

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное агентство по образованию
«Санкт-Петербургский государственный университет телекоммуникаций им.
проф. М. А. Бонч-Бруевича (СПбГУТ)»

СПб ГУТ)))

Системы цифровой звукозаписи

ЛАБОРАТОРНАЯ РАБОТА № 4

Выполнили:

Балан К. А.

Валиахметов В. А.

Студенты группы:

РЦТ-22

Преподаватель:

Прасолов А. А.

Санкт-Петербург

2025

Цель работы

Научиться читать заголовки wav файлов

Полученные результаты

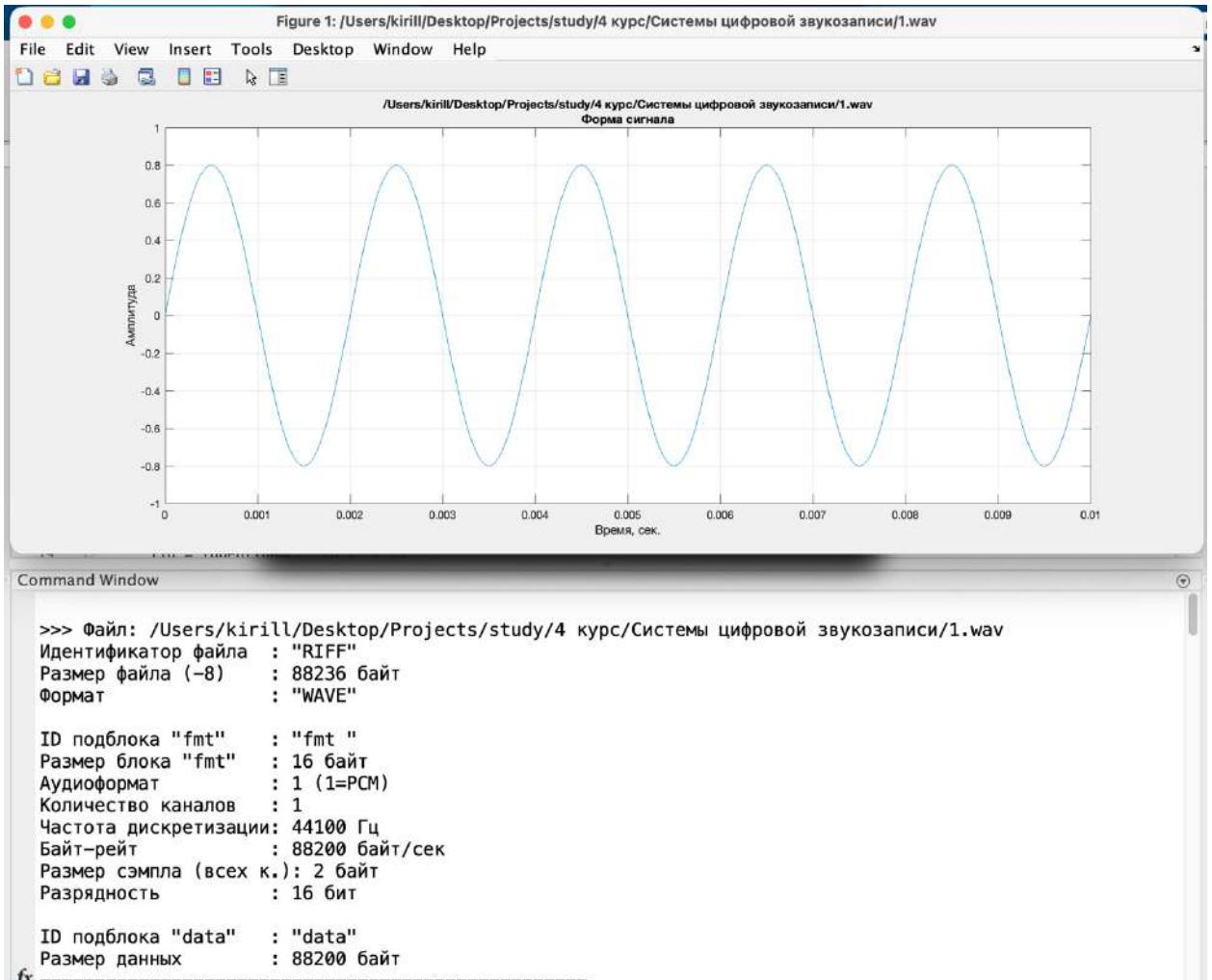


Рисунок 1 – Сигнал с 1 каналом, частотой дискретизации 44100 Гц и разрядностью 16 бит, сгенерированный в audacity и прочитанный в matlab

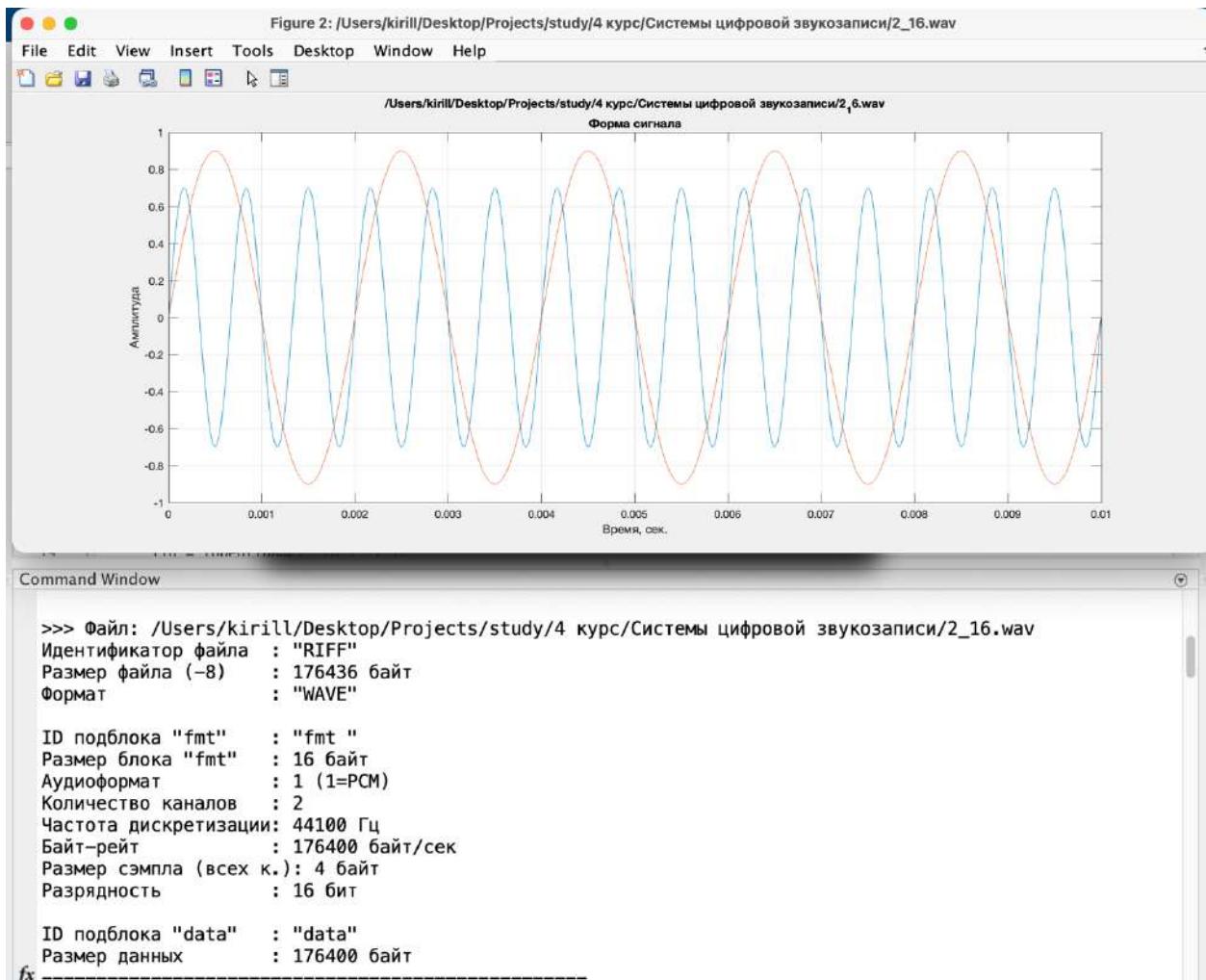


Рисунок 2 – Сигнал с 2 каналами, частотой дискретизации 44100 Гц и разрядностью 16 бит, сгенерированный в audacity и прочитанный в matlab

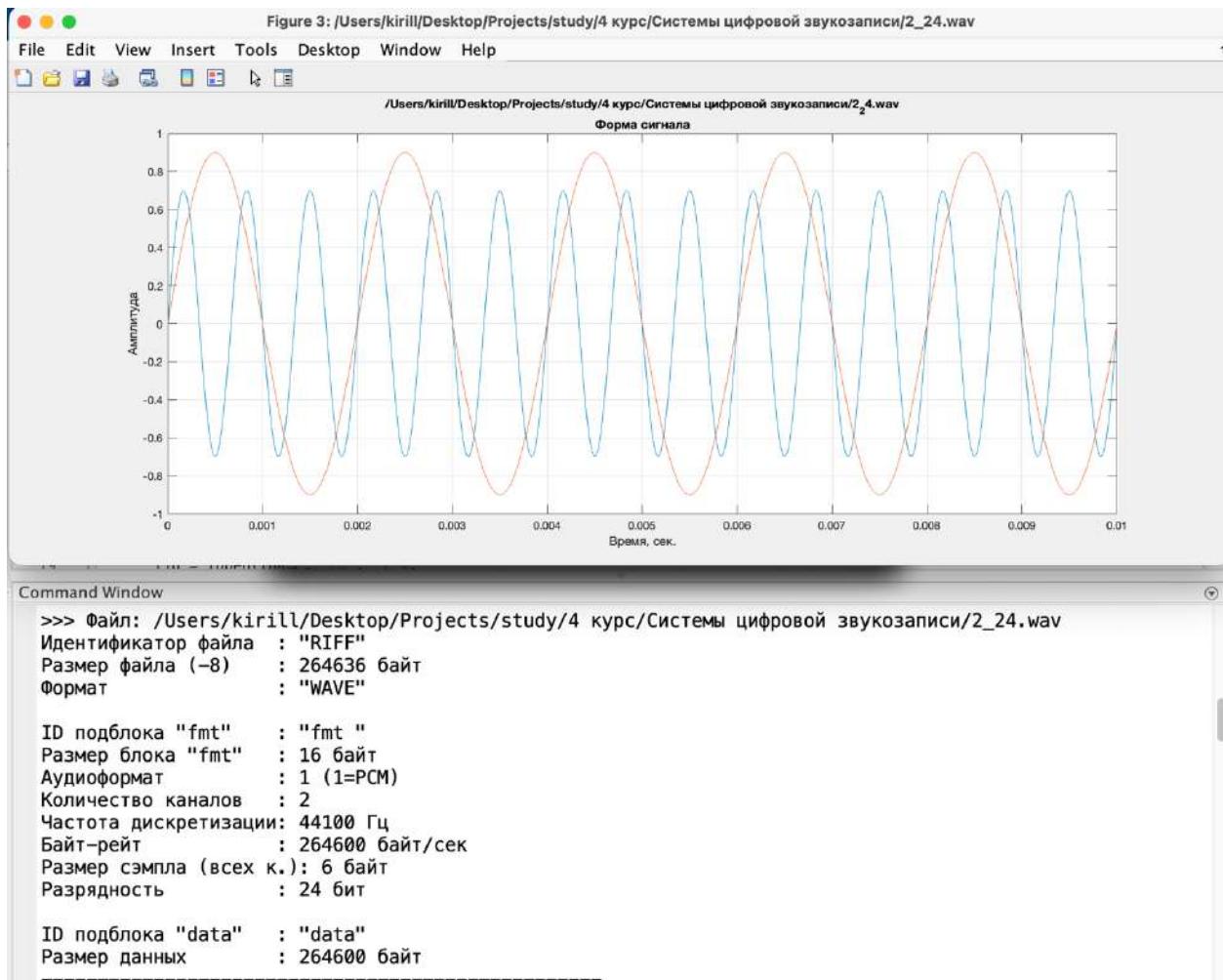


Рисунок 3 – Сигнал с 2 каналами, частотой дискретизации 44100 Гц и разрядностью 24 бит, сгенерированный в audacity и прочитанный в matlab

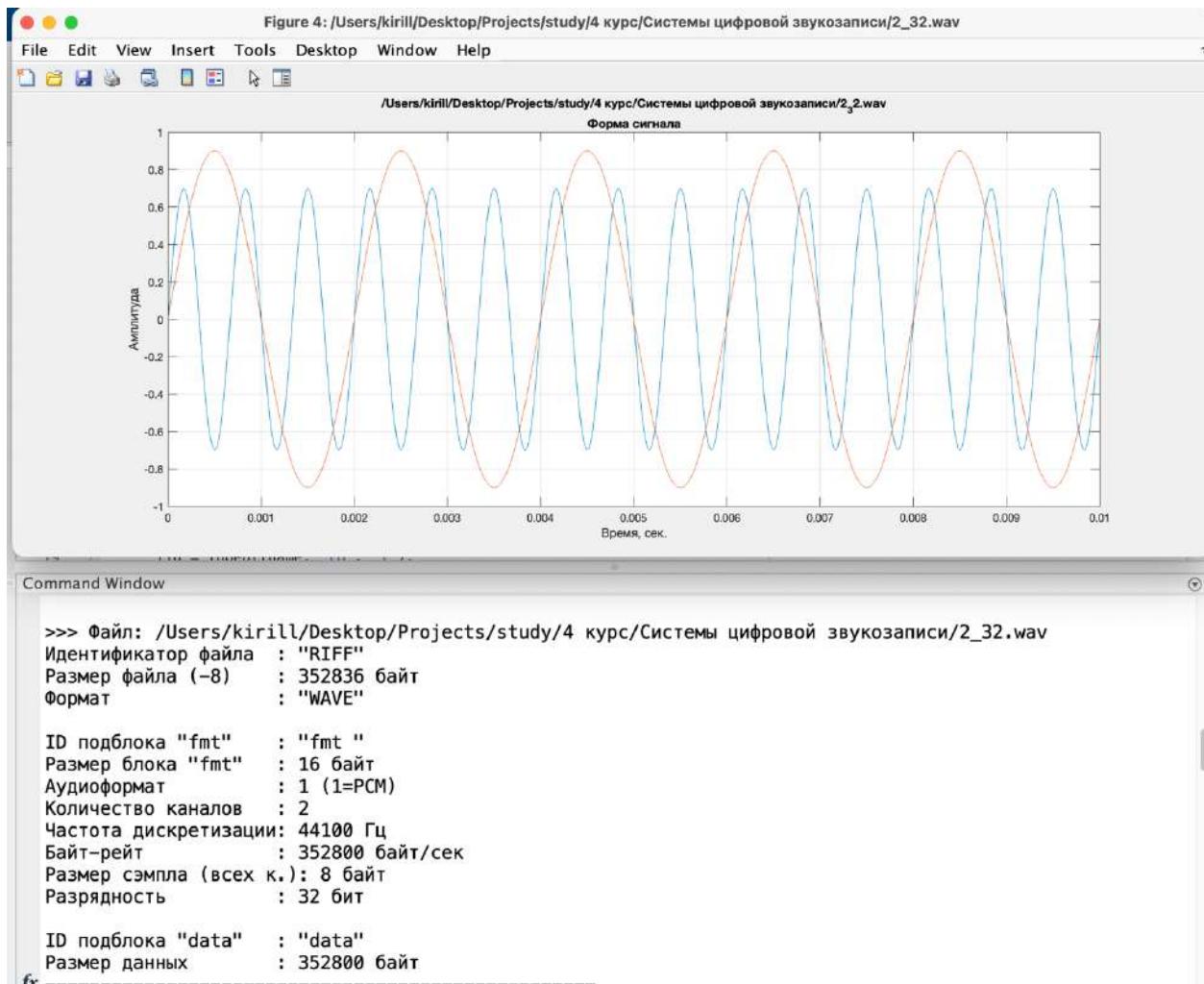


Рисунок 4 – Сигнал с 2 каналами, частотой дискретизации 44100 Гц и разрядностью 32 бит, сгенерированный в audacity и прочитанный в matlab

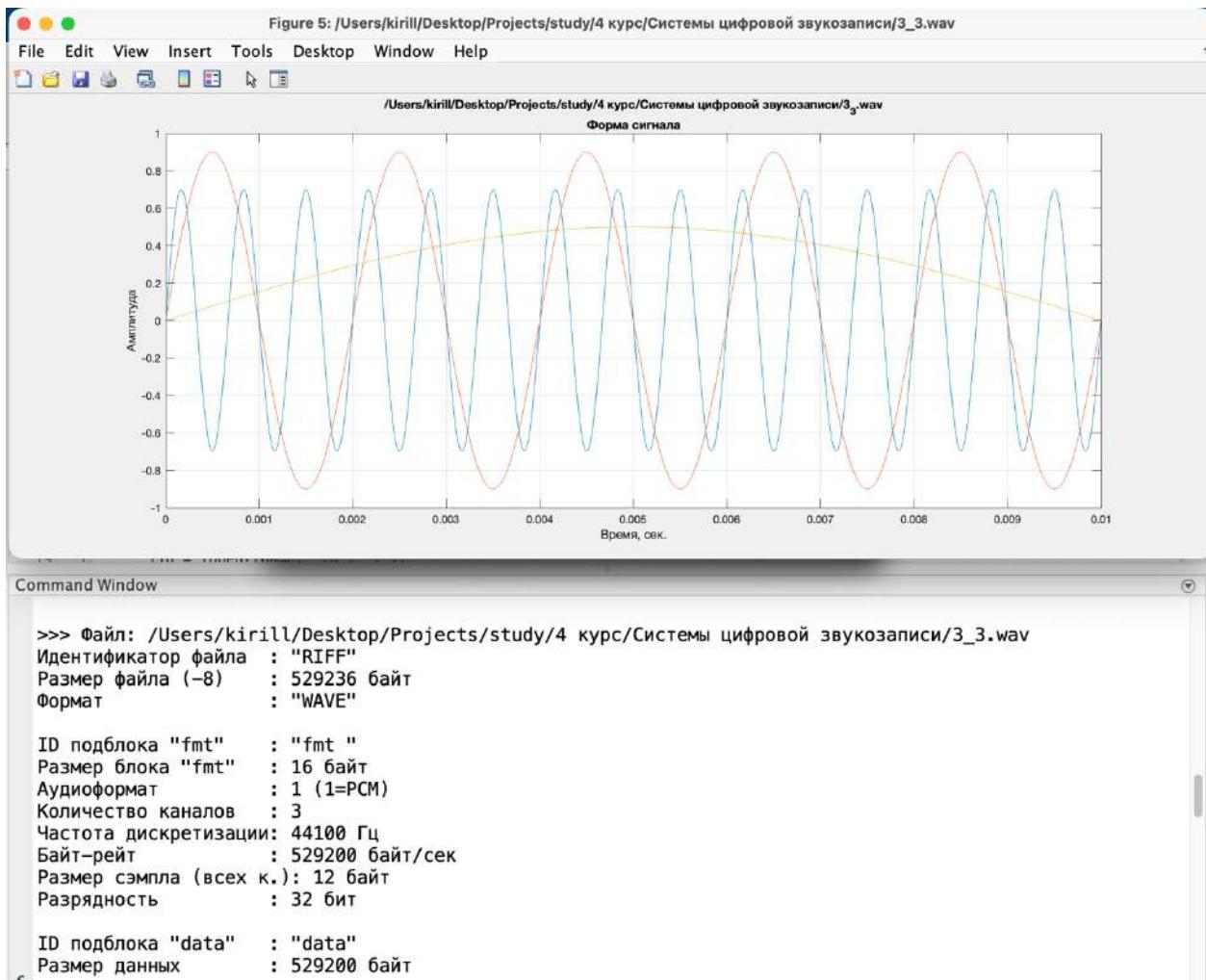


Рисунок 5 – Сигнал с 3 каналами, частотой дискретизации 44100 Гц и разрядностью 32 бит, сгенерированный в audacity и прочитанный в matlab

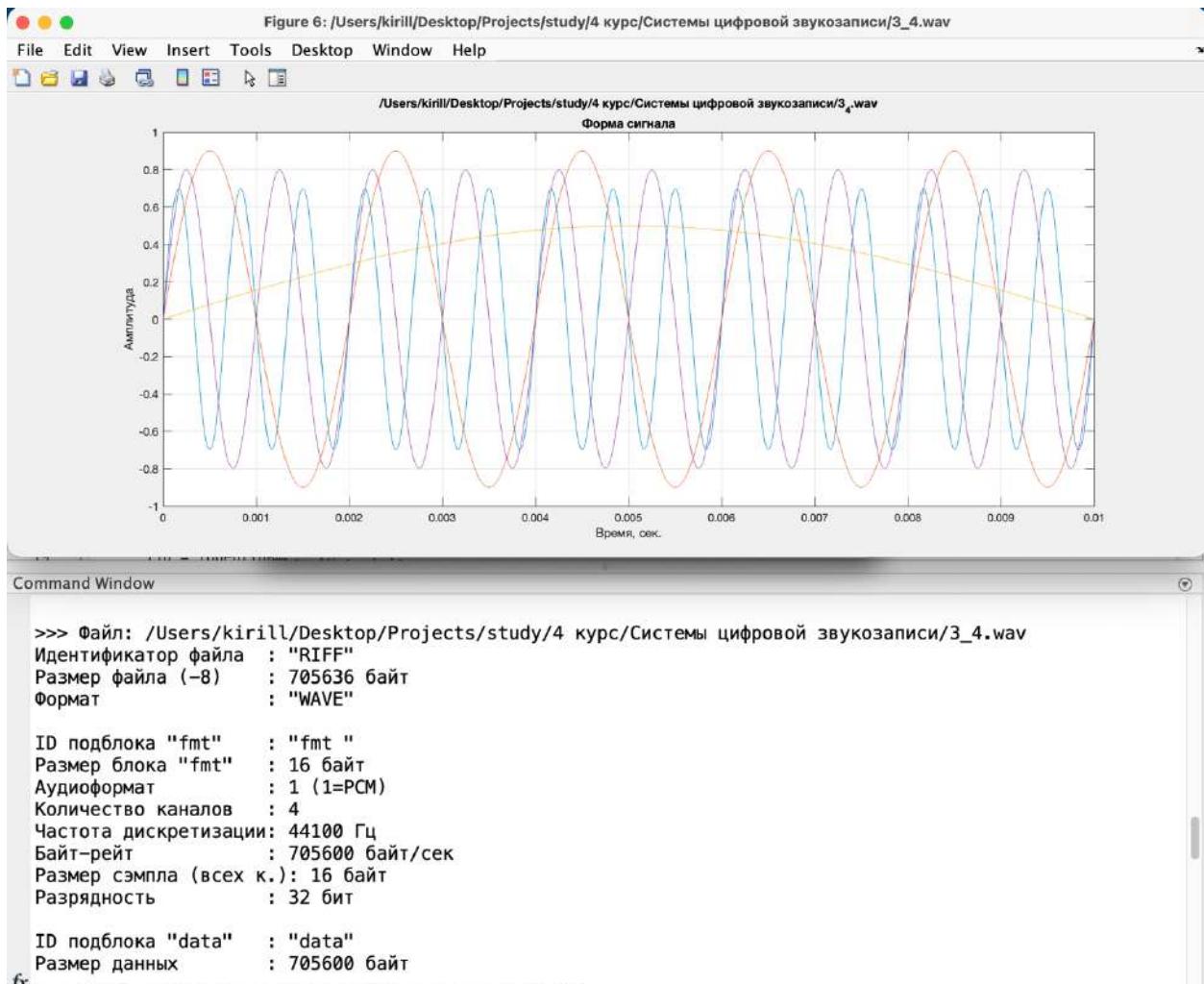


Рисунок 6 – Сигнал с 4 каналами, частотой дискретизации 44100 Гц и разрядностью 32 бит, сгенерированный в audacity и прочитанный в matlab

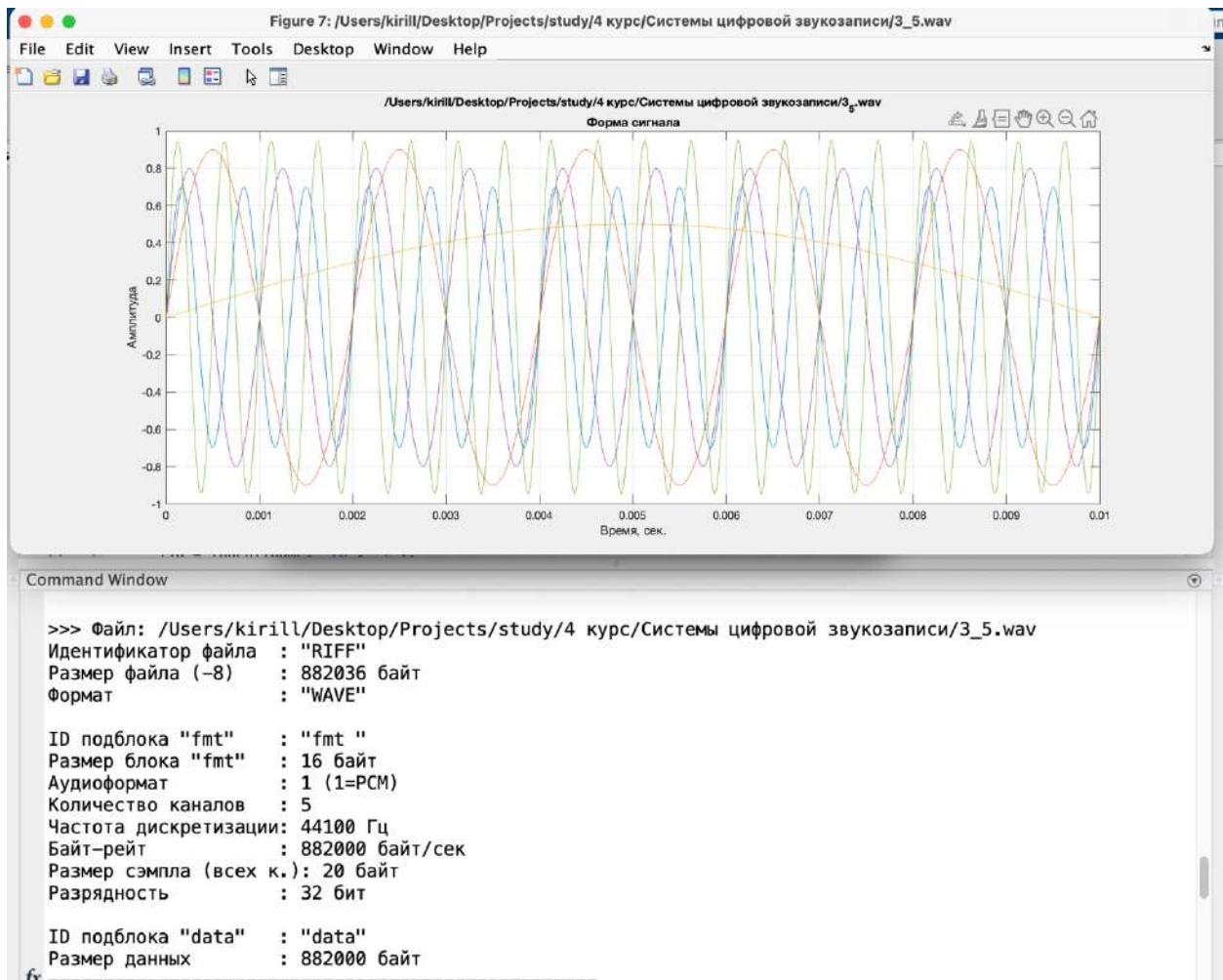


Рисунок 7 – Сигнал с 5 каналами, частотой дискретизации 44100 Гц и разрядностью 32 бит, сгенерированный в audacity и прочитанный в matlab

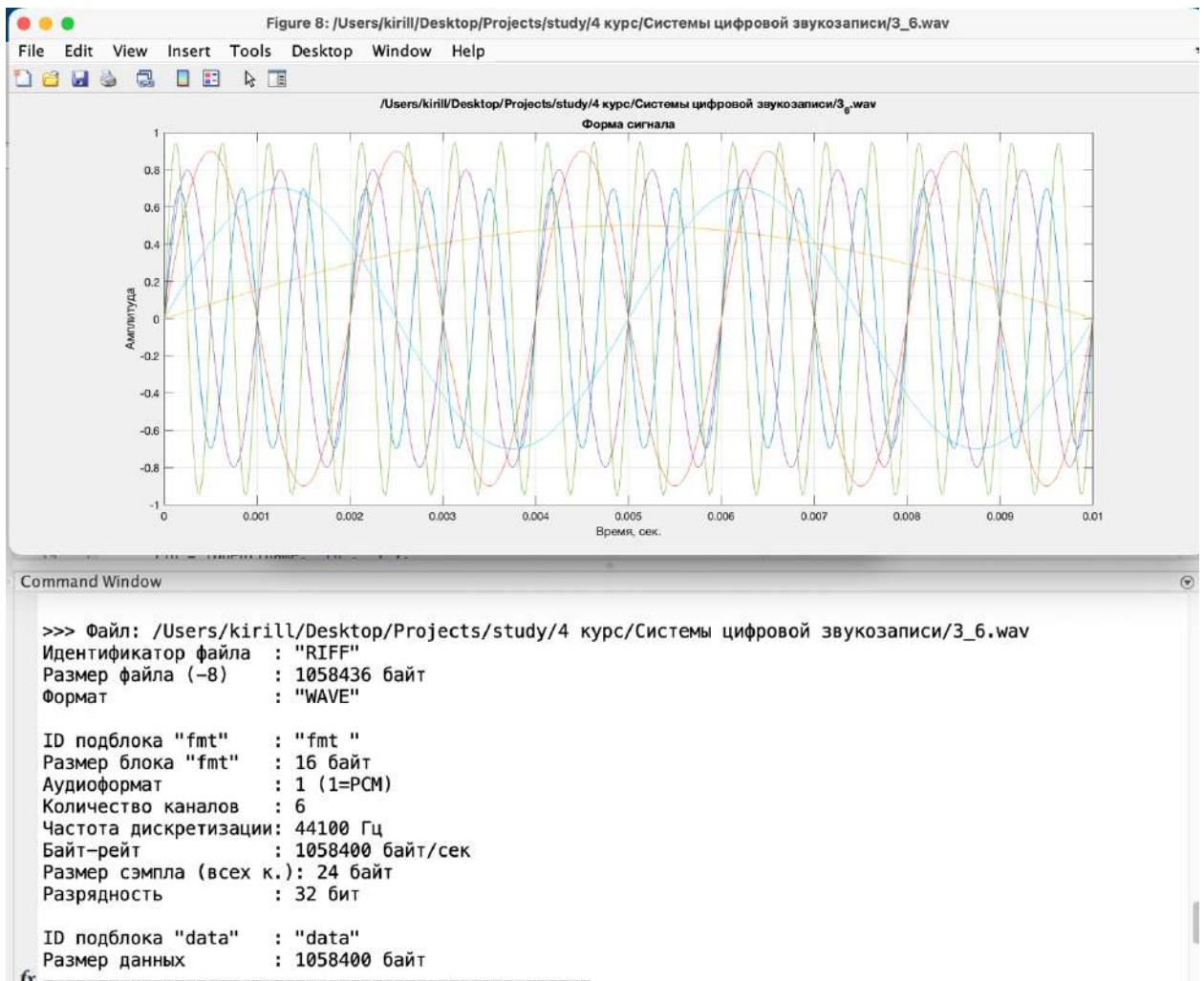


Рисунок 8 – Сигнал с 6 каналами, частотой дискретизации 44100 Гц и разрядностью 32 бит, сгенерированный в audacity и прочитанный в matlab

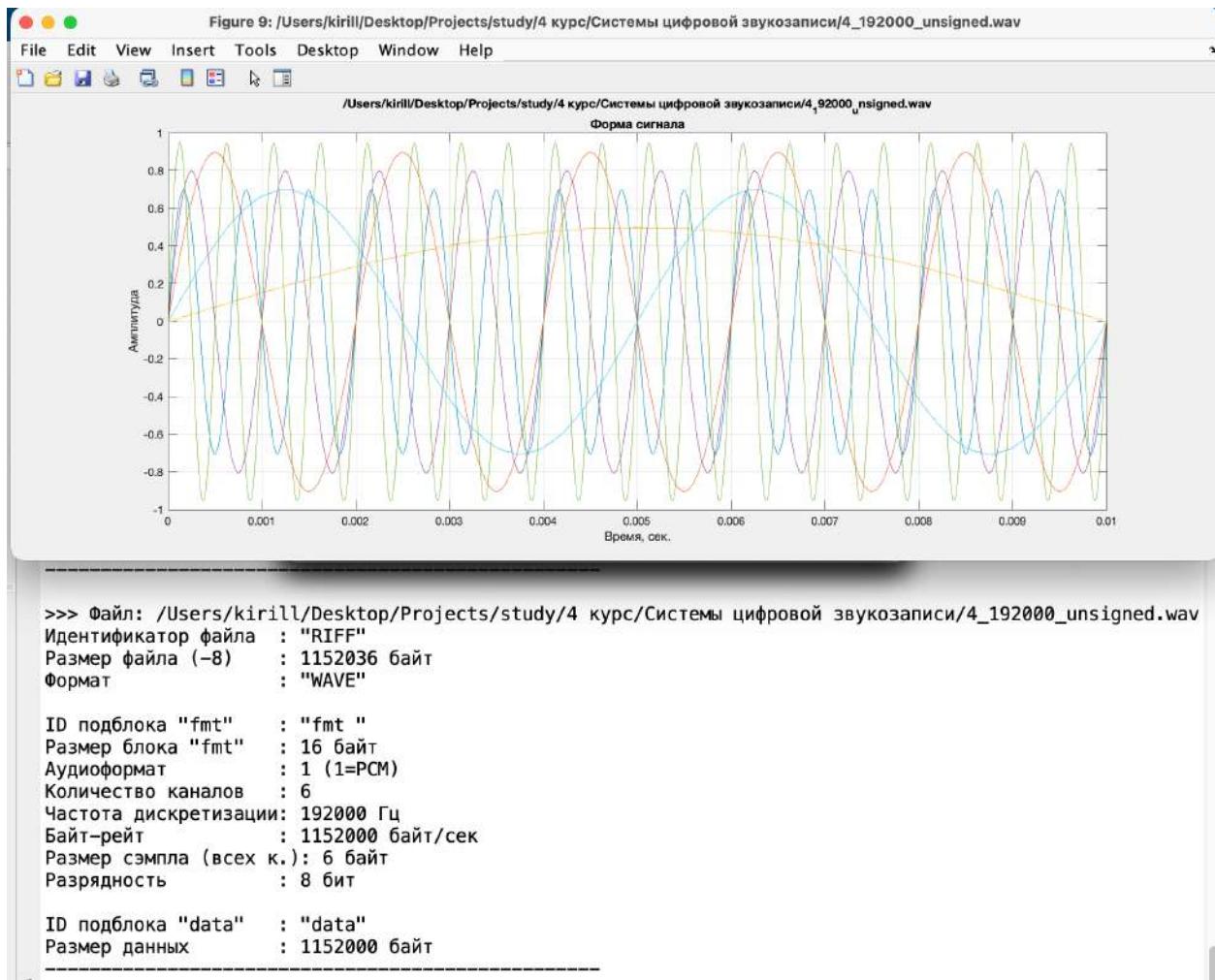


Рисунок 9 – Сигнал с 6 каналами, частотой дискретизации 192000 Гц и разрядностью 8 бит, сгенерированный в audacity и прочитанный в matlab

При чтении файла в audacity всегда ставилось значение разрядности 32 бит с плавающей точкой

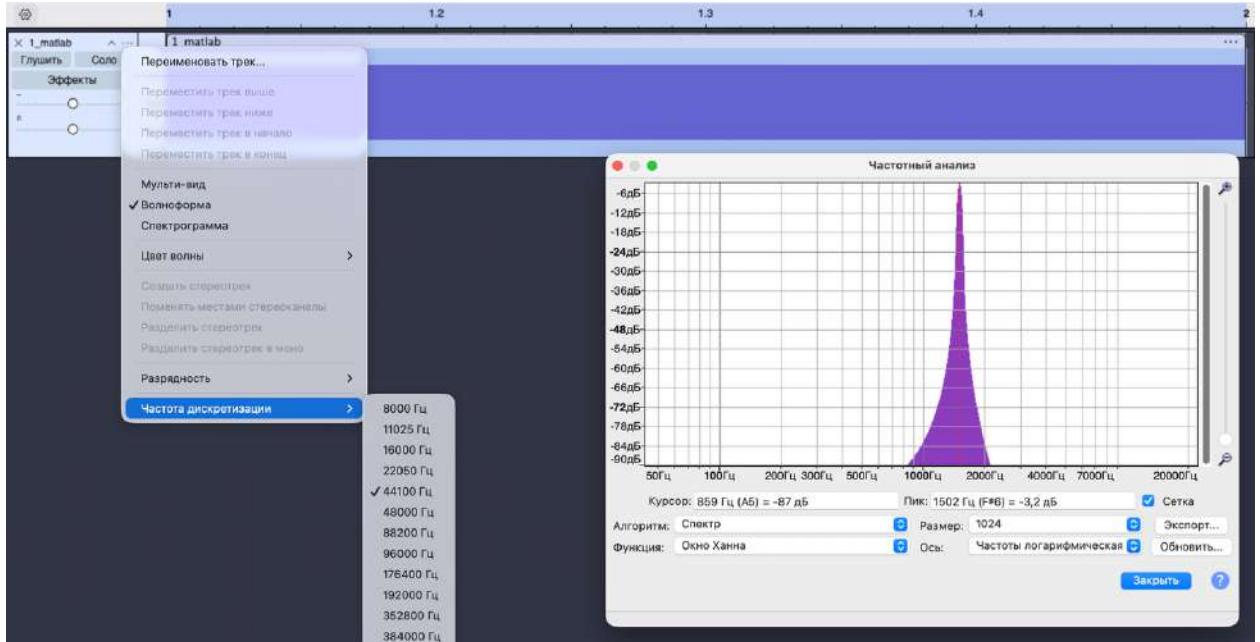


Рисунок 10 – Сигнал с 1 каналом, частотой дискретизации 44100 Гц и разрядностью 16 бит, сгенерированный в matlab и прочитанный в audacity

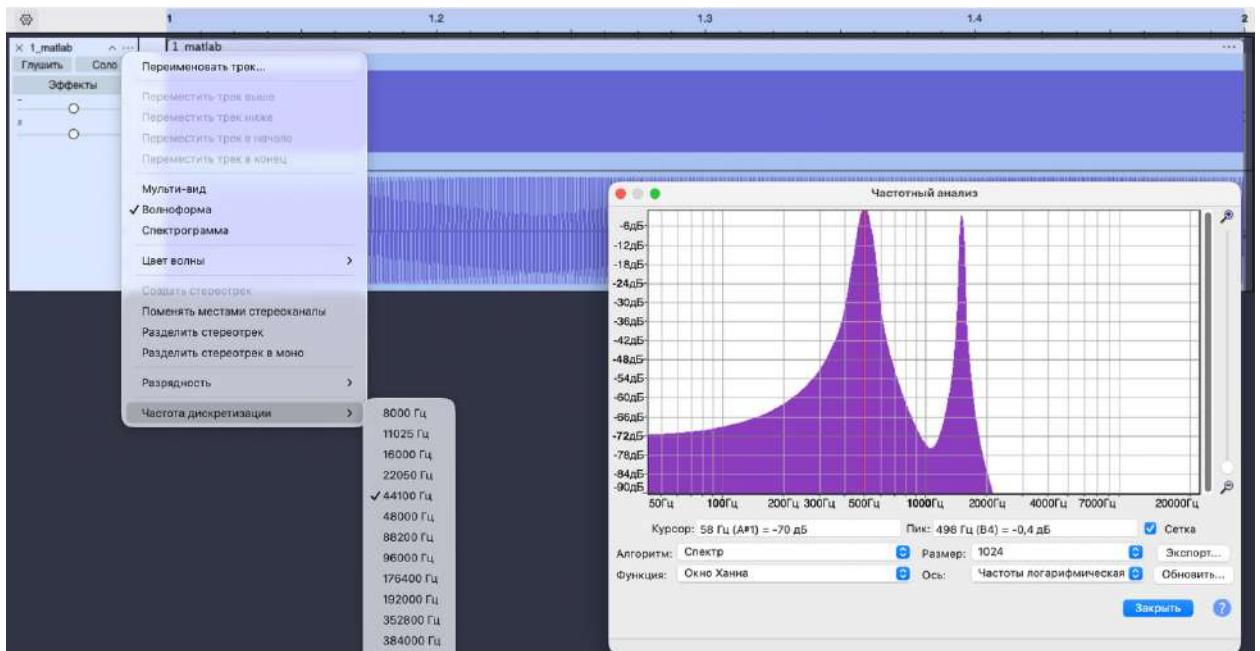


Рисунок 11 – Сигнал с 2 каналами, частотой дискретизации 44100 Гц и разрядностью 16 бит, сгенерированный в matlab и прочитанный в audacity

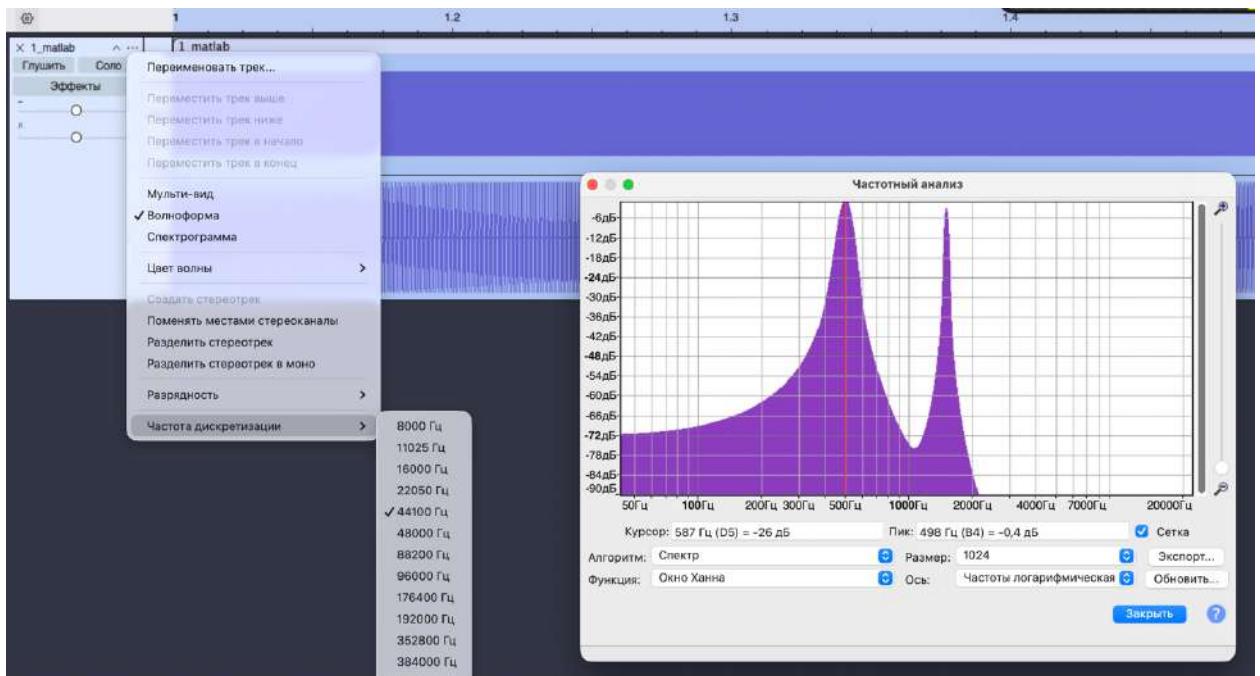


Рисунок 12 – Сигнал с 2 каналами, частотой дискретизации 44100 Гц и разрядностью 24 бит, сгенерированный в matlab и прочитанный в audacity

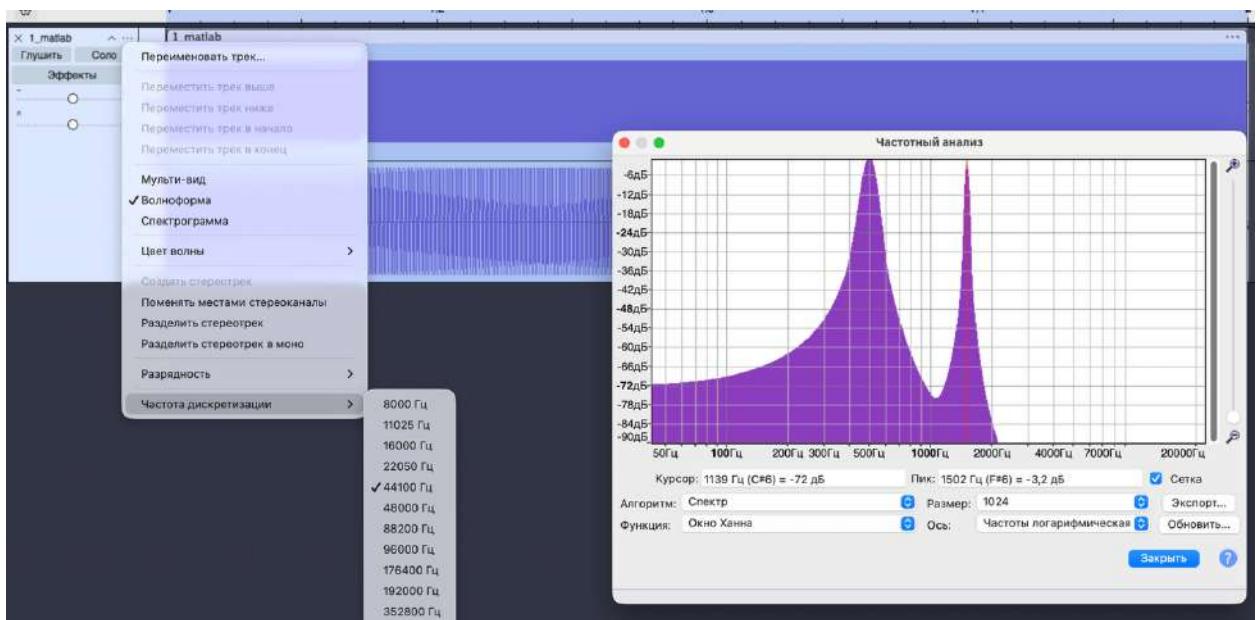


Рисунок 13 – Сигнал с 2 каналами, частотой дискретизации 44100 Гц и разрядностью 32 бит, сгенерированный в matlab и прочитанный в audacity

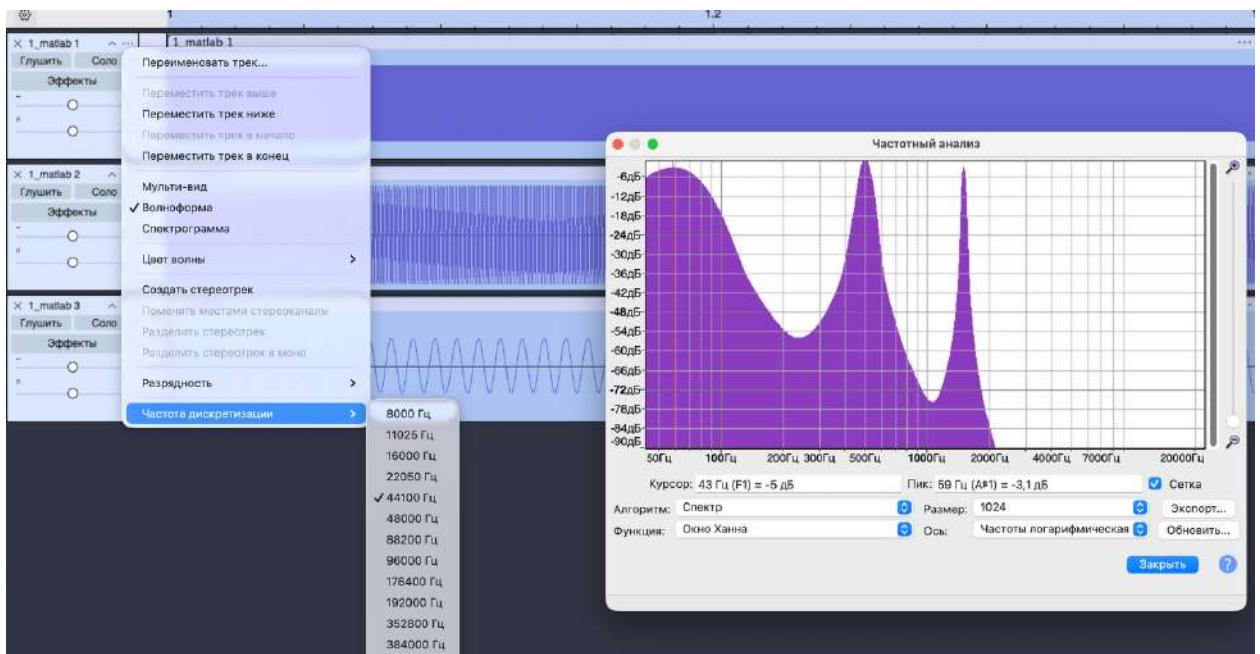


Рисунок 14 – Сигнал с 3 каналами, частотой дискретизации 44100 Гц и разрядностью 32 бит, сгенерированный в matlab и прочитанный в audacity

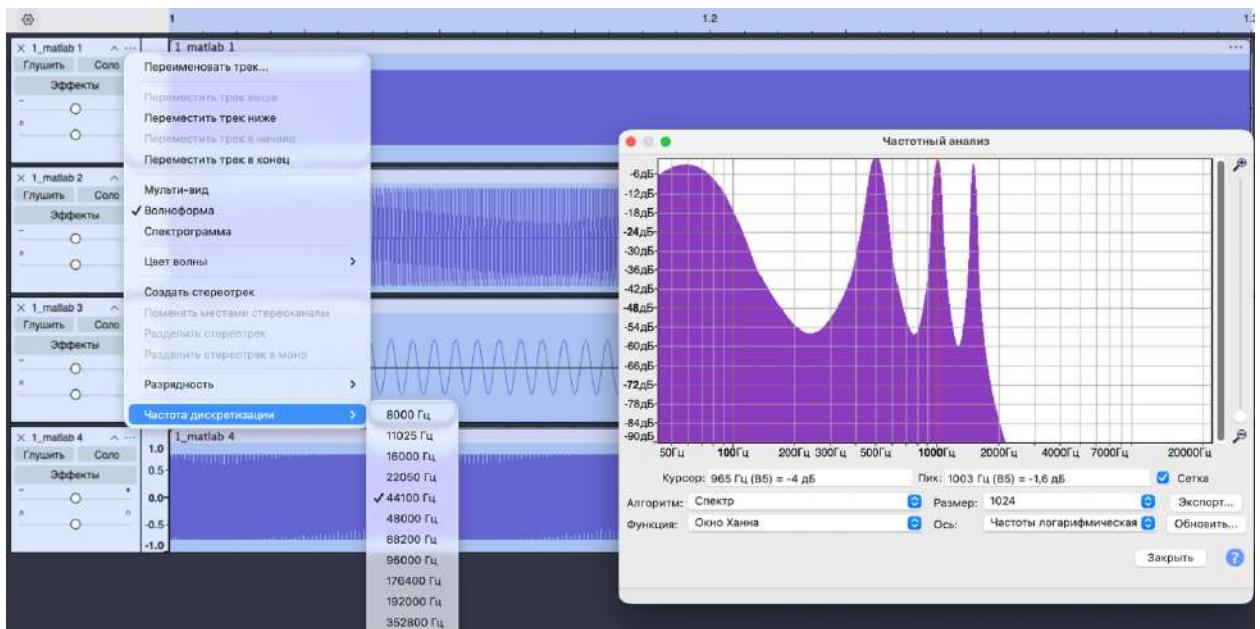


Рисунок 15 – Сигнал с 4 каналами, частотой дискретизации 44100 Гц и разрядностью 8 бит, сгенерированный в matlab и прочитанный в audacity

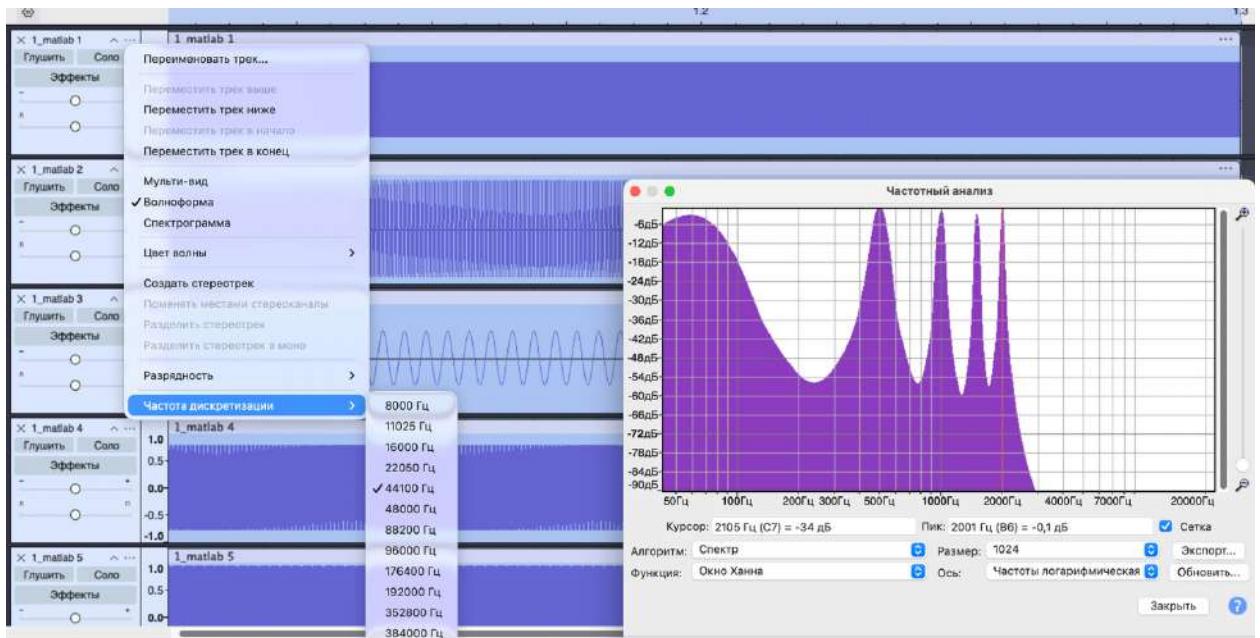


Рисунок 16 – Сигнал с 5 каналами, частотой дискретизации 44100 Гц и разрядностью 8 бит, сгенерированный в matlab и прочитанный в audacity

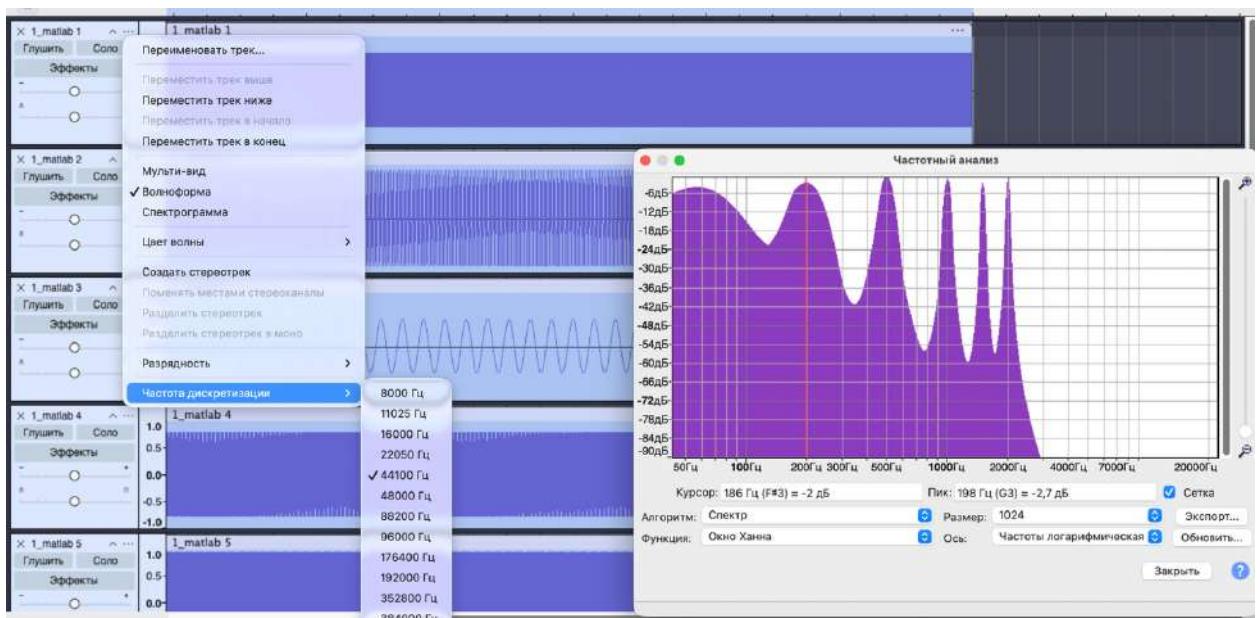


Рисунок 17 – Сигнал с 6 каналами, частотой дискретизации 44100 Гц и разрядностью 8 бит, сгенерированный в matlab и прочитанный в audacity

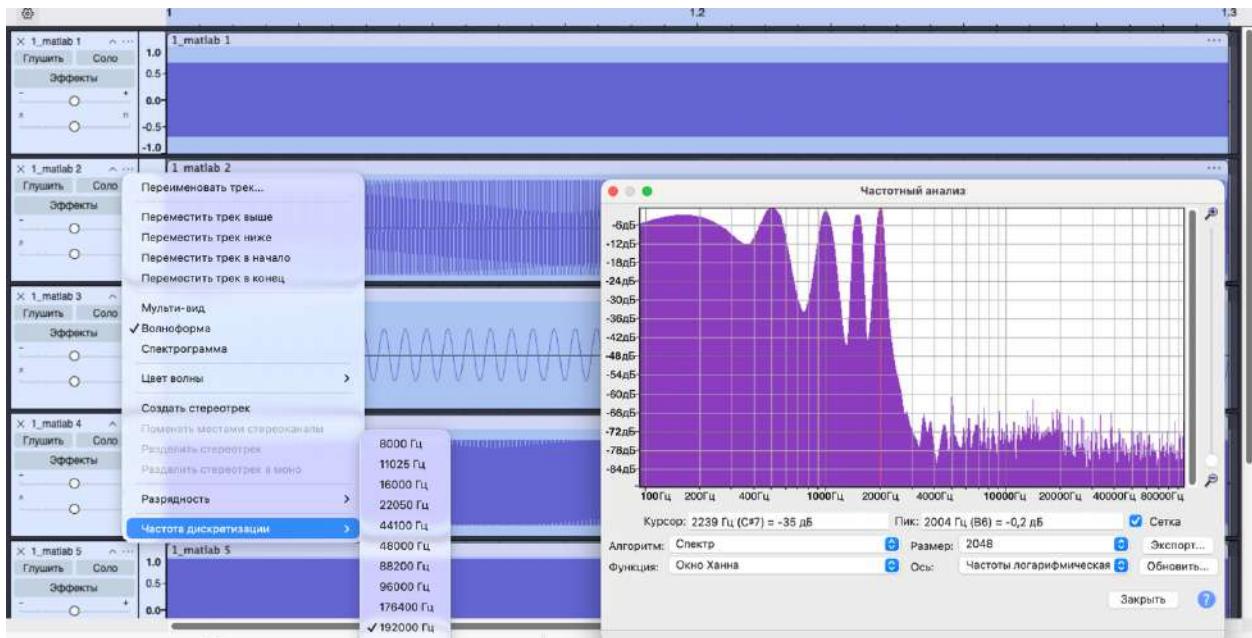


Рисунок 18 – Сигнал с 6 каналами, частотой дискретизации 192000 Гц и разрядностью 8 бит, сгенерированный в matlab и прочитанный в audacity

Вывод

В ходе лабораторной работы были сгенерированы и проанализированы аудиосигналы с различным числом каналов, частотой дискретизации и разрядностью как в MATLAB, так и в Audacity. Были выявлены следующие особенности:

- Audacity при чтении WAV-файлов всегда использует 32-битные числа с плавающей точкой, независимо от исходного формата файла.
- Небольшие расхождения (1–2 Гц) в положении спектральных пиков между сгенерированным и восстановленным сигналами обусловлены погрешностью вычисления спектра.

Код программы 1

```
clear;
clc;

wavFiles = {
    '/Users/kirill/Desktop/Projects/study/4 курс/Системы цифровой звукозаписи/1.wav',
    '/Users/kirill/Desktop/Projects/study/4 курс/Системы цифровой звукозаписи/2_16.wav',
    '/Users/kirill/Desktop/Projects/study/4 курс/Системы цифровой звукозаписи/2_24.wav',
    '/Users/kirill/Desktop/Projects/study/4 курс/Системы цифровой звукозаписи/2_32.wav',
    '/Users/kirill/Desktop/Projects/study/4 курс/Системы цифровой звукозаписи/3_3.wav',
    '/Users/kirill/Desktop/Projects/study/4 курс/Системы цифровой звукозаписи/3_4.wav',
    '/Users/kirill/Desktop/Projects/study/4 курс/Системы цифровой звукозаписи/3_5.wav',
    '/Users/kirill/Desktop/Projects/study/4 курс/Системы цифровой звукозаписи/3_6.wav',
    '/Users/kirill/Desktop/Projects/study/4 курс/Системы цифровой звукозаписи/4_192000_unsigned.wav'
};

for idx = 1:numel(wavFiles)
    fname = wavFiles{idx};

    fid = fopen(fname, 'rb', 'l');
    cleanup = onCleanup(@() fclose(fid));

    chunkID = char(fread(fid, 4, 'uchar'));
    chunkSize = fread(fid, 1, 'uint32');
    format = char(fread(fid, 4, 'uchar'));
    subchunk1ID = char(fread(fid, 4, 'uchar'));
    subchunk1Size = fread(fid, 1, 'uint32');
    audioFormat = fread(fid, 1, 'uint16');
    numChannels = fread(fid, 1, 'uint16');
    sampleRate = fread(fid, 1, 'uint32');
    byteRate = fread(fid, 1, 'uint32');
    blockAlign = fread(fid, 1, 'uint16');
    bitsPerSample = fread(fid, 1, 'uint16');

    if subchunk1Size > 16
        fread(fid, subchunk1Size - 16, 'uint8');
    end

    subchunk2ID = '';
    subchunk2Size = 0;
    while true
        id = char(fread(fid, 4, 'uchar'));
        sz = fread(fid, 1, 'uint32');
        if strcmp(id, 'data')
            subchunk2ID = id;
            subchunk2Size = sz;
            break;
        end
        fseek(fid, sz, 'cof');
    end
end
```

```

fprintf('\n>> Файл: %s\n', fname);
fprintf('Идентификатор файла : "%s"\n', chunkID);
fprintf('Размер файла (-8) : %d байт\n', chunkSize);
fprintf('Формат : "%s"\n', format);
fprintf('\n');
fprintf('ID подблока "fmt" : "%s"\n', subchunk1ID);
fprintf('Размер блока "fmt" : %d байт\n', subchunk1Size);
fprintf('Аудиоформат : %d (1=PCM)\n', audioFormat);
fprintf('Количество каналов : %d\n', numChannels);
fprintf('Частота дискретизации: %d Гц\n', sampleRate);
fprintf('Байт-рейт : %d байт/сек\n', byteRate);
fprintf('Размер сэмпла (всех к.): %d байт\n', blockAlign);
fprintf('Разрядность : %d бит\n', bitsPerSample);
fprintf('\n');

fprintf('ID подблока "data" : "%s"\n', subchunk2ID);
fprintf('Размер данных : %d байт\n', subchunk2Size);
fprintf('%s\n', repmat('-', 1, 50));

nSamp = floor(subchunk2Size / blockAlign);

switch bitsPerSample
    case 8
        audio = double(fread(fid, [numChannels, nSamp], 'uint8'));
    case 16
        audio = fread(fid, [numChannels, nSamp], 'int16');
    case 32
        audio = fread(fid, [numChannels, nSamp], 'int32');
    case 24
        totalBytes = nSamp * blockAlign;
        rawBytes = fread(fid, totalBytes, 'uint8');
        byteMatrix = reshape(rawBytes, blockAlign, nSamp);
        audio = zeros(numChannels, nSamp);
        for ch = 1:numChannels
            startIdx = (ch-1)*3 + 1;
            endIdx = ch*3;
            chanBytes = byteMatrix(startIdx:endIdx, :);
            padded = [chanBytes; zeros(1, nSamp, 'uint8')];
            asInt32 = typecast(padded(:), 'int32');
            audio(ch, :) = bitshift(bitshift(asInt32, 8), -8);
        end
    end

t = (0:nSamp-1)' / sampleRate;
if bitsPerSample == 8
    normAudio = (audio - 128) / 128;
else
    normAudio = audio / (2^(bitsPerSample - 1));
end

figure('Name', fname);
plot(t, normAudio');
xlabel('Время, сек.');
ylabel('Амплитуда');
title({fname, 'Signal waveform'});
grid on;
if t(end) > 0.05
    xlim([0 0.01]);
    title({fname, 'Форма сигнала'});
end
end

```

Код программы 2

```
clear;
clc;

outputFile = '/Users/kirill/Desktop/Projects/study/4 курс/Системы цифровой
звукозаписи/1_matlab.wav';

channelFrequencies = [1500, 500, 50, 1000, 2000, 200];
channelAmplitudes = [0.7, 0.9, 0.5, 0.8, 0.95, 0.7];

numChannels = 6;
bitDepth = 8;
sampleRate = 192000;
duration = 1;

t = (0 : 1/sampleRate : duration - 1/sampleRate)';
audio = zeros(length(t), numChannels);

for ch = 1:numChannels
    freq = channelFrequencies(ch);
    amp = channelAmplitudes(ch);
    audio(:, ch) = amp * sin(2 * pi * freq * t);
end

fid = fopen(outputFile, 'wb', 'l');

numSamples = size(audio, 1);
blockAlign = numChannels * bitDepth / 8;
byteRate = sampleRate * blockAlign;
dataSize = numSamples * blockAlign;
dataSize = round(dataSize);
chunkSize = 36 + dataSize;

fwrite(fid, 'RIFF', 'char');
fwrite(fid, uint32(chunkSize), 'uint32');
fwrite(fid, 'WAVE', 'char');
fwrite(fid, 'fmt ', 'char');
fwrite(fid, uint32(16), 'uint32');
fwrite(fid, uint16(1), 'uint16');
fwrite(fid, uint16(numChannels), 'uint16');
fwrite(fid, uint32(sampleRate), 'uint32');
fwrite(fid, uint32(byteRate), 'uint32');
fwrite(fid, uint16(blockAlign), 'uint16');
fwrite(fid, uint16(bitDepth), 'uint16');
fwrite(fid, 'data', 'char');
fwrite(fid, uint32(dataSize), 'uint32');

if bitDepth == 8
    clipped = max(-1, min(1, audio));
    samples8 = uint8(round((clipped + 1) * 127.5));
    interleaved = reshape(samples8.', 1, []);
    fwrite(fid, interleaved, 'uint8');

elseif bitDepth == 16
    maxVal16 = 2^15 - 1;
    clipped = max(-1, min(1, audio));
    samples16 = int16(clipped * maxVal16);
    interleaved = reshape(samples16.', 1, []);
    fwrite(fid, interleaved, 'int16');
```

```

elseif bitDepth == 32
    maxVal32 = 2^31 - 1;
    clipped = max(-1, min(1, audio));
    samples32 = int32(clipped * maxVal32);
    interleaved = reshape(samples32.', 1, []);
    fwrite(fid, interleaved, 'int32');

elseif bitDepth == 24
    maxVal24 = 2^23 - 1;
    clipped = max(-1, min(1, audio));
    samples32_full = int32(clipped * maxVal24);
    rawData = zeros(numSamples * numChannels * 3, 1, 'uint8');
    for i = 1:numSamples
        for ch = 1:numChannels
            val = samples32_full(i, ch);
            bytesAll = typecast(uint32(typecast(val, 'uint32')), 'uint8');
            idx = (i - 1) * numChannels * 3 + (ch - 1) * 3 + (1:3);
            rawData(idx) = bytesAll(1:3);
        end
    end
    fwrite(fid, rawData, 'uint8');
end

fclose(fid);
fprintf('\nФайл сохранён: %s\n', outputFile);

```