

Universidade Federal de Santa Catarina - UFSC  
Departamento de Engenharia Elétrica e Eletrônica

Gabriel Martins dos Santos Cunico

## **Embaralhador de Voz**

Santa Catarina - SC  
2025/Set

## Sumário

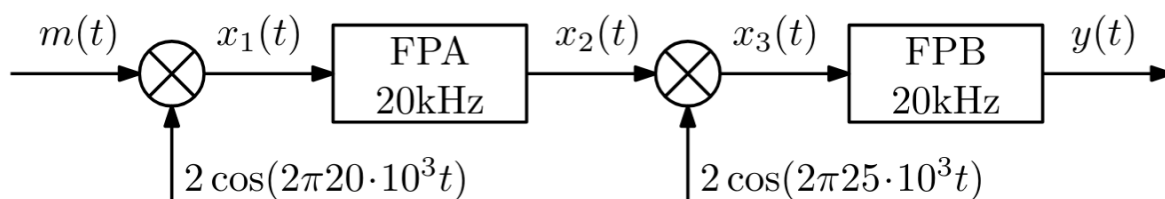
<b>1. Introdução .....</b>	<b>3</b>
<b>1.1 Sistema de Embaralhamento .....</b>	<b>3</b>
<b>1.2 Amostragem .....</b>	<b>3</b>
<b>2. Metodologia .....</b>	<b>4</b>
<b>2.1 Importação .....</b>	<b>4</b>
<b>2.2 Upscaling .....</b>	<b>4</b>
<b>2.3 Embaralhamento - Modulação .....</b>	<b>4</b>
<b>2.4 Desembaralhamento - Demodulação .....</b>	<b>6</b>
<b>2.5 Visualizações .....</b>	<b>6</b>
<b>3. Chirp .....</b>	<b>7</b>
<b>4. 19 William Tell Overture .....</b>	<b>8</b>
<b>5. Megalovania .....</b>	<b>9</b>
<b>6. Ta Vendo Aquela Lua .....</b>	<b>10</b>
<b>7. Conclusão .....</b>	<b>11</b>

## 1. Introdução

Este relatório apresenta o desenvolvimento e aplicação de um sistema de embaralhamento de voz. Serão descritos os métodos utilizados, a ferramenta desenvolvida e seu uso, os testes realizados e os resultados obtidos.

### 1.1 Sistema de Embaralhamento

O sistema de embaralhamento de voz utilizado baseia-se no deslocamento do sinal por frequências específicas, acompanhado da aplicação de filtros segundo o seguinte diagrama:



Onde, considerando  $m(t)$  um sinal em banda base com largura 5kHz:

- $m(t)$  é deslocado até 20kHz;
- A cópia do sinal  $m(t)$  vinda das frequências negativas é removida por um filtro passa altas em 20kHz;
- O sinal é deslocado em 25kHz, trazendo o espelho de  $x_2(t)$  das frequências negativas até o intervalo 0-5kHz;
- Um filtro passa baixas elimina as altas frequências resultantes do deslocamento de 25kHz aplicado à parte do sinal em torno de 20kHz;
- Por fim, obtém-se a faixa entre 0-5kHz de  $m(t)$ , espelhada em relação ao eixo da amplitude.

### 1.2 Amostragem

O sistema descrito para embaralhamento de sinais conta com multiplicações de sinais em torno de 25kHz. Entretanto, arquivos de áudio costumam ser salvos com amostragem na faixa de 44,1kHz, o que, pelo *Teorema da Amostragem*, pode gerar perdas de informação na aplicação do sistema (frequências acima de 22,05kHz são geradas durante as transformações). Portanto, é necessário empregar o processo de *upsampling* para contornar o problema.

Realizar *upsampling* consiste em gerar amostras sintéticas entre os pontos conhecidos, de modo a aumentar a taxa de amostragem do sinal. Para isso, os seguintes passos são realizados sobre o sinal:

1. Calcula-se a razão  $L$  entre a frequência atual e a desejada, arredondando para o inteiro imediatamente superior;
2. Inserem-se  $L - 1$  zeros entre cada amostra do sinal original;
3. Um filtro passa baixas é aplicado para suavizar as descontinuidades. Conhecido como filtro de interpolação, ele substitui os zeros com base em amostras vizinhas.

## 2. Metodologia

Para aplicar as devidas transformações nos sinais de áudio, foi desenvolvida a ferramenta Audio Scrambler, que é distribuída junto a este relatório.

Para utilizar a ferramenta, basta adicionar os arquivos de áudio de interesse no diretório `"/samples"` e executar o script `"run.py"`. Após a execução, imagens e áudios dos sinais serão gerados na pasta `"/output"`, organizados por arquivo de áudio de referência. Os links para arquivos locais presentes neste PDF podem deixar de funcionar caso o PDF seja aberto em um *browser* e, neste caso, os áudios gerados devem ser acessados diretamente em `"/output"`.

Os arquivos gerados serão espectograma, representação no tempo e áudios em formato `".wav"`. Cada arquivo terá informações referentes ao sinal original após *upsampling* ( $m(t)$ ), sinal embaralhado ( $y(t)$ ) e sinal desembaralhado ( $m_{rec}(t)$ ). Ainda, este relatório será atualizado após o processamento, incluindo seções referentes a cada arquivo de entrada.

Atente-se que algumas etapas de processamento exigem filtros de ordem elevada, o que, combinado com a alta taxa de amostragem, pode levar a um processamento lento - proporcional a quantidade de áudios a serem processados.

### 2.1 Importação

A importação dos áudios é realizada com auxílio da ferramenta *ffmpeg*, de modo que formatos de áudio diversos são aceitos. Caso haja algum problema com os arquivos de entrada, por favor converta o arquivo para um dos formatos testados durante o desenvolvimento: MP3 ou WAV.

### 2.2 Upscaling

Para realizar o *upsampling* do áudio de entrada, o filtro de interpolação utilizado foi uma arquitetura Butterworth passa baixas de quinta ordem, projetado para a frequência de Nyquist do sinal de entrada. Como a banda efetiva para aplicação do embaralhamento (0-5kHz) ocorre distante da frequência de corte do filtro (geralmente 22kHz), sua seletividade é reduzida e sua ordem pode ser pequena.

A ferramenta desenvolvida é configurada por padrão para realizar o *upsampling* para 96kHz, mas também pode trabalhar com outras taxas de amostragem. A configuração desta taxa pode ser realizada no arquivo `"run.py"`.

### 2.3 Embaralhamento - Modulação

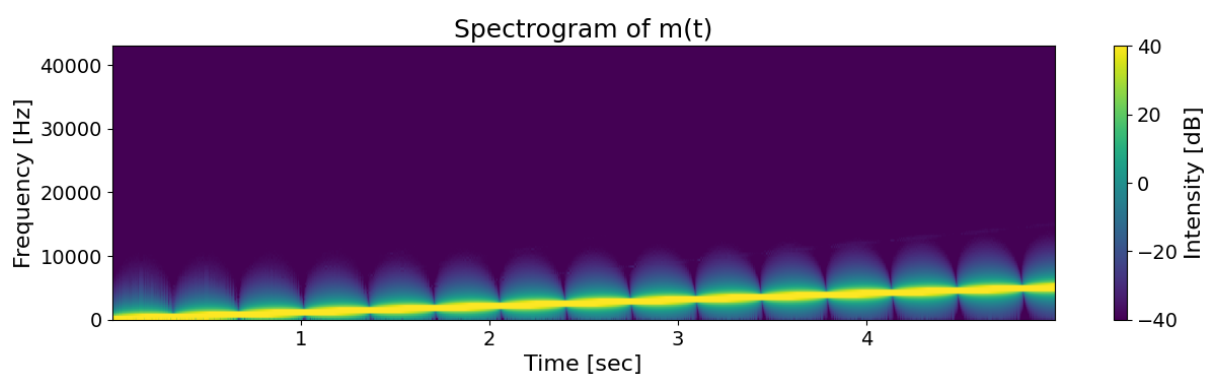
Para aplicar o sistema de embaralhamento, foram gerados cossenos de 20kHz e 25kHz a partir de um vetor de tempo. O deslocamento de frequências foi obtido pela multiplicação direta dos sinais com esses cossenos.

Quanto aos filtros projetados, a arquitetura utilizada foi Butterworth com frequência de corte 20kHz tanto para o filtro passa baixas quanto passa altas. Entretanto, diferente do filtro de interpolação, a seletividade deste filtro é alta, devendo atenuar frequências imediatamente após a frequência de corte.

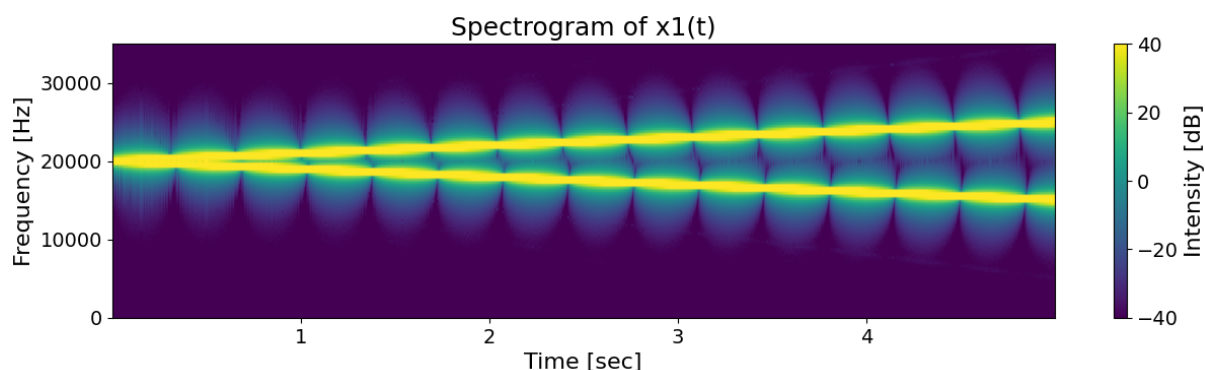
Diferentes configurações foram testadas e, a partir da observação de espectrogramas dos sinais gerados pelo sistema a partir de uma função chirp, resultados aceitáveis foram obtidos da ordem 100 em diante. Foi escolhido um filtro Butterworth de ordem 150 para reduzir ainda mais a banda de transição.

A partir dos sinais e filtros descritos, as etapas do sistema de embaralhamento foram aplicadas:

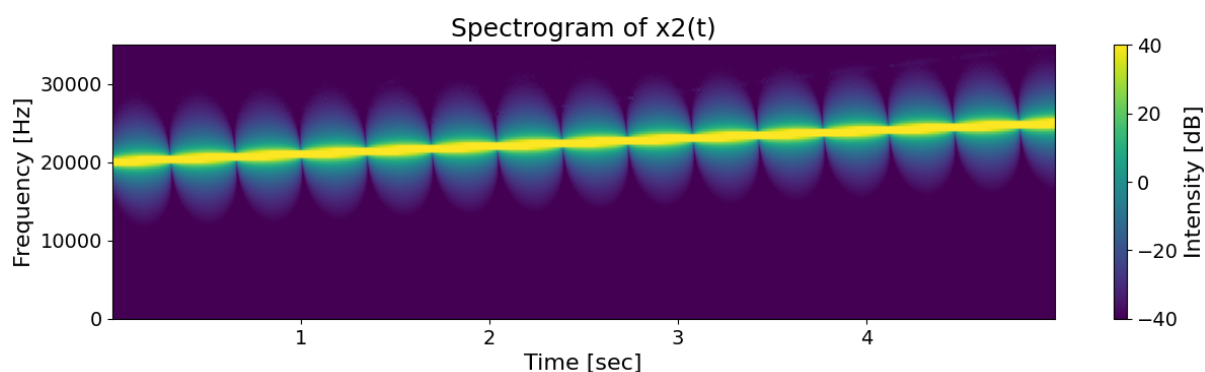
0. Utilizando uma função *chirp* de 20 a 5kHz como sinal de entrada  $m(t)$ :



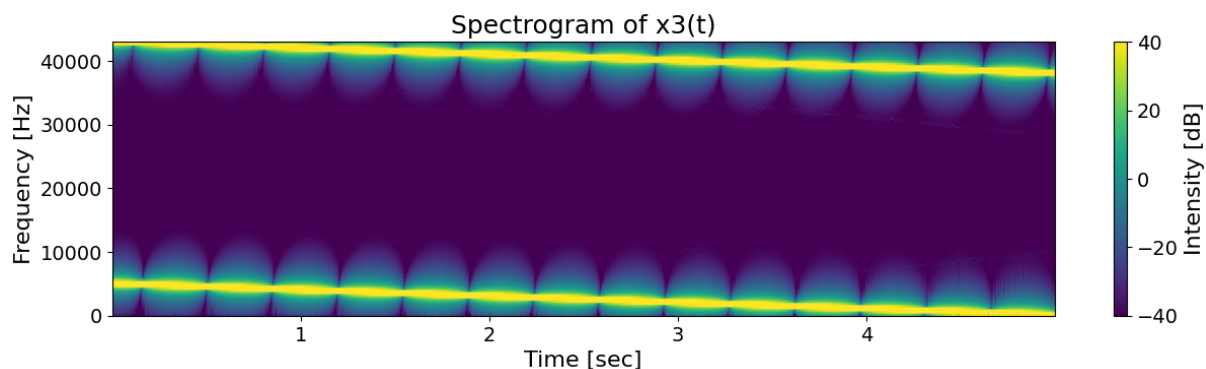
1. Multiplicação do sinal  $m(t)$  pelo cosseno de 20kHz, obtendo o sinal  $x_1(t)$ :



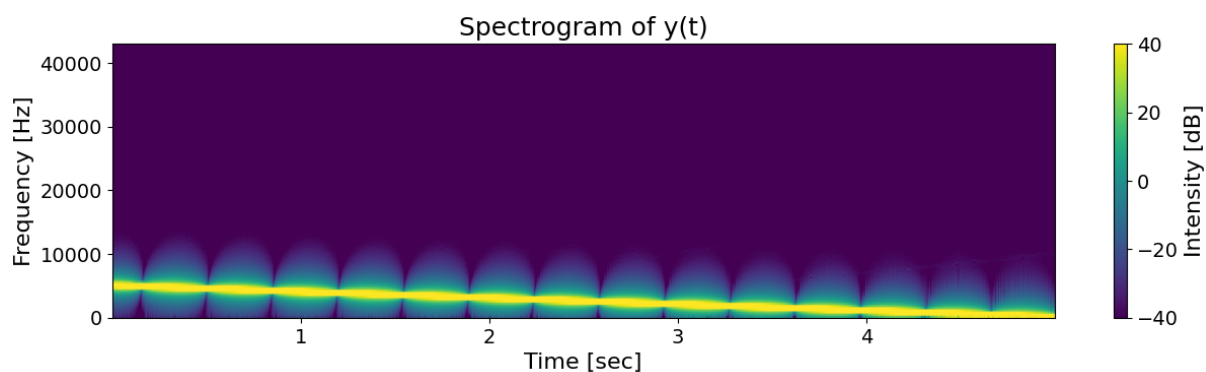
2. Filtragem de  $x_1(t)$  pelo filtro passa altas com frequência de corte 20kHz, obtendo o sinal  $x_2(t)$ :



3. Multiplicação do sinal  $m(t)$  pelo cosseno de 25kHz, obtendo o sinal  $x_3(t)$ :



4. Filtragem de  $x_3(t)$  pelo filtro passa baixas com frequência de corte 20kHz, obtendo o sinal  $y(t)$ :



## 2.4 Desembaralhamento - Demodulação

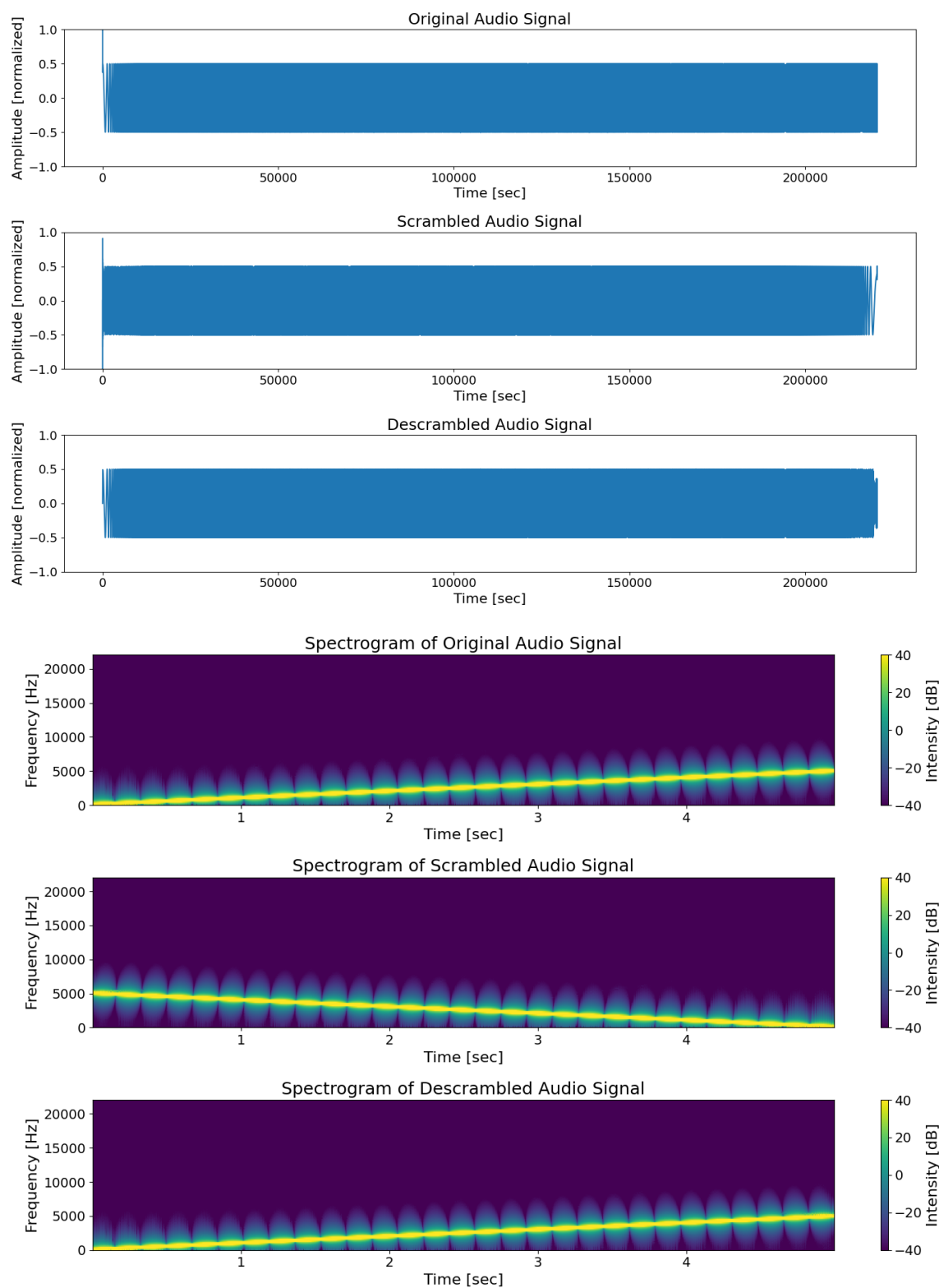
A demodulação dos sinais embaralhados é realizada de forma análoga à modulação, dado que o sistema é auto inversor. Assim, os mesmos cossenos e filtros são utilizados para deslocar as frequências de volta à banda original e reconstruir o sinal, aplicados na mesma ordem.

## 2.5 Visualizações

Os sinais de áudio utilizados internamente são gerados diretamente dos sinais processados em 96kHz. Contudo, a plotagem do sinal ao longo do tempo e espectrogramas demandam muito processamento e memória, de modo que a geração destas visualizações, bem como a escrita dos arquivos de áudio de saída, ocorrem após realizar *downsampling* do sinal processado para 44,1kHz.

### 3. Chirp

Visualizações referentes ao arquivo Chirp.



Os áudios podem ser ouvidos clicando nos links abaixo:

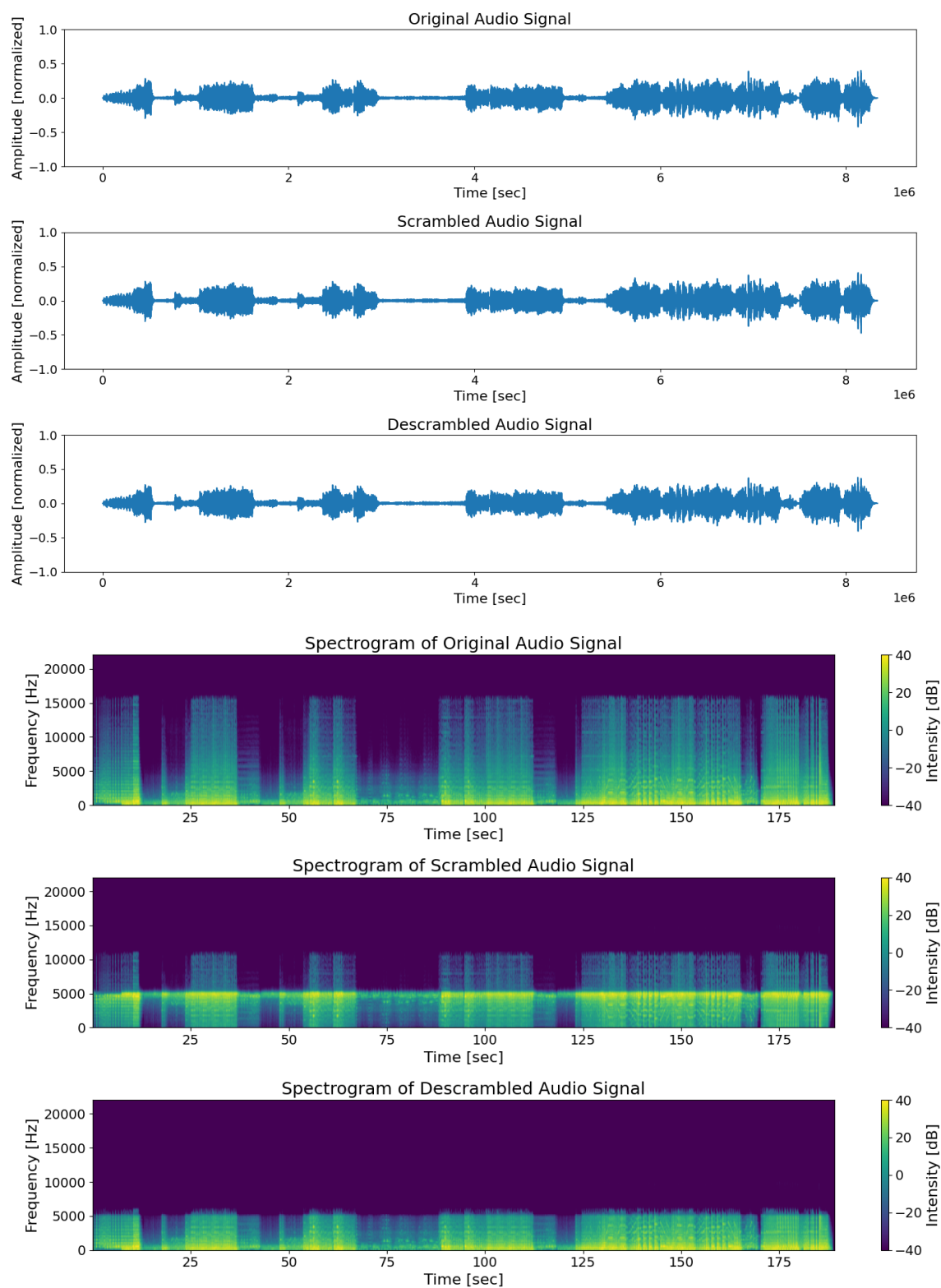
[Áudio Original](#)

[Áudio Embaralhado](#)

[Áudio Desembaralhado](#)

## 4. 19 William Tell Overture

Visualizações referentes ao arquivo 19 William Tell Overture.



Os áudios podem ser ouvidos clicando nos links abaixo:

[Áudio Original](#)

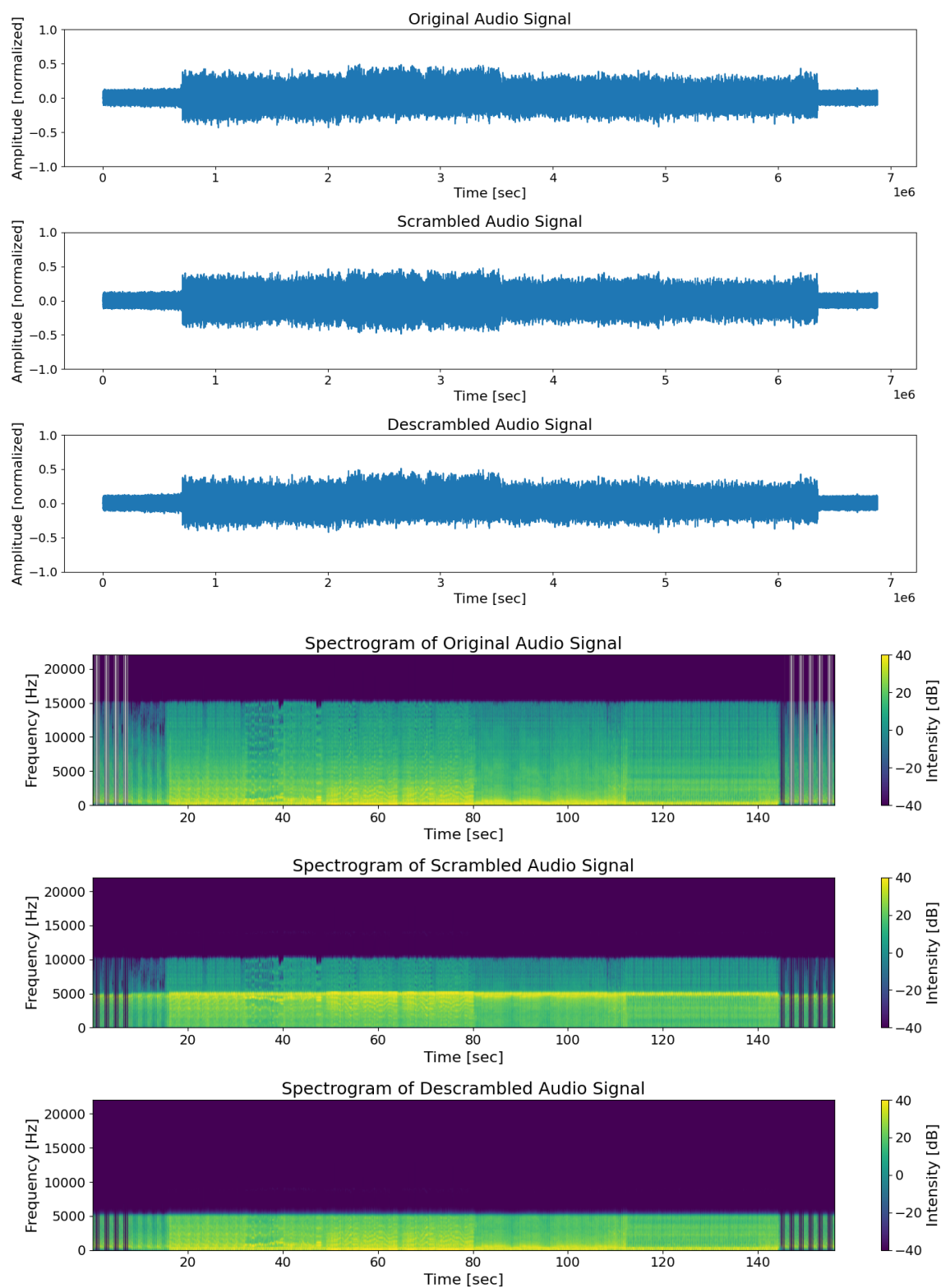
[Áudio Embaralhado](#)

[Áudio Desembaralhado](#)



## 5. Megalovania

Visualizações referentes ao arquivo Megalovania.



Os áudios podem ser ouvidos clicando nos links abaixo:

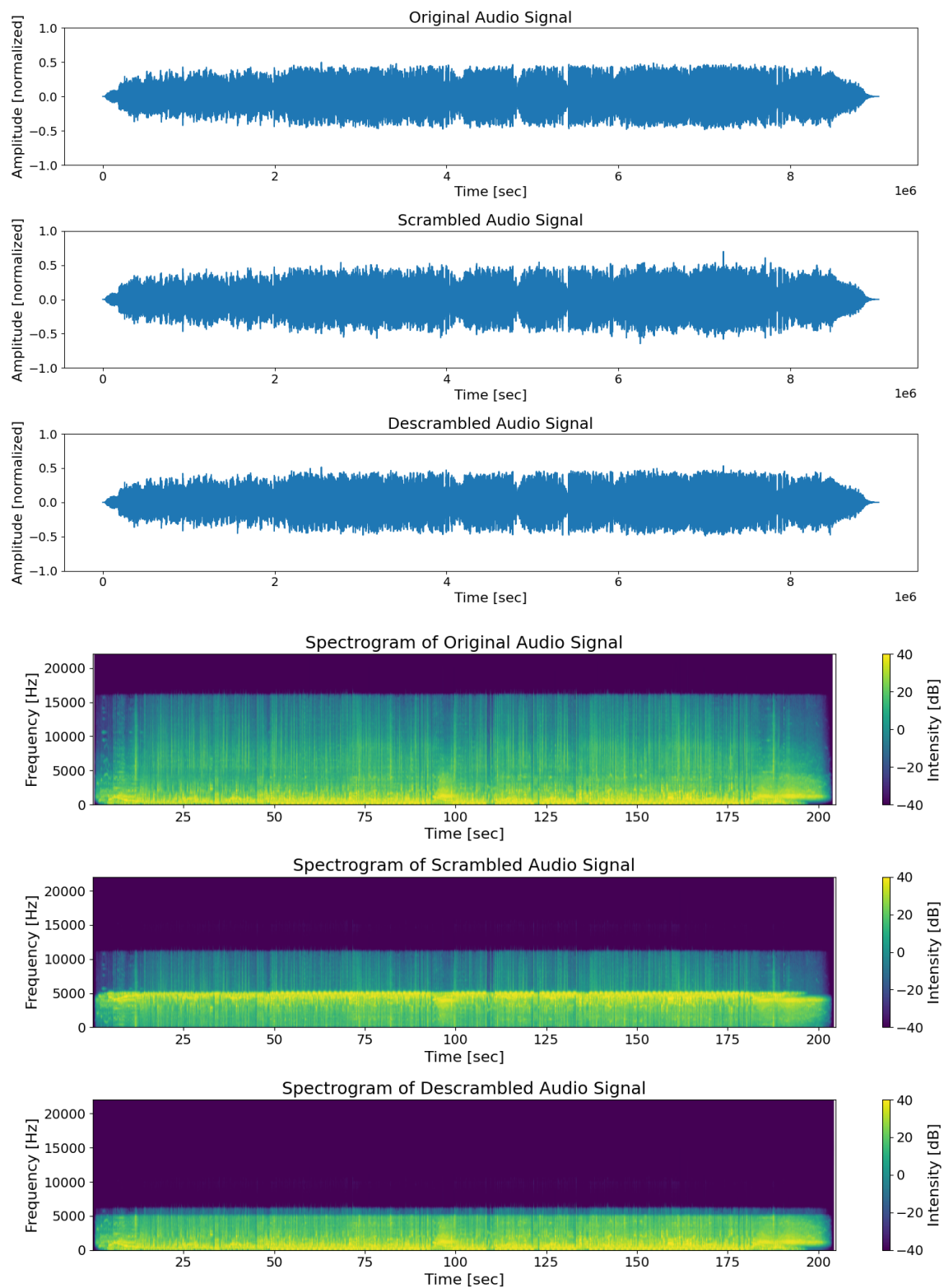
[Áudio Original](#)

[Áudio Embaralhado](#)

[Áudio Desembaralhado](#)

## 6. Ta Vendo Aquela Lua

Visualizações referentes ao arquivo Ta Vendo Aquela Lua.



Os áudios podem ser ouvidos clicando nos links abaixo:

[Áudio Original](#)

[Áudio Embaralhado](#)

[Áudio Desembaralhado](#)

## 7. Conclusão

O sistema de embaralhamento e desembaralhamento de sinais de áudio foi implementado com sucesso. Através da utilização de técnicas de processamento de sinais, foi possível atingir os objetivos propostos no início do projeto. Entretanto, dada a banda estreita em que o sistema opera, a qualidade do áudio processado é afetada, principalmente a parte instrumental ou eletrônica dos sinais de áudio. Assim, embora funcione corretamente (principalmente com vozes humanas), as frequências mais agudas do espectro não são preservadas e um ruído perceptível é introduzido.

Enquanto os espectros dos sinais exibem diferenças notáveis após aplicação do embaralhamento e desembaralhamento, a representação de cada sinal no domínio do tempo permanece visualmente similar.

Ainda, os algoritmos de *upsampling* e *downsampling* implementados foram efetivos e possibilitaram, com auxílio de ferramentas de codificação e decodificação de áudio, o tratamento local dos arquivos e seus sinais.