



Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Московский государственный технический университет
имени Н.Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н.Э. Баумана)

ФАКУЛЬТЕТ «Информатика и системы управления»

КАФЕДРА «Программное обеспечение ЭВМ и информационные технологии»

РАСЧЕТНО-ПОЯСНИТЕЛЬНАЯ ЗАПИСКА
К НАУЧНО-ИССЛЕДОВАТЕЛЬСКОЙ РАБОТЕ
НА ТЕМУ:

«Классификация методов воспроизведения потокового аудио в
мобильном приложении на операционной системе iOS»

Студент группы **ИУ7-73Б**

(Подпись, дата)

Топорков П.А.

(И.О. Фамилия)

Руководитель

(Подпись, дата)

Терентьев Ю.И.

(И.О. Фамилия)

2022 г.

РЕФЕРАТ

Расчетно-пояснительная записка 25 с., 3 рис., 3 табл., 16 ист., 1 прил.

Классифицированы методы воспроизведения потокового аудио в мобильном приложении на операционной системе iOS. Представлены достоинства и недостатки описанных методов.

СОДЕРЖАНИЕ

РЕФЕРАТ	3
ОПРЕДЕЛЕНИЯ, ОБОЗНАЧЕНИЯ И СОКРАЩЕНИЯ	6
ВВЕДЕНИЕ	7
1 Аналитическая часть	8
1.1 Анализ предметной области	8
1.2 Цифровое представления аудиоданных	8
1.3 Анализ форматов хранения аудиоданных	11
1.3.1 Необработанные звуковые файлы	11
1.3.2 WAV (WAVE) формат	11
1.3.3 MP4 формат	12
1.3.4 MP3 формат	13
1.3.5 AAC формат	13
1.3.6 Сравнение форматов хранения аудиоданных	14
1.4 Анализ протоколов передачи потоковых данных	14
1.4.1 HLS (HTTP Live Streaming)	15
1.4.2 Low-Latency HLS	16
1.4.3 RTSP (Real-Time Streaming Protocol)	16
1.4.4 Сравнение протоколов передачи потоковых данных	17
1.5 Анализ существующих средств воспроизведения аудиоданных в операционной системе iOS	18
1.5.1 Воспроизведение загруженных в оперативную паять аудио- данных	18
1.5.2 Воспроизведение аудиоданных с помощью системного сегментатора	19

1.5.3	Воспроизведение аудиоданных с помощью взаимодействия с звуковой картой	20
1.5.4	Сравнение средств воспроизведения аудиоданных в операционной системе iOS	20
	ЗАКЛЮЧЕНИЕ	22
	СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ	23
	ПРИЛОЖЕНИЕ А	25

ОПРЕДЕЛЕНИЯ, ОБОЗНАЧЕНИЯ И СОКРАЩЕНИЯ

- 1) Стриминг (англ. streaming) — потоковая передача данных по сети интернет;
- 2) Битрейт (от англ. bitrate) — количество бит, используемых для передачи или обработки данных в единицу времени;
- 3) iOS (iPhone OS) — операционная система для мобильных устройств, выпускаемых американской компанией «Apple»;
- 4) Аналого-цифровой преобразователь — устройство, преобразующее входной аналоговый сигнал в дискретный формат;

ВВЕДЕНИЕ

За последние 10 лет рынок аудиостриминга значительно вырос и претерпел сильные изменения[1]. Всё больше людей пользуются сервисами потокового прослушивания музыки, подкастов и аудиокниг. Исследование [2] показало, что число пользователей мобильных устройств превысило 2.5 миллиарда человек. Ежегодный прирост в 5 – 10% пользователей способствует стремительному увеличению популярности мобильных сервисов. В результате возникает потребность в поддержке воспроизведения потоковых аудиоданных на мобильных устройствах.

Цель работы – провести классификацию методов воспроизведения потокового аудио в мобильном приложении на операционной системе iOS.

Для достижения поставленной цели в ходе работы требуется решить следующие задачи:

- рассмотреть цифровое представление аудиоданных;
- провести анализ форматов хранения аудиоданных;
- провести анализ протоколов передачи потоковых данных;
- провести анализ существующих средств воспроизведения аудиоданных в операционной системе iOS;

1 Аналитическая часть

В данном разделе рассматривается задача воспроизведения потокового аудио. Происходит постановка задачи, описывается предметная область. Рассматриваются прикладные протоколы передачи потоковых данных. Дается анализ существующих средств воспроизведения аудиоданных в операционной системе iOS.

1.1 Анализ предметной области

Воспроизведение потоковых аудиоданных на мобильных устройствах зависит от протоколов потоковой передачи данных, форматов хранения аудиоданных, их обработки, а также от программных интерфейсов, предоставляемых разработчикам.

Зачастую используемые решения могут быть не универсальными, а их выбор продиктован спецификой области применения и поставленными задачами. Так при построении системы транслирования радиозфиров в реальном времени, можно пренебречь качеством (разрешением звука) записи, а для системы прослушивания аудиокниг, в которых важна дикция и подача чтеца, задержка передачи данных не является важным критерием.

1.2 Цифровое представления аудиоданных

Звук состоит из слышимых изменений давления воздуха. Звукозаписывающие устройства преобразуют изменение давления воздуха в переменное напряжение. Чтобы представить звук в цифровом виде, необходимо преобразовать это изменяющееся напряжение в ряд чисел, представляющих его амплитуду. Этот процесс известен как аналого-цифровое преобразование или дискретизация звука. Числа, выдаваемые аналого-цифровым преобразователем, в общем случае произвольны.

Изменение давления воздуха и, следовательно, соответствующее напряжение, создаваемое микрофоном, является непрерывным в двух измерениях. То есть значения непрерывно изменяются, и они существуют в каждый момент

времени. Однако цифровая система, такая как компьютер, не может напрямую представлять непрерывный сигнал. Вместо этого применяется ряд процессов для оцифровки звука:

- Дискретизация — преобразование сигнала в конечное множество дискретных моментов времени;
- Разрешение звука — количество используемых дискретных уровней амплитуды сигнала;
- Квантирование — округление значения сигнала к одному из конечных дискретных уровней амплитуды. Обычно выражается в битах, то есть как логарифм по основанию 2 фактического числа;

В результате вышеприведённых преобразований волна из своей непрерывной формы переходит в дискретизированную.

Ниже, на рисунке 1 представлена непрерывная формы волны:

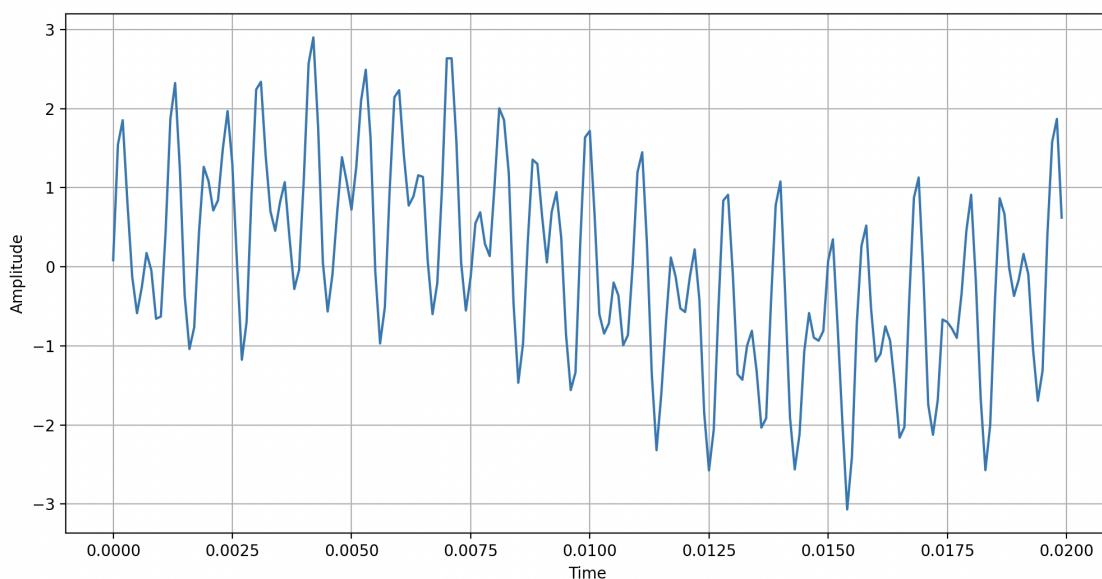


Рисунок 1 – Непрерывная форма волны.

На рисунке 2 представлено наложение дискретных уровней, полученных в ходе дискретизации, на непрерывную форму волны:

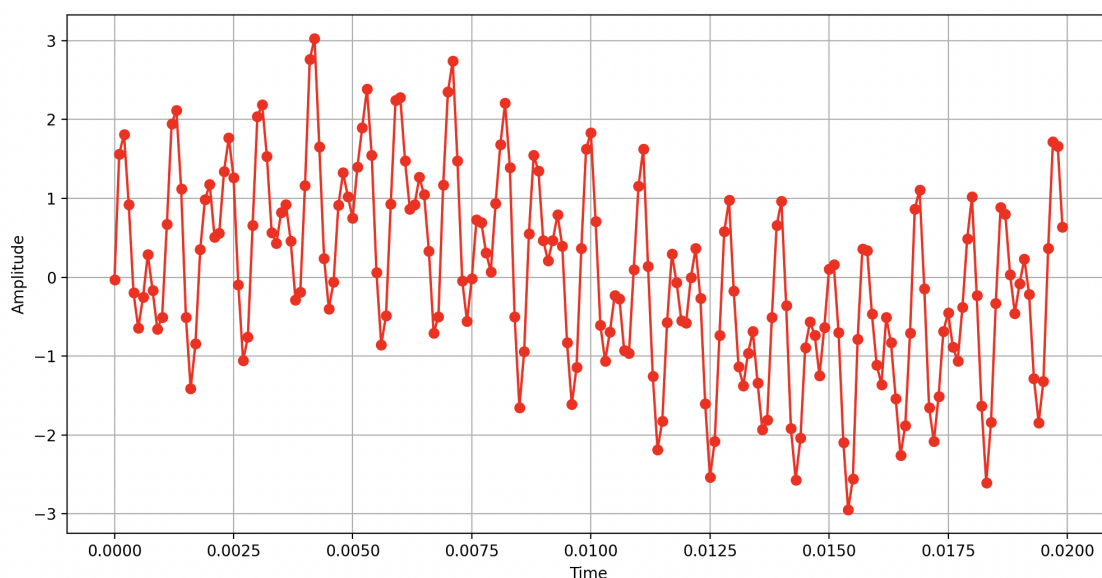


Рисунок 2 – Наложение дискретных уровней на непрерывную форму волны.

Частота дискретизации

Частота дискретизации — это количество измерений амплитуды сигнала в секунду. Чем выше частота дискретизации, тем точнее дискретизированный сигнал будет представлять исходный сигнал.

Выбор частоты дискретизации определяется теоремой Котельникова о связи непрерывных и дискретных сигналов. Эта теорема утверждает, что если максимальная частота, при которой исходный сигнал содержит энергию, равна f_c , то, если он дискретизируется со скоростью, строго превышающей $2f_c$, можно будет полностью восстановить исходный сигнал из дискретизированного сигнала, то есть дискретизированный сигнал будет содержать всю информацию исходного сигнала.

Другими словами, теорема Котельникова говорит о том, что непрерывный сигнал можно представить в виде интерполяционного ряда Фурье. Данное представление отображено в формуле 1:

$$\begin{cases} x(t) = \sum_{k=-\infty}^{\infty} x(k\Delta) \operatorname{sinc}\left[\frac{\pi}{\Delta}(t - k\Delta)\right], \\ 0 < \Delta \leq \frac{1}{2f_c}, \end{cases} \quad (1)$$

где Δ – интервал дискретизации (сек.).

1.3 Анализ форматов хранения аудиоданных

В общем случае аудиофайлы могут быть не сжаты, а также не иметь явного формата. Чаще всего аудиоданные обрабатывают для хранения по специальному формату, хранят в файле специального формата, представляющим «заголовок», а также тело — выборку целых чисел, характеризующих цифровой сигнал для воспроизведения аудио.

Заголовок может содержать:

- тип файла;
- разрешение звука;
- частота дискретизации;
- информацию о сжатие, если таковое имеется;
- количество байтов, следующих за заголовком;
- информацию о названии и исполнителе;

1.3.1 Необработанные звуковые файлы

Звуковые файлы, которые не имеют заголовка, а содержат только числовую выборку, характеризую цифровой сигнал, называются необработанными. Отсутствие явной информации о частоте дискретизации и разрешение позволяет использовать аудиоданные только по принятым соглашениям, которые принимаются между теми, кто упаковывает и использует данные аудио-файлы.

1.3.2 WAV (WAVE) формат

Для данного формата не применяется сжатие к битовому потоку, аудиозаписи хранятся с разными частотами дискретизации и битрейтами. Однако WAV [3] файлы имеют больший размер по сравнению с другими популярными фор-

матами, такими как MP3, в которых используется сжатие с потерями для уменьшения размера файла при сохранении того же качества звука. Разрешение звука может быть как беззнаковым 8-битным, так и знаковым 16-битным (фрагменты дублируют информацию, найденную в других фрагментах). Засчёт дубликатов блоков данных, характеризую цифровой сигнал в конкретный момент времени, обработка информации может занять дополнительное время.

Заголовок файла содержит:

- размер файла в байтах;
- длину блока данных;
- количество аудиоканалов;
- битрейд;
- частота дискретизации;

1.3.3 MP4 формат

MP4 [4] — это международный стандарт кодирования аудиовизуальных материалов, из-за высокой степени сжатия, результирующие файлы имеют меньший размер с почти полным сохранением исходного качества. Данные в файле MP4 разделены на два раздела, первый из которых содержит мультимедийные данные, а второй содержит метаданные. Структуры в MP4 обычно называются атомами или блоками. Минимальный размер атома составляет 8 байт (первые 4 байта определяют размер, а следующие 4 байта определяют тип).

Первый раздел содержит следующую информаию:

- таблица выборок значений цифрового сигнала в момент времени;
- заголовок аудиоданных, содержащий частоты дискретизаии (до 48 кГц) и разрешение звука;
- количество аудиоканалов;
- битрейд;

Второй раздел содержит следующую информаию:

- дескриптор файла;
- информацию о сжатие;

1.3.4 MP3 формат

Файл MP3 [5] состоит из фреймов, каждый из которых состоит из заголовка и блока данных. Фреймы не являются независимыми и обычно не могут быть извлечены на произвольных границах переходов фреймов. Блоки данных файла содержат информацию об аудио с точки зрения частот и амплитуд. Заголовок идентифицирует начало допустимого кадра. Затем следуют 3 бита, где первый бит показывает, что это стандарт MPEG, а оставшиеся 2 бита показывают, что используется уровень MPEG-1 Audio Layer [7]. После этого значения будут различаться в зависимости от содержимого конкретного аудиофайла.

Заголовок фрейма содержит следующую информацию:

- информация о переходе на текущий фрейм;
- идентификатор версии MPEG Audio;
- описание фрейма;
- бит, который показывает, что фрейм зашифрован;
- битрейт;
- частота дискретизации (от 8 кГц до 48 кГц);
- приватный ключ шифрования;

1.3.5 AAC формат

AAC (Advanced Audio Coding) [6] относится к стандарту цифрового кодирования звука, который представляет аудиофайлы на основе сжатия звука с потерями. Он был запущен как преемник формата файла MP3 с учетом того, что у сторонних разработчиков возникли проблемы с реализацией новых идей в процессе кодирования, основанных на развитии методов сжатия данных. AAC обеспечивает лучшее качество звука по сравнению с MP3 при той же скорости передачи данных. Основные различия между AAC и MP3 с точки зрения улучшений заключаются в поддержке более широкого диапазона и частот дискретизации (от 8 кГц до 96 кГц). Разбиение по фреймам и содержание заголовка фрейма идентичны с MP3 форматом.

1.3.6 Сравнение форматов хранения аудиоданных

Сравнение рассмотренных форматов хранения аудиоданных рассмотрены в таблице 1. Обозначения:

- 1 — поддержка частоты дискретизации более 48 кГц;
- 2 — наличие заголовка с метаданной;
- 3 — поддержка сжатия данных;
- 4 — фрагментация данных цифрового сигнала;

Таблица 1 – Сравнение форматов хранения аудиоданных

Формат	1	2	3	4
WAV	+	-	-	-
MP3	-	+	+	+
MP4	-	+	+	-
ACC	+	+	+	+

По результатам сравнения можно сделать вывод, что наиболее оптимальными форматами хранения аудиоданных являются ACC и MP3. Принципиальное отличие заключается в диапазоне поддерживаемых частот дискретизации. При требовании поддержки частоты дискретизации до 48 кГц форматы равноэффективны, но при необходимости в поддержке частоты до 96 кГц, необходимо использовать формат ACC. Стоит отметить, что формат WAV подходит в случае заранее принятых стандартов звукозаписи, при которых метаданные, хранящиеся в заголовках других форматов, будут известны.

1.4 Анализ протоколов передачи потоковых данных

Для передаче данных при стриминге необходимо использовать прикладные протоколы передачи данных, сжимающие данные для быстрой транспортировки, а также форматирующие данные в специальный формат.

1.4.1 HLS (HTTP Live Streaming)

На сегодняшний день HLS [8] является наиболее часто используемым протоколом для потоковой передачи. Данный протокол создан на основе протокола прикладного уровня HTTP, использующий протокол транспортного уровня TCP [11]. Концептуально HTTP Live Streaming состоит из трех частей:

- серверный компонент;
- компонент доставки;
- клиентское программное обеспечение.

В классической конфигурации аппаратный кодировщик принимает на вход аудио-файл. Происходит кодировка аудио в АСС формат, затем программный сегментатор разбивает поток на серии коротких медиафайлов, размещающихся на сервере. Сегментатор создает и поддерживает файл-индекс, содержащий список медиафайлов. Клиент считывает индекс, затем запрашивает перечисленные файлы по порядку и отображает их без остановок, разрывов между сегментами. На рисунке 3 изображена схема принципа работы протокола HLS.

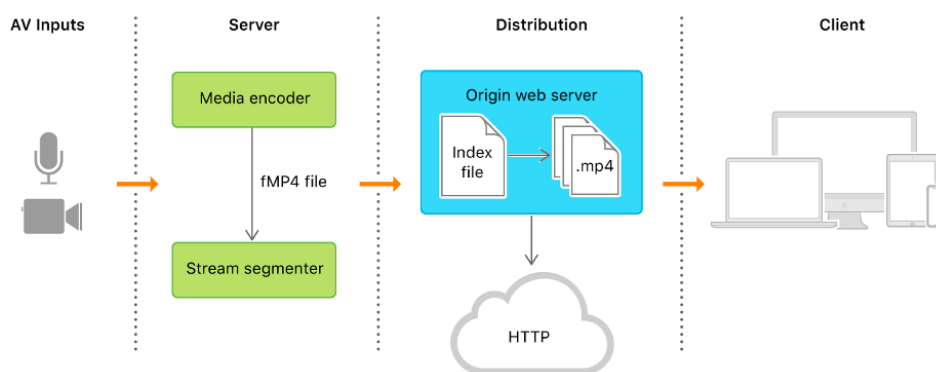


Рисунок 3 – Схема принципа работы протокола HLS.

К достоинствам данного протокола можно отнести:

- сегментация и индексация потока файлов, полученных в результате работы сегментатора;
- поддержка АСС формата, что позволяет передавать аудиоданные с частотой дискретизации от 8 до 96 кГц;
- гарантированная последовательная передачи пакетов без потерь;

К недостаткам данного протокола можно отнести:

- предварительная обработка исходного файла, которая заключается в разбиение и сжатие на подфайлы специального формата;
- сильная задержка при передаче данных, которая не позволяет транслировать данные в режиме реального времени;

1.4.2 Low-Latency HLS

Low-Latency HLS [9] позволяет передавать медиа файлы на параллельных каналах, с помощью разделения медиафайлов на большее количество маленьких файлов CMAF Chunks. Эти файлы называются частичными сегментами HLS. Т.к. каждый частичный сегмент имеет меньшую длительность, он может быть упакован, передан или добавлен быстрее чем его родительский сегмент. Пока стандартный медиасегмент может быть длительностью 6 секунд каждый, частичный сегмент может быть до 1 секунды.

Ускорение передачи данных достигается с помощью усложнения процесса предварительной обработки файла. Происходит дробления подфайлов, полученных сегментатором, на частичные сегменты. Создаётся список контекста сегментов, которые находятся в очереди на отправку данных. Параллельно происходит фильтрация списка, которая отбрасывает сегменты, которые считаются незначительными.

По сравнению с стандартной организацией HLS, данный протокол обеспечивает более быструю передачу пакетов, но требует дополнительной фильтрации сегментов, на которые делится исходный файл. Также может происходить потеря какого-то блока данных цифрового сигнала, что приведёт к искажению воспроизведения аудиоданных.

1.4.3 RTSP (Real-Time Streaming Protocol)

RTSP [10] – протокол прикладного уровня. Реализует потоковую передачу медиа данных в реальном времени, а также устанавливает и управляет либо одним, либо несколькими синхронизированными во времени потоками. Источники данных могут быть как источниками реального времени, так и храни-

мыми. RTSP поддерживает как передачу данных, сегментируемых аппаратным сегментатором, так и передачу с помощью датаграм, благодаря чему совместим и с протоколом транспортного уровня TCP [11], и с UDP [12].

Механизм передачи данных в реальном времени обусловлен отсутствием предварительной фильтрации сегментов аудиофайлов. Аналогово-цифровой преобразователь полностью перенаправляет поток данных на RTSP сервер, который сегментирует поступающие данные, сжимает и конвертирует их в AAC или MP3 формат.

К достоинствам данного протокола можно отнести:

- отсутствие предварительной обработки поступающих данных;
- передача данных в режиме реального времени с задержкой до 2 секунд;
- гарантированная последовательная передачи пакетов;

К недостаткам данного протокола можно отнести:

- потеря пакетов данных при обработке аналогово-цифрового сигнала;
- сжатие данных обуславливает частоту дискретизации до 48 кГц;

1.4.4 Сравнение протоколов передачи потоковых данных

Сравнение рассмотренных протоколов передачи потоковых данных рассмотрены в таблице 2. Обозначения:

- 1 — поддержка частоты дискретизации более 48 кГц;
- 2 — трансляция в режиме реального времени;
- 3 — гарантированная последовательная передача пакетов без потерь;
- 4 — необходимость предварительной обработки аудиофайлов перед трансляцией;

Таблица 2 – Сравнение протоколов передачи потоковых данных

Протокол	1	2	3	4
HLS	+	-	+	+
Low-Latency HLS	+	-	-	+
RTSP	-	+	-	-

По результатам сравнения можно сделать вывод, протоколы семейства HLS хорошо подходят для транслирования предварительно обработанных аудио-файлов с частотой дискретизации более 48 кГц, но не поддерживают передачу данных в режиме реального времени. Протокол RTSP, напротив, лучше подходит для трансляции в реальном времени, но плохо поддерживает файлы с большой частотой дискретизации. Стоит отметить, что при использовании Low-Latency HLS и RTSP возможна потеря данных при трансляции.

1.5 Анализ существующих средств воспроизведения аудиоданных в операционной системе iOS

Воспроизведение аудиоданных зависит от программных компонентов, а также доступных ресурсов, предоставляемых разработчикам. Ниже рассмотрены доступные подходы и механизмы для воспроизведения аудио в операционной системе iOS [13].

1.5.1 Воспроизведение загруженных в оперативную память аудиоданных

Один из подходов к воспроизведению аудиоданных в операционной системе iOS заключается в обработке и загрузке аудио-файла целиком в оперативную память устройства. Работа происходит с аудио-файлами целиком, в оперативную память устройства выгружаются заголовки и блоки данных цифрового сигнала, что напрямую связывает возможность проигрывания аудио с доступным размером оперативной памяти.

Программным компонентом который позволяет воспользоваться данным подходом является AVPlayer, позволяющий работать с локальными файлами

устройства, имеющими форматы MP3, MP4, AAC. Поддержка частоты дискретизации вплоть до 96 кГц, а также разрешение звука до 32 бит даёт возможность воспроизведение и управления аудиоданных высокого качества.

К плюсам данного подхода можно отнести:

- поддержка аудио-файлов с частотой дискретизации до 96 кГц;
- поддержка аудио-файлов с разрешением звука до 32 бит;
- отсутствие задержек, обусловленных получением пакетов данных при потоковой передаче;
- поддержка форматов AAC, MP3;

К недостаткам данного подхода можно отнести:

- полная выгрузка аудиоданных в оперативную память, отсутствие сегментации файла;
- отсутствует поддержка потокового воспроизведение данных:

1.5.2 Воспроизведение аудиоданных с помощью системного сегментатора

Системный сегментатор позволяет обрабатывать пакеты данных при потоковой передаче аудиоданных. Происходит обработка сегментов, сформированных протоколами потоковой передачи данных (HLS, RTSP), которая заключается в получении заголовка фрейма блока данных, который подаётся для воспроизведения. В операционной системе iOS сегментацию пакетов аудиоданных содержат два программных компонента: AVAudioPlayer [14], AVAudioEngine [15].

AVAudioPlayer поддерживает протокол потоковой передачи данных HLS, частоту дискретизации до 48 кГц, а также разрешение звука до 16 бит. Основной особенностью является управление воспроизведением аудио.

AVAudioEngine поддерживает протоколы HLS и RTSP, а также разрешение звука до 32 бит. Данный программный компонент не предполагает управление воспроизведением аудиоданных, но даёт возможность наложения эффектов аудиофильтров и работы с нотной матрицей.

1.5.3 Воспроизведение аудиоданных с помощью взаимодействия с звуковой картой

Работа с звуковой картой устройства поддерживается с помощью программного интерфейса CoreAudio [16], который поддерживает обработку пакетов аудиоданных для потокового воспроизведения, различные частоты дискретизации и разрешения звука, наложение звуковых эффектов, работу с нотной матрицей и обработку выделения памяти для хранения сегментов данных цифрового сигнала. При работе с аудиокартой необходимо реализовывать собственный сегментатор, так как системная поддержка стандартов протоколов потоковой передачи данных отсутствует.

К плюсам данного подхода можно отнести:

- поддержка аудио-файлов с различными частотами дискретизации;
- поддержка аудио-файлов с различными разрешениями звука;
- поддержка обработки пакетов аудиоданных для потокового воспроизведения;

К недостаткам данного подхода можно отнести:

- отсутствие поддержки системного сегментатора;
- отсутствие поддержки форматов аудио-файлов, все звуковые файлы принимаются в качестве необработанных;
- отсутствие поддержки обработки стандартов протоколов потоковой передачи данных;

1.5.4 Сравнение средств воспроизведения аудиоданных в операционной системе iOS

Сравнение рассмотренных средств воспроизведения аудиоданных в операционной системе iOS рассмотрены в таблице 3. Обозначения:

- 1 — поддержка частоты дискретизации более 48 кГц;
- 2 — поддержка воспроизведения потоковых данных;
- 3 — воспроизведение в режиме реального времени;

- 4 — наложение звуковых эффектов;
- 5 — наличие встроенного сегментатора для обработки потоковых данных;
- 6 — поддержка MP3 или AAC форматов ауди-файлов;

Таблица 3 – Сравнение протоколов передачи потоковых данных

Подход	1	2	3	4	5	6
Воспроизведение загруженных в оперативную память аудиоданных	+	-	-	+	-	+
Воспроизведение аудиоданных с помощью системного сегментатора	-	+	+	-	+	+
Воспроизведение аудиоданных с помощью взаимодействия с звуковой картой	+	+	+	+	-	-

По результатам сравнения можно сделать вывод, что для воспроизведения потокового аудио подход с воспроизведением загруженных в оперативную память аудиоданных не подходит. Системный сегментатор позволяет обрабатывать пакеты потоковых аудиоданных, с использованием протоколов HLS и RTSP. Для воспроизведение аудио с частотой дискретизации более 48 кГц, то необходимо использовать программный компонент CoreAudio для взаимодействия с аудио-картой устройства.

ЗАКЛЮЧЕНИЕ

Среди рассмотренных методов воспроизведения потокового аудио в мобильном приложении на операционной системе iOS наиболее подходящим решением, с точки зрения поддерживаемых форматов ауди-файлов (AAC и MP3), протоколов передачи потоковых аудиоданных HLS и RTSP, является метод воспроизведение аудиоданных с помощью системного сегментатора.

В результате выполнения научной исследовательской работы была достигнута поставленная цель — проведена классификация методов воспроизведения потокового аудио в мобильном приложении на операционной системе iOS. Для достижения поставленной цели в ходе работы были решены следующие задачи:

- рассмотрено цифровое представление аудиоданных;
- проведён анализ форматов хранения аудиоданных;
- проведён анализ протоколов передачи потоковых данных;
- проведён анализ существующих средств воспроизведения аудиоданных в операционной системе iOS;

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Streaming is the future of media [Электронный ресурс]. Режим доступа URL: <https://www.robeco.com/en-int/insights/2020/07/streaming-is-the-future-of-media> (Дата обращения: 18.10.2022).
2. Mobile APPs and Global Markets [Электронный ресурс]. Режим доступа URL: <https://www.scirp.org/journal/paper-information-asp?paperid=85209> (Дата обращения: 13.10.2022).
3. Wav/wave audio format [Электронный ресурс]. Режим доступа URL: <https://docs.fileformat.com/audio/wav/> (Дата обращения: 18.10.2022).
4. MP4 audio format [Электронный ресурс]. Режим доступа URL: <https://docs.fileformat.com/video/mp4/> (Дата обращения: 18.10.2022).
5. MP3 audio format [Электронный ресурс]. Режим доступа URL: <https://docs.fileformat.com/audio/mp3/> (Дата обращения: 18.10.2022).
6. AAC audio format [Электронный ресурс]. Режим доступа URL: <https://docs.fileformat.com/audio/aac/> (Дата обращения: 18.10.2022).
7. MPEG Audio [Электронный ресурс]. Режим доступа URL: <https://sound.media.mit.edu/resources/mpeg4/audio/mpeg1> (Дата обращения: 18.10.2022).
8. HTTP Live Streaming [Электронный ресурс]. Режим доступа URL: <https://developer.apple.com/streaming/> (Дата обращения: 18.10.2022).
9. HTTP Live Streaming Low Latency [Электронный ресурс]. Режим доступа URL: <https://datatracker.ietf.org/doc/html/draft-pantos-hls-rfc8216bis> (Дата обращения: 18.10.2022).

10. Real Time Streaming Protocol [Электронный ресурс]. Режим доступа URL: <https://www.ietf.org/rfc/rfc2326.txt> (Дата обращения: 18.10.2022).
11. TRANSMISSION CONTROL PROTOCOL [Электронный ресурс]. Режим доступа URL: <https://www.ietf.org/rfc/rfc0793.txt> (Дата обращения: 18.10.2022).
12. User Datagram Protocol [Электронный ресурс]. Режим доступа URL: <https://www.ietf.org/rfc/rfc768.txt> (Дата обращения: 18.10.2022).
13. iOS mobile operating system [Электронный ресурс]. Режим доступа URL: <https://developer.apple.com/ios/> (Дата обращения: 18.10.2022).
14. AVAudioPlayer [Электронный ресурс]. Режим доступа URL: <https://developer.apple.com/documentation/avfaudio-avaudioplayer> (Дата обращения: 18.10.2022).
15. AVAudioEngine [Электронный ресурс]. Режим доступа URL: <https://developer.apple.com/documentation/avfaudio-avaudioengine> (Дата обращения: 18.10.2022).
16. Core Audio [Электронный ресурс]. Режим доступа URL: <https://developer.apple.com/documentation/coreaudio> (Дата обращения: 18.10.2022).

ПРИЛОЖЕНИЕ А