

МИНОБРНАУКИ РОССИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ОБРАЗОВАНИЯ
“ВОРОНЕЖСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ”

Факультет компьютерных наук

Кафедра цифровых технологий

Цифровой широкополосный стетоскоп

ВКР Бакалаврская работа

02.04.01 Математика и компьютерные науки

Распределённые системы и искусственный интеллект

Допущено к защите в ГЭК

Зав. кафедрой _____ С.Д. Кургалин, д.ф.-м.н., профессор ___. ___. 2017

Обучающийся _____ А.А. Родионов, 4 курс, д/о

Руководитель _____ Я.А. Туровский, к. мед. н, доцент

Воронеж 2017

МИНОБРНАУКИ РОССИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ОБРАЗОВАНИЯ
«ВОРОНЕЖСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»

Факультет компьютерных наук
Кафедра цифровых технологий

УТВЕРЖДАЮ
заведующий кафедрой
Кургалин С. Д.,
подпись расшифровка подписи

**ЗАДАНИЕ
НА ВЫПОЛНЕНИЕ ВЫПУСКНОЙ КВАЛИФИКАЦИОННОЙ РАБОТЫ
ОБУЧАЮЩЕГОСЯ РОДИОНОВА АЛЕКСАНДРА АЛЕКСАНДРОВИЧА**

фамилия, имя, отчество

1. Тема работы «Цифровой широкополосный стетоскоп», утверждена решением ученого совета факультета компьютерных наук от _____._____.20____
2. Направление подготовки 02.03.01 Математика и компьютерные науки.
шифр, наименование
3. Срок сдачи студентом законченной работы _____._____.20____
4. Календарный план:

№	Структура ВКР	Сроки выполнения	Примечание
1	Введение	03.09.2016	
2	Разработка усилителя для микрофона	10.09.2016	
3	Разработка аппаратной части готового устройства	12.10.2016	
4	Разработка ПО для АЦП LCard E14-140M	02.02.2017	
5	Разработка ПО для АЦП LA-n10-12USB	13.02.2017	
6	Разработка ПО для АЦП Arduino Due	16.03.2017	
7	Разработка ПО для CUDA-сервера	20.04.2017	
8	Заключение	13.05.2017	
9	Список использованных источников	03.06.2017	

Обучающийся Родионов А. А.
подпись расшифровка подписи

Руководитель Туровский Я. А.
подпись расшифровка подписи

Реферат

Бакалаврская работа 50с., 32 рисунка, 25 источников

МЕДИЦИНСКАЯ ДИАГНОСТИКА, СТЕТОСКОП, АНАЛОГО ЦИФРОВЫЕ ПРЕОБРАЗОВАТЕЛИ, ЦИФРОВАЯ ОБРАБОТКА СИГНАЛОВ, ВИЗУАЛИЗАЦИЯ ЗВУКОВОГО СИГНАЛА, БЫСТРОЕ ПРЕОБРАЗОВАНИЕ ФУРЬЕ

Целью данной работы является создание доступного и простого в производстве широкополосного цифрового стетоскопа. Стетоскоп должен быть способен регистрировать высокое качество звука. Он должен иметь высокую частоту дискретизации (больше 100кГц) и способен регистрировать сигнал в ультразвуковом диапазоне.(больше 20кГц)

В процессе выполнения работы тестировались 3 различных аналого цифровых преобразователя. Были выявлены плюсы и минусы каждого из АЦП применительно к поставленной задаче. Также был создан усилитель сигнала для микрофона.

В результате работы был создан рабочий прототип цифрового широкополосного стетоскопа. Стетоскоп способен регистрировать сигнал с частотой дискретизации 660кГц и способен анализировать ультразвук до 100кГц. Также было написано программное обеспечения для этого стетоскопа, позволяющего визуализировать сигнал, обрабатывать сигнал с помощью быстрого преобразования фурье (БПФ), а также визуализировать сигнал после обработки БПФ. В возможности программного обеспечения входит обработка сигнала на удаленном высокопроизводительном сервере с помощью технологии Nvidia CUDA.

Содержание

Введение	4
1 Цели и задачи	6
1.1 Цель	6
1.2 Задачи	6
2 Анализ предметной области	7
2.1 3M™ Littmann® Electronic Stethoscope Model 3200	7
2.2 CMS-VESD SPO2 PR	8
2.3 CMS-VE	9
2.4 CMS-M	11
2.5 Cardionics E-scope II	12
2.6 Thinklabs One	13
2.7 Eko Core	14
2.8 Eko Duo	15
3 Описание аппаратной части разработанной системы широкополосного стетоскопа	17
3.1 Описание микрофона	19
3.2 Описание усилителя	19
3.3 АЦП	23
3.3.1 LCard E14-140M	23
3.3.2 ЛА-н10-12USB	24
3.3.3 Arduino Due	26
4 Описание программного обеспечения	28
4.1 Програмное обеспечение для ЛА-н10-12USB	28
4.1.1 Инструкция по установке ПО	28
4.1.2 Описание интерфейса пользователя ПО	29
4.1.3 Описание кода программы	30
4.1.4 Измерение времени работы АЦП	32

4.1.5	Паралельные вычисления в программе	32
4.2	Программное обеспечение для Arduino Due	32
4.2.1	Программа для запуска на Arduino	33
4.2.2	Программа для запуска на компьютере	37
4.3	Программное обеспечение для CUDA-сервера	41
	Выводы	43
	Заключение	44
	Список использованных источников	46

Введение

Аускультация (выслушивание) звуков, исходящих от различных органов - одна из областей медицинской диагностики. Прибор для выслушивания звуков называется стетоскоп. Стетоскопы бывают акустические и электронные. Акустический стетоскоп передает звук от пациента непосредственно в ухо врачу. У электронных есть микрофон, который передает звук либо через динамики врачу, либо записывает для дальнейшего анализа.

Медицинская диагностика сердечно-сосудистых заболеваний является одной из важнейших отраслей современной медицины. По данным Всемирной Организации Здравоохранения (ВОЗ) сердечно-сосудистые заболевания - это основная причина смертей в мире. Максимальное количество смертей - это сердечно-сосудистые заболевания. Процентное соотношение этих смертей составляет примерно 31% или в абсолютных значениях 17.5 млн. человек. Сердечно-сосудистые заболевания нуждаются в раннем обнаружении, диагностике, консультировании и лечении.

Медицинская диагностика заболеваний легочной системы является не менее важной. Заболевания легочной системы, такие как респираторные инфекции дыхательных путей, хроническая обструктивная болезнь лёгких или ХОБЛ, рак легких и туберкулез входят в 10 болезней, от которых чаще всего умирают люди. (статистика ВОЗ на 2015 год).

Обычно люди, страдающие заболеваниями сердечно-сосудистой системы или легочной системы обращаются к участковому врачу в поликлинику (первичное звено). В первичном звене врач принимает решение о дальнейших действиях по лечению больного. Проблема заключается в том что в первичном звене РФ почти нет автоматизации. Из за этого врач первичного звена может принять неправильные решения из за несовершенства оборудования. Эту проблему можно решить, обеспечив врачей первичного звена оборудованием, позволяющим очень точно и автоматически определять со-

стояние пациента и помогать принять однозначное решение о дальнейшем лечении пациента. Основным инструментом диагностики у врачей первичного звена РФ является стетоскоп, поэтому совершенствование стетоскопов должно быть основным трендом в развитии диагностики первичного звена. Совершенствование должно вестись не только в сторону улучшения обработки слышимого звука, но также и в сторону новых частотных диапазонов (ультразвука).

Проблемой существующих на рынке электронных стетоскопов является невысокое качество звука. Под невысоким качеством звука подразумевается низкая частота дискретизации получаемого на выходе сигнала и как следствие неспособность выдавать данные о высокочастотном диапазоне звука (в частности ультразвука). Эти стетоскопы имеют частоту, не охватывающую весь звуковой диапазон, слышимый человеком, следовательно они теряют массу информации, которая может быть полезна врачам для осуществления медицинской диагностики пациентов.

1 Цели и задачи

1.1 Цель

Целью данной работы является создание доступного и простого в производстве широкополосного цифрового стетоскопа.

1.2 Задачи

1. обеспечить диапазон принятия сигналов во всем слышимом человеком диапазоне ($20\text{Гц} - 20\text{kГц}$)
2. обеспечить диапазон принятия сигналов в диапазоне от 20kГц до 100kГц
3. обеспечить одновременную работу в двух указанных выше диапазонах
4. обеспечить обработку полученного сигнала с использованием спектральных методов оценивания
5. обеспечить возможность локальной обработки сигнала
6. обеспечить возможность обработки сигнала на удаленном сервере

2 Анализ предметной области

В настоящий момент на рынке есть много решений в области цифровых стетоскопов. Приведем несколько примеров и проанализируем плюсы и минусы.

2.1 3M™ Littmann® Electronic Stethoscope Model 3200



Рисунок 1 – 3M™ Littmann® Electronic Stethoscope Model 3200

Технические характеристики:[1]

- Диапазон принимаемого сигнала: 10 - 2000Гц
- Возможность записи 12ти 30-секундных звуковых дорожек
- Передача данных через Bluetooth
- Удаленная прослушка с помощью технологии 3M™ Littmann® TeleSteth™
- Удаление 85% (в среднем) окружающего шума

- 24x кратное усиление сигнала
- Цена - 22000 руб
- Вес - 185г

Это один из самых лучших стетоскопов, существующих на рынке. Стетоскоп обеспечивает высокое качество звука, но только в нижнем диапазоне (до 2кГц). Также к плюсам нужно отнести высокое качество подавления окружающего шума. Минусом является высокая цена.

2.2 CMS-VESD SPO2 PR



Рисунок 2 – CMS-VESD SPO2 PR

Технические характеристики:[2]

- Диапазон принимаемого сигнала для сердца: 20 - 230Гц
- Диапазон принимаемого сигнала для легких: 100 - 800Гц
- Диапазон принимаемого сигнала для сердца и легких: 20 - 800Гц
- Измерение пульса: 30 - 300 ударов в минуту

- Точность измерения пульса: +- 2 уд/мин
- Диапазон измерения насыщения кислородом (SpO₂, процентное содержание оксигемоглобина в артериальной крови): 70% - 100%
- Дисплей, отображающий в реальном времени звуковую волну, пульс, SpO₂
- Возможность загрузки данных на компьютер через USB
- ПО для анализа данных
- Цена - 5800руб
- Вес - 100г

Данный стетоскоп обладает невысокой разрешающей способностью (маленький диапазон принимаемого сигнала для сердца и легких) и невысокой ценой.

2.3 CMS-VE



Рисунок 3 – CMS-VE

Технические характеристики:[3]

- Диапазон принимаемого сигнала для сердца: 20 - 230Гц
- Диапазон принимаемого сигнала для легких: 100 - 800Гц
- Диапазон принимаемого сигнала для сердца и легких: 20 - 800Гц
- Измерение пульса: 30 - 300 ударов в минуту
- Точность измерения пульса: +- 2 уд/мин
- Диапазон измерения насыщения кислородом (SpO2, процентное содержание оксигемоглобина в артериальной крови): 70% - 100%
- Дисплей, отображающий в реальном времени звуковую волну, пульс, SpO2
- возможность загрузки данных на компьютер через USB
- ПО для анализа данных
- Цена - 5646руб
- вес - 100г

Преимуществом данного прибора является возможность записывать процентное содержание оксигемоглобина в артериальной крови (SpO2). Недостатками являются невысокая разрешающая способность 20 - 800Гц

2.4 CMS-M



Рисунок 4 – CMS-M

Технические характеристики:[4]

- Измерение пульса: 30 - 300 ударов в минуту
- Точность измерения пульса: +- 2 уд/мин
- Диапазон измерения насыщения кислородом (SpO_2 , процентное содержание оксигемоглобина в артериальной крови): 70% - 100%
- Дисплей, отображающий в реальном времени звуковую волну, пульс, SpO_2
- возможность загрузки данных на компьютер через USB
- ПО для анализа данных
- Цена - 6500руб

Преимуществом данного прибора является возможность записывать процентное содержание оксигемоглобина в артериальной крови.) Недостатком является невысокая разрешающая способность 20 - 800Гц

2.5 Cardionics E-scope II



Рисунок 5 – Cardionics E-scope II

Технические характеристики:[5]

- Диапазон 45-900Hz для звуков сердца и 50-2000Hz
- 30-ти кратное усиление сигнала по сравнению с акустическим стетоскопом.
- вход для наушников
- возможность подключиться к компьютеру через USB-кабель для визуализации сигнала
- переключатель между звуками сердца и звуками легких
- Цена - 20900руб

Преимуществами данного прибора являются достаточно высокий, по сравнению с другими аналогами, диапазон принимаемого сигнала (20 - 2000Гц) и возможность подключиться к компьютеру для визуализации сигнала) Недостатком является высокая цена.

2.6 Thinklabs One



Рисунок 6 – Thinklabs One

Технические характеристики:[6]

- диапазон 10-1000Гц
- более чем 100 кратное усиление сигнала по сравнению с акустическим стетоскопом.
- система аудио-фильтров: несколько фильтров дают возможность контролировать множество нюансов выходного сигнала
- шумоподавление
- возможность подключиться к iPhone, iPad, Android, планшету, компьютеру.
- управление устройством и запись сигнала через мобильное приложение

Преимуществом данного прибора является наличие качественных приложений под основные операционные системы (iPhone, iPad, Android, macOS, Windows) Недостатком является то что приложения с закрытым исходным кодом, что делает невозможным модификацию программного обеспечения под свои нужды.

2.7 Eko Core



Рисунок 7 – Eko Core

Технические характеристики:[7]

- 40-кратное усиление сигнала
- Частота дискретизации 4000Гц

- Диапазон 20Гц – 2кГц
- Запись в .WAV формат
- Подключение через Bluetooth 4.0 low-energy
- Програмное обеспечение для iOS, Android, Windows

Преимуществом данного прибора является беспроводная передача данных через Bluetooth и наличие приложений под основные операционные системы (iPhone, iPad, Android, macOS, Windiws). Недостатком является то что приложения с закрытым исходным кодом, что делает невозможным модификацию программного обеспечения под свои нужды.

2.8 Eko Duo



Рисунок 8 – Eko Core

Технические характеристики:[8]

- 60-кратное усиление сигнала
- подавление окружающего шума

- Частота дискретизации 4000Гц
- Диапазон 20Гц – 2кГц
- 4 аудиофильтра
- Запись в .WAV формат
- Подключение через Bluetooth 4.0 low-energy
- Програмное обеспечение для iOS, Android, Windows
- Возможность снимать ЭКГ

Преимуществом данного прибора является относительно высокая частота дискретизации (4000Гц), наличие аудиофильтров и качественное шумоподавление. Недостатком является программное обеспечение с закрытым исходным кодом без возможности модифицировать его для своих задач.

Как видно из списка товаров, большинство стетоскопов не имеют возможность анализировать сигнал выше 1000 Гц. Те же которые имеют - дорого стоят. Многие дешевые стетоскопы вообще нацелены на измерение пульса, а не на подробный анализ звука.

3 Описание аппаратной части разработанной системы широкополосного стетоскопа

Разработанная аппаратная часть должна отвечать следующим условиям. Прием сигнала стетоскопом осуществляется через микрофон. Далее, сигнал с микрофона должен подаваться на усилитель, усиливающий сигнал с микрофона до значений, в которых работает аналого цифровой преобразователь. Аналого цифровой преобразователь принимает сигнал и подключается к компьютеру через USB порт для передачи данных. Врач должен иметь возможность прослушивать пациента, видеть визуализацию сигнала на компьютере. Также врач должен иметь возможность увидеть спектр сигнала после преобразования Фурье. Преобразование Фурье должно производиться как на локальном компьютере, так и на удаленном сервере.

Разработанная информационная система состоит из следующих частей:

1. Врач
2. Пациент (диагностируемый)
3. Микрофон
4. Усилитель
5. Аналого цифровой преобразователь
6. Компьютер
7. Удаленный сервер для высокопроизводительных вычислений

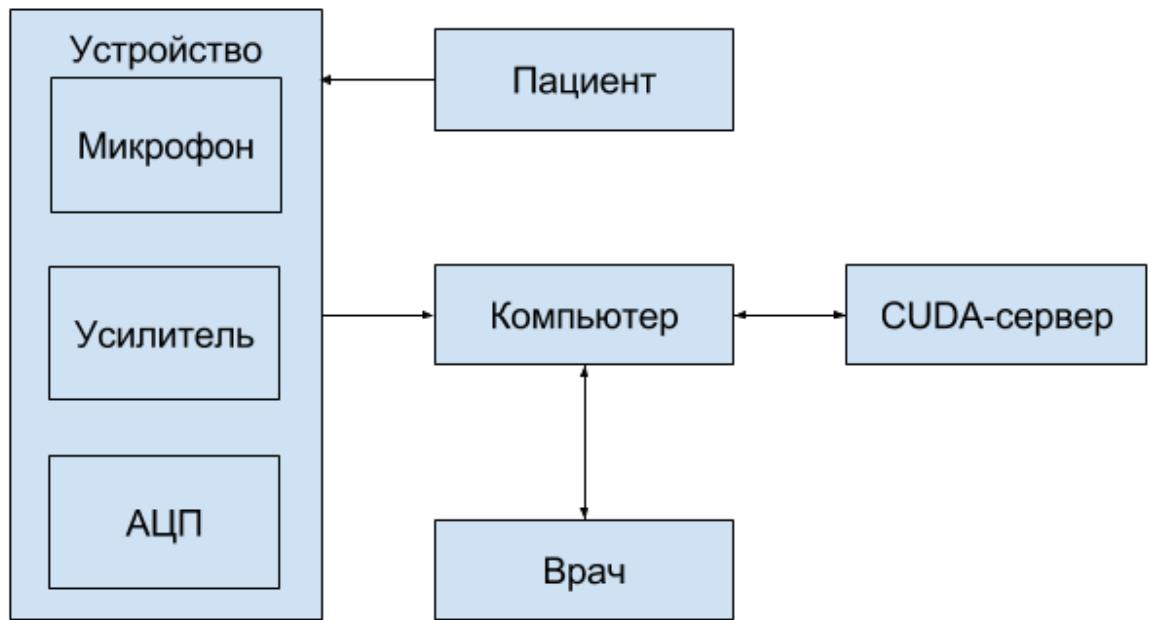


Рисунок 9 – Схема програмно-аппаратной системы



Рисунок 10 – Собранный прототип

3.1 Описание микрофона

В качестве микрофона был выбран SWEN MK-200. Микрофон был вынут из стандартного корпуса, чтобы лучше соединиться с трубкой, ведущей к мембране.

Таблица 1 – Технические характеристики SWEN MK-200

Чувствительность, дБ	-60 ± 3
Диапазон частот, Гц	50 – 16 000
Размер микрофонного модуля, мм	9×7
Тип разъема	мини-джек Ø 3,5 мм (3 pin)
Длина кабеля, м	1,8
Вес, г	63

Наилучшая чувствительность данного микрофона достигается в диапазоне частот от 50 до 16000 Гц. Тем не менее, микрофон с ослаблением принимает сигнал вплоть до 40кГц. Поэтому данный микрофон подходит для данного проекта.

Для подавления шумов и лучшей передачи звука от сердца, легких и других органов к микрофону присоединяются мембрана и соединительная трубка от аналогового стетоскопа.

3.2 Описание усилителя

Усилитель для микрофона был создан самостоятельно в рамках данной работы.

Была выбрана следующая схема усилителя:

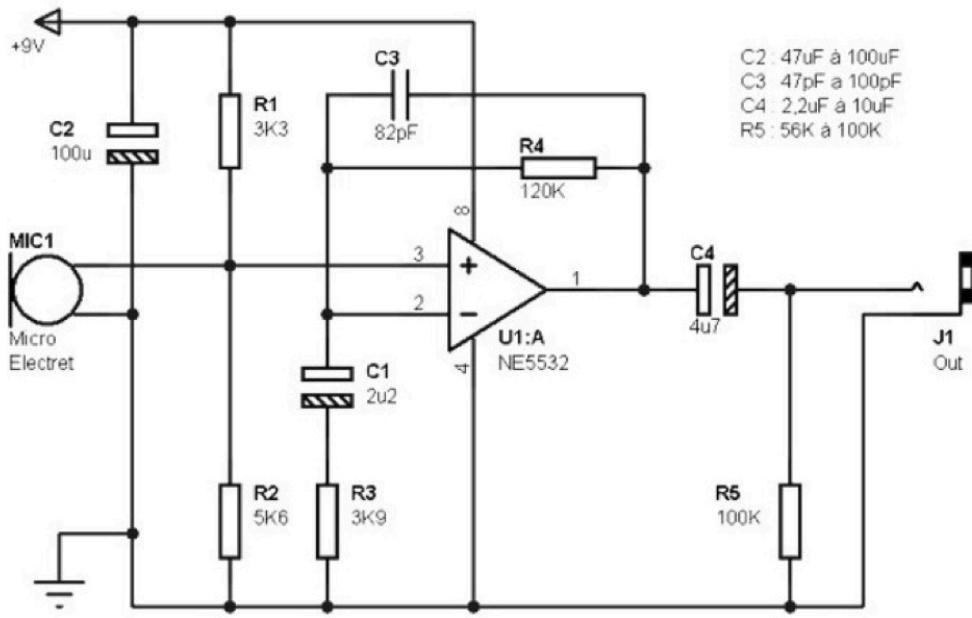


Рисунок 11 – Схема усилителя сигнала

В качестве операционного усилителя был выбран **MCP6022** от производителя Microchip. Это усилитель типа Rail-to-Rail SO-8.

SOIC или просто SO (small-outline-integrated-circuit), а также SOP (Small-Outline Package) корпус микросхем, предназначенный для поверхностного монтажа, занимающий на печатной плате на 30-50% меньше площади чем аналогичный корпус DIP, а также имеющий на 50-70% меньшую толщину. Обычно в обозначении также указывается число выводов.

Ниже приводятся параметры операционного усилителя.

Таблица 2 – Технические характеристики операционного усилителя MCP6022

Полоса частот	10МГц
Уровень шума	8.7 нВ/ $\sqrt{\text{Гц}}$
Количество каналов	2
Напряжение питания	2.5В — 5.5В
Напряжение смещения	$\pm 500\mu\text{V}$
Гармонические искажения	0.00053%
Температурный диапазон	-40°C — +85°C
Тип корпуса	SO-8

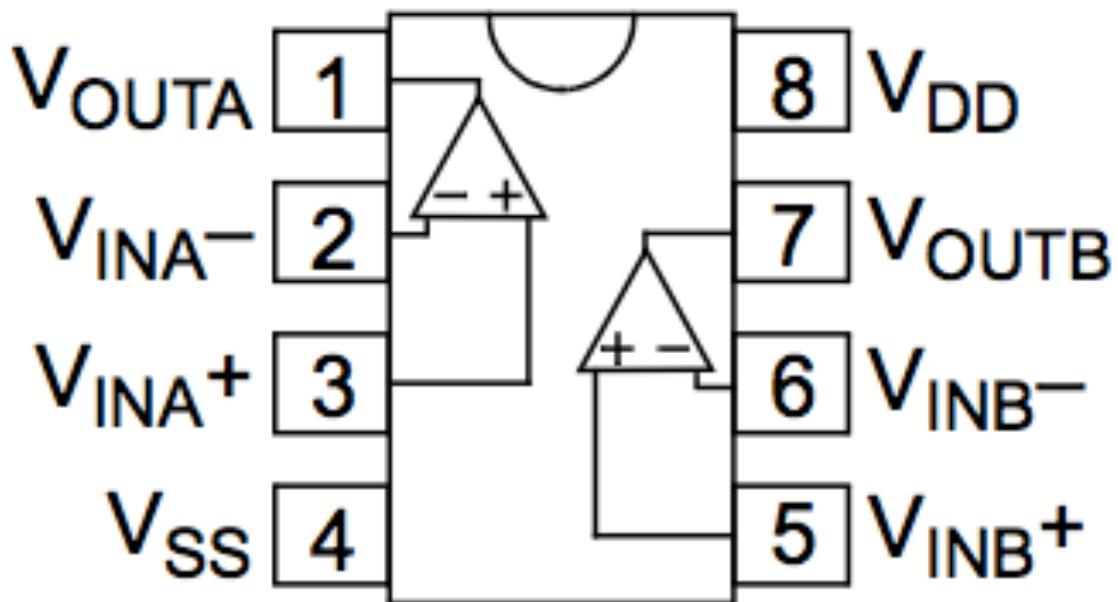
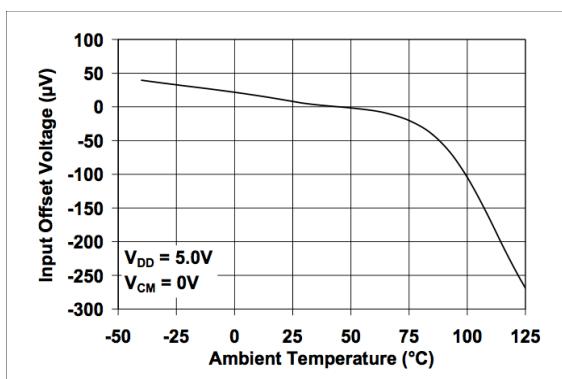
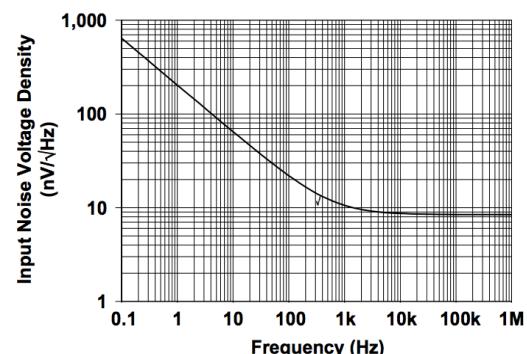


Рисунок 12 – Распиновка операционного усилителя

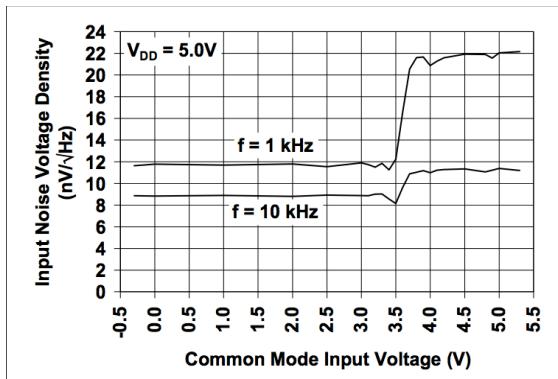
В данный усилитель встроены High-Pass и Low-Pass фильтры. High-Pass фильтрует частоты сигнала меньше 1Гц. Low-Pass фильтрует частоты выше 100кГц. Меняя конденсатор C3, можно менять частоту среза LowPass фильтра. Усиление схемы зависит от резисторов R3 и R4. На текущий момент усиление составляет порядка 100.



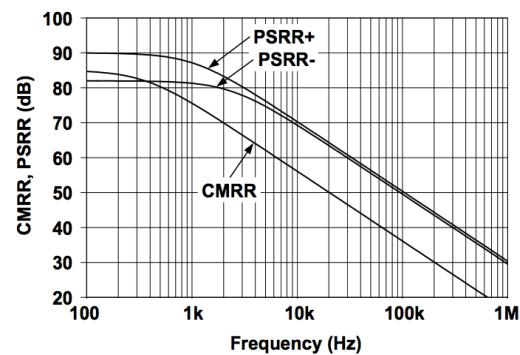
(a) Напряжение смещения — Температура



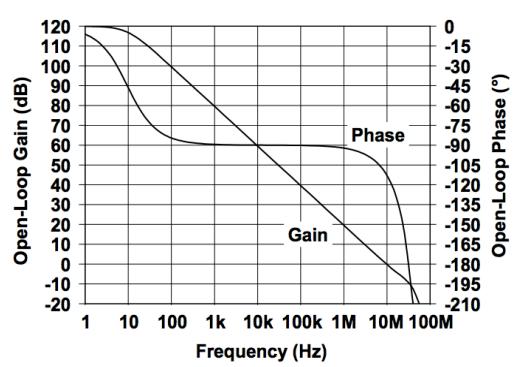
(b) Шум — Частота



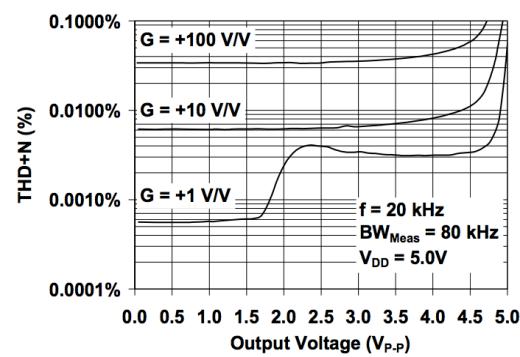
(a) Шум - Напряжение смещения



(b) CMRR — Частота



(c) Коэффициент усиления - Частота



(d) Гармонические искажения — Вых. напряжение ($f=20$ кГц)

Рисунок 14 – Параметры операционного усилителя

Разработка платы усилителя велась в программе Sprint Layout. Ниже представлен скриншот макета платы из этой программы.

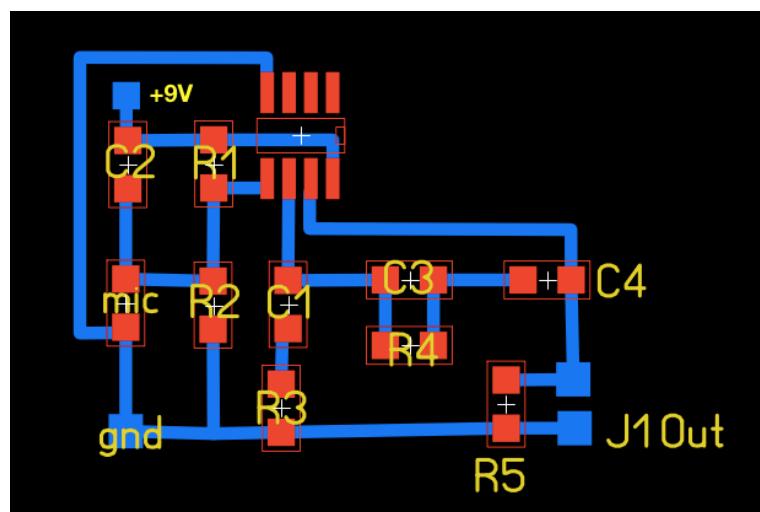


Рисунок 15 – Схема печатной платы усилителя

Печатная плата изготавливалась при помощи печати на лазерном принтере методом травления текстолита.

3.3 АЦП

Предназначение АЦП – преобразование непрерывных (аналоговых) входных сигналов в цифровую форму для дальнейшей обработки с помощью компьютера.

В качестве аналого цифрового преобразователя были опробованы 3 устройства:

1. LCard E14-140M
2. ЗАО "Руднев-Шиляев"ЛА-н10-12USB
3. Arduino Due

3.3.1 LCard E14-140M

Таблица 3 – Технические характеристики АЦП LCard E14-140M

Количество каналов	16 дифференциальных или 32 с "общей землей"
Максимальная частота дискретизации	200 кГц
Объем буфера памяти	64 Кбайт
Разрядность	14 (бит)
Эффективная разрядность	13.3 бит (при частоте дискр. 100 кГц)
Входное сопротивление	не меньше 10 МОм
Диапазоны входного напряжения	±10V; ±2.5V; ±0.6V; ±0.15V
Синхронизация	От внешнего синхро-сигнала, по уровню аналогового, внутренняя. Возможна многомодульная.
Защита входов	30V (при вкл. питании); 10В (при выкл. питания и в suspend-mode)



Рисунок 16 – АЦП LCard E14-140M

При работе с данным АЦП возникли трудности при написании программного обеспечения. Средства разработки ПО для данного ацп давно не обновлялись. Работа с АЦП плохо документирована производителем и приведено мало примеров использования. Тем не менее, удалось запустить данный АЦП на одном из примеров производителя. Данный пример позволяет записывать данные с АЦП в бинарный файл.

3.3.2 ЛА-н10-12USB

Таблица 4 – Технические характеристики АЦП ЛА-н10-12USB

Число аналоговых входов	2
Минимальная частота дискретизации	1.25МГц
Максимальная частота дискретизации	80МГц
Объем буфера памяти	$2 \times 10^{19} = 524288$
Разрядность	12бит (4096 значений)
Входное сопротивление	50Ом
Разъем	BNC
Диапазоны входного напряжения	$\pm 2V; \pm 1V; \pm 0.4V; \pm 0.2V$
Задержка по входному напряжению	$\pm 5V$
Дифференциальная нелинейность	± 1.2 МЗР
Интегральная нелинейность	± 1.5 МЗР
Ошибка сдвига	$\pm 0.15\%$
Интерфейс	USB
Потребляемая мощность	12В, 0.7А
Масса	400г



Рисунок 17 – АЦП ЛА-н10-12USB

Интегральная нелинейность - это отклонение точек по вертикальной оси реальной хар-ки от идеальной хар. преобразования, которое делит пополам расстояние по оси X между средними значениями пороговых значений характеристики преобразования. Измеряется в процентах или МЗР.

Дифференциальная нелинейность - отклонение разности двух аналоговых сигналов от значения, соответствующего единице МЗР.

Для данного АЦП было написано программное обеспечение, позволяющее принимать данные с АЦП и визуализировать их в реальном времени.

Особенности реализации програмно-аппаратных решений данного АЦП, не позволяющие его использовать в нашем проекте:

Данный АЦП работает в режиме старт-стоп. Другими словами он периодически производит запуск, сбор данных(в буффер) и остановку. Это один полный цикл сбора данных. При этом полезное время - это сбор данных, а старт и стоп - бесполезное.

Время, за которое АЦП совершает полный цикл сбора данных и соотношение полезного и бесполезного времени отличается в зависимости от частоты дискретизации и размера буфера. Были произведены замеры времени для различных значений частоты дискретизации и размера буфера и были составлены следующие таблицы.

Во всех таблицах ось X: размер буфера, ось Y: частота дискр.

	2^{19}	2^{18}	2^{17}	2^{16}	2^{15}	2^{14}	2^{13}	2^{12}	2^{11}	2^{10}
$8 \cdot 10^7$	190	96	48	25	13	7	4	2	2	1
$8 \cdot 10^6$	348	174	88	44	23	12	6	4	2	2
$8 \cdot 10^5$	851	426	214	107	54	27	14	8	4	3

Рисунок 18 – Время на полный цикл (мс)

	2^{19}	2^{18}	2^{17}	2^{16}	2^{15}	2^{14}	2^{13}	2^{12}	2^{11}	2^{10}
$8 \cdot 10^7$	6.55	3.27	1.6	0.81	0.4	0.2	0.1	0.05	0.02	0.01
$8 \cdot 10^6$	65.5	32.7	16.3	8.19	4.09	2.04	1.02	0.51	0.25	0.12
$8 \cdot 10^5$	655	327	163	81.9	40.9	20.4	10.2	5.12	2.56	1.28

Рисунок 19 – Полезное время (мс)

	2^{19}	2^{18}	2^{17}	2^{16}	2^{15}	2^{14}	2^{13}	2^{12}	2^{11}	2^{10}
$8 \cdot 10^7$	0.034	0.034	0.033	0.032	0.031	0.029	0.025	0.025	0.010	0.010
$8 \cdot 10^6$	0.188	0.188	0.185	0.186	0.178	0.170	0.170	0.128	0.125	0.060
$8 \cdot 10^5$	0.770	0.768	0.762	0.765	0.757	0.756	0.729	0.640	0.640	0.427

Рисунок 20 – Отношение полезного времени к полному

Как видно из таблицы, даже при самых оптимальных значениях размера буфера и частоты дискретизации, отношение полезного времени к полному составляет 77%. (полезное время 655мс, полное - 851мс) То есть в конце каждого цикла образуется дырка длиной 196мс. При таком долгом времени на перезапуск теряется очень много информации, что неприемлемо для данного проекта. Поэтому было принято решение заменить его. Данный же подходит для коротких сигналов (меньше секунды), которые нужно оцифровывать в сверхвысоком качестве.

3.3.3 Arduino Due

Самым оптимальным вариантом АЦП для данного проекта оказался Arduino Due. Он сочетает в себе как простоту в использовании так и возможность

оцифровывать сигнал высокого качества. Максимальная частота дискретизации Arduino Due составляет 1МГц. В ходе данной работы удалось достичь максимума в 672кГц. Обычно среднее значение частоты дискретизации составляло 666кГц. Максимальное значение частоты дискретизации зависит также от производительности компьютера. Данное устройство тестировалось на MacBook Air 2014.

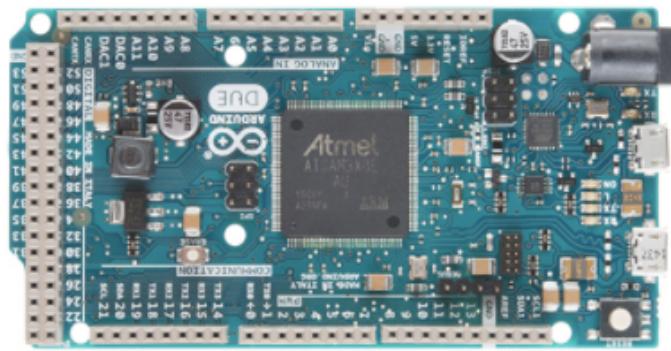


Рисунок 21 – Технические характеристики АЦП Arduino Due

Таблица 5 – Технические характеристики АЦП Arduino Due

Число аналоговых входов	12
Максимальная частота дискретизации	1МГц
Объем буфера памяти	512 КВ
Разрядность	12бит (4096 значений)
Рабочее напряжение	3.3V
Диапазоны входного напряжения	7-12V
Защита по входному напряжению	6-16V
Интерфейс	USB
Микроконтроллер	AT91SAM3X8E
Масса	36г

4 Описание программного обеспечения

В рамках данной работы было написано программное обеспечение для работы с ультразвуковым стетоскопом. Были разработаны программы под 2 разных АЦП: ЗАО ”Руднев-Шиляев”ЛА-н10-12USB и Arduino Due. Также была написана программа для вычисления быстрого преобразования Фурье на удаленном высокопроизводительном сервере с помощью технологии NVidia CUDA.

Ниже приводятся описания программ для работы с двумя разными АЦП а также описание программы, запускаемой на CUDA-сервере.

4.1 Програмное обеспечение для ЛА-н10-12USB

Програмное обеспечение написано на языке C# для платформы Windows.

В функционал программного обеспечения входит:

- Получение сигнала от АЦП ЛА-н10-12USB
- Отображение сигнала в реальном времени на экране в виде графика
- Отображение спектра сигнала полученного в результате быстрого преобразования Фурье
- Отображение скользящего среднего сигнала
- Возможность записи сигнала на жесткий диск для последующей обработки

4.1.1 Инструкция по установке ПО

- Скачать драйверы для АЦП ЛА-н10-12USB по ссылке:[10]

- Запустить и установить файл RShUSBDriver_XP_Vista_7_8_x86_x64_SDK2_2 (драйверы для USB устройств)
- Запустить и установить файл RshSDKSetup_2.0.x.y.exe (средства разработки ПО (SDK))
- Скачать проект для работы со стетоскопом по ссылке [20]
- Скомпилировать проект с помощью IDE Microsoft Visual Studio.

4.1.2 Описание интерфейса пользователя ПО

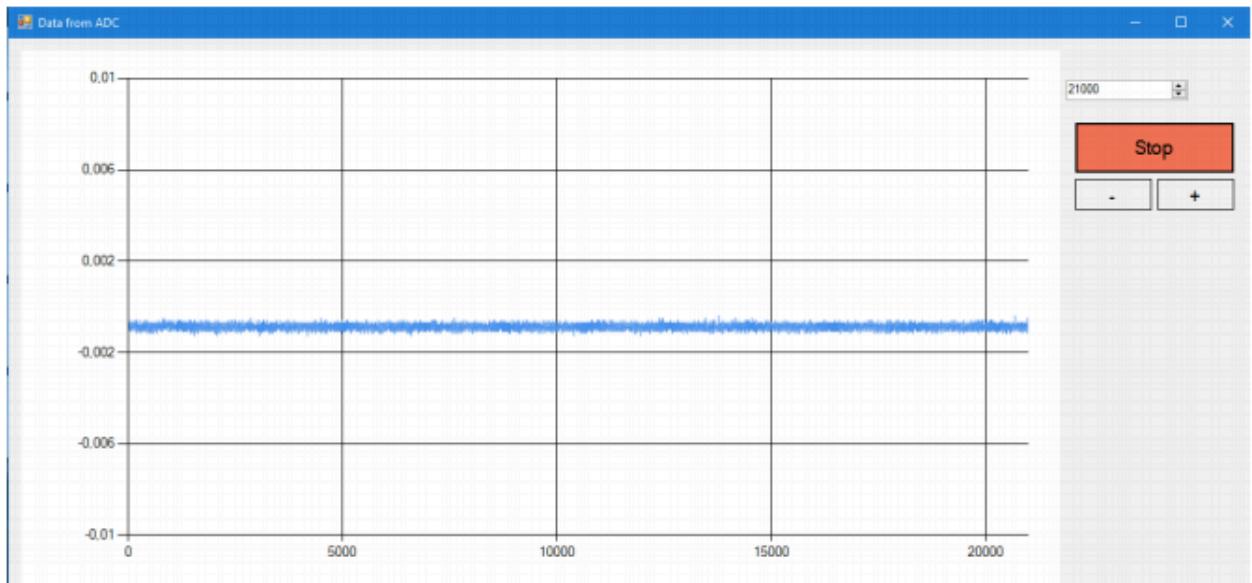


Рисунок 22 – Скриншот работы программы

В центре находится график сигнала с АЦП. Справа находится кнопка старт/стоп, которая управляет запуском и остановкой сбора сигнала с АЦП. Справа вверху есть числовое поле, которое отвечает за количество точек по горизонтальной оси на графике. Кнопки “+/-”Отвечают за увеличение и уменьшение размеров графика (zoom).

4.1.3 Описание кода программы

В начале программы задаются параметры размера буфера и частоты дискретизации (RATE).

```
const uint    BSIZE          = 524288; // размербуфера_
const double RATE          = 8.0e+7;
```

Далее при нажатии пользователем на кнопку старт, программа начинает инициализацию АЦП. Она проверяет подключено ли устройство, проверяет его работоспособность и начинает сбор данных.

За инициализацию отвечают следующие строчки в коде:

```
//===== INITIALISATION =====
st = device.EstablishDriverConnection(BOARD_NAME); //load and
connect to device's library
if (st != RSH_API.SUCCESS) SayGoodBye(st);
st = device.Connect(1); // connecting to device
if (st != RSH_API.SUCCESS) SayGoodBye(st);
p.startType      = (uint)RshInitMemory.StartTypeBit.Program;
p.bufferSize     = BSIZE; // Internal buffer block size
p.frequency      = RATE; // Sampling rate
p.channels[0].control = (uint)RshChannel.ControlBit.Used; // make 0 channel active
p.channels[0].gain   = 10; // gain coefficient for 0 channel
[1, 2, 5, 10] ~ [+0.2V, +0.4V, +1V, +2V]
st = device.Init(p); // Initialisation
if (st != RSH_API.SUCCESS) SayGoodBye(st);
```

Далее программа ждет пока пользователь нажмет кнопку старт. При нажатии на кнопку старт вызывается функция button1_Click, которая в свою очередь запускает новый процесс backgroundWorker1_DoWork. Этот процесс собирает данные с АЦП в бесконечном цикле пока пользователь не нажмет на кнопку стоп. Он записывает значения с АЦП в структуру данных очередь, доступ к которой имеет UI-процесс (процесс, отвечающий за отображение

и обработку элементов пользовательского интерфейса). По мере обновления данных UI процесс обновляет график.

За старт, сбор данных и остановку АЦП отвечает следующий блок кода:

```
double[] buffer = new double[p.bufferSize]; // buffer from adc
uint waitTime = 100000; // wait time (in miliseconds) until
    buffer overflow // default = 100000
while (getting_data)
{
    stopwatch.Restart();

    st = device.Start();
    if (st != RSH_API.SUCCESS) SayGoodBye(st);

    st = device.Get(RSH_GET.WAIT_BUFFER_READY_EVENT, ref waitTime
    );
    if (st != RSH_API.SUCCESS) SayGoodBye(st);

    st = device.GetData(buffer); // very big amount of data
    if (st != RSH_API.SUCCESS) SayGoodBye(st);

    device.Stop();

    // Queue Dequeue and Enqueue implementation with arrays
    double[] values_to_draw_copy = (double[])values_to_draw.Clone
        ();
    for (int i = 0; i < x_axis_points - r_buffer_size; i++)
        values_to_draw[i] = values_to_draw_copy[r_buffer_size + i
            ];

    for (int i = 0; i < r_buffer_size; i++)
        values_to_draw[x_axis_points - r_buffer_size + i] =
            buffer[BSIZE / r_buffer_size * i];

    stopwatch.Stop();
```

```
        Console.WriteLine("GetData() time:\t" +
    stopwatch.ElapsedMilliseconds + "ms");
}
```

Также процесс backgroundWorker замеряет время каждого цикла сбора данных. (stopwatch.Restart() и stopwatch.Stop()) Это время процесс пользовательского интерфейса выводит на экран.

4.1.4 Измерение времени работы АЦП

Как уже было сказано выше, в зависимости от различных значений размера буфера и частоты дискретизации у АЦП уходит различное время на перезапуск и сбор данных. Эти вычисления производились при помощи стандартной функции-секундомера из стандартной библиотеки C#.

4.1.5 Паралельные вычисления в программе

В программе используются паралельные вычисления. Присутствуют два потока. Один поток работает с АЦП: принимает данные и записывает во временный буфер. Другой поток - отрисовывает данные из буфера на графике а также обрабатывает команды пользователя.

Паралельные вычисления используются из-за необходимости непрерывно получать данные с АЦП, чтобы избежать потери данных. Также, если не использовать паралельные вычисления, то приложение может не реагировать на команды пользователя из за того что процессор будет занят работой с АЦП.

4.2 Програмное обеспечение для Arduino Due

Програмное обеспечение для микроконтроллера Arduino Due состоит из программы, запускаемой на Arduino и программы, запускаемой на компьютере. Возможности программного обеспечения:

1. Прием сигнала от АЦП Arduino через Native-USB порт
2. Визуализация сигнала в реальном времени
3. Запись на диск в бинарный файл
4. Отправка записанного сигнала на CUDA-сервер для вычисления быстрого преобразования Фурье.
5. Возможность посчитать быстрое преобразование Фурье локально

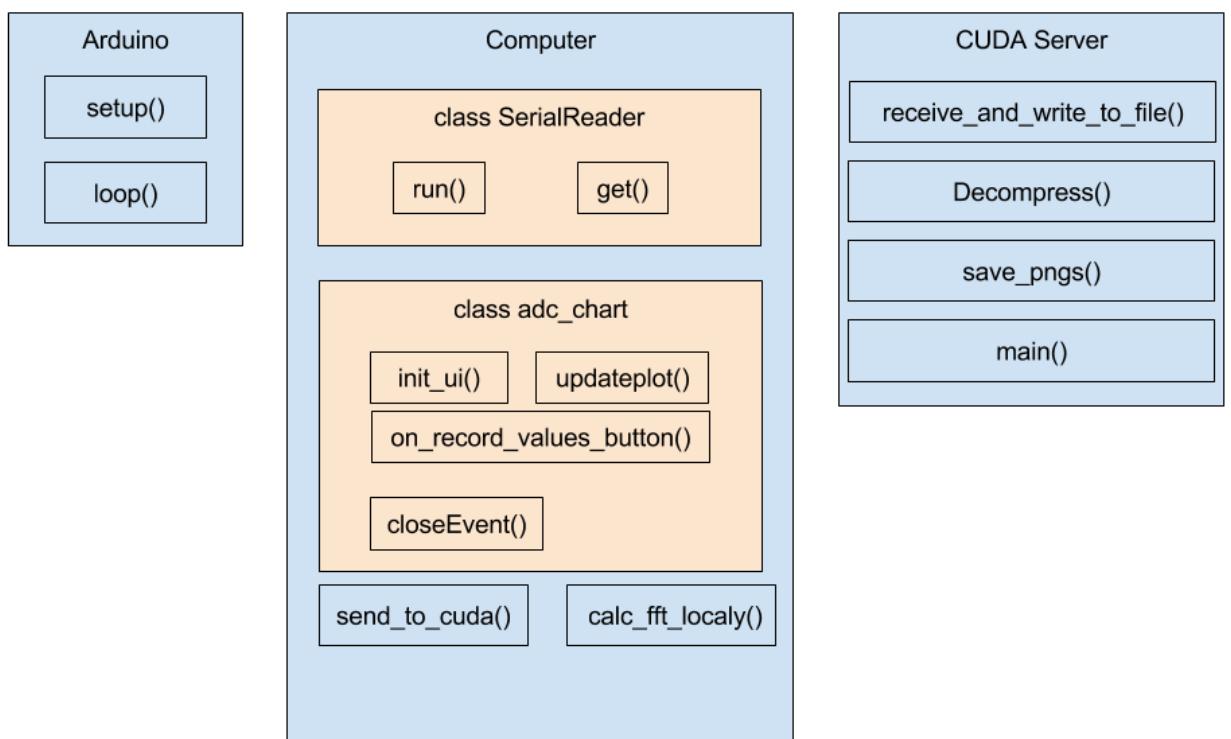


Рисунок 23 – Диаграмма классов ПО для Arduino Due

4.2.1 Программа для запуска на Arduino

Для того, чтобы ардуино работало в нужном режиме, на него нужно записать следующую программу:

```

#define HID_ENABLED
volatile int bufn,obufn;
uint16_t buf[4][256]; // 4 buffers of 256 readings

```

```

void ADC_Handler(){      // move DMA pointers to next buffer
    int f=ADC->ADC_ISR;
    if (f&(1<<27)){
        bufn=(bufn+1)&3;
        ADC->ADC_RNPR=(uint32_t)buf[bufn];
        ADC->ADC_RNCR=256;
    }
}

void setup(){
    SerialUSB.begin(0);
    while(!SerialUSB);
    pmc_enable_periph_clk(ID_ADC);
    adc_init(ADC, SystemCoreClock, ADC_FREQ_MAX, ADC_STARTUP_FAST);
    ADC->ADC_MR |=0x80; // free running

    ADC->ADC_CHER=0x80;

    NVIC_EnableIRQ(ADC_IRQn);
    ADC->ADC_IDR=~(1<<27);
    ADC->ADC_IER=1<<27;
    ADC->ADC_RPR=(uint32_t)buf[0]; // DMA buffer
    ADC->ADC_RCR=256;
    ADC->ADC_RNPR=(uint32_t)buf[1]; // next DMA buffer
    ADC->ADC_RNCR=256;
    bufn=obufn=1;
    ADC->ADC_PTCR=1;
    ADC->ADC_CR=2;
}

void loop(){
    while(obufn==bufn); // wait for buffer to be full
    SerialUSB.write((uint8_t *)buf[obufn],512); // send it - 512
        bytes
= 256 uint16_t

```

```
    obufn=(obufn+1)&3;  
}  
% \end{lstinlineatim}
```

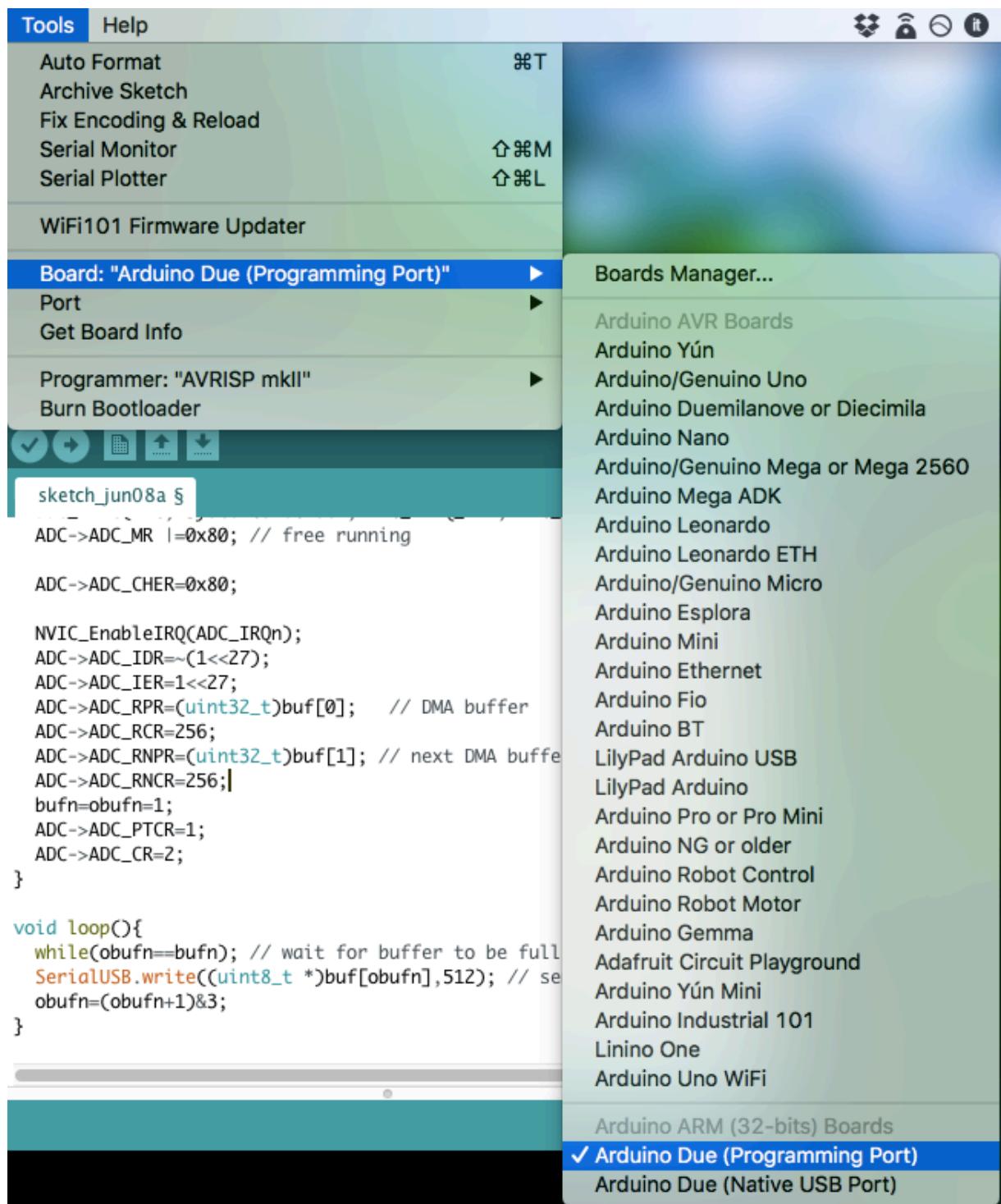
Данный код запускает АЦП в так называемом Free Running Mode. Это самый быстрый режим работы данного АЦП. Его особенность заключается в том, что нет необходимости вручную собирать данные через определенное количество времени. Программа пользователя запускает только первый сбор данных, а дальше АЦП сам автоматически начинает преобразование сразу как только закончилось предыдущее. Таким образом не возникает простоев АЦП и не теряются данные.

Также здесь используется прямой доступ к памяти микроконтроллера AT91SAM3X8E. (Direct Memory Access / DMA). Этот режим позволяет обращаться напрямую к памяти микроконтроллера, таким образом не отнимая процессорное время самого микроконтроллера. Процессор не прерывается на обработку запроса к памяти и постоянно занят сбором данных с АЦП. Это позволяет приблизиться к максимальной частоте дискретизации.

Чтобы установить данное ПО на Arduino Due, необходимо скачать Arduino IDE с официального сайта Arduino[24]

Устройство нужно подключить к компьютеру через USB порт. Данное устройство имеет 2 USB порта: программируемый и native. Чтобы записать программу нужно подключаться через программируемый.

В Arduino IDE в меню Tools - Board нужно выбрать "Arduino Due (Programming Port)" как показано на скриншоте ниже.



В главное окно программы нужно вставить код, приведенный выше и нажать кнопку Upload. После этого программа будет успешно записана на микроконтроллер.

4.2.2 Программа для запуска на компьютере

Данное ПО написано на языке Python и может быть запущено на любой операционной системе. Используются библиотеки PyQtgraph, PyQt, numpy, matplotlib и pyserial. Для запуска программы необходимо установить эти библиотеки:

1. Скачать и установить последнюю версию python с официального сайта[18]
2. открыть командную строку и установить необходимые библиотеки с помощью команд:
3. sudo pip install pyqtgraph
4. pip install pyqt5
5. pip install numpy
6. pip install matplotlib
7. pip install pyserial
8. скачать саму программу для работы с ацп по ссылке: [?]

Далее в командной строке необходимо перейти в директорию, где находится скачанная программа app.py. Теперь нужно подключить устройство к компьютеру через Native-USB порт и запустить программу с помощью команды:

```
python app.py
```

При запуске программы можно указывать параметры децимации (downsampling). Децимация позволяет отрисовывать на графике не все значения с АЦП, а в N раз меньше. Это позволяет увеличить скорость отрисовки (Более высокое значение FPS (frames per second, количество кадров в секунду), но при этом

теряется точность сигнала, можно не увидеть очень высокие частоты на графике. Значение децимации задается с помощью аргумента `-d`. Допустим, команда

```
python app.py -d 100
```

запустит программу со значением параметра децимации=100.

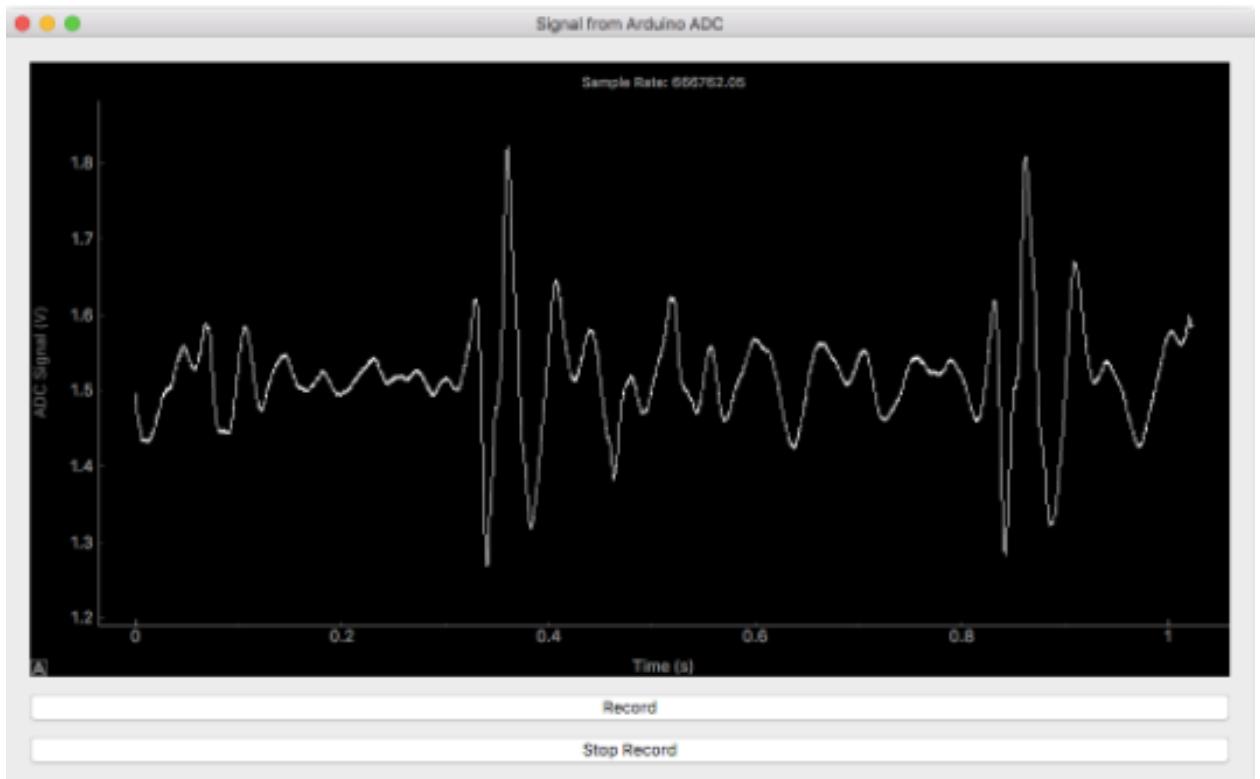


Рисунок 24 – Скриншот работы программы, прослушивается сердце. Сигнал отображается с децимацией=100. Частота дискретизации 666кГц.

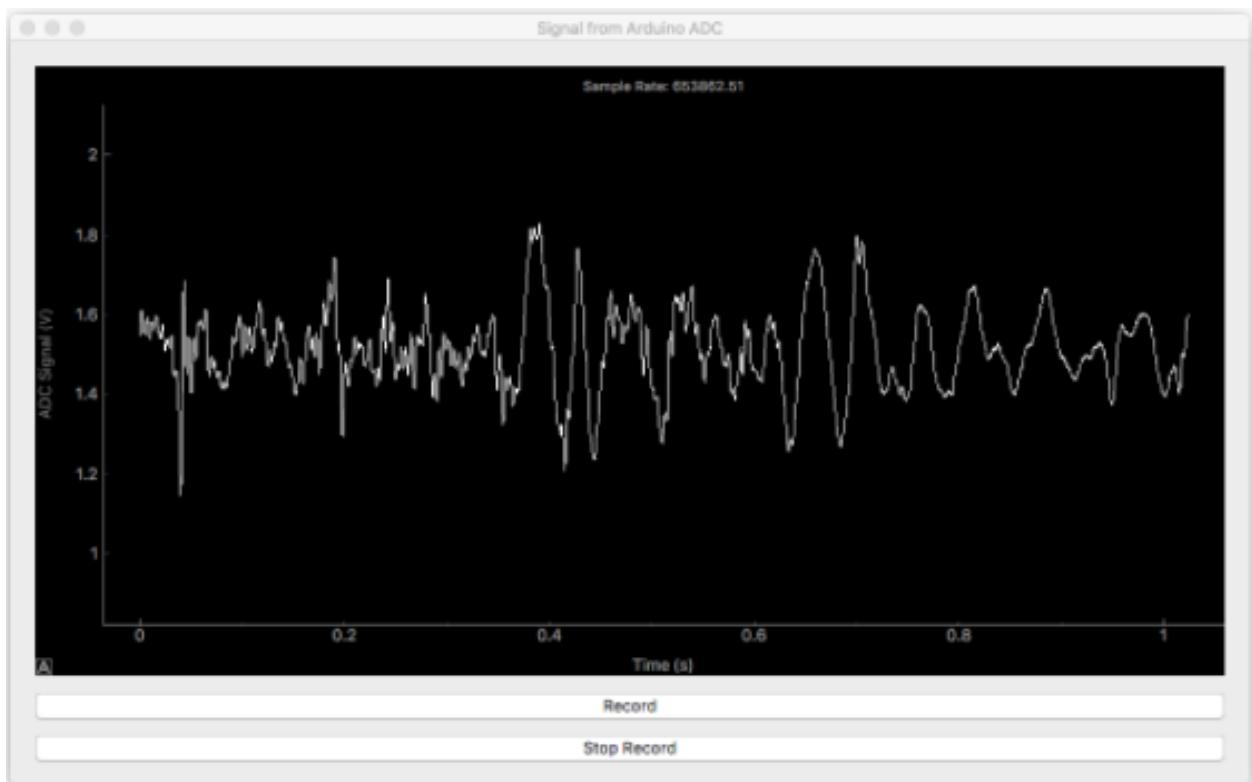


Рисунок 25 – Скриншот работы программы, прослушиваются легкие. Сигнал отображается с децимацией=100. Частота дискретизации 666Гц.

Для записи данных в файл и отправки их на сервер нужно выбрать сколько значений записывать и нажать кнопку Record Values.

Основные компоненты программы:

- Класс `SerialReader` - принимает данные с ацп через serial порт. В этом классе есть 2 основных метода: метод `run()`, который в бесконечном цикле собирает данные с АЦП и, если нажата кнопка записи, записывает данные в файл. Метод `get(num, downsample=1)` - возвращает num последних значений с ацп. Этот метод использует второй класс `adc_chart`
- Класс `adc_chart` - работает с интерфейсом пользователя. Основные методы:
 - `init_ui` - инициализация пользовательского интерфейса
 - `updateplot` - метод получает последние значения с АЦП и отрисовывает их на графике

- `on_record_values_button` - обработка кнопки записи сигнала.
- `closeEvent` - обработка закрытия окна программы. Завершает все запущенные процессы.
- `send_to_cuda` - метод вызываемый, когда завершена запись данных. Отправляет данные на CUDA-сервер.
- `calc_fft_localy` - метод позволяющий посчитать быстрое преобразование Фурье на локальном компьютере, без отправки на сервер. Для вычисления и отрисовки используются библиотеки Numpy и Matplotlib.

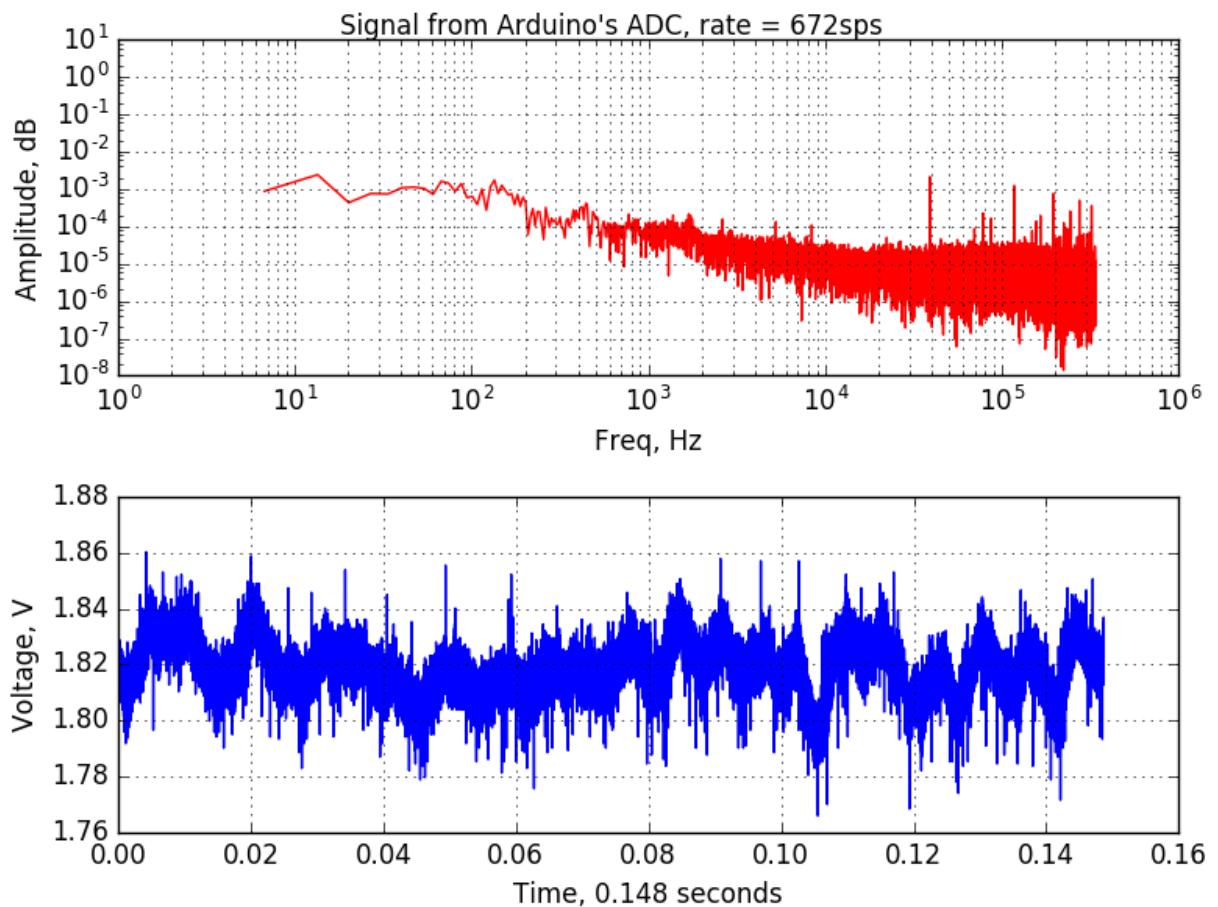


Рисунок 26 – Изображения, получающиеся при локальном преобразовании Фурье

4.3 Програмное обеспечение для CUDA-сервера

Програмное обеспечение для сервера написано на языке C# под операционную систему Windows. ПО позволяет считать быстрое преобразование Фурье на видеокартах NVidia с помощью технологии CUDA (библиотека для работы с высокопроизводительными математическими вычислениями на видеокартах NVidia).

Быстрое преобразование Фурье (FFT) — алгоритм, используемый в цифровой обработки сигнала для получения амплитудно частотной характеристики сигнала из амплитудно временной характеристики.

Пусть дан массив значений функции X . Тогда значения элементов выходного массива после FFT будут равны:

$$F(x) = f$$
$$f[k] = \sum_{j=0}^{N-1} x[j] \left(e^{-2\pi ik/N}\right)^j$$
$$0 \leq k < N$$

Возможности данного ПО:

1. Прием сжатого сигнала от клиента
2. Декомпрессия сигнала и вычисление быстрого преобразования Фурье (FFT)
3. Построение графиков сигнала и его спектра и сохранение в .png файл.
4. Отправка полученных изображений обратно клиенту

Для запуска, необходимо поместить в папку с программой .dll - библиотеку CUFT.dll для работы с видеокартой. Библиотеку можно скачать по ссылке [23]

Основные методы главного класса программы:

1. `receive_and_write_to_file` - принимает сжатый сигнал от клиента и записывает его в файл.
2. `Decompress` - распаковка .gz архива с сигналом. Метод возвращает массив с сигналом.
3. `Main` - основной метод программы, в котором считается fft и вызываются остальные методы.
4. `save_pngs` - Построение графиков сигнала и его спектра и сохранение в .png файл(ы).

В методе `Main` программа в бесконечном цикле пытается принять данные от клиента. Если клиент подключен, то сигнал принимается. Если сигнал большой, то он делится на блоки. Для каждого блока считается FFT и создается график спектра сигнала, который записывается в отдельный файл-изображение. Когда все блоки обработаны, сервер продолжает ждать приема нового сигнала.

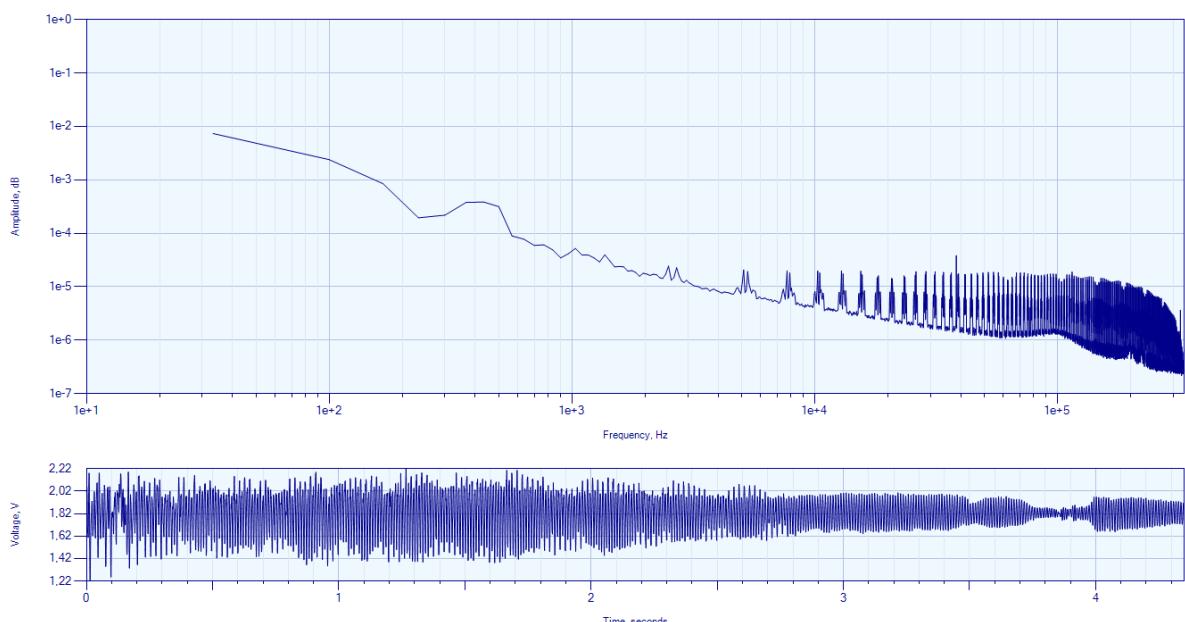


Рисунок 27 – Изображения, получающиеся при преобразовании Фурье на удаленном CUDA-сервере

Выводы

1. Создан рабочий прототип устройства, состоящий из микрофона, усилителя, АЦП, компьютера и удаленного сервера.
2. Обеспечен диапазон принятия сигналов во всем слышимом человеком диапазоне ($20\text{Гц} - 20\text{kГц}$)
3. Обеспечен диапазон принятия сигналов в диапазоне от 20kГц до 100kГц
4. Обеспечена одновременная работа в двух указанных выше диапазонах
5. Обеспечена обработка полученного сигнала с использованием спектральных методов оценивания
6. Обеспечена возможность локальной обработки сигнала
7. Обеспечена возможность обработки сигнала на удаленном сервере в виде быстрого преобразования фурье

Заключение

В результате данной работы был создан рабочий прототип устройства, позволяющего анализировать в реальном времени звуковой сигнал высокого качества. Также было написано программное обеспечение к этому прибору. Программное обеспечение позволяет записывать и анализировать звуковые сигналы в реальном времени. Есть возможность рассмотреть различные характеристики сигнала, такие как спектр Фурье, скользящее среднее. Также есть возможность записывать аудиосигнал на жесткий диск для его последующей обработки. Сигнал записывается с частотой дискретизации 660кГц. Данная частота позволяет анализировать ультразвуковой сигнал до 330кГц.

Данное устройство предназначено для получения дополнительной информации из звука поступающего от сердца, легких и других внутренних органов, которую нельзя услышать на обычном стетоскопе. Данная дополнительная информация может быть использована врачом для осуществления более качественной медицинской диагностики.

Сфера применения не ограничена медициной, устройство позволяет анализировать любые типы ультразвуковых сигналов.

Развитие проекта можно продолжить в направлении улучшения качества звука. Для этого нужно использовать более качественный микрофон, который будет обладать более высокой границей воспринемаемых частот и более качественный усилитель. Также нужно произвести опрос врачей о том, какие именно характеристики звука со стетоскопа важны.

Улучшений в программном обеспечении можно достичнуть путем оптимизации алгоритмов распаралеливания на нескольких ядрах процессора или на видеокарте. Также, на основе информации от докторов, можно сделать систему распознавания различных заболеваний легких и сердца.

Разработка данного проекта велась с помощью системы контроля версий

git. Исходный код программного обеспечения, этапы создания и документация доступны по адресу: [31]

Список использованных источников

1 Стетоскоп 3M™ Littmann® Electronic Stethoscope Model 3200

http://www.littmann.com/3M/en_US/littmann-stethoscopes/products/~/3M-Littmann-Electronic-Stethoscope-Model-3200?N=5932256+8711017+3293188392&rt=rud

2 Стетоскоп CMS-VESD SPO2 PR

http://www.contecmed.com/index.php?page=shop.product_details&flypage=flypage.tpl&product_id=26&category_id=13&option=com_virtuemart&Itemid=600

3 Стетоскоп CMS-VE

http://www.contecmed.com/index.php?page=shop.product_details&flypage=flypage.tpl&product_id=21&category_id=13&option=com_virtuemart&Itemid=600

4 Стетоскоп CMS-M

http://www.contecmed.com/index.php?page=shop.product_details&flypage=flypage.tpl&product_id=27&category_id=13&option=com_virtuemart&Itemid=600

5 Стетоскоп Cardionics E-scope II

<http://www.cardionics.com/product/clinical-systems/hearing-impaired-e-scope>

6 Стетоскоп Thinklabs One

<http://www.thinklabs.com/one-digital-stethoscope>

7 Стетоскоп Eko Core

<https://ekodevices.com/core/>

8 Стетоскоп Eko Duo

<https://ekodevices.com/duo/>

9 Аналогово цифровой преобразователь LA-n10-12USB

<http://www.rudshel.ru/show.php?dev=14>

10 Драйверы и программное обеспечение для устройств ЗАО "Руднев-Шиляев"

<http://rudshel.ru/software.html>

11 Документация по программированию устройств ЗАО "Руднев-Шиляев"

http://www.rudshel.ru/soft/SDK2/Doc/CPP_USER_RU/html/index.html

12 Руководство пользователя LA-n10-12USB

[http://www.rudshel.ru/pdf/LA-n10-12USB\(y\).rar](http://www.rudshel.ru/pdf/LA-n10-12USB(y).rar)

13 Внешний USB АЦП/ЦАП E14-140-M

<http://www.lcard.ru/products/external/e-140m>

14 АЦП Arduino Due

<https://www.arduino.cc/en/Main/ArduinoBoardDue>

15 Схема усилителя для микрофона

<http://full-chip.net/analogovaya-elektronika/70-usilitel-dlya-elektretnogo-mikrofona-s-nizkim-urovнем-shuma-shema.html>

16 Руководство пользователя и технические характеристики операционного усилителя MCP6022

<https://lib.chipdip.ru/291/DOC000291231.pdf>

17 Проект усилителя для программы Sprint Layout

<https://github.com/tandav/ultrasonic-stethoscope/blob/master/Circuit/Amplifier.lay6>

18 Язык программирования Python

<https://www.python.org/downloads/>

19 Программа для запуска на компьютере для АЦП Е14-140-М

<https://github.com/tandav/ultrasonic-stethoscope/tree/master/Lcard%20ADC>

20 Программа для запуска на компьютере для АЦП ЛА-н10-12USB

<https://github.com/tandav/ultrasonic-stethoscope/tree/master/Rudshel/forms-timer-label>

21 Программа для запуска на компьютере для АЦП Arduino Due

<https://github.com/tandav/ultrasonic-stethoscope/blob/master/arduino/app.py>

22 Технология Nvidia CUDA

<http://www.geforce.com/hardware/technology/cuda>

23 Библиотека для работы с Nvidia CUDA для языка C#

<https://github.com/tandav/ultrasonic-stethoscope/blob/master/Server/CUFT-lib/CUFT.dll>

24 Программа Arduino IDE

<https://www.arduino.cc/en/Main/Software>

25 Исходный код, документация и этапы создания проекта ультразвукового стетоскопа

<https://github.com/tandav/ultrasonic-stethoscope>

26 Горшков, Ю. Г. Обработка речевых и акустических биомедицинских сигналов на основе вейвлетов // Издательство Радиотехника, 2017.

27 Дьяконов, В. П. Вейвлеты. От теории к практике. М.: СОЛОН-Р. 2002.

- 28 Чистович Л. А. Физиология речи. Восприятие речи человеком. АН СССР.
Л.: Наука. 1976.
- 29 Дворянкин С.В. Взаимосвязь цифры и графики, звука и изображения //
Открытые системы. 2000.
- 30 Блаттер К. Вейвлет-анализ. Основы теории. М.: Техносфера. 2006
- 31 Галышина Е.И. Речь под микроскопом // Компьютерра 1999