25.04.2023	
Написание отчета	15.05.2023	
Компоновка текста работы и сдача на нормоконтроль	16.05.2023	
Подготовка презентации и доклада	24.05.2023	

Руководитель работы _____ /И.Л. Кафтанников /

Студент _____ /С.В. Буторин /

Аннотация

С.В. Буторин. Прототип многорежимного слухового аппарата. – Челябинск: ФГАОУ ВО «ЮУрГУ (НИУ)», ВШ ЭКН; 2023, 50 с., 17 ил., библиогр. список –20 наим.

В рамках выпускной квалификационной работы производится аналитический обзор современной научно-технической, нормативной, методической литературы, затрагивающей научно-техническую проблему, и рассмотрения технических решений, касающихся проблем пожилых людей, страдающих тугоухостью, был разработан прототип модели слухового преобразователя.

После изучения теоретических аспектов природного явления звука были сформулированы требования к модели, такие как обработка звукового сигнала и организация его понижения и повышения частоты.

В ходе выполнения практической работы была осуществлена реализация преобразователя и произведено сравнение исходного и полученного звукового отрезка.

ОГЛАВЛЕНИЕ

ВВЕДЕНИЕ	7
1 ОБЗОР ЛИТЕРАТУРЫ	8
1.1 Причины возникновения пресбиакузиса и негативные последствия.....	8
1.2 Методы слухопротезирования.....	10
1.3 Типы понижения частоты	12
2 ПРОЕКТИРОВАНИЕ МОДЕЛИ СЛУХОВОГО ПРЕОБРАЗОВАТЕЛЯ	20
2.1 Теория физического явления звука.....	20
2.2 Прототип преобразователя	23
2.3 Проектирование слухового преобразователя	26
3 РЕАЛИЗАЦИЯ СЛУХОВОГО ПРЕОБРАЗОВАТЕЛЯ.....	30
4 ТЕСТИРОВАНИЕ ПРОТОТИПА СЛУХОВОГО ПРЕОБРАЗОВАТЕЛЯ.....	44
ЗАКЛЮЧЕНИЕ	47
БИБЛИОГРАФИЧЕСКИЙ СПИСОК	48

ВВЕДЕНИЕ

Тугоухость является одной из актуальных проблем оториноларингологии и геронтологии. Количество людей, которые живут с нарушениями органов слуха, неуклонно увеличивается, что ведет к широкому изучению вопроса создания наиболее эффективных методов медицинской и социальной реабилитации. По данным ВОЗ на 2018 год, по всему миру насчитывалось около полумиллиона человек, страдающих пресбиакузисом, и если не принимать никаких действий, то к 2050 году количество людей с тяжелыми формами потери слуха может превысить 900 млн [1].

Для решения такой важной проблемы применяются слуховые аппараты. При этом могут использоваться как обычные устройства, которые увеличивают громкость звука, так и такие, которые изменяют частоту. Последние направлены на улучшение слышимости звуков и голосов высокой тональности, например женских или детских. Использование слуховых аппаратов при тугоухости оказывает положительную динамику на качество жизни пациентов. Кроме того, у таких людей значительно лучше память, чем у тех, кто не пользуется такими устройствами. Большая часть пользователей слуховых аппаратов отмечают такие положительные моменты, как более высокую самостоятельность и независимость.

Целью работы является создание прототипа слухового преобразователя.

Для написания выпускной квалификационной работы были сформулированы следующие задачи:

- 1) изучить причины и специфику заболевания;
- 2) проанализировать принцип работы модели слухового преобразователя;
- 3) разработать модель слухового преобразователя;
- 4) Протестировать прототип.

1 ОБЗОР ЛИТЕРАТУРЫ

1.1 Причины возникновения пресбиакузиса и негативные последствия

Пресбиакузис — нарушение слуха вследствие потери восприятия преимущественно высоких частот, наблюдающееся у людей пожилого возраста. Рассматривается рядом исследователей как дегенеративный процесс. Причиной этому, главным образом, служит рост урбанизации: с увеличением доли населения возрастает количество людей, страдающих данным недугом [2]. Виной этому становится шум, который является неотъемлемой частью городской среды. Главными его источниками можно назвать транспорт, производства, отсутствие достаточного количества деревьев, которые могли бы поглощать громкие звуки. Кроме этого существуют и вполне естественные причины возникновения тугоухости [3]. К ним относится старение, особенности организма конкретного человека, другие заболевания, которые могут повлечь потерю слуха. Статистические данные говорят о глобальности проблемы, связанных с данным заболеванием [4]. Так, среди людей, страдающих тугоухостью 41% старше 70 лет, 28% – от 60 до 69 лет, а в свою очередь от 14 до 49 лет лишь 12%. Это объясняется тем, что в пожилом возрасте у человека теряется возможность слышать звуки с высокой частотностью (рисунок 1).

Люди любых возрастов, имеющие подобные нарушения органов слуха, могут встретиться в обычной жизни с такими проблемами как депрессивное состояние, уход в себя, изоляция от внешнего мира, потеря интереса к окружающей жизни [5]. Пресбиакузис в пожилом возрасте определяется ухудшением восприятия того что человек слышит в шумной обстановке, замедлением работы области мозга, отвечающей за обработку слуховых

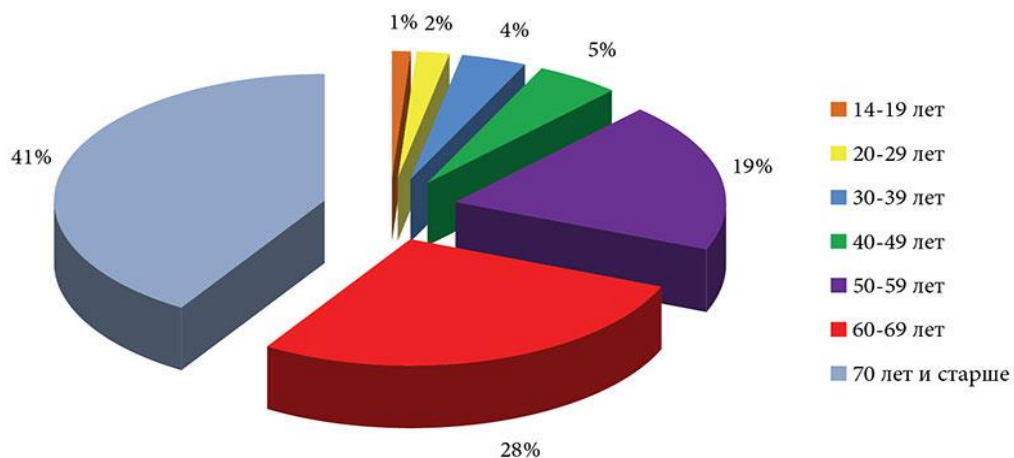


Рисунок 1 – Возрастное распределение пациентов с тугоухостью

сигналов, а также нарушением понимания источника звука. Помимо этого, может наблюдаться снижение понимания речи не только в шумной, но и в тихой обстановке, человеку приходится опираться не на слышимые, а на видимые источники информации. Также, есть сопутствующая старению проблема потери зрения, что может еще больше усугубить проблему тугоухости. Из-за этого человек, страдающий таким заболеванием, может еще больше уйти в себя и потерять всякую социализацию. Одной из главных особенностей пресбиакузиса является снижение слуха не на одно, а сразу на оба уха. Стоит отметить, что потеря слуха из-за пресбиакузиса является одной из причин деменции. Считается, что вплоть до 9% случаев возникновения этого заболевания как раз и связано с тем, что человек страдает от тугоухости. Этот процесс обратимый: правильная коррекция слуховых органов может привести к ремиссии и восстановлению психического здоровья.

Выделяют следующие виды пресбиакузиса [6]:

- *сенсорный тип* — высокочастотная тугоухость, которая не так много затрагивающая речевую зону и при этом медленно развивающаяся. В основе этого типа лежит потеря волосковых и поддерживающих клеток с преимущественным поражением базальных отделов улитки.

- *невральный тип* — при таком типе нарушение разборчивости речи выражено гораздо сильнее, чем изменение тональных порогов слышимости. Для такого типа характерно то, что заболевание развивается очень быстро. Гистологически определяется снижение количества нейронов на всем протяжении структур улитки.

- *метаболический тип* — при этом типе тональная аудиограмма имеет плоский характер. Проблемы со слухом соответствует с проблемами с разборчивостью речи, то есть деградация и слуха, и речи идут равномерно.

- *механический тип* — нисходящая аудиограмма в частотном диапазоне выше 1—2 кГц, разборчивость речи обратно пропорциональна крутизне наклона аудиограммы, при этом отсутствуют морфологические изменения.

1.2 Методы слухопротезирования

Главным и самым эффективным методом реабилитации при пресбиакузисе может являться слухопротезирование. Подобное исправление слуха у людей, страдающих тугоухостью, имеет свои нюансы и достаточно сложна. Считается, что если эффективность слухопротезирования у людей 50–60 лет условно принять за 100%, то к 70 годам она падает до 80%, к 80 годам составляет уже 40, а к 90 годам – всего 20%. По этой причине должны учитываться, кроме потери слуха, его физическое, умственное и сенсорное состояние, которое необходимо определять еще на стадии подготовки к слухопротезированию. Существенное негативное воздействие на успех такого вмешательства оказывают возрастные изменения таких способностей, как внимание и память.

Для улучшения социализации людей, страдающих тугоухостью, используются слуховые аппараты. Они направлены не на лечение заболевания, а на устранение его последствий. В одну из ушных раковин вставляется

устройство, которое путем изменения частоты звука улучшает его восприятие. Настраивается оно при непосредственном участии специалиста, учитывая степень способности человека слышать различные частоты. Также встречаются кохлеарные аппараты, которые устанавливаются под кожей в височной области. Они используются при тугоухости тяжелой степени. Третьим методом слухопротезирования является стволомозговая имплантация. Она представляет собой модифицированный кохлеарный имплантат, который устанавливается при помощи хирургической операции в ствол головного мозга на поверхность улиткового ядра.

В результате изучения методов слухопротезирования был произведен сравнительный анализ (таблица 1). Было определено, что наиболее безопасным и универсальным является слуховой аппарат.

Таблица 1 – Сравнительный анализ методов слухопротезирования

Параметр	Слуховой аппарат	Кохлеарный имплант	Стволомозговая имплантация
Простота использования	+	-	-
Удобство	+	+	+
Время автономной работы	-	+	+
Стоимость	+	-	-
Доступность	+	-	-
Неоперационное решение	+	-	-
Отсутствие риска полной потери слуха	+	-	-
Избавление от проблем, связанных с пресбиакузисом	+	+	+

1.3 Типы понижения частоты

Для решения проблемы потери слуха используется частотная модификация.

В литературе вместо этого термина может встречаться общее понятие «частотное понижение» (Frequency Lowering). Тем не менее более понятным и правильным будет именно термин «частотная модификация».

Частотная модификация дает возможность слабослышащим людям распознать те высокочастотные компоненты сигнала, которые им не слышны, в более низком частотном диапазоне. Каждый производитель реализует этот алгоритм по-своему.

Рассмотрим 3 типа понижения частоты (рисунок 2):



Рисунок 2 – Типы понижения частот

- **частотная компрессия.** Принцип работы заключается в сжатии участка выше начального порога частоты, приводящее к смещению предельного порога частоты в сторону понижения. В результате, все сигналы между начальной и верхней предельной частотой будут сжаты и преобразованы. В процессе изменения происходит значительное уменьшение доли высокочастотных

отрезков, применяемое для дополнительного подавления процесса самовозбуждения. В виду весомого изменения сигнала время адаптации к слуховым аппаратам данного типа повышается.

Динамика человеческой речи может быстро и резко меняться при разговоре или пении. Компрессия помогает минимизировать разницу между самыми тихими и громкими звуками путем ослабления размаха амплитуды сигнала и его всплесков, находящихся выше определенного порога громкости.

Взамен того, чтобы изменять звуки с повышенной частотой, приводящих к искажению и эффекту обратной связи, этот тип понижает частоту до такого уровня, чтобы такие звуки могли восприниматься без чрезмерного усиления. Однако у этой функции есть определенный минус, связанный с появлением механических нот при переходе в высокочастотный режим.

Примерами аппаратов, работающих на основе данного метода являются Phonak Audeo B-R, Signia Primax, Starkey Halo, ReSound LiNX 3D [7].

- **частотный перенос.** Тип основан на переносе области с повышенными частотами в среднечастотный диапазон. При переносе наблюдается определенное наложение сигналов, что ведет к появлению артефактов и преобразованию тембра голоса. Незатронутая часть диапазона остается без изменений. В данном методе преобразование звука осуществляется не так значительно, чем при компрессии. Схожим признаком этих методов является ухудшение качества звучания высоких частот. Такой метод отличается относительной простотой реализации, в том числе и для разработки слуховых аппаратов [8].

- **частотная композиция.** Данный тип заключается в копировании оригинальной высокочастотной области в целевую область. Главной отличительной особенностью от предыдущих методов, здесь сохраняется весь

частотный диапазон. В результате, сигналы передаются с наименее значительными искажениями звука. Иными словами, это разложение сложного по своему внутреннему содержанию звука на те или иные составные части. Можно сказать, что при этом типе изменения частот расщепляется звук [9].

Метод частотной композиции от Oticon называется Speech Rescue. Принцип основан на копировании высокочастотных компонентов, неслышимых ухом человека, но с незначительным наложением частотного диапазона.

Разберем основные принципы работы и алгоритмы типов понижения частоты.

Частотная компрессия

Известно несколько подходов и принципов компрессии и цифровизации речи [10]. Одним из наиболее действенных методов является подход, основанный на спектральном представлении речи. Его использование в совокупности с принципами антропоморфической обработки сигнала дает возможность заметно понизить число компонент, служащих основой для синтеза речи.

Такой подход позволяет осуществить предварительную обработку звука и обеспечивает:

- разложение на спектральные составные части (применяется быстрое преобразование Фурье);
- нахождение избыточных спектральных составляющих с учетом принципов психоакустики (вычисляются пороги маскирования и предельный порог слышимости для критических полос);
- отбор доминирующих спектральных компонент (проводится сравнение принадлежности к избыточным компонентам);

- синтез речи основан с помощью найденных спектральных составляющих (применяется обратное быстрое преобразование Фурье).

Алгоритм математической модели спектрального сжатия речи состоит из нескольких этапов:

Этап 1. Определение начальных условий связано с установлением частоты дискретизации f_s и длиной окна N_f быстрого преобразования Фурье (БПФ) и состоит из:

- разбиения на критические полосы;
- определения центров критических полос.

Этап 2. Выделение из начальной последовательности речи очередного отрезка – фрейма.

Этап 3. Нахождение дискретного преобразования Фурье для текущего фрейма.

Этап 4. Расчет тональных маскеров. Для каждого отсчета дискретного преобразования Фурье из r -го фрейма вычисляется его энергия в дБ и формируется его принадлежность к тональному маркеру исходя из выражения:

$$\begin{aligned} P_Y(i) > P_Y(i \pm 1), \\ P_Y(i) > P_Y(i \pm \Delta_i) + 7 \text{ дБ}, \end{aligned} \quad \text{где } \Delta_i = \begin{cases} 2 & \text{при } 0,17 \text{ кГц} < \omega_i < 5,5 \text{ кГц}; \\ [2;3] & \text{при } 5,5 \text{ кГц} \leq \omega_i < 11 \text{ кГц}; \\ [2;6] & \text{при } 11 \text{ кГц} \leq \omega_i < 20 \text{ кГц}. \end{cases}$$

Здесь i – частота, соответствующая i -му элементу частотной характеристики.

Этап 5. Нахождение шумовых маскеров. Находится энергия шума в пределах вычисленных речевых критических полос с помощью выражения:

$$P_{NM}(i) = \sum_{j=k_i}^{k_b} P_Y(j) \text{ (дБ)},$$

где i – номер критической полосы;

k_1 и k_n – номера первого и последнего элемента i -той критической полосы.

Этап 6. Прореживание маскеро́в. Анализируются все ранее найденные тональные и шумовые маскеро́в. Для каждого из двух маскеро́в, находящихся друг от друга на расстоянии не более 0,5 барка, выбирается только один с наибольшей энергией.

Этап 7. Расчет абсолютного порога маскирования. Для каждого оставшегося маскера вычисляется абсолютный порог маскирования, на основе его функции распространения во всем диапазоне частот быстрого преобразования Фурье. Функция распространения SF , которая имеет вид треугольника с вершиной, находящейся в центре тонального или шумового маскера, и рассчитывается по формуле:

$$SF(i, j) = \begin{cases} 17\Delta_z - 0,4P_{NM}(j) + 11 & \text{при } -3 \leq \Delta_z < -1; \\ (0,4P_{NM}(j) + 6) \cdot \Delta_z & \text{при } -1 \leq \Delta_z < 0; \\ -17\Delta_z & \text{при } 0 \leq \Delta_z < 1; \\ (0,15P_{NM}(j) - 17) \cdot \Delta_z - 0,15P_{NM}(j) & \text{при } 1 \leq \Delta_z < 8, \end{cases} \quad (\text{дБ})$$

где i – номер отсчета, для которого вычисляется индивидуальный порог маскирования;

j – номер отсчета, соответствующий маскеру;

P – шумовой или тональный маскер;

$\Delta_z = z(i) - z(j)$ – разность частот маскирующего и маскируемого сигналов выраженных в барках.

Этап 8. Расчет абсолютного порога слышимости вычисляется по формуле:

$$T_q(f) = 3,64 \cdot (f/1000)^{-0,8} - 6,5 \cdot e^{-0,6 \cdot (f/1000 - 3,3)^2} + 10^{-3} \cdot (f/1000)^4 \quad (\text{дБ}),$$

где f – частота в Гц.

Этап 9. Выявление доминирующих спектральных компонентов. Осуществляется выбор максимального значения из найденных порогов и энергии для каждого отсчета быстрого преобразования Фурье. Если энергия гармоники больше абсолютного порога маскирования и абсолютного порога слышимости, тогда гармоника проходит отбор.

Этап 10. Из отобранных отсчетов восстанавливается r -фрейм на основе обратного быстрого преобразования Фурье.

Частотный перенос

Механизм частотного переноса заключается в следующем. Для переноса частот используются две частотных полосы: целевая и начальная. Их выбор нужно производить как можно тщательнее и внимательно. При выборе слишком низкой целевой полосы, есть вероятность нарушить низкочастотный диапазон, что может привести к негативным последствиям для человека и повреждению его остаточного слуха. В обратном случае, при подборе чрезмерно высокой частоты для целевой полосы этот метод становится неприменимым в использовании, потому как этот тип не будет хранить высокочастотную информацию на определенных низких частотах, слышимых людьми с остаточным слухом. Первый спектральный момент, представляющий центроид спектра, определяет только согласные звуки. При обработке фреймов, применяется начальная адаптивная частотная полоса с центральной частотой f_c , которая вычисляется по формуле, представленной ниже:

$$f_c = \frac{\sum_{l=1}^{N/2} P(l) (f_s * \frac{1}{N})}{\sum_{l=1}^{N/2} P(l)}$$

Где $P(l)$ – мощность l -го частотного отсчёта,

f – частота дискретизации,

N – размер БПФ.

Частотная композиция

Примером частотной композиции является от Oticon, является Speech Rescue [11]. Он основан на идеях Brain Hearing, и в свою очередь использует три принципа:

Максимальное содержание в целевой области. Speech Rescue в своей работе полагается на применение естественных характеристик речи: при наличии высокочастотных компонентов речи в интервале средних частот речевая энергия за счет возникновения гласных практически отсутствует. Поэтому Speech Rescue не сжимает речь, а копирует высокочастотные компоненты из исходной области и вносит их в частотный диапазон, который может воспринимать слабослышащий. Два-три скопированных «частотных пакета» вставляются из дальнего диапазона высоких частот в более узкую целевую область, тем самым фактически складываются в один ряд (одно из значений английского слова «compose» – складывать). За счет этого, целевая область перекрывается в наименьшей степени.

Максимальное сохранение естественности. Метод Speech Rescue работает без прямой компрессии, что дает возможность не изменять гармонические обертоны из целевой и изначальной областей. Гармонические

обертоны высокочастотных согласных присоединяются к обертонам гласных с сохранением структуры. Таким образом, звук становится более естественным.

Максимальное сохранение оригинальной области. Speech Rescue продолжает акустическую стимуляцию оригинальной области. Параллельно происходит передача как высоких, так и средних частот. При этом обрабатывается максимальное количество информации, что дает возможность слабослышащему человеку получать качественный звук и оптимальную широкополосную передачу.

Подводя итог всему вышесказанному, можно отметить, что наиболее доступным средством избавления от проблем, связанных с тугоухостью является использование слухового аппарата. Он не может в полной мере вернуть человеку возможность слышать все звуки, так как они звучат на самом деле, однако являются, пожалуй, единственной доступной альтернативой. Кроме этого учитывая специфику заболевания, недостаточно с помощью слухового аппарата просто увеличить громкость. Необходимо чтобы все звуки находились в том диапазоне частот, который способно распознать ухо человека. Для решения этой проблемы можно использовать один из трех методов понижения частоты звука: компрессия, перенос и композиция.

2 ПРОЕКТИРОВАНИЕ МОДЕЛИ СЛУХОВОГО ПРЕОБРАЗОВАТЕЛЯ

2.1 Теория физического явления звука

Звук – это колебательное движение частиц среды, которое распространяется в виде волн во всех средах (газообразной, жидкой и твёрдой). Иначе можно сказать, что, звук – это явление, которое живой организм способен воспринимать одним из органов чувств. Человеческое ухо слышит звук с частотой в диапазоне от 16 Гц до 20000 Гц. Неслышимые колебания разделяют на инфразвук, то есть те, частота которых ниже 16 Гц и ультразвук, с частотой выше 20000 Гц.

Источниками звука может быть любой событие, которое относится с каким-либо изменением давления или напряжения. Пожалуй, самыми встречающимися источниками можно назвать колеблющиеся твердые тела, такие как музыкальные инструменты, мембраны колонок или даже самые обычные предметы, используемые в повседневной жизни. Звук может возникать при условиях ограниченной среды, из-за ветра и т.д.

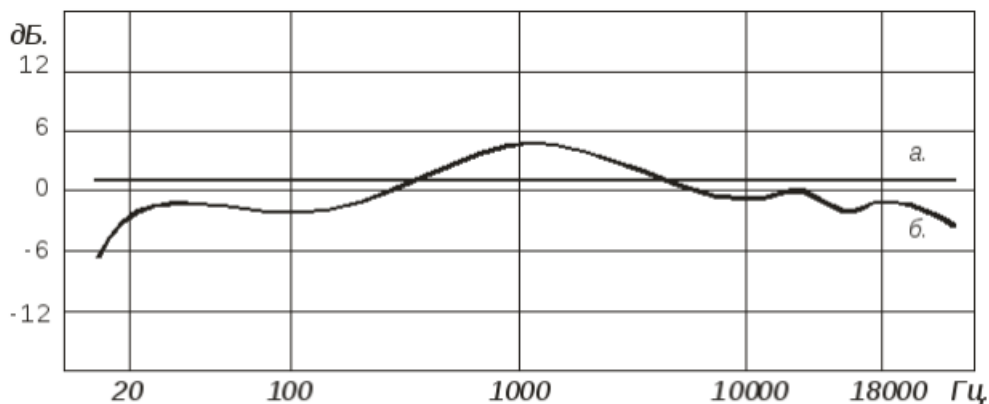
Особо выделяются звуковые колебания, которые возбуждаются человеческими органами. Голосовые связки человека способны издавать в среднем от 85 до 500 Гц [12].

Также необходимо особенно обратить внимание на наиболее важные физические свойства звука: высоту, громкость и спектр.

- Высота звука зависит от его частоты колебаний, чем выше тон голоса человека, тем большая частота характерна звуку и наоборот. Также существует зависимость между громкостью и частотой (рисунок 3). Частота звука измеряется в герцах – количество колебаний за 1 секунду.

- Громкость (сила) зависит от амплитуды. Зависимость также прямая: чем больше показатель амплитуды, тем громче звук.

- Спектр – это совокупность дополнительных колебаний, на которые можно разложить звуковую волну. Наиболее линейные спектры характерны для музыкальных произведений, а хаотичное – для обычных и повседневных звуков.



Частотные характеристики: а - идеальная;
б - возможный пример реальной.

Рисунок 3 – Зависимость громкости от частоты

В данной работе наиболее значимым является вопрос изменения частоты звука. Также необходимо рассмотреть понятия семпла и частоты дискретизации.

Семпл – это небольшой оцифрованный звуковой фрагмент, из которых состоит музыкальный файл. Важной характеристикой семпла является количество бит, которое в нем содержится – чем больше битов, тем выше точность. На рисунке 4 показано как выглядят семплы в одном из звуковых редакторов.

Частота дискретизации – свойство звукового файла, характеризующее количество семплов в одной секунде. Высокая частота дискретизации может

позволить воспроизводить высокочастотный звук, в том числе и ультразвуковой.

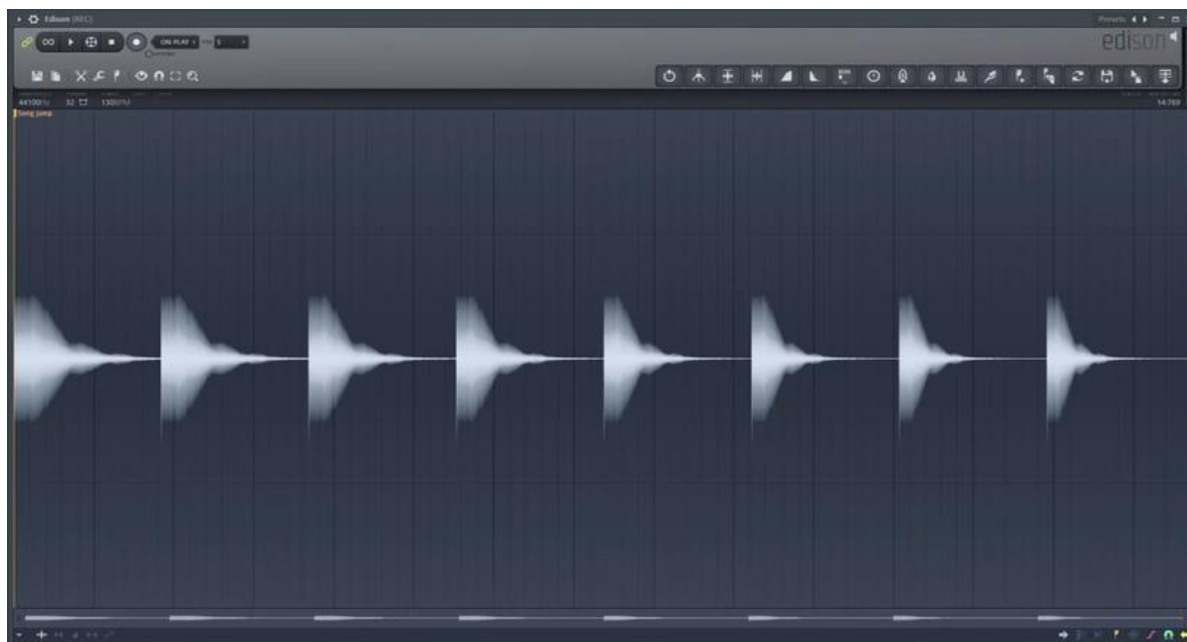


Рисунок 4 – Отображение семплов

При понижении частоты непосредственно необходимо изменять семплы. Для того, чтобы человек мог услышать реальное изменение тона голоса, также важно, чтобы частота дискретизации была высокой.

На рисунке 5 показано, в каком виде представлены семплы в среде программирования. Здесь в массиве y длиной 4644000 хранятся все семплы выбранного звукового отрезка. При этом, они могут быть как положительными, так и отрицательными. В переменной sr хранится частота дискретизации – 22050. Это значит, что в течение одной секунды звук делится на такое количество фрагментов.

```
sr = (int) 22050
y = (ndarray: (4644000,)) [ 1.04034320e-04 1.78020447e-04 1.31316483e-04 4.94718552e-06 -1.77055597e-04 -3.04989517e-04
```

Рисунок 5 – Представление семплов и частоты дискретизации

2.2 Прототип преобразователя

Для создания слухового преобразователя необходимо создать его прототип. Нужно выбрать микросхему, моно-наушник, микрофон, а также цифровой усилитель мощности звука.

Аппарат должен иметь микрофон для того, чтобы передать сигнал в аналогово-цифровой преобразователь. Сигнал преобразуется в цифровой вид и поступает в процессор, который изменяет характеристики звука, после чего он возвращается в аналоговый вид и поступает на динамик.

Для того, чтобы аппарат был удобен для использования в обычной жизни, следует изменять частоту звука только при попадании на микрофон звуковых сигналов высокой частоты. При этом желательно, чтобы помимо голоса, аппарат преобразовывал любые высокие частоты, так как подобное устройство используется пациентом в тех случаях, когда ему действительно необходимо расслышать что-либо.

В качестве микросхемы была выбрана STM Cortex [13]. Это достаточно удобная и популярная схема среди российских разработчиков электроники (рисунок 6).



Рисунок 6 – Микросхема STM32 Cortex

STM32 – это микроконтроллер, одним из самых главных преимуществ которого является универсальность. При использовании такого продукта можно пользоваться языками программирования высокого уровня.

Еще одним достоинством выбранного микропроцессора является хорошее соотношение цены и доступного функционала. Также в семействе STM32 стоит отметить наличие высококлассной периферии. Кроме того, архитектура ARM CORTEX получила большую популярность за счет низкого уровня электропотребления.

На рисунке 7 показано как выглядит аппаратная модель Cortex.

Среди всех компонентов модели Cortex можно выделить 3 наиболее важных. Это вычислительное ядро (CM3Core), контроллер вложенных прерываний (NVIC) и шинная матрица (Bus matrix).

Функцией вычислительного ядра является непосредственное выполнение инструкций, выполняя операции в своем арифметико-логическом устройстве. Особенностью ядра Cortex-M3 поддерживает как 32-битные, так и 16-битные инструкции для сокращения объема кода, в случае перехода и имеет 13 регистров общего назначения.

NVIC генерирует прерывания на различные события. Он поддерживает до 240 прерываний и до 256 уровней приоритета, прерывания могут быть вложенными.

Шинная матрица – контроллер, управляющий множеством шин. В современных процессорах используется множество шин, которые соединяют остальные устройства системы. Это позволяет работать с несколькими устройствами одновременно, что значительно повышает скорость работы.

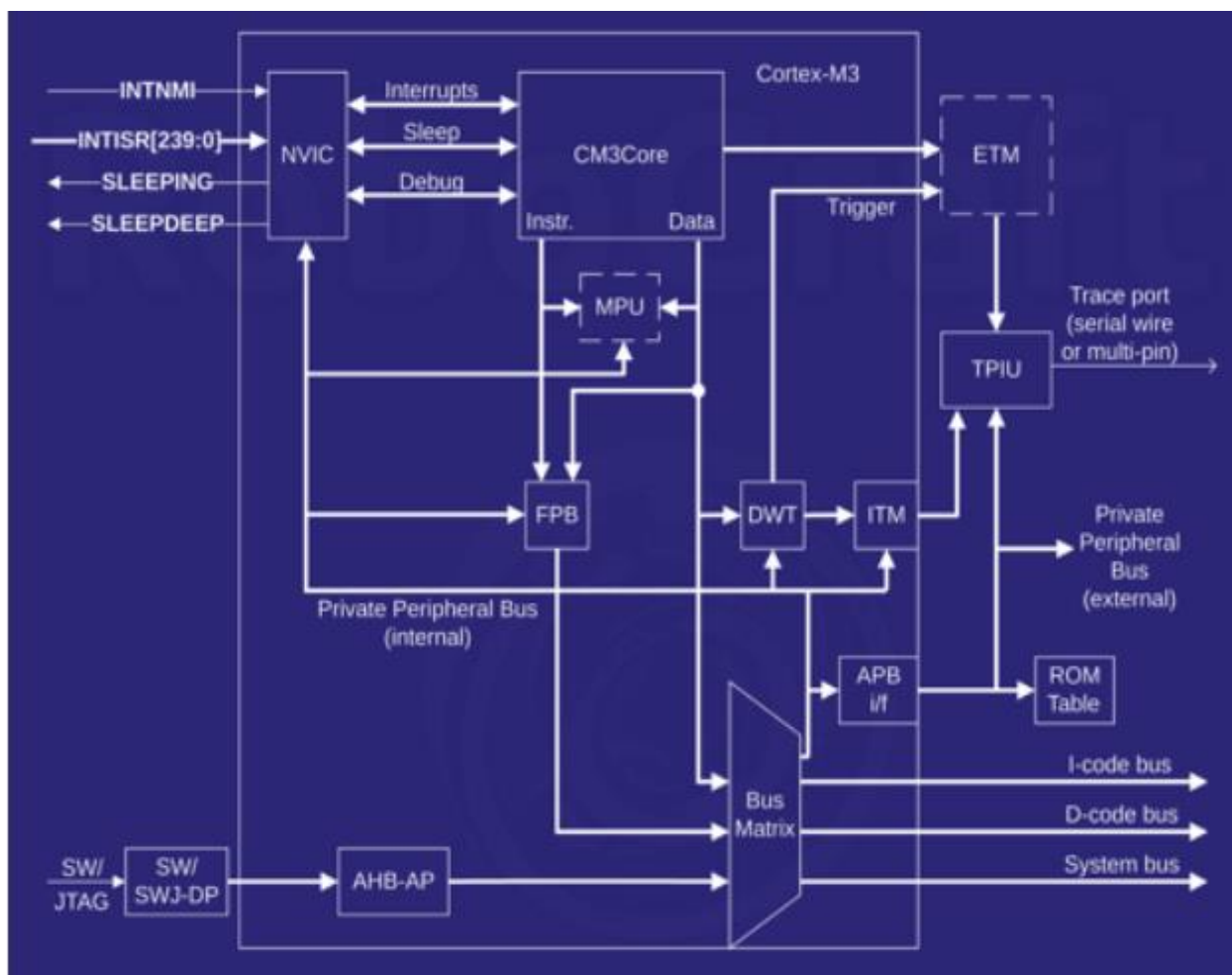


Рисунок 7 – Схема Cortex-M3

Помимо микросхемы для комфортного использования аппарата нужен цифровой усилитель звука. В качестве такой платы подойдет одна из разработок компании FGHGF (рисунок 8).

Также для работы аппарата подойдет любой моно-наушник и микрофон.

В результате получается полноценное устройство, способное преобразовать исходный сигнал в более удобный для пользователя. При этом он сможет в любой момент вынуть наушник из уха, если в его услугах не будет необходимости.



Рисунок 8 – Цифровой усилитель звука от FGHGF

2.3 Проектирование слухового преобразователя

Для создания слухового преобразователя, нужно сначала сформулировать требования, которые будут применимы к нашему устройству. Выделим функциональные и нефункциональные требования.

Функциональные требования:

- 1) преобразователь должен принять аналоговый сигнал на микрофон;
- 2) преобразователь должен оценивать какой из звуковых отрезков является высокочастотным или низкочастотным, а какой нет;
- 3) преобразователь должен изменить высокочастотный сигнал таким образом, чтобы пользователь смог слышать тот же самый сигнал на другой частоте, более низкой или высокой;
- 4) преобразователь должен вернуть аналоговый сигнал на моно-наушник.

Нефункциональные требования:

- 1) преобразователь должен быть написан на языке программирования Python;
- 2) в качестве среды разработки следует использовать PyCharm;
- 3) при создании устройства должны использоваться библиотеки librosa и pyaudio.

На основе сформулированных требований была составлена диаграмма вариантов использования (рисунок 9).

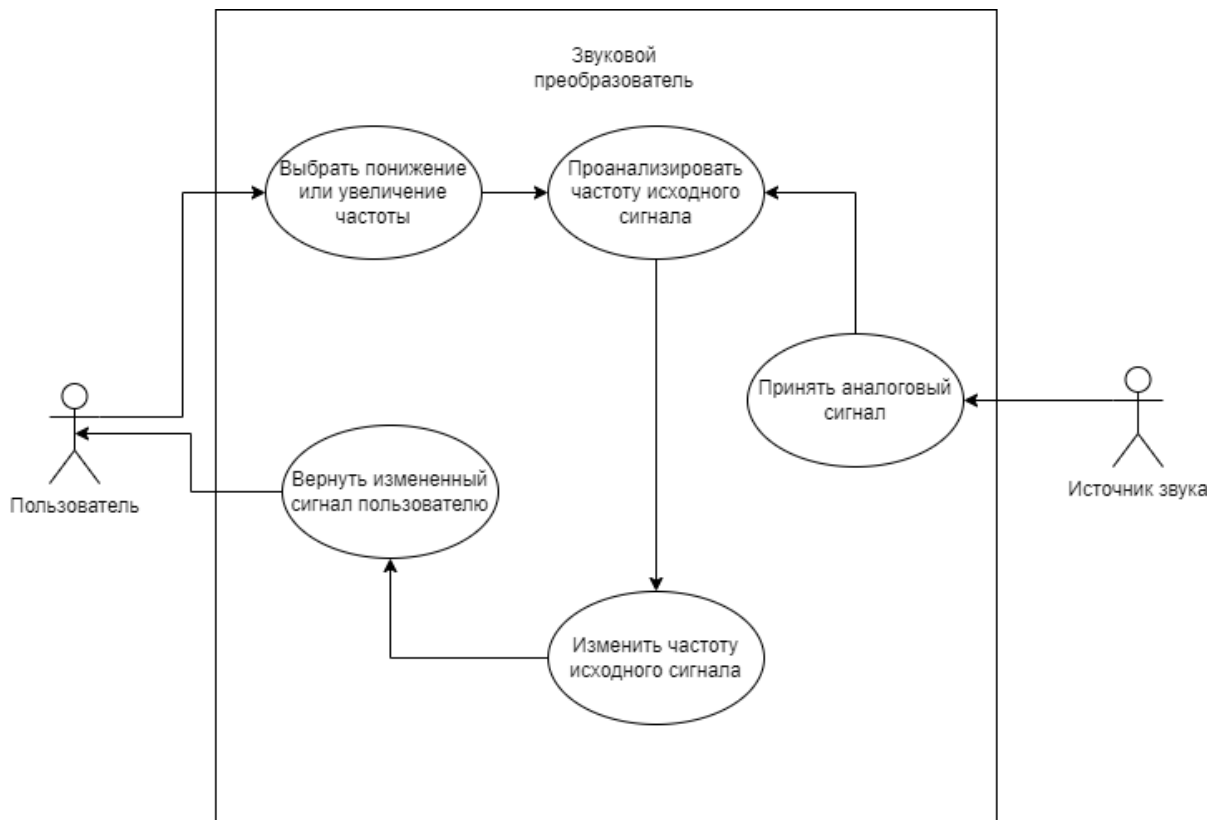


Рисунок 9 – Диаграмма вариантов использования

Диаграмма вариантов использования является, пожалуй, самым наглядным способом демонстрации представления функциональных требований к системе. К диаграмме обычно прилагается краткое описание, а

именно то, как тот или иной актер относится к определенной функции устройства.

Воспользуемся для этого редактором «draw io» [14].

На диаграмме присутствуют 2 актера: пользователь и некий источник звука.

Пользователь – это человек, который использует преобразователь звука, может выбирать режим работы аппарата: либо понижение тона голоса и окружающих звуков, либо их повышение.

Источник звука – любой объект, который способен каким-либо образом создавать звуковые колебания. В качестве таких источников могут выступать голоса людей, музыкальные инструменты, звуки, которые издают животные и т.д.

Описание диаграммы вариантов использования:

1) принять аналоговый сигнал – преобразователь должен принять исходный аналоговый сигнал от источника звука и сохранить его.

2) проанализировать частоту исходного сигнала – преобразователь анализирует частоту полученного звука и определяет имеются ли отрезки, где частота повышена или понижена (в зависимости от режима работы).

3) изменить частоту исходного сигнала – возможность слухового преобразователя изменить значения семплов в той части массива, где частота звука слишком высока (или низка).

4) вернуть измененный сигнал пользователю – возможность преобразовать полученные данные обратно в аналоговый сигнал, понятный пользователю.

5) выбрать понижение или увеличение частоты – возможность выбирать режим, в котором будет работать преобразователь. Если пользователь страдает тугоухостью, то он должен использовать понижение частоты, а если человеку

нужно слышать низкий тон, то будет использоваться режим повышения частоты.

В данной главе были рассмотрены теоретическая часть природного явления звука, придуман прототип слухового преобразователя, а также приведены требования к системе. В теоретической части были рассмотрены свойства самого звука, а также его программного представления. Для прототипа была выбрана соответствующая микросхема, цифровой усилитель звука, микрофон и моно-наушник. После этого были сформулированы функциональные и нефункциональные требования к системе, а также составлена диаграмма вариантов использования.

3 РЕАЛИЗАЦИЯ СЛУХОВОГО ПРЕОБРАЗОВАТЕЛЯ

После изучения теоретической части и проектирования модели слухового преобразователя следует приступить к реализации программной части. Для того, чтобы грамотно настроить устройство для конкретного пациента необходимо заключение врача. Опираясь на это нужно выбрать сдвиг частот, который будет устанавливаться для конкретного пользователя. В случае пресбиакузиса такой порог в среднем составляет 2 кГц. При реализации выберем именно такую частоту как максимальную.

В режиме повышения частоты остановимся на 500 Гц.

Для реализации слухового аппарата наиболее удобным является язык программирования Python.

Python – это интерпретированный, объектно-ориентированный высокоуровневый язык программирования с динамической семантикой. Он достаточно гибкий для разработки различных программных продуктов. Кроме того, является неплохим средством для связи существующих компонентов вместе. Также Python имеет достаточно читаемый синтаксис. Python поддерживает модули и пакеты, которые поощряют программную модульность и повторное использование кода.

Для того, чтобы решить поставленную задачу необходимо использовать несколько библиотек. Среди них librosa, soundfile, pyaudio, wave и sys.

Librosa [15] – библиотека, с помощью которой можно изменять элементы массива семплов в зависимости от поставленной задачи. С помощью ее методов возможно анализировать исходный аудиоряд и осуществлять с ним множество операций. В нашем случае нужно изменять высоту тона и анализировать частоту сигнала.

Pyaudio [16] – программный пакет, который позволяет изменять, записывать и воспроизводить аналоговый сигнал. Он может использовать микрофон и наушник в качестве входа и выхода звукового сигнала.

Soundfile [17] – библиотека для работы со звуком. Для записи аудиофайла в файл нужно использовать метод write, доступный в данном пакете.

Wave [18] – еще одна библиотека, с помощью которой возможно открывать файл, который в дальнейшем будет использоваться в обработке звукового сигнала.

Sys [19] дает возможность выбрать устройство, с которого будет считываться звуковой ряд. По умолчанию это микрофон. Однако можно выбрать иное устройство.

В качестве среды программирования был выбран PyCharm [20] . Это наиболее удобное средство для разработки приложений на Python (рисунок 10).



```
main.py x
1 | This is a sample Python script.
2
3 # Press Shift+F10 to execute it or replace it with your code.
4 # Press Double Shift to search everywhere for classes, files, tool windows, actions, and settings.
5
6
7 usage
7 def print_hi(name):
8     # Use a breakpoint in the code line below to debug your script.
9     print(f'Hi, {name}') # Press Ctrl+F8 to toggle the breakpoint.
10
11
12 # Press the green button in the gutter to run the script.
13 if __name__ == '__main__':
14     print_hi('PyCharm')
15
16 # See PyCharm help at https://www.jetbrains.com/help/pycharm/
17
```

Рисунок 10 – Среда программирования PyCharm

Для поставленной задачи нужно разработать и реализовать программу, которая изменит частоту звукового отрезка.

Для начала следует выбрать режим работы приложения. В случае, когда требуется понижать частоту, переменная `ShiftToDown` принимает значение истины, иначе ложь (листинг 1). В качестве сдвига частоты выберем параметр равный трем.

Листинг 1 – Выбор режима работы.

```
ShiftToDown = True
frequency = 2000
steps = -3
if not ShiftToDown:
    frequency = 500
    steps = 3
```

Для того, чтобы записать звуковой сигнал нам требуется функционал библиотеки `Pyaudio`. Звук подается с микрофона и записывается. Этот процесс отображен в листинге 2.

Листинг 2. Запись сигнала.

```
with wave.open('output.wav', 'wb') as wf:
    p = pyaudio.PyAudio()
    wf.setnchannels(CHANNELS)
    wf.setsampwidth(p.get_sample_size(FORMAT))
    wf.setframerate(RATE)

    stream = p.open(format=FORMAT,
channels=CHANNELS, rate=RATE, input=True)

    print('Recording...')
    for _ in range(0, round(RATE // CHUNK *
RECORD_SECONDS)):
```


Продолжение листинга 2

```
        wf.writeframes(stream.read(CHUNK))
    print('Done')

    stream.close()
    p.terminate()
```

Здесь используются функции, описываемые в листинге 3.

Листинг 3. Функции `setnchannels`, `setsampwidth`, `setframerate`.

```
def setnchannels(self, nchannels):
    if self._datawritten:
        raise Error('cannot change parameters
after starting to write')
    if nchannels < 1:
        raise Error('bad # of channels')
    self._nchannels = nchannels

def setsampwidth(self, sampwidth):
    if self._datawritten:
        raise Error('cannot change parameters
after starting to write')
    if sampwidth < 1 or sampwidth > 4:
        raise Error('bad sample width')
    self._sampwidth = sampwidth

def setframerate(self, framerate):
    if self._datawritten:
```

Продолжение листинга 3

```
        raise Error('cannot change parameters
after starting to write')
    if framerate <= 0:
        raise Error('bad frame rate')
    self._framerate = int(round(framerate))
```

Далее необходимо определить какие отрезки не входят в подходящий для нас диапазон, а какие нет. Для этого мы будем использовать метод `yin`, который предоставлен `Librosa`. С его помощью несколько десятков раз в секунду определяется частота звука. Если на какой-то период времени она будет выше максимальной (или ниже минимальной), то произойдет смещение тона на выбранный ранее параметр. При этом смещение будет действовать на протяжении следующей секунды во избежание неудобства при использовании устройства. Если сделать такой отрезок слишком маленьким, то вместо желаемого результата человек будет слышать треск вместо звука. На листинге 4 описан алгоритм изменения тональности.

Листинг 4 – Обработка звукового отрезка.

```
array = librosa.yin(y, fmin=440, fmax=3000)
i = 0
j = 0
y_tritone = []
mypar = len(y)//len(array) + 1
while i < len(y):
    if i % mypar == 0:
        if ShiftToDown and array[i//mypar] > frequency or
not ShiftToDown and array[i//mypar] < frequency:
            y_tritone = y[i:i+mypar*43]
```

Продолжение листинга 4

```
        y_tritone =
librosa.effects.pitch_shift(y_tritone, sr=sr,
n_steps=steps)
        for k in range (len(y_tritone)):
            if i+k >= len(y):
                break
            else:
                y[i+k] = y_tritone[k]
            i += mypar * 43
            j += 1 + 43
        j += 1
    i += 1
```

В приведенном листинге используется метод `yin`. Это алгоритм оценки основной частоты, основан на автокорреляции. Сначала нормализованная разностная функция вычисляется по коротким кадрам звука. Затем в качестве оценки периода сигнала выбирается первый период разностной функции. После этого расчетный период уточняется с помощью параболической интерполяции перед преобразованием в соответствующую частоту.

На листинге 5 представлен метод `yin`.

Листинг 5. Метод `yin`.

```
def yin(
    y: np.ndarray,
    *,
    fmin: float,
    fmax: float,
    sr: float = 22050,
```

Продолжение листинга 5

```
    frame_length: int = 2048,
    win_length: Optional[int] = None,
    hop_length: Optional[int] = None,
    trough_threshold: float = 0.1,
    center: bool = True,
    pad_mode: _PadMode = "constant",
) -> np.ndarray:

    if fmin is None or fmax is None:
        raise ParameterError('both "fmin" and "fmax"
must be provided')

    if win_length is None:
        win_length = frame_length // 2

    if win_length >= frame_length:
        raise ParameterError(
            f"win_length={win_length} cannot exceed
given frame_length={frame_length}"
        )

    if hop_length is None:
        hop_length = frame_length // 4

    util.valid_audio(y, mono=False)
```

Продолжение листинга 5

```
    if center:
        padding = [(0, 0)] * y.ndim
        padding[-1] = (frame_length // 2,
frame_length // 2)
        y = np.pad(y, padding, mode=pad_mode)

    y_frames = util.frame(y,
frame_length=frame_length, hop_length=hop_length)

    min_period = max(int(np.floor(sr / fmax)), 1)
    max_period = min(int(np.ceil(sr / fmin)),
frame_length - win_length - 1)

    yin_frames =
_cumulative_mean_normalized_difference(
        y_frames, frame_length, win_length,
min_period, max_period
    )

    parabolic_shifts =
_parabolic_interpolation(yin_frames)

    is_trough = util.localmin(yin_frames, axis=-2)
    is_trough[..., 0, :] = yin_frames[..., 0, :] <
yin_frames[..., 1, :]
```

Окончание листинга 5

```
is_threshold_trough = np.logical_and(is_trough,
yin_frames < trough_threshold)

target_shape = list(yin_frames.shape)
target_shape[-2] = 1

global_min = np.argmin(yin_frames, axis=-2)
yin_period = np.argmax(is_threshold_trough,
axis=-2)

global_min = global_min.reshape(target_shape)
yin_period = yin_period.reshape(target_shape)

no_trough_below_threshold =
np.all(~is_threshold_trough, axis=-2, keepdims=True)
yin_period[no_trough_below_threshold] =
global_min[no_trough_below_threshold]

yin_period = (
    min_period
    + yin_period
    + np.take_along_axis(parabolic_shifts,
yin_period, axis=-2)
)[..., 0, :]
f0: np.ndarray = sr / yin_period
return f0
```

Также в листинге 2 используется метод `pitch_shift`. С его помощью можно пошагово изменять высоту тона волны. Отрицательные значения переменная `n_steps` принимает в случаях, когда тон нужно понизить и соответственно положительные, когда тон следует повысить. На листинге 6 показан метод `pitch_shift`.

Листинг 6. Метод `pitch_shift`.

```
def pitch_shift(
    y: np.ndarray,
    *,
    sr: float,
    n_steps: float,
    bins_per_octave: int = 12,
    res_type: str = "soxr_hq",
    **kwargs: Any,
) -> np.ndarray:

    if not util.is_positive_int(bins_per_octave):
        raise ParameterError(
            f"bins_per_octave={bins_per_octave} must
be a positive integer."
        )

    rate = 2.0 ** (-float(n_steps) / bins_per_octave)
    y_shift = core.resample(
        time_stretch(y, rate=rate, **kwargs),
        orig_sr=float(sr) / rate,
        target_sr=sr,
        res_type=res_type,
```

Продолжение листинга 6

)

```
return util.fix_length(y_shift, size=y.shape[-1])
```

Кроме этого было необходимо подключить к нашей системе микрофон. Он должен был записывать происходящее извне и передавать полученные данные устройству. Осуществляется это с помощью программного пакета `pyaudio`. На листинге 7 описываются характеристики, соответствующие звуковому отрезку.

Среди таких характеристик:

- количество кусков звука, на которые разделяется секунда;
- формат аудио;
- устройство, с помощью которого происходит запись;
- частота дискретизации;
- временной отрезок, на продолжении которого идет запись.

Листинг 7 – Характеристики звукового отрезка.

```
CHUNK = 1024
FORMAT = pyaudio.paInt16
CHANNELS = 1 if sys.platform == 'darwin' else 2
RATE = 22050
RECORD_SECONDS = 3
```

После того, как данный отрезок поступает в процессор, идет его обработка, описанная ранее.

Далее обработанный сигнал нужно вывести в моно-наушник. Делается это как показано на листинге 8.

Листинг 8. Вывод обработанного аудиосигнала.

```
p = pyaudio.PyAudio()
    stream =
p.open(format=p.get_format_from_width(wf.getsampwidth()),
        channels=wf.getnchannels(),
        rate=wf.getframerate(),
        output=True)

    while len(data := wf.readframes(CHUNK)):
        stream.write(data)

    stream.close()
    p.terminate()
```

В данном случае используются методы `getsampwidth`, `getchannels` и `getframerate`, а также `readframes`. Их описание приведено на листинге 9.

Листинг 9. Методы `getsampwidth`, `getchannels`, `getframerate` и `readframes`.

```
def getsampwidth(self):
    return self._sampwidth

def getframerate(self):
    return self._framerate

def getnchannels(self):
    return self._nchannels
```

Продолжение листинга 9

```
def readframes(self, nframes):
    if self._data_seek_needed:
        self._data_chunk.seek(0, 0)
        pos = self._soundpos * self._framesize
        if pos:
            self._data_chunk.seek(pos, 0)
        self._data_seek_needed = 0
    if nframes == 0:
        return b''
    data = self._data_chunk.read(nframes *
self._framesize)
    if self._sampwidth != 1 and sys.byteorder ==
'big':
        data = audioop.byteswap(data,
self._sampwidth)
    if self._convert and data:
        data = self._convert(data)
    self._soundpos = self._soundpos + len(data)
// (self._nchannels * self._sampwidth)
    return data
```

Помимо этого, для вывода звука необходимо в функцию `open` в качестве аргумента объявить переменную `format`, которая вычисляется в методе `get_format_from_width`. Результат будет зависеть от ширины выборки в байтах, которая находится внутри переменной `width` (листинг 10).

Листинг 10. Метод `get_format_from_width`.

```
def get_format_from_width(width, unsigned=True):
```

Продолжение листинга 10

```
if width == 1:
    if unsigned:
        return paUInt8
    return paInt8
if width == 2:
    return paInt16
if width == 3:
    return paInt24
if width == 4:
    return paFloat32
raise ValueError(f"Invalid width: {width}")
```

В этой главе рассмотрена реализация модели слухового преобразователя. Был выбран язык программирования, среда разработки, а также библиотеки, которые необходимы для решения поставленной задачи. Кроме того, было проведено сравнение графиков исходного звукового отрезка и измененного.

4 ТЕСТИРОВАНИЕ ПРОТОТИПА СЛУХОВОГО ПРЕОБРАЗОВАТЕЛЯ

В данной главе рассматривается тестирование прототипа слухового преобразователя, на основе функциональных требований, представленных ранее.

После реализации прототипа слухового преобразователя требуется проверить правильность его работы. Аппарат должен принять на вход звуковой сигнал и преобразовать его в цифровой вид. Прототип слухового преобразователя представляет отрезок в качестве массива семплов, так как именно они являются базой для работы звука. На рисунке 11 показано представление такого массива в PyCharm.

	0	1	2	3	4	5	6	7	8	9	10	11
0	-0.00137759	-0.009040709	-0.0156473	-0.019611642	-0.023240803	-0.025670998	-0.027885318	-0.030769076	-0.03351152	-0.035404004	-0.036767952	-0.03604117

Рисунок 11 – Массив семплов

Далее требуется проанализировать сигнал и проверить нужно ли его изменять. После проверки семплы меняют свое значение на нужном отрезке. При этом там, где частота остается в пределах выбранной нормы, модификация не вносится. На рисунках 12 и 13 продемонстрирован отрезок массива семплов до и после изменения.

	0	1	2	3	4	5	6	7	8	9	10	11
	-0.054416962	-0.036136024	-0.009636095	0.021799674	0.05412246	0.08377199	0.10748996	0.12233102	0.12676756	0.120598	0.104523435	0.08008002

Рисунок 12 – Массив семплов до изменения

	0	1	2	3	4	5	6	7	8	9	10
	-0.05508187	-0.06388806	-0.046956934	-0.031148627	-0.018012427	-0.009709271	0.00027584098	0.015501802	0.032860775	0.047551848	0.05884985

Рисунок 13 – Массив семплов после изменения

После этого измененный отрезок можно возвращать обратно пользователю.

На рисунке 14 показан отрезок высокочастотного звука до обработки, а на рисунке 15 – после. При сравнении графиков видно, что частота сигнала на них различается. Если на первом колебания визуально имеют одинаковую частоту, то на втором, после определенного момента, частота понижается на некоторое время для более понятного уровня слышимости для пользователя.



Рисунок 14 – Сигнал до понижения частоты

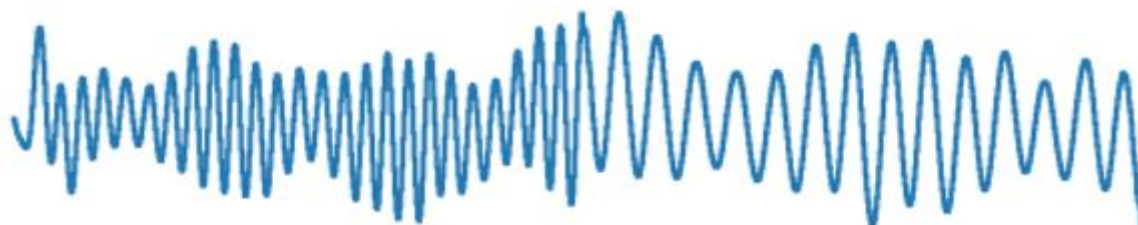


Рисунок 15 – Сигнал после понижения частоты

В данном случае преобразователь работал в режиме понижения частоты. В результате одной из проверок потребовалось понизить тон звука, что отразилось на втором графике.

Теперь рассмотрим вариант, когда устройство работает в режиме повышения тона звука. На рисунке 16 показан сигнал до повышения частоты, а на рисунке 17 – после. Здесь также визуально можно заметить, что на первом из ниже представленных рисунках колебания имеют примерно одну и ту же частоту. На втором из графиков после обработки цифрового сигнала частота увеличивается. Аналогично с первым примером, первая часть графиков на рисунках совпадает до того момента, как частота перестает входить в

допустимые пределы. После этого колебания сигнала учащаются – высота тона увеличивается.



Рисунок 16 – Сигнал до повышения частоты



Рисунок 17 – Сигнал после повышения частоты

Также следует заметить, что на графиках, представленных выше, частота сигнала изменяется на протяжении достаточно долгого периода. Если на рисунках 14-17 расположено 1000 семплов, то для более удобного использования слухового преобразователя в ходе понижения (повышения) высоты тона изменяется 22000 семплов. В разрезе времени это составляет примерно одну секунду. Сделано это для того, чтобы звук слышался более равномерно, не было каких-либо тресков. Иначе использование такого аппарата может не улучшить, а даже ухудшить восприятие аудиосигналов у пользователя.

В данной главе было рассмотрено тестирование слухового преобразователя, в частности проверено изменение значений семплов при обработке сигнала, работа аппарата в режимах повышения и понижения высоты тона.

ЗАКЛЮЧЕНИЕ

В результате выполнения данной работы были изучены причины и специфика заболевания тугоухостью. Был произведен анализ существующих методов слухопротезирования, в результате которого был выбран слуховой аппарат.

В итоге были получены следующие результаты:

1. изучены причины и специфика заболевания пресбиакузисом;
2. проанализирован принцип работы модели слухового преобразователя;
3. спроектирована модель слухового преобразователя для плохослышащих людей. Особенностью разрабатываемого устройства является возможность выбора режима работы, позволяющего либо повышать, либо понижать частоту исходного звукового сигнала. Также в ходе работы была реализована программа, изменяющая частоты звуковых отрезков;
4. протестирован прототип слухового преобразователя.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. World Health Organization / Deafness and hearing loss – URL: <http://www.who.int/deafness/worldhearing-day/World-Hearing-Day-Banner-RU.pdf> (дата обращения: 06.03.2023).

2. Девятловский, Д.Н. Влияние шумового воздействия на здоровье человека / Д.Н. Девятловский // Актуальные проблемы лесного комплекса. – 2010. – № 27. – С. 52-55.

3. Бобошко, М.Ю. Разборчивость речи при тугоухости у лиц пожилого и старческого возраста / М.Ю. Бобошко, Л.Е. Голованова, Е.В. Жилинская и др. // Успехи геронтологии – 2016. – Т. 29. – № 4. – С. 663-669.

4. Вишняков, В.В. Старческая тугоухость (пресбиакузис), или особенности нарушения слуха у пожилых людей / В.В. Вишняков // Эффективная фармакотерапия. – 2010. – №27. – С. 80-83.

5. Гуненко, А.В. Возрастные изменения слуха (пресбиакузис). Современные подходы к старой проблеме / А.В. Гуненко // Вестник оториноларингологии. – 2007. – №3. – С. 33-35.

6. Кунельская Л.Н. Пресбиакузис — актуальная проблема стареющего населения / Л.Н. Кунельская, Ю.В. Левина, Е.В. Гаров // Вестник оториноларингологии. – 2019. – Т.84. – №4. – С. 67-71.

7. Ю-Мед Центр слуха / Частотная компрессия в слуховых аппаратах. – URL: [https://you-med.ru/stati/slukovyie-apparati/chastotnaya-kompressiya-v-slukovyix-](https://you-med.ru/stati/slukovyie-apparati/chastotnaya-kompressiya-v-slukovyix-apparatax#:~:text=%D0%9F%D1%80%D0%B8%D0%BC%D0%B5%D1%80%D1%8B%20%D1%82%D0%B0%D0%BA%D0%B8%D1%85%20%D1%81%D0%BB%D1%83%D1%85%D0%BE%D0%B2%D1%8B%D1%85%20%D0%B0%D0%BF%D0%BF%D0%B0%D1%80%D0%B0%D1%82%D0%BE%D0%B2%3A%20Phon)

[apparatax#:~:text=%D0%9F%D1%80%D0%B8%D0%BC%D0%B5%D1%80%D1%8B%20%D1%82%D0%B0%D0%BA%D0%B8%D1%85%20%D1%81%D0%BB%D1%83%D1%85%D0%BE%D0%B2%D1%8B%D1%85%20%D0%B0%D0%BF%D0%BF%D0%B0%D1%80%D0%B0%D1%82%D0%BE%D0%B2%3A%20Phon](https://you-med.ru/stati/slukovyie-apparati/chastotnaya-kompressiya-v-slukovyix-apparatax#:~:text=%D0%9F%D1%80%D0%B8%D0%BC%D0%B5%D1%80%D1%8B%20%D1%82%D0%B0%D0%BA%D0%B8%D1%85%20%D1%81%D0%BB%D1%83%D1%85%D0%BE%D0%B2%D1%8B%D1%85%20%D0%B0%D0%BF%D0%BF%D0%B0%D1%80%D0%B0%D1%82%D0%BE%D0%B2%3A%20Phon)

ak,Starkey%20Halo%2C%20ReSound%20LiNX%203D (дата обращения 12.03.2023).

8. Порхун М.И. Перенос частот в речевом сигнале для коррекции слуховых патологий. / М.И. Порхун, М.И. Вашкевич // Доклады 21-й Международной конференции – 2019. – Вып.101. – С. 112-116.

9. Исток-аудио группа компаний / Частотная модификация в слуховом аппарате – залог лучшего понимания речи. – URL: https://www.istok-audio.com/info/articles/slukhovye_apparaty/chastotnaya_modifikatsiya_v_slukhovo_m_apparate_zalog_luchshego_ponimaniya_rechi/ (дата обращения 12.03.2023).

10. Мальцев, С.В. Компрессия речи на основе доминирующих спектральных компонентов / С.В Мальцев, В.М. Чертков // Вестник полоцкого государственного университета. – 2010. – № 8. – С 92-97.

11. Проект: Аудиология / Частотная модификация в слуховом аппарате – залог лучшего понимания речи [Электронный ресурс] URL: <https://pro-audiologia.ru/categories/hearing-aids/item/32-frequency-modification>. (дата обращения 01.05.2023).

12. Филологический факультет МГУ имени М.В. Ломоносова [Электронный ресурс] URL: <https://www.philol.msu.ru>. (дата обращения 01.04.2023).

13. STM32 32-bit Arm Cortex MCUs [Электронный ресурс] URL: <https://www.st.com/en/microcontrollers-microprocessors/stm32-32-bit-arm-cortex-mcus.html>. (дата обращения 01.04.2023).

14. Flowchart Maker and Online Diagram Software [Электронный ресурс] URL: <https://app.diagrams.net>. (дата обращения 01.04.2023).

15. Librosa [Электронный ресурс] URL: <https://librosa.org/doc/latest/index.html> (дата обращения 05.04.2023).

16. Pyaudio 0.2.13 [Электронный ресурс] URL:
<https://pypi.org/project/PyAudio> (дата обращения 05.04.2023).
17. Soundile 0.12.1 [Электронный ресурс] URL:
<https://pypi.org/project/soundfile> (дата обращения 05.04.2023).
18. Wave – Read and Write WAV files [Электронный ресурс] URL:
<https://docs.python.org/3/library/wave.html> (дата обращения 05.04.2023).
19. Sys – System-specific parameters and functions [Электронный ресурс]
URL: <https://docs.python.org/3/library/sys.html> (дата обращения 05.04.2023).
20. PyCharm IDE для профессиональной разработки на Python
[Электронный ресурс] URL: <https://www.jetbrains.com/ru-ru/pycharm> (дата
обращения 05.04.2023).