Universidade Federal do Ceará
Centro de Tecnologia
Departamento de Engenharia de Teleinformática
Processamento Digital de Sinais - TIP7200

Lucas Abdalah

Exercício Computacional 2

Professor: André Almeida
Data de Entrega: 08 de maio de 2023

Sumário

Introdução

O presente trabalho é dividido em seções intituladas de "Problema", enumerados de 1 a 6. Em, apresenta um apêndice com o comando dos problemas propostos e o código de solução dessas questões, gerado com função *publish* para exportar códigos do Matlab.

Por fim, o código[1], as figuras[2] e os áudios[3] gerados no decorrer deste relatório são disponibilizados em repositório público do Github.

[1]Código: `https://github.com/lucasabdalah/Courses-HWs/blob/master/Master/TIP7200-PROCESSAMENTO_DIGITAL_DE_SINAIS/hw2/code/script.m`

[2]Figuras: `https://github.com/lucasabdalah/Courses-HWs/tree/master/Master/TIP7200-PROCESSAMENTO_DIGITAL_DE_SINAIS/hw2/figures`

[3]Áudios: `https://github.com/lucasabdalah/Courses-HWs/tree/master/Master/TIP7200-PROCESSAMENTO_DIGITAL_DE_SINAIS/hw2/audio`

Problema 1

A questão apresenta o arquivo de áudio ”bipsIN.wav”como entrada $x[n]$ de um sistema H. O arquivo tem duração total de 5 segundos e consiste em uma combinação de três tons de senoides, cada um com uma frequência distinta e duração de 1 segundo, como pode ser observado no domínio do tempo (Figura 1).

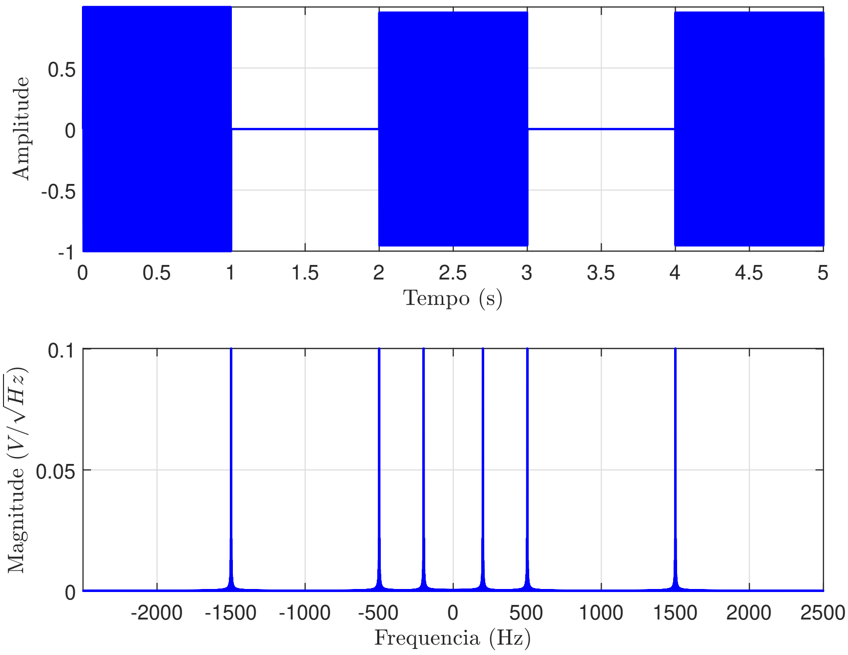


Figura 1: Sinal de entrada representado no domínios do tempo e da frequência.

O sinal também pode ser representado matematicamente por meio da seguinte equação:

$$x[n] = \sum_{n=0}^{Fs-1} \text{sen}(\omega_1 n) + \sum_{n=2Fs}^{3Fs-1} \text{sen}(\omega_2 n) + \sum_{n=4Fs}^{5Fs-1} \text{sen}(\omega_3 n) \tag{1}$$

onde $\omega_k = 2\pi f_k$ e f_1, f_2 e f_3 representam as frequências de 200, 500 e 1500 Hz, respectivamente.

A representação do sinal no domínio da frequência pode ser obtida através de sua transformada de Fourier (TF), representada por $\mathcal{F}\{\cdot\}$. A TF é uma ferramenta matemática utilizada para decompor um sinal complexo em um conjunto de componentes mais simples. No caso de uma onda senoidal, que é uma função periódica, a TF expressa o sinal como uma soma de ondas senoidais de diferentes frequências. Para simplificar a visualização, o desenvolvimento é apresentado apenas para uma entrada $\text{sen}(\omega_k t)$, mas que se estende para o sinal $x[n]$, dado a propriedade da linearidade da TF [1].

$$\mathcal{F}\{\text{sen}(\omega_k t)\} = \frac{\pi}{j}[\delta(\omega - \omega_k) - \delta(\omega + \omega_k)] \tag{2}$$

A equação 2 demonstra que o espectro de frequência da entrada considerada apresenta dois impulsos deslocados nas frequências $\pm\omega_k$, que diferem apenas no sinal, ambos com a presença do número complexo j.

Visando aplicar o conceito apresentado na equação 2 para o sinal $x[n]$ fornecido pelo problema e que é modelado pelo autor como a equação 1, podemos aplicar a TF e obter a seguinte expressão:

$$
\begin{aligned}
\mathcal{F}\{x[n]\} =& \mathcal{F}\{\sum_{n=0}^{Fs} \text{sen}(\omega_1 n)\} + \mathcal{F}\{\sum_{n=2Fs}^{3Fs} \text{sen}(\omega_2 n)\} + \mathcal{F}\{\sum_{n=4Fs}^{5Fs} \text{sen}(\omega_3 n)\} \\
=& \frac{\pi}{j}[\delta(\omega-\omega_1) - \delta(\omega+\omega_1)] + \dots \\
& \frac{\pi}{j}[\delta(\omega-\omega_2) - \delta(\omega+\omega_2)] + \dots \\
& \frac{\pi}{j}[\delta(\omega-\omega_3) - \delta(\omega+\omega_3)]
\end{aligned} \tag{3}
$$

O resultado é uma soma de impulsos deslocados nas frequências $\pm\omega_1$, $\pm\omega_2$ e $\pm\omega_3$.

Para ilustrar o resultado das equações anteriores e compará-lo com o que é obtido no domínio da frequência na figura 1, pode-se desenhar a TF da função $\text{sen}(\omega_k t)$, como na figura 2.

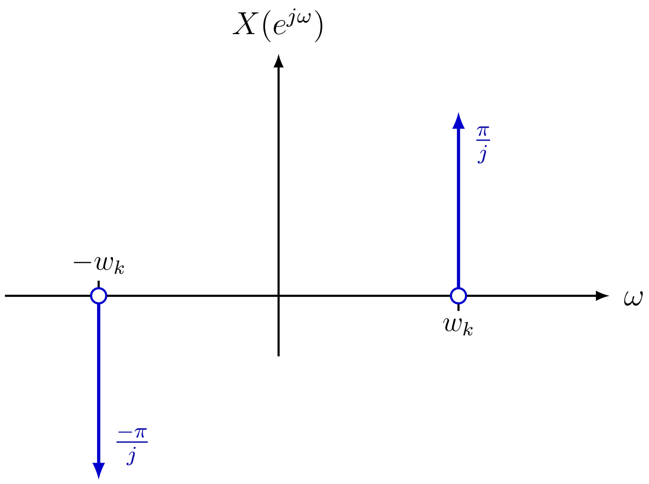


Figura 2: Gráfico da transformada de Fourier da função $\text{sen}(\omega_k t)$.

O modelo matemático apresentado corrobora os resultados experimentais obtidos com $x[n]$. Entretanto, é evidente que o lado negativo do espectro de frequência está invertido, pois quando aplicamos a TF a uma onda senoidal, obtemos um espectro de frequência que mostra a amplitude dos diferentes componentes de frequência que compõem o sinal. O espectro terá um pico na frequência da onda senoidal, indicando a intensidade desse componente de frequência particular.

No entanto, como a onda senoidal não é simétrica e é uma função ímpar, a TF também produzirá uma imagem espelhada do espectro de frequência, representando frequências negativas. Isso ocorre porque a TF assume que o sinal de entrada é periódico, e as frequências negativas representam uma continuação retroativa do sinal além do intervalo no qual ele é definido.

Portanto, a TF de uma onda senoidal terá tipicamente dois picos de magnitude igual, um positivo e um negativo, representando os componentes de frequência positiva e negativa do sinal.

No caso computacional da Transformada Rápida de Fourier (*Fast Fourier Transform*, FFT), trabalha-se com a parte real e a magnitude do sinal, logo, o espectro de frequência apresentado na figura 1 indica a presença de três senoides nas frequências indicadas no modelo, juntamente com suas réplicas na parte negativa do domínio da frequência.

Problema 2

A questão apresenta o arquivo de áudio "bipsOUT.wav" como saída $y[n]$ de um sistema H. O arquivo tem duração total de 5 segundos e consiste em uma combinação de um tom de senoide mais perceptível, grave e dois tons mais agudos fortemente atenuados, praticamente inaudíveis (Figura 3).

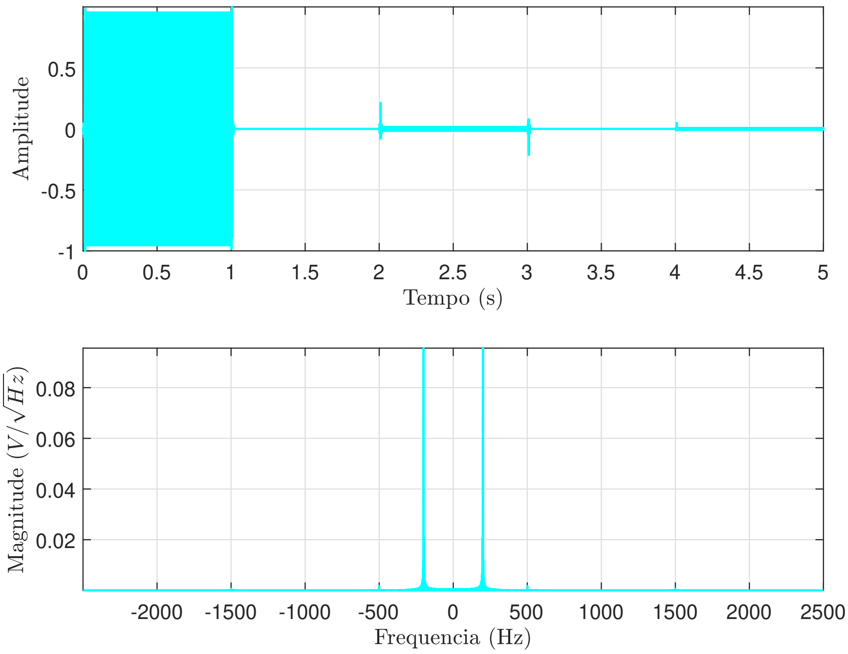


Figura 3: Sinal de saída representado no domínios do tempo e da frequência.

O sinal também pode ser representado matematicamente por meio da adaptação da equação 1, de modo que uma constante c_k é introduzida para atenuação de termos de alta frequência e manutenção da senoide de frequência f_1:

$$y[n] = c_1 \sum_{n=0}^{Fs-1} \text{sen}(\omega_1 n) + c_2 \sum_{n=2Fs}^{3Fs-1} \text{sen}(\omega_2 n) + c_3 \sum_{n=4Fs}^{5Fs-1} \text{sen}(\omega_3 n) \tag{4}$$

onde c_1 é unitário para preservar a senoide desta frequência, enquanto c_2 e $c_3 \approx 0$.

O problema também traz a tona que o sinal $y[n]$ estudado nesse caso é resultado de uma transformação $h[n]$ de um sistema Linear e Invariante no Tempo (LIT) aplicada ao sinal $x[n]$ apresentada na eqn.1, Problema 1. Para isto, faz-se necessário a implementação de um filtro de passa-baixas (Low-pass Filter, LPF).

O modelo utilizado é o *Butterworth*, que apresenta banda de passagem plana, entretanto apresenta larga banda de transição. Para sua concepção, são exigidos os parâmetros frequência de corte f_c utilizada foi a de 250 Hz.

Parâmetro	Valor
N	7
f_c	250 Hz

Tabela 1: Parâmetros do modelo LPF Butterworth.

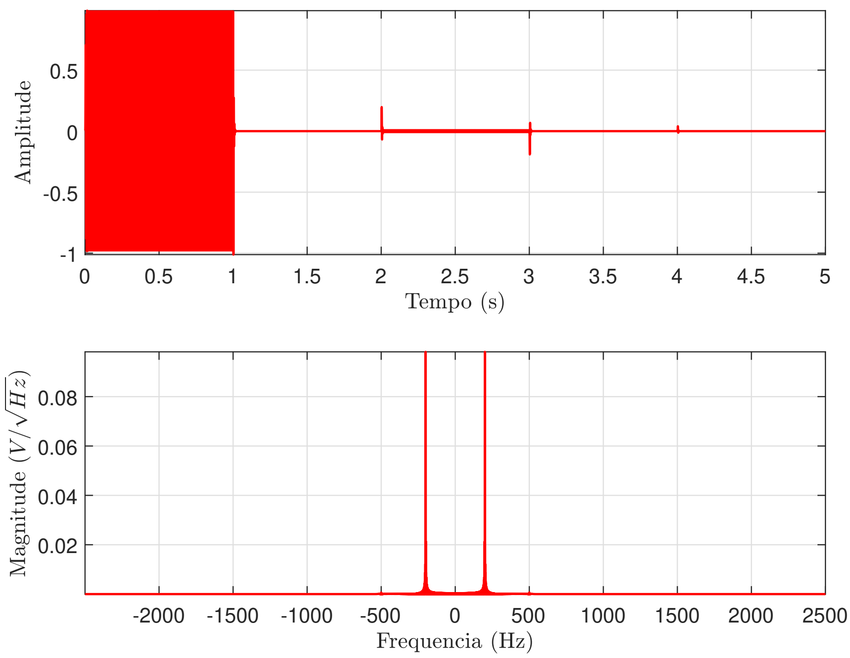


Figura 4: Sinal $\hat{y}[n]$, saída do LPF Butterworth tendo como entrada $x[n]$, "bipsIN.wav".

É necessário uma pequena operação obter uma versão normalizada (ω_c) de f_c para computar os coeficientes da função de transferência do filtro, que estão relacionados com teorema de Nyquist e a frequência de amostragem (F_s) do sinal [1], [2]. A expressão seguinte ilustra a manipulação matemática necessária:

$$\omega_c = 2\frac{f_c}{Fs} \tag{5}$$

A saída da função é composta de dois vetores b e a, que apresentam tamanho $N+1$, i.e, 8 coeficientes cada.

Com os coeficientes do filtro, para obter a saída desejada ($\hat{y}[n]$), basta aplicar a operação de convolução ($*$) entre o sistema $h[n]$ e o sinal $x[n]$:

$$\hat{y}[n] = x[n] * h[n] \tag{6}$$

É possível observar que as figuras 3 e 4 apresentam forte semelhança, tanto no domínio do tempo quanto na frequência. Essas evidências reforçam a hipótese que o sistema LTI utilizado foi um LPF.

Por fim, a figura 5 mostra as características de resposta do sistema no domínio do tempo e da frequência.

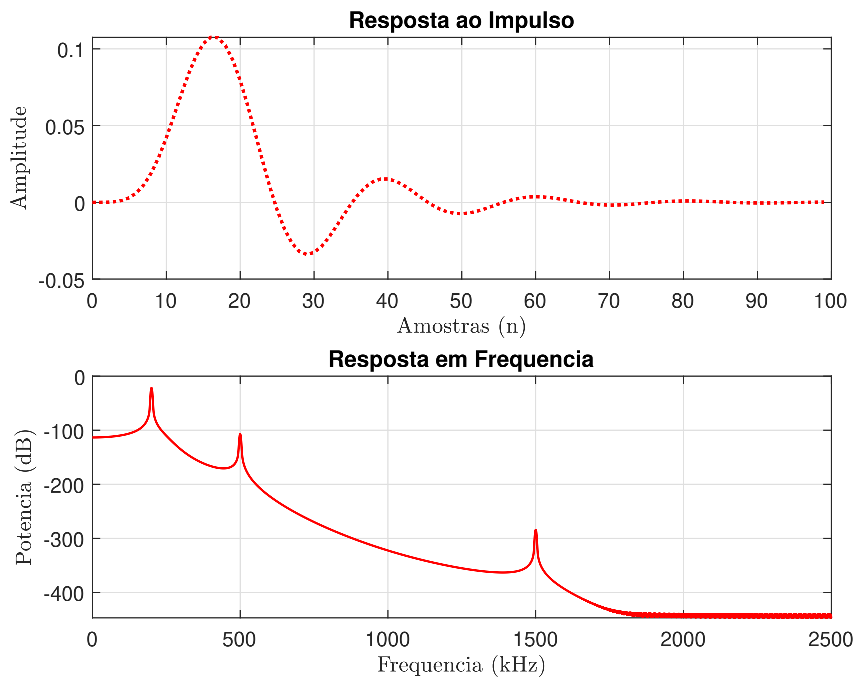


Figura 5: Resposta ao impulso e a resposta em frequência do sistema $h[n]$ projetado.

Problema 3

A questão apresenta o arquivo de áudio "bipsIN-mixed.wav" como entrada $x[n]$ de um sistema H. O arquivo tem duração total de 5 segundos e consiste na mistura de três tons de senoides, cada um com uma frequência distinta e duração de 5 segundos, como pode ser observado no domínio do tempo (Figura 6).

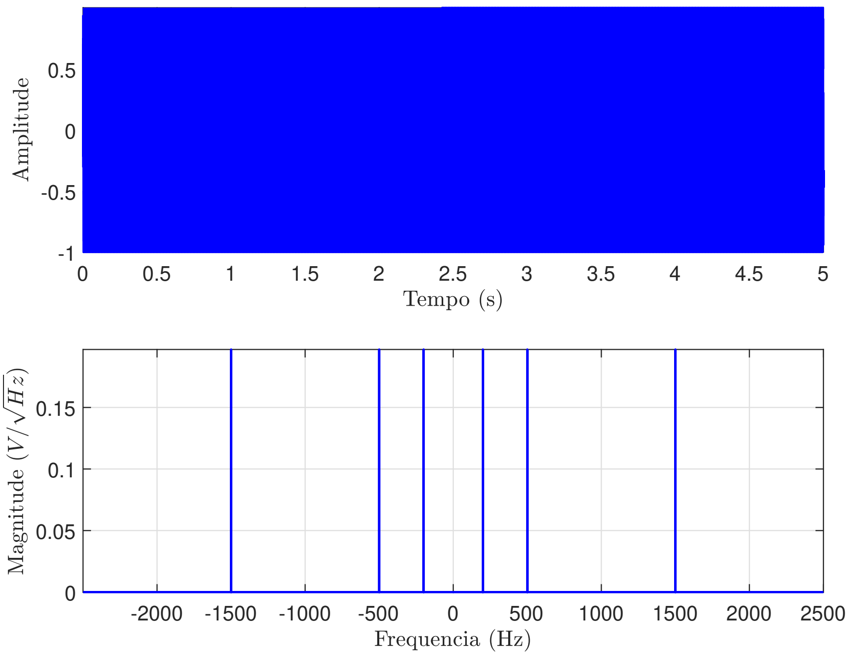


Figura 6: Sinal de entrada representado no domínios do tempo e da frequência.

O sinal também pode ser representado matematicamente por meio da seguinte equação:

$$x[n] = \sum_{n=0}^{5Fs-1} \text{sen}(\omega_1 n) + \text{sen}(\omega_2 n) + \text{sen}(\omega_3 n) \tag{7}$$

com os mesmos valores de ω_k, para $k = 1, 2, 3$, apresentados no problema 1.

A questão também apresenta o arquivo de áudio "bipsOUT-mixed.wav" como saída $y[n]$ de um sistema H. O arquivo tem duração total de 5 segundos e consiste em uma combinação de um tom de senoide mais perceptível, grave e dois tons mais agudos fortemente atenuados, praticamente inaudíveis (Figura 7).

Neste caso, os tons do sinal estão misturados no domínio do tempo, de modo que a representação matemática é uma adaptação da equação 7, de modo que uma constante c_k é introduzida para atenuação de termos de alta frequência e manutenção da senoide de frequência f_1:

$$y[n] = \sum_{n=0}^{5Fs-1} c_1\text{sen}(\omega_1 n) + c_2\text{sen}(\omega_2 n) + c_3\text{sen}(\omega_3 n) \tag{8}$$

onde c_1 é unitário para preservar a senoide desta frequência, enquanto c_2 e $c_3 \approx 0$. Na figura 8 é possível observar essa atenuação na escala de decibéis(dB), além da aparição de pequenos picos que representa as harmônicas, dadas a interação entre as senoides que tocam simultaneamente.

O problema também traz a tona que o sinal $y[n]$ estudado nesse caso é resultado de uma transformação $h[n]$ de um sistema LIT aplicada ao sinal $x[n]$ apresentada na eqn. 7. Para isto, faz-se necessário a implementação de um LPF novamente.

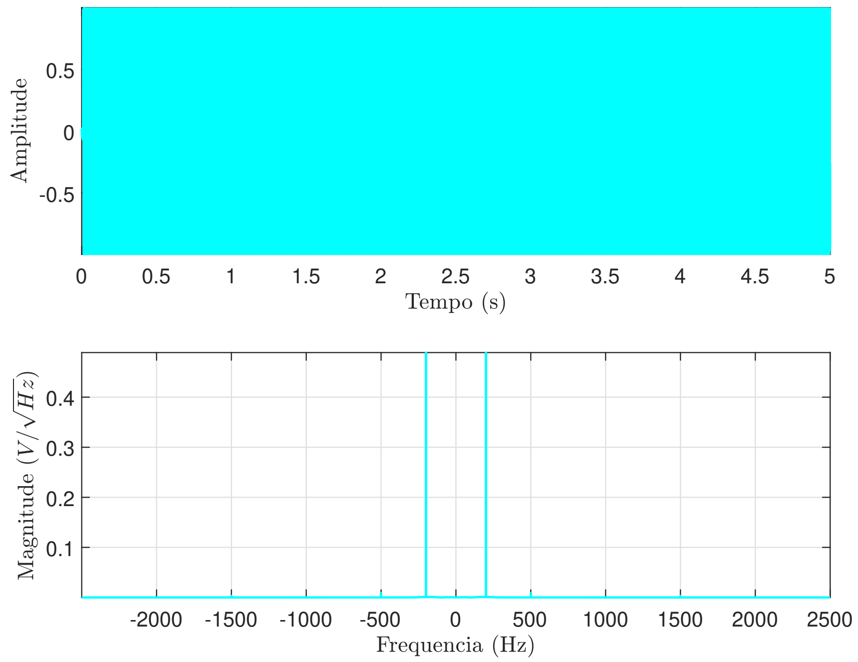


Figura 7: Sinal de saída representado no domínios do tempo e da frequência.

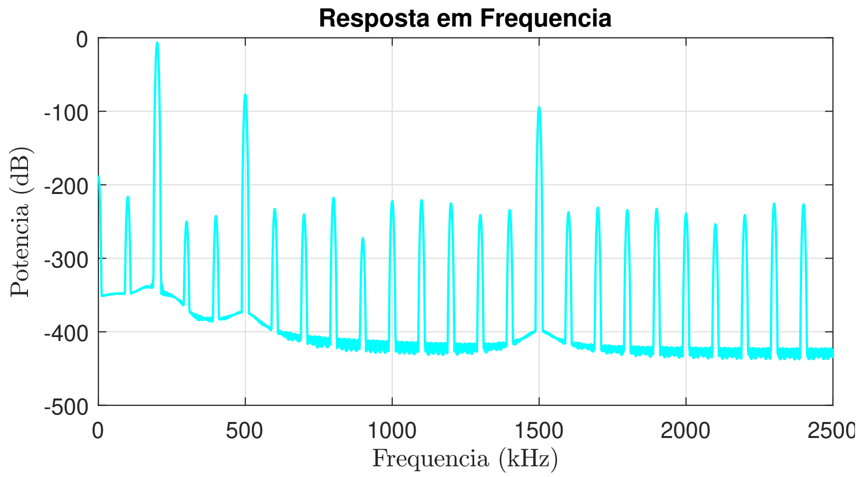


Figura 8: Sinal de saída representado no domínios da frequência em dB.

O modelo utilizado neste segundo momento é um filtro digital de resposta ao impulso finita (*finite impulse response*, FIR) é um tipo de filtro que é caracterizado por possuir coeficientes não recursivos, o que o torna uma opção mais simples e estável em comparação com os filtros IIR (*infinite impulse response*), que é o caso do filtro Butterworth.

Foi utilizada a janela de Kaiser é para projetar o filtro apresentado, pois implica na suavização das transições abruptas dos coeficientes e redução da distorção do filtro na banda de transição, mantendo uma boa rejeição de frequência. Além disso, é bastante flexível, permitindo um controle mais preciso da largura da banda de transição e da atenuação da banda de rejeição.

Para sua concepção, são exigidos os parâmetros frequência de corte f_c utilizada foi a de 450 Hz.

Seguindo os mesmos passos apresentados no Problema 1, obtém-se uma versão normalizada (ω_c) de f_c para computar os coeficientes da função de transferência do filtro,

Parâmetro	Valor
N	11
f_c	450 Hz
Janela	Kaiser

Tabela 2: Parâmetros do modelo LPF FIR (Kaiser).

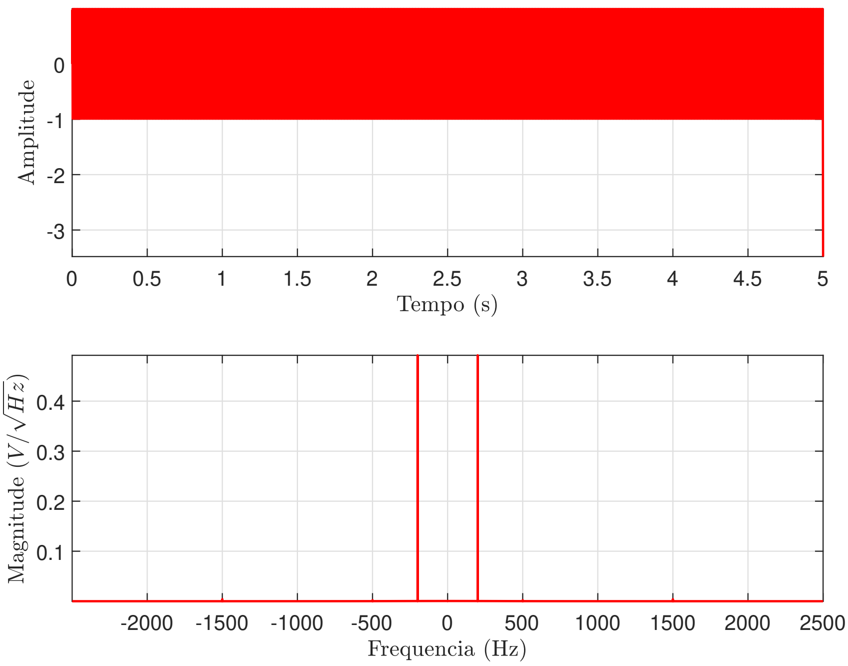


Figura 9: Sinal $\hat{y}[n]$, saída do LPF FIR (Kaiser) tendo como entrada $x[n]$, "bipsIN-mixed.wav".

que estão relacionados com teorema de Nyquist e a frequência de amostragem (F_s) do sinal [1], [2]. A expressão da eqn. 5 ilustra a manipulação matemática necessária.

A saída da função é composta de um vetor b, que apresentam tamanho $N + 1$, i.e, 12 coeficientes. Em posse desses coeficientes, para obter a saída desejada ($\hat{y}[n]$), basta aplicar a operação de convolução da eqn. 6 entre o sistema $h[n]$ e o sinal $x[n]$. É possível observar que as figuras 7 e 9 apresentam forte semelhança, tanto no domínio do tempo quanto na frequência. Essas evidências reforçam a hipótese que o sistema LTI utilizado foi um LPF.

Por fim, a figura 5 mostra as características de resposta do sistema no domínio do tempo e da frequência.

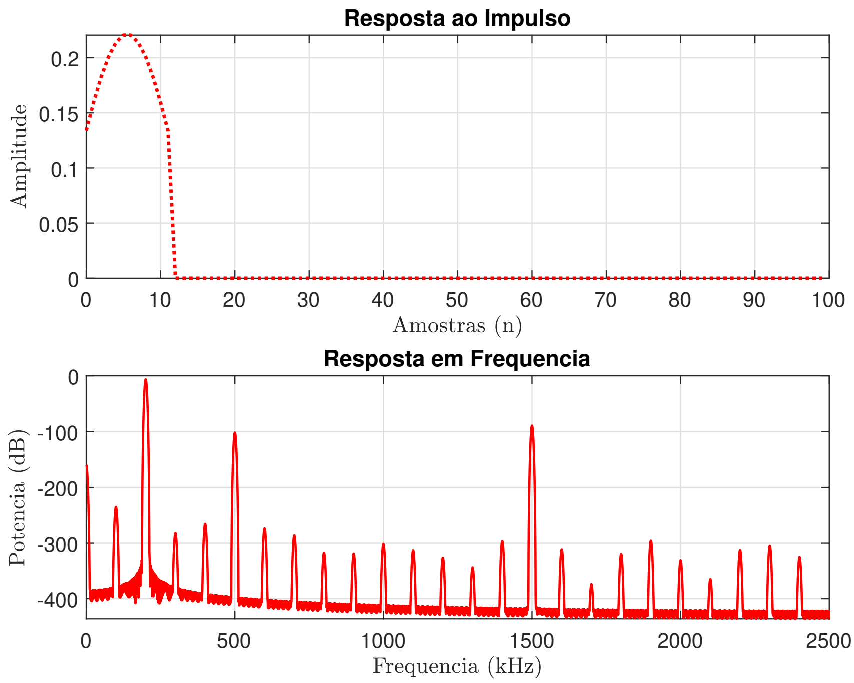


Figura 10: Resposta ao impulso e a resposta em frequência do sistema $h[n]$ projetado.

Problema 4

A questão apresenta o arquivo de áudio “bomdia.wav” como entrada $x[n]$ de um sistema H. O arquivo tem cerca de 1 segundo de duração e consiste numa voz masculina falando e expressão “bom dia”, em português, como pode ser observado no domínio do tempo e frequência (Figura 11).

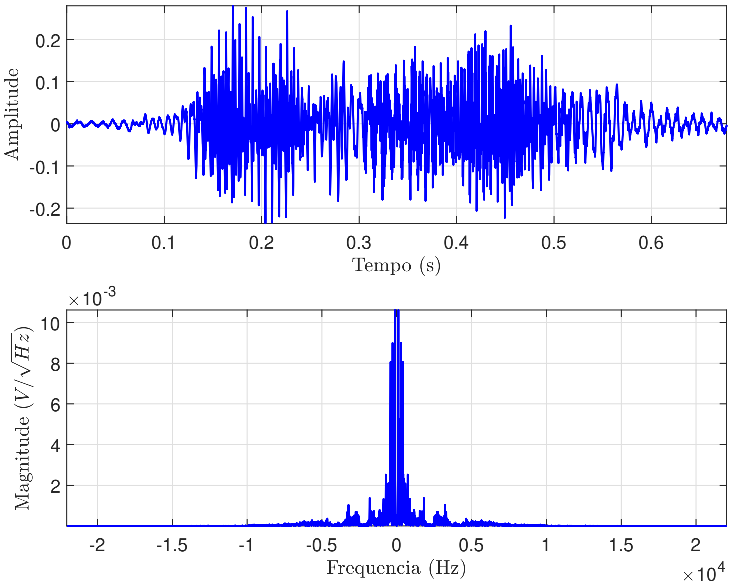


Figura 11: Sinal de entrada representado no domínios do tempo e da frequência.

Também é fornecido o arquivo de áudio “bomdia-reverb.wav” como saída $y[n]$ de um sistema H. O arquivo é semelhante à $x[n]$, porém é perceptível a presença de um efeito que distorce o sinal original, apresentando um tipo de eco ou reverberação. O sinal é apresentado no domínio do tempo e frequência (Figura 12). É importante observar que a transformação impacta no espectro da frequência de modo que realça as componentes da faixa de frequência de 2 à 5 kHz.

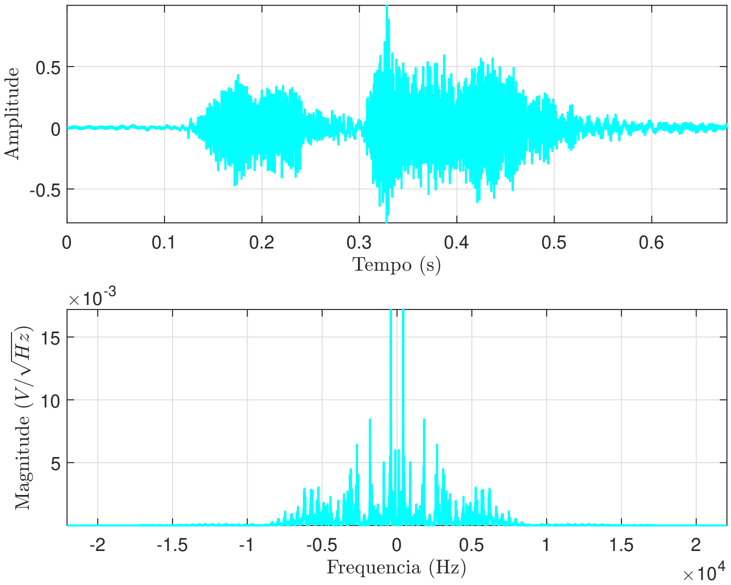


Figura 12: Sinal de saída representado no domínios do tempo e da frequência.

Um terceiro arquivo fornecido, “imp-resp.mat”, é indicado como os coeficientes do sistema $h[n]$. Ele é responsável pela transformação de $x[n]$ em $y[n]$, por via da convolução, apresentado na eqn. 6.

A resposta ao impulso de $h[n]$ é apresentado na figura 13, com um comportamento interessante, são produzidas 3 cópias espaçadas em 100 amostras e com desvanecimento, i.e, amplitude que diminui a medida que o sinal se propaga. Isso é esperado ao reproduzir o áudio de $y[n]$, que apresenta essa espécie de eco.

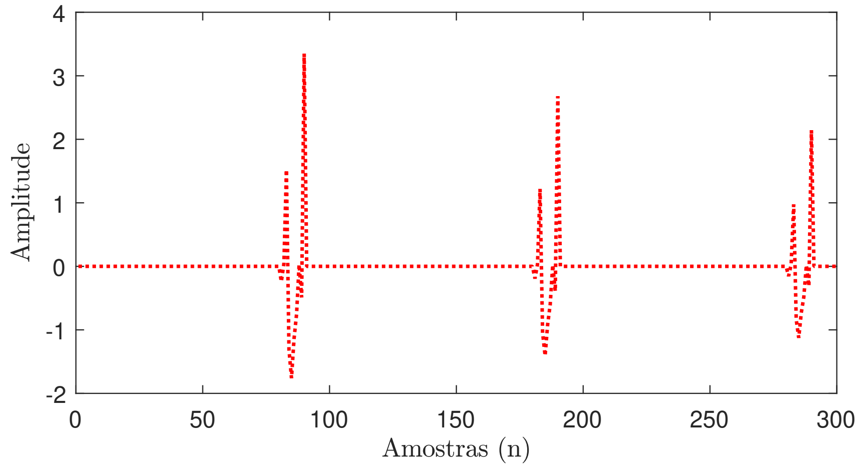


Figura 13: Resposta ao impulso do sistema $h[n]$ fornecido.

Tendo conhecimento de $h[n]$, o procedimento para obter o sinal desejado $\hat{y}[n]$ é o mesmo que o apresentado nos problemas anteriores, convolução entre os coeficientes do filtro e o sinal $x[n]$.

Ao comparar as figuras 12 e 14, é possível observar que os sinais representados são idênticos tanto no domínios do tempo quanto na frequência.

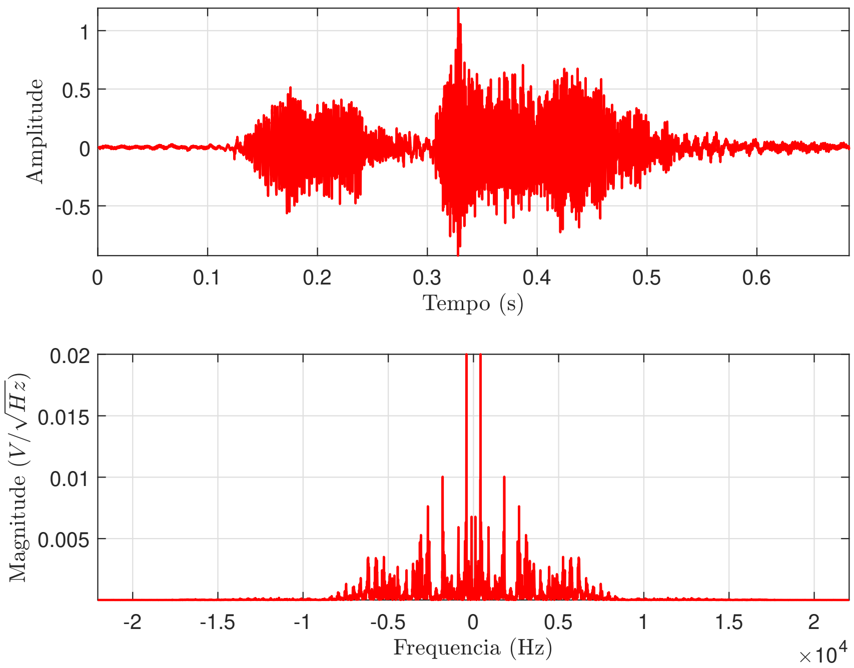


Figura 14: Geração do Sinal $\hat{y}[n]$, saída de $h[n]$ tendo como entrada $x[n]$, “bomdia.wav”.

Problema 5

O sinal original $x[n]$ pode ser estimado/reconstruído como $\hat{x}[n]$ a partir de um sistema $w[n]$ que equaliza o sinal observado $y[n]$. O objetivo do sistema é inverter os efeitos de distorção causados por $h[n]$.

$$\hat{x}[n] = y[n] * w[n] \tag{9}$$

O problema é apresentado no domínio do tempo na eqn. 9, entretanto é realizado no domínio da frequência para se aproveitar do teorema de convolução. É um importante resultado na teoria de processamento de sinais que relaciona a convolução no domínio do tempo com a multiplicação no domínio da frequência. Especificamente, o teorema estabelece que a TF da convolução de duas funções é igual ao produto da TF de cada uma dessas funções [1], [2].

$$\begin{aligned} H(e^{j\omega}) &= \mathcal{F}\{h[n]\} \\ W(e^{j\omega}) &= \frac{1}{H(e^{j\omega})} \\ w[n] &= \mathcal{F}^{-1}\{H(e^{j\omega})^{-1}\} \end{aligned} \tag{10}$$

A eqn. 10 apresenta como obter os coeficientes do equalizador $w[n]$ a partir do conhecimento do comportamento do sistema $h[n]$. Isto permite a aplicação da eqn. 9 e finalmente estimar $\hat{x}[n]$ (Figura 15).

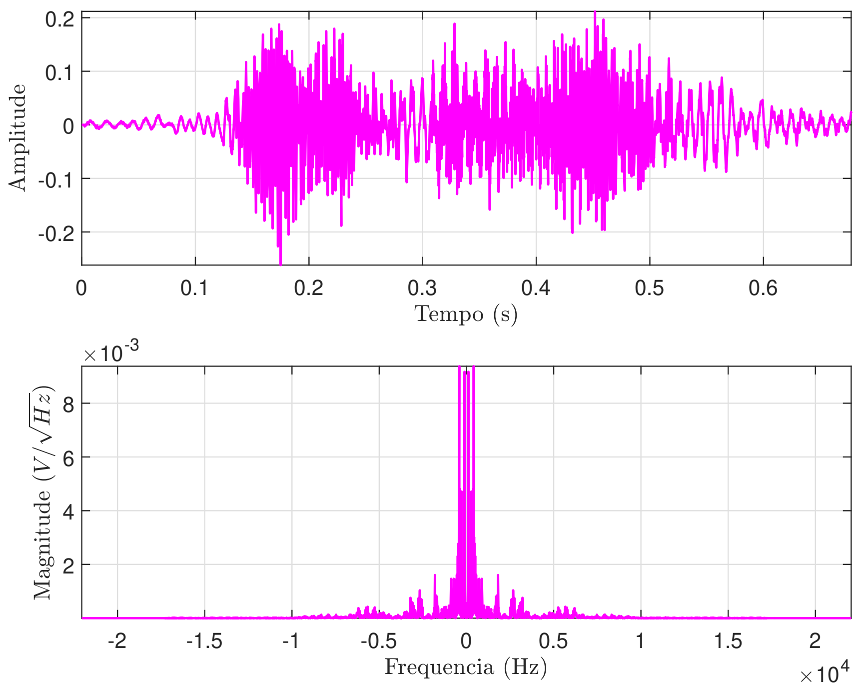


Figura 15: Estimação do Sinal $\hat{x}[n]$, saída de $w[n]$ tendo como entrada $y[n]$, "bomdia-reverb.wav".

Para avaliar a qualidade da estimação $\hat{x}[n]$ comparado ao sinal original $x[n]$ pode-se calcular uma métrica quantitativa chamada **erro de aproximação** (e), que consiste em obter o quadrado da diferença amostra a amostra, i.e, calcular a norma/distância entre o ponto obtido na estimação e o valor original:

$$e = \sum_{n=1}^{N} ||x[n] - \hat{x}[n]||^2 \tag{11}$$

A figura 16 apresenta o resultado para todas as amostras. O erro médio da ordem de 7×10^{-3}, o que implicaria que os erros no geral afetam até a 2ª casa decimal (0.0X). O valor pontos de máximo de erro não ultrapassa 0.15.

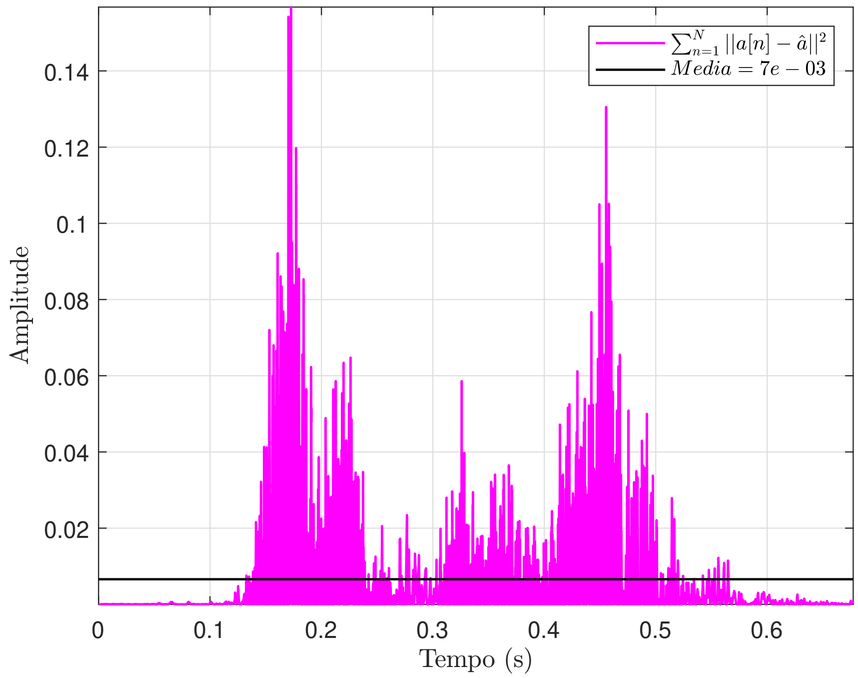


Figura 16: Erro de aproximação.

Problema 6

A questão apresenta o arquivo de áudio "preamble.wav" como entrada $x[n]$ de um sistema H. O arquivo tem cerca de 1 segundo de duração e consiste numa voz masculina falando um trecho da constituição estadunidense "We the People of the United States, in Order to form a more perfect Union, establish Justice, insure domestic Tranquility, provide for the common defense", em inglês, pode ser observado no domínio do tempo e frequência (Figura 17).

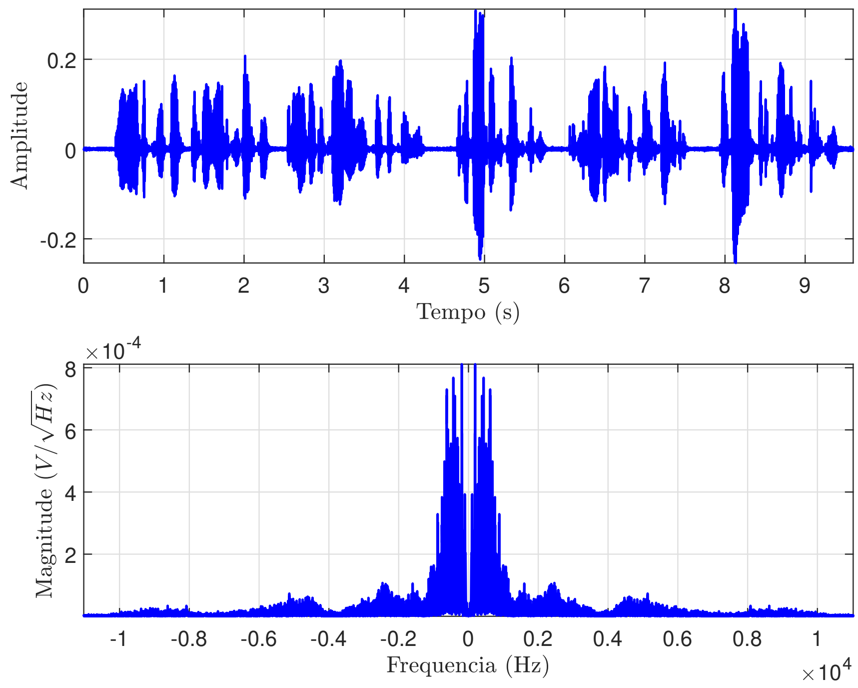


Figura 17: Sinal de entrada representado no domínios do tempo e da frequência.

Também é fornecido o arquivo de áudio "preamble-reverb.wav" como saída $y[n]$ de um sistema H. O arquivo é semelhante à $x[n]$, porém é perceptível a presença de um efeito que distorce o sinal original, como nos problemas anteriores. O sinal é apresentado no domínio do tempo e frequência (Figura 18). É importante observar que a transformação impacta no espectro da frequência de modo que realça as componentes da faixa de frequência de 2 à 5 kHz.

A abordagem para estimar o sinal original através do equalizador é o mesmo do apresentado no Problema 5, que utiliza o teorema de convolução. A eqn. 10 apresenta como obter os coeficientes do equalizador $w[n]$ a partir do conhecimento do comportamento do sistema $h[n]$. Isto novamente permite a aplicação da eqn. 9 e finalmente estimar $\hat{x}[n]$ (Figura 19).

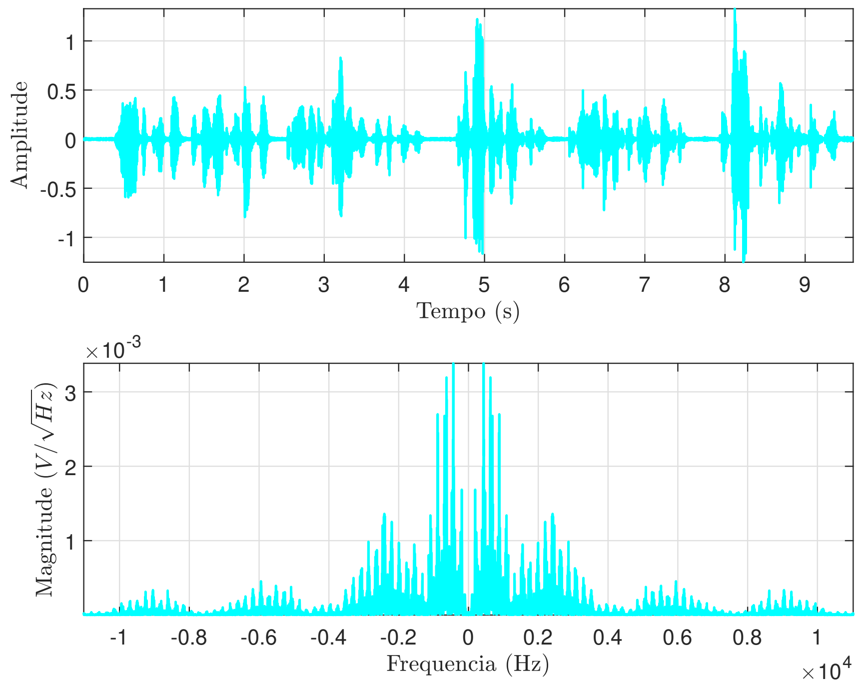


Figura 18: Sinal de saída representado no domínios do tempo e da frequência.

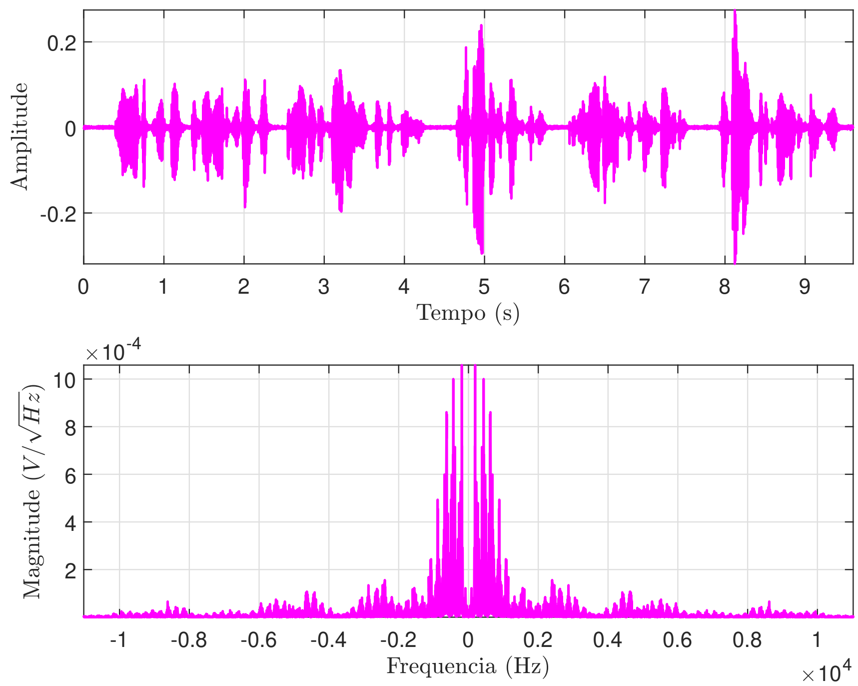


Figura 19: Estimação do Sinal $\hat{x}[n]$, saída de $w[n]$ tendo como entrada $y[n]$, “preamble-reverb.wav”.

A figura 20 apresenta o resultado para todas as amostras. O erro médio da ordem de 1×10^{-3}, o que implicaria que os erros no geral afetam até a $3^{\underline{a}}$ casa decimal (0.00X). O valor pontos de máximo de erro não ultrapassa 0.10.

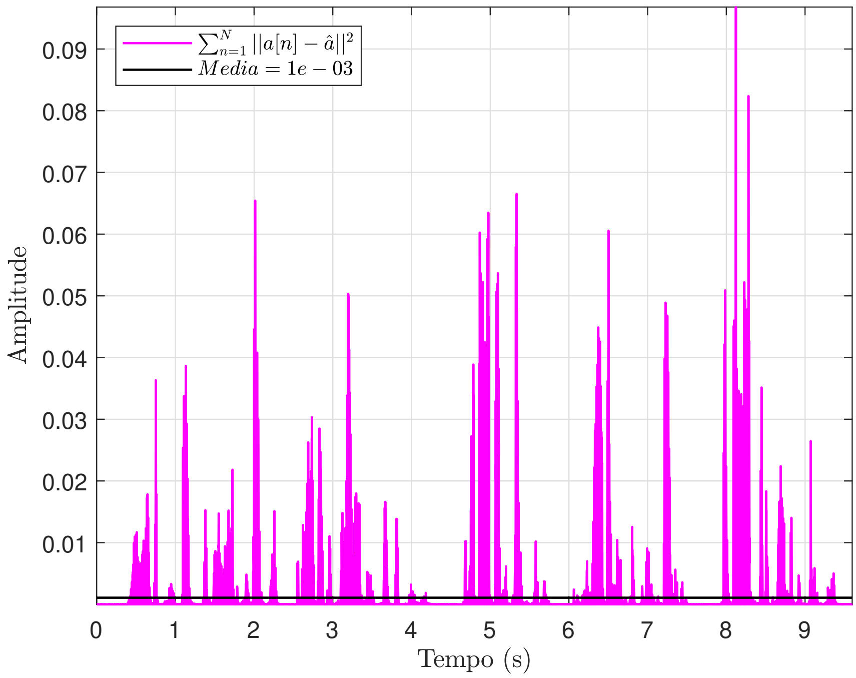


Figura 20: Erro de aproximação.

Referências

[1] A. V. Oppenheim, A. S. Willsky e S. H. Nawab, *Signals & Systems (2nd Ed.)* USA: Prentice-Hall, Inc., 1996.

[2] A. V. Oppenheim, R. W. Schafer e J. R. Buck, *Discrete-Time Signal Processing (2nd Ed.)* USA: Prentice-Hall, Inc., 1999.

Apêndice

Apêndice A - Exercícios Propostos

Exercício Computacional II

Processamento Digital de Sinais

Engenharia de Telecomunicações

INSTRUÇÕES

- Os resultados dos exercícios computacionais devem ser enviados para o instrutor de apoio da disciplina na forma de relatórios contendo eventuais gráficos acompanhados de uma breve discussão.
- Durante as aulas haverá espaço reservado para discussão dos resultados e esclarecimentos de dúvidas.

Problema 1 Carregue o arquivo "bipsIN.wav" no Matlab. Plote o sinal no domínio do tempo e da frequência. Conclua sobre o sinal você está analisando. Detalhe sua resposta. Forneça a expressão para o sinal $x[n]$ em questão, bem como a expressão para sua transformada de Fourier $X(e^{j\omega})$.

Dica: Para plotar corretamente o sinal nos nois domínios, você deverá definir o suporte de tempo e da frequência em função da taxa de amostragem do sinal (que é fornecida com parâmetro de saída da função "audioread" que você irá utilizar. Sabendo que a taxa de amostragem é F_s, seu suporte temporal deverá ter um espaçamento $T = 1/F_s$ entre as amostras. O mesmo vale para o suporte frequencial.

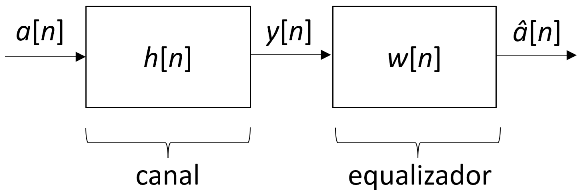
Problema 2 Carregue agora o arquivo "bipsOUT.wav" no Matlab. Plote o sinal no domínio do tempo e da frequência. O que é possível concluir sobre este sinal, em comparação com o sinal do problema anterior? Sabendo que o sinal de saída "bipsOUT.wav"foi produzido a partir do sinal de entrada "bipsIN.wav" através de uma transformação por um sistema LIT, o que é possível concluir sobre este sistema, em particular sobre sua resposta em frequência? Implemente um possível sistema LTI que seja capaz de gerar o mesmo sinal "bipsOUT.wav" (ou uma aproximação deste sinal) a partir do sinal "bipsIN.wav". Plote a resposta ao impulso e a resposta em frequência do sistema eque você escolheu.

Problema 3 Repita os passos dos problemas 1 e 2, utilizando agora os sinais de áudio "bipsIN-mixed.wav" and "bipsOUT-mixed.wav'. O que é possível concluir comparando os novos resultados com os anteriores?

Problema 4 Carregue o arquivo "bomdia.wav" e "bomdia-reverb.wav" no Matlab. Plote ambos os sinais no domínio do tempo e da frequência. Escute os áudios dos dois sinais. O que você pode concluir através da comparação dos

gráficos tempo/frequência de ambos os sinais, bem como sobre a qualidade de áudio dos mesmos?
Carregue agora o arquivo "imp-resp.mat" e plote sua resposta ao impulso $h[n]$. Gere o sinal de saída deste sistema assumindo como entrada o sinal "bomdia.wav". Compare o sinal obtido com sinal "bomdia-reverb.wav". O que se pode concluir?

Problema 5 A figura abaixo ilustra uma possível solução para tentar restaurar o sinal original "bomdia.wav" a partir do sinal "bomdia-reverb.wav" (que é uma versão distorcida do sinal original) usando um sistema denominado "equalizador".



Considerando $a[n]$ como o sinal original ("bomdia.wav") e $y[n]$ como o sinal distorcido ("bomdia-reverb.wav"), determine uma solução para a resposta ao impulso do sistema equalizador $w[n]$. Em seguida, obtenha o sinal "equalizado" $\hat{a}[n]$ e compare-o com o sinal original $a[n]$. O que se pode concluir? Calcule ainda o erro de aproximação, definido por $erro = \sum_{n=1}^{N} \|a[n] - \hat{a}[n]\|^2$.

Problema 6 Carregue o arquivo "preamble.wav" no Matlab. Plote sua resposta ao impulso e sua resposta em frequência. Utilizando a resposta ao impulso "imp-resp.mat" que modela o sistema distorcivo $h[n]$ da figura acima, obtenha o sinal de saída $y[n]$. Grave este sinal como "preamble-reverb.wav". Em seguida, seguindo os passos da figura, projete o equalizador $w[n]$ e obtenha o sinal equalizado. Plote a resposta em frequência deste sinal e verifique o resultado do áudio resultante. Discuta sobre os resultados obtidos.

Apêndice B - Código

TIP7200 - Processamento Digital de Sinais - HW 2

Table of Contents

Author: Lucas Abdalah

script.m

2023/04/22 - v1

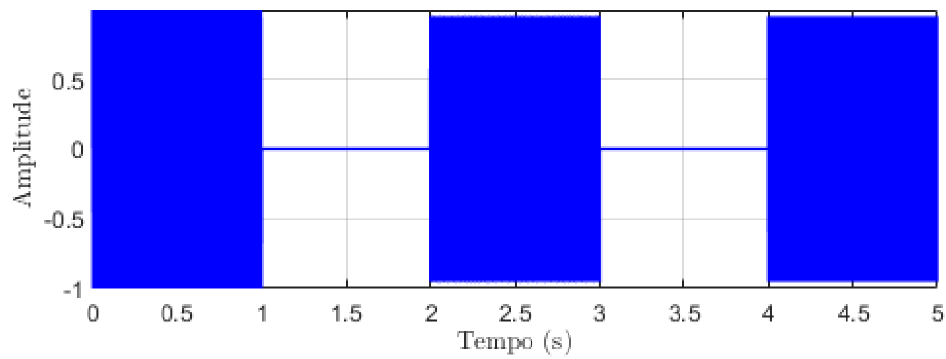
Preamble

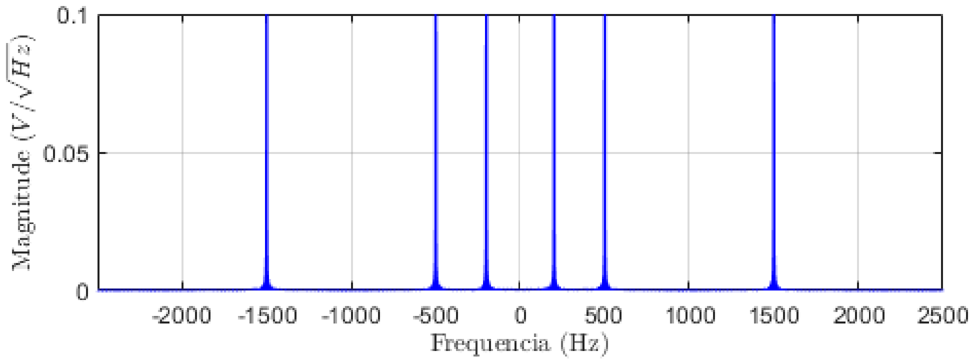
```
close all;
clearvars;
% clc;
pause(0.1);
% pause off;

savefigPath = '..\figures\';
color_ = struct('X', 'blue', ...
  'Y', 'cyan', ...
  'Yfilt', 'red', ...
  'Xfilt', 'magenta', ...
  'Ex2', [0.3010 0.7450 0.9330], ...
  'Ex3', [0.6350 0.0780 0.1840]);
```

Problem 1

```
fileName = 'bipsIN';
fprintf('Problema 1 - %s \n\n', fileName);
[x, Fs] = audioread(['data\', fileName, '.wav']);
% Time and Frequencia Domain Analysis
hw2p1fig1 = plot_signal(x, Fs, 'shifted', fileName, color_.X);
```

```
Problema 1 - bipsIN
```

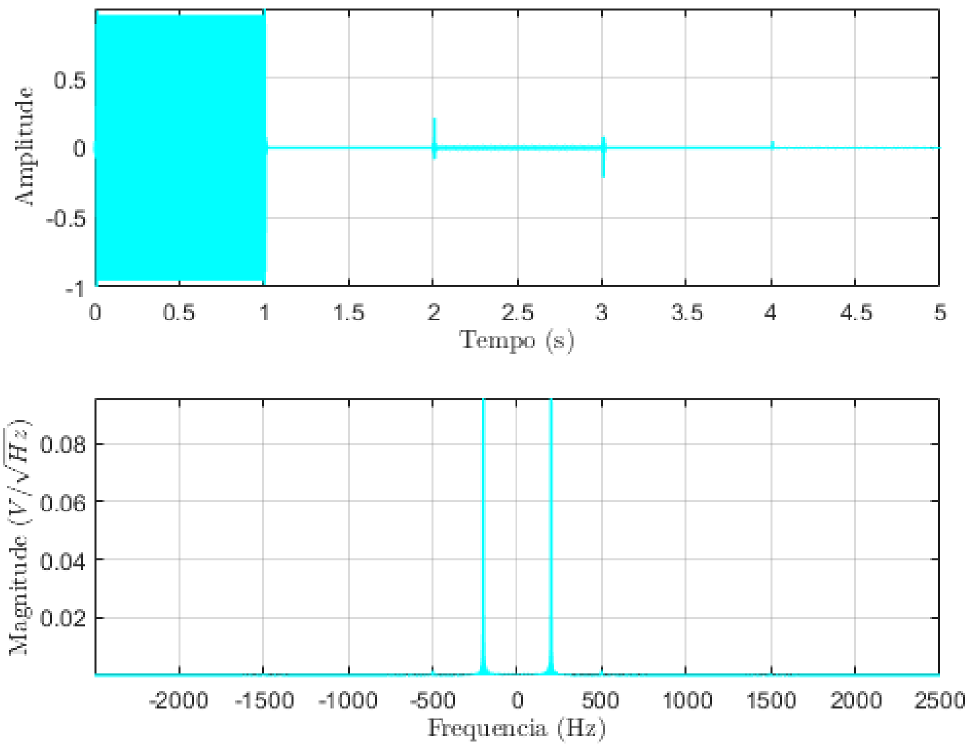


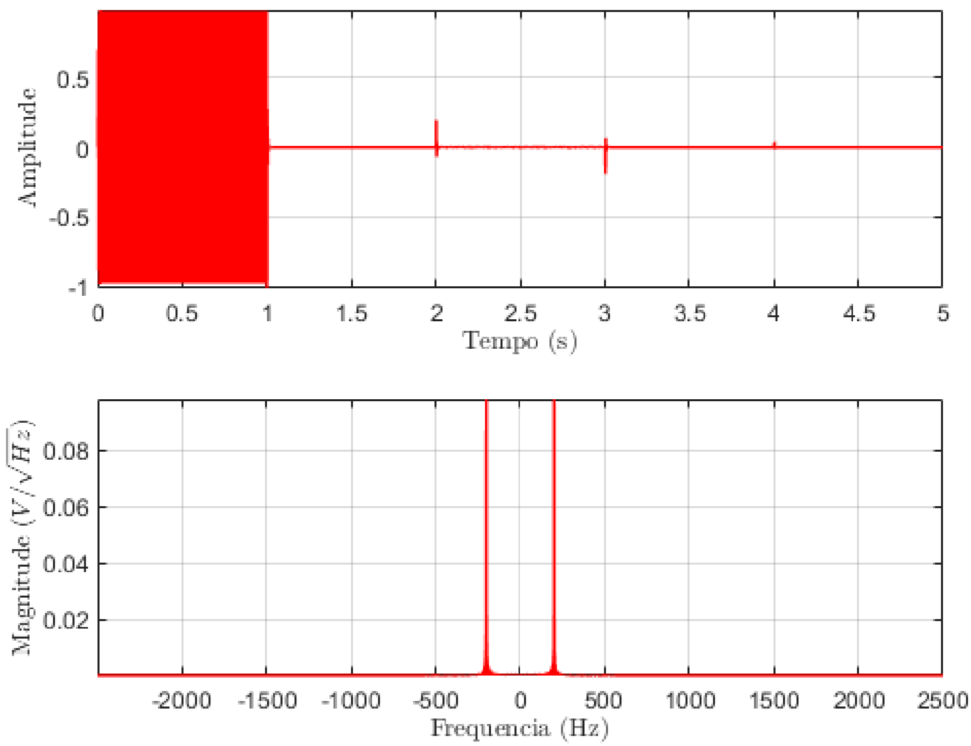


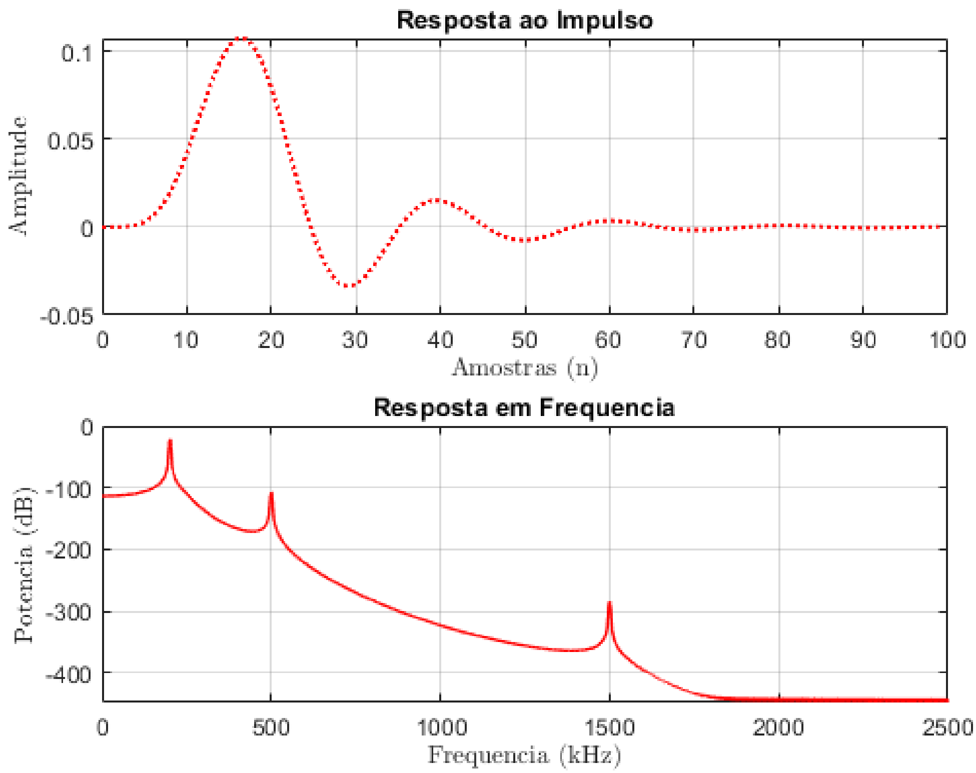
Problem 2

```
fileName = 'bipsOUT';
fprintf('Problema 2 - %s \n\n', fileName)
[y, Fs] = audioread(['data\', fileName, '.wav']);
% Time and Frequencia Domain Analysis
hw2p2fig1 = plot_signal(y, Fs, 'shifted', fileName, color_.Y);

% Low Pass Filter (LPF) - Butterworth
Fcutoff = 250;
Fc_norm = Fcutoff/(Fs/2);
[b,a] = butter(7, Fc_norm);
yfilt = filter(b,a,x);

% Time and Frequencia Domain Analysis
hw2p2fig2 = plot_signal(yfilt, Fs, 'shifted', 'Filter Output',
 color_.Yfilt);
hw2p2fig3 = figure('name', 'Resposta ao Impulso e em Frequencia');
subplot(2,1,1);
plot_impz(b,a, 'Resposta ao Impulso', color_.Yfilt)
title('Resposta ao Impulso')
subplot(2,1,2)
plot_pspectrum(yfilt, Fs, 'Resposta em Frequencia', color_.Yfilt)
title('Resposta em Frequencia')
```

Problema 2 - bipsOUT


Amplitude
0.5
0
-0.5
-1
0
0.5
1
1.5
2
2.5
3
3.5
4
4.5
5
Tempo (s)
Magnitude $(V/\sqrt{Hz})$
0.08
0.06
0.04
0.02
-2000
-1500
-1000
-500
0
500
1000
1500
2000
2500
Frequencia (Hz)


Amplitude
Tempo (s)
Magnitude $(V/\sqrt{Hz})$
Frequencia (Hz)


Resposta ao Impulso
Amplitude
Amostras (n)
Resposta em Frequencia
Potencia (dB)
Frequencia (kHz)

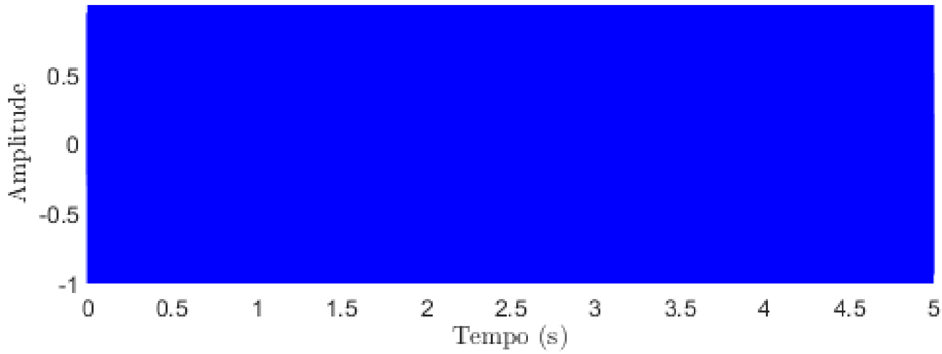
Problem 3

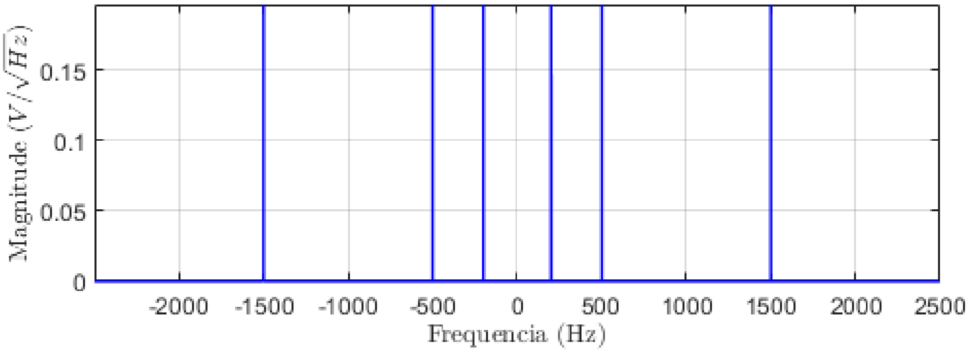
Input

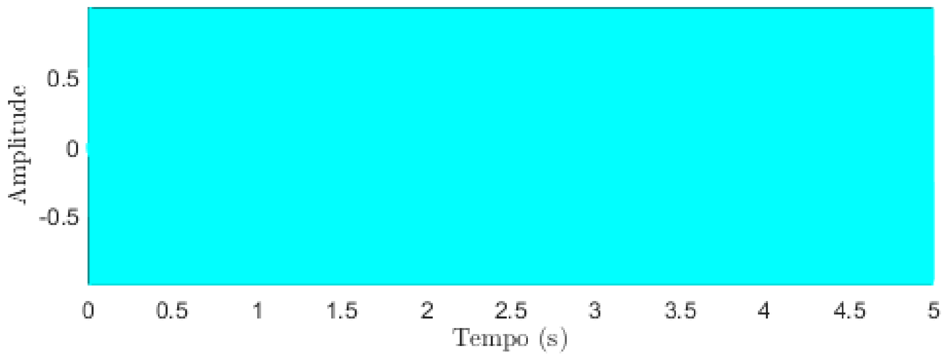
```
fileName = 'bipsIN_mixed';
fprintf('Problema 3 - %s \n\n', fileName)
[x, Fs] = audioread(['data\', fileName, '.wav']);
% Input - Time and Frequencia Domain Analysis
hw2p3fig1 = plot_signal(x, Fs, 'shifted', fileName, color_.X);

% Output - Time and Frequencia Domain Analysis
fileName = 'bipsOUT_mixed';
[y, Fs] = audioread(['data\', fileName, '.wav']);
fprintf('- %s \n\n', fileName)
hw2p3fig2 = plot_signal(y, Fs, 'shifted', fileName, color_.Y);
hw2p3fig3 = figure('name', 'Resposta em Frequencia');
plot_pspectrum(y, Fs, 'Resposta em Frequencia', color_.Y)
pbaspect([2 1 1]);

% Low Pass Filter (LPF) - Butterworth
Fcutoff = 450;
order = 11;
b = fir1(order, Fcutoff/Fs, 'low', kaiser(order
+1)); % ,chebwin(35,30))
a = Fcutoff*1e-3;
yfilt = filtfilt(b, a, x);

% Time and Frequencia Domain Analysis
hw2p3fig4 = plot_signal(yfilt, Fs, 'shifted', 'Filter Output',
 color_.Yfilt);
hw2p3fig5 = figure('name', 'Resposta ao Impulso e em Frequencia');
subplot(2,1,1);
plot_impz(b, a, 'Resposta ao Impulso', color_.Yfilt)
subplot(2,1,2)
plot_pspectrum(yfilt, Fs, 'Resposta em Frequencia', color_.Yfilt)
```

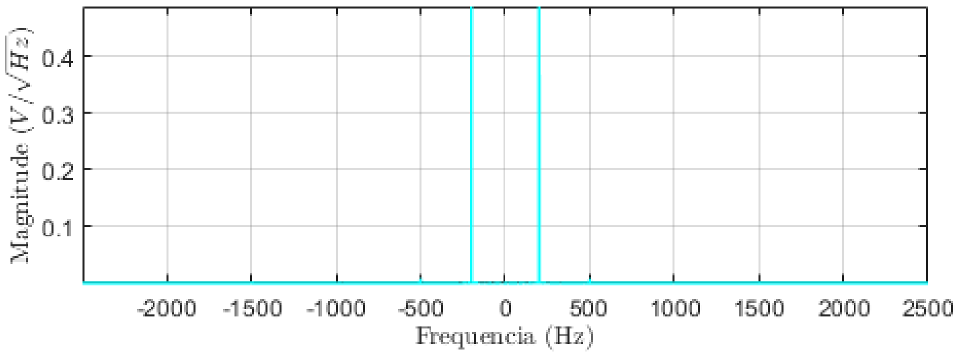
Problema 3 - bipsIN_mixed

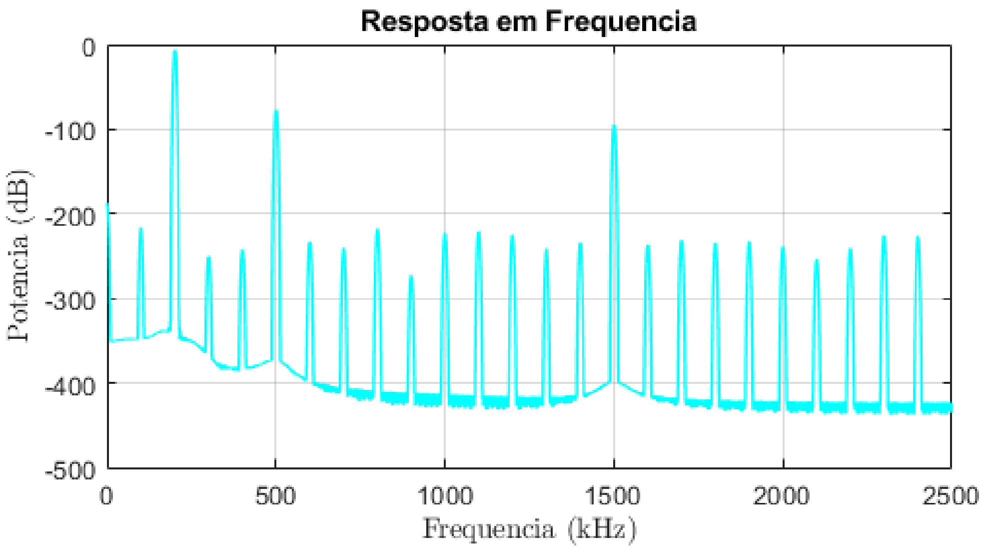
- bipsOUT_mixed

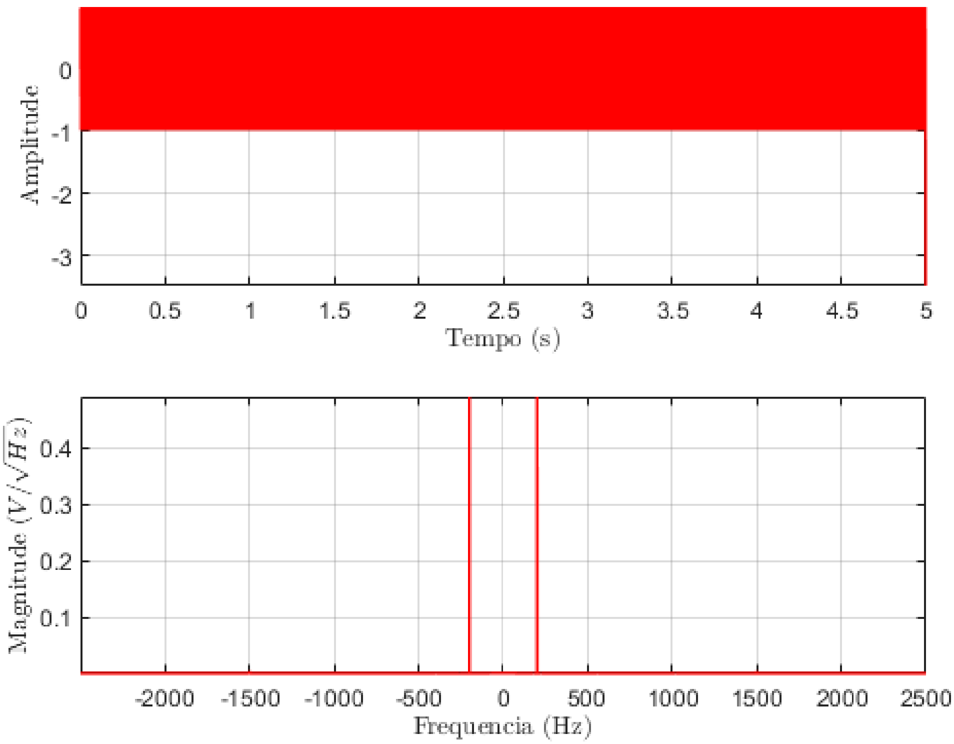

Amplitude
0.5
0
-0.5
-1
0 0.5 1 1.5 2 2.5 3 3.5 4 4.5 5
Tempo (s)

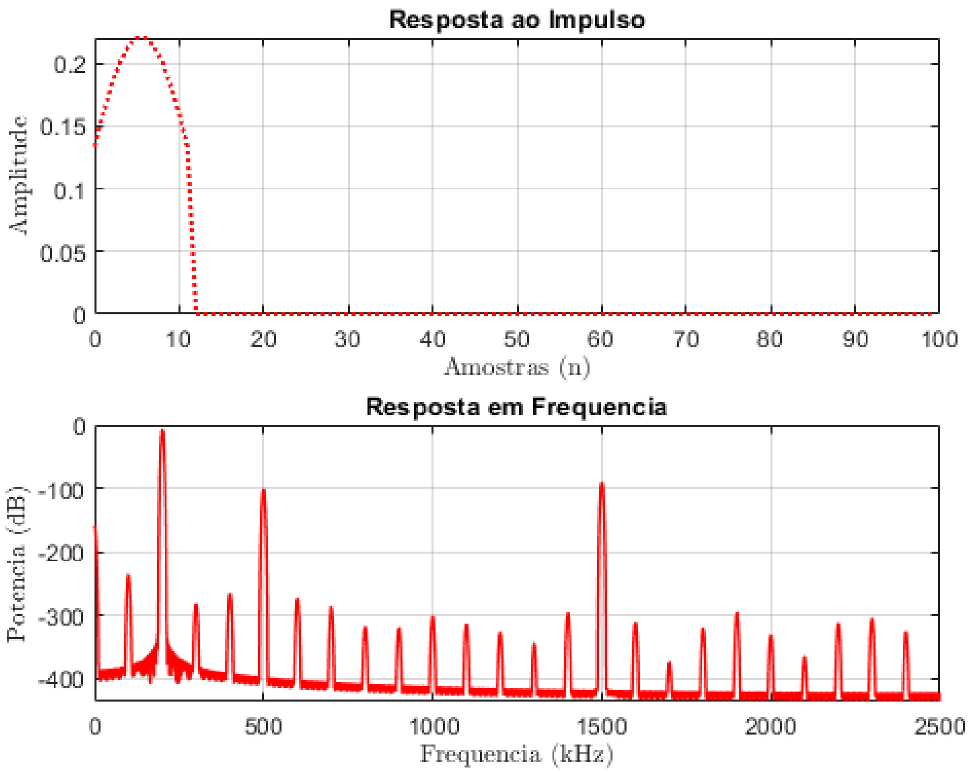

Magnitude $(V/\sqrt{Hz})$
0.15
0.1
0.05
0
-2000 -1500 -1000 -500 0 500 1000 1500 2000 2500
Frequencia (Hz)


Amplitude
0.5
0
-0.5
0 0.5 1 1.5 2 2.5 3 3.5 4 4.5 5
Tempo (s)


Magnitude $(V/\sqrt{Hz})$
0.4
0.3
0.2
0.1
-2000 -1500 -1000 -500 0 500 1000 1500 2000 2500
Frequencia (Hz)


Resposta em Frequencia
0
-100
-200
-300
-400
-500
Potencia (dB)
0
500
1000
1500
2000
2500
Frequencia (kHz)


0
-1
-2
-3
Amplitude
0
0.5
1
1.5
2
2.5
3
3.5
4
4.5
5
Tempo (s)
0.4
0.3
0.2
0.1
Magnitude $(V/\sqrt{Hz})$
-2000
-1500
-1000
-500
0
500
1000
1500
2000
2500
Frequencia (Hz)



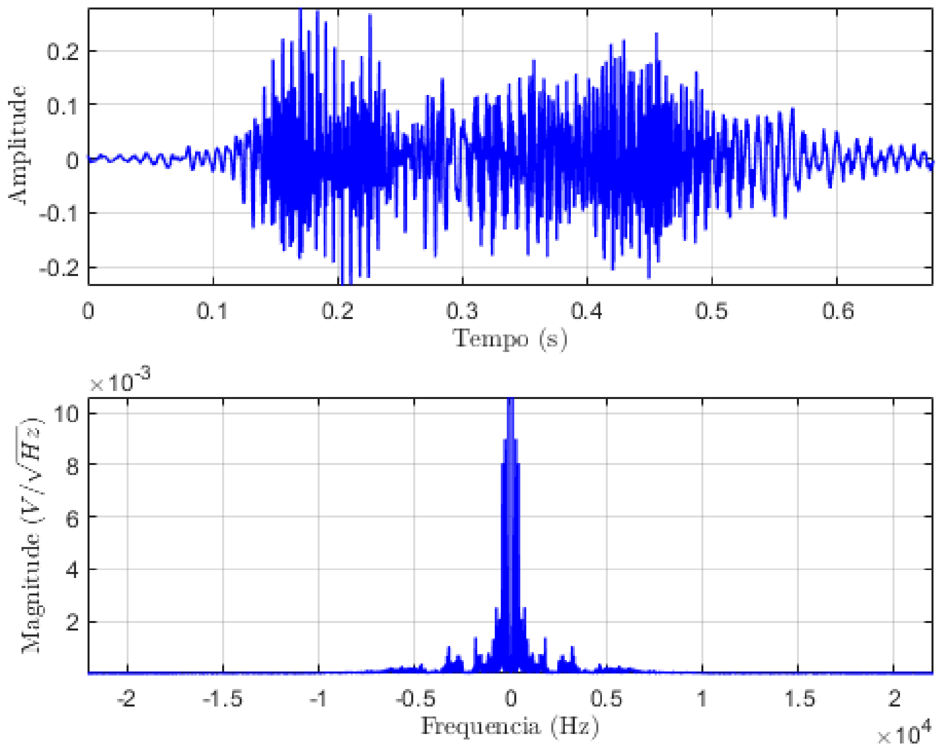
Problem 4

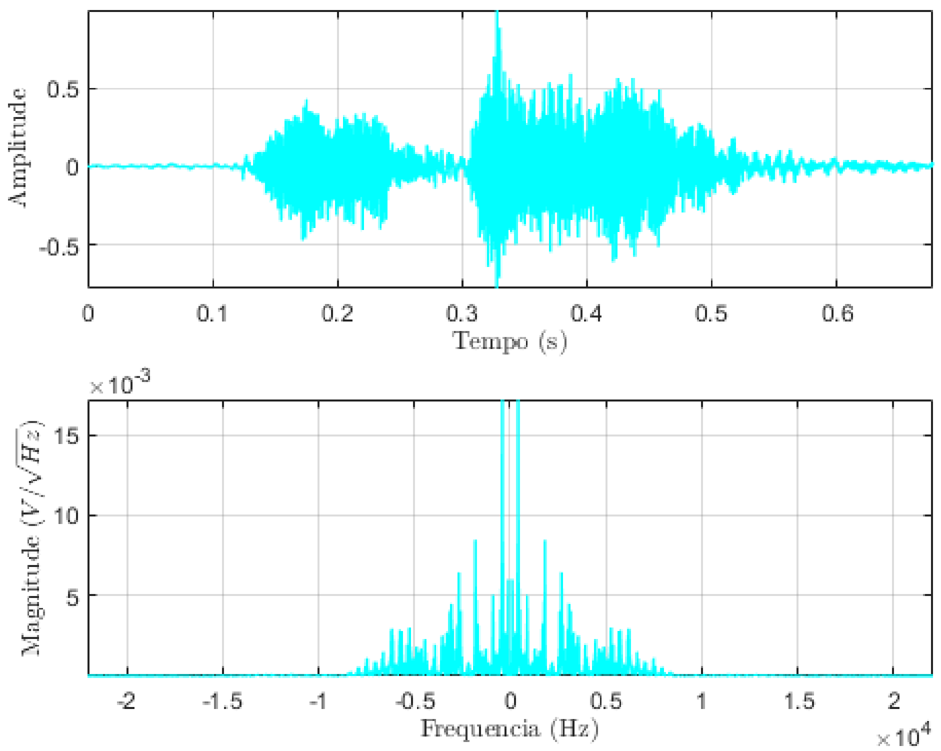
Input audio

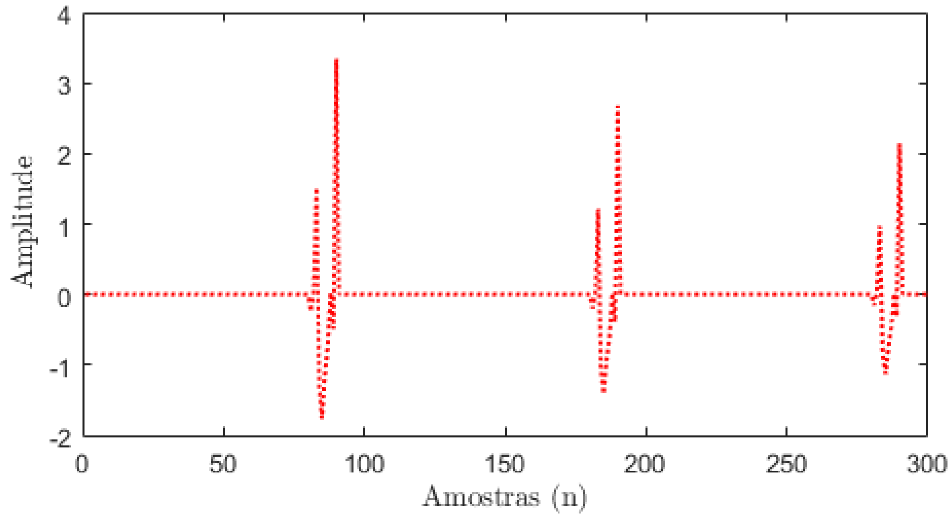
```
fileName = 'bomdia';
fprintf('Problema 4 - %s \n\n', fileName);
[x, Fs] = audioread(['data\', fileName, '.wav']);
% Time and Frequencia Domain Analysis
hw2p4fig1 = plot_signal(x, Fs, 'shifted', fileName, color_.X);

% Output audio
fileName = 'bomdia_reverb';
[y, Fs] = audioread(['data\', fileName, '.wav']);
% Time and Frequencia Domain Analysis
hw2p4fig2 = plot_signal(y, Fs, 'shifted', fileName, color_.Y);

% Impulsive response
fileName = 'imp_resp';
reverb = load('data/imp_resp.mat');
hw2p4fig3 = figure('name', 'Resposta ao Impulso');
plot(reverb.h, 'Color', color_.Yfilt, 'LineStyle', ':', 'LineWidth',
 1.5);
% title('Resposta ao Impulso')
xlabel('Amostras (n)', 'interpreter', 'Latex');
ylabel('Amplitude', 'interpreter', 'Latex');
pbaspect([2 1 1]);
```

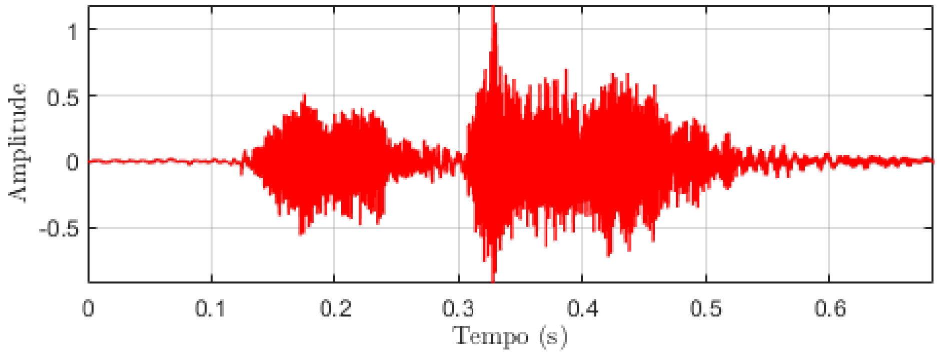
```
% Convolve input x and system h
yfilt = conv(x, reverb.h);
hw2p4fig4 = plot_signal(yfilt, Fs, 'shifted', [fileName, '
 Convolution'], color_.Yfilt);
```

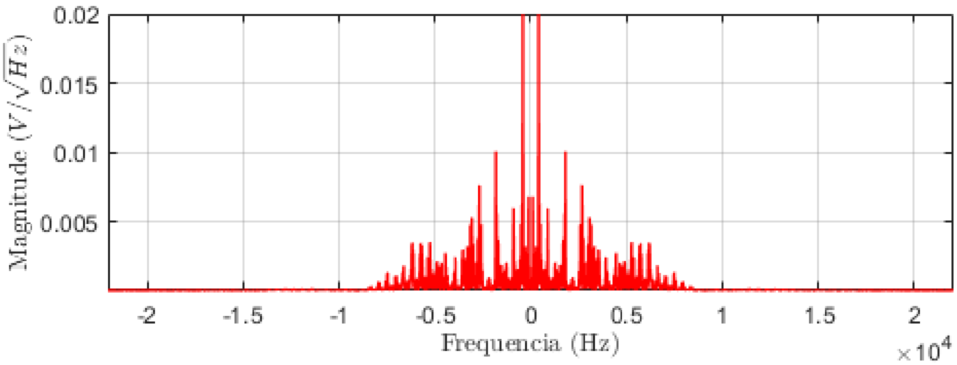
Problema 4 - bomdia




Amplitude
0.5
0
-0.5
0
0.1
0.2
0.3
0.4
0.5
0.6
Tempo (s)
×10^-3
Magnitude (V/√Hz)
15
10
5
-2
-1.5
-1
-0.5
0
0.5
1
1.5
2
Frequencia (Hz)
×10^4


4
3
2
1
0
-1
-2
Amplitude
0
50
100
150
200
250
300
Amostras (n)



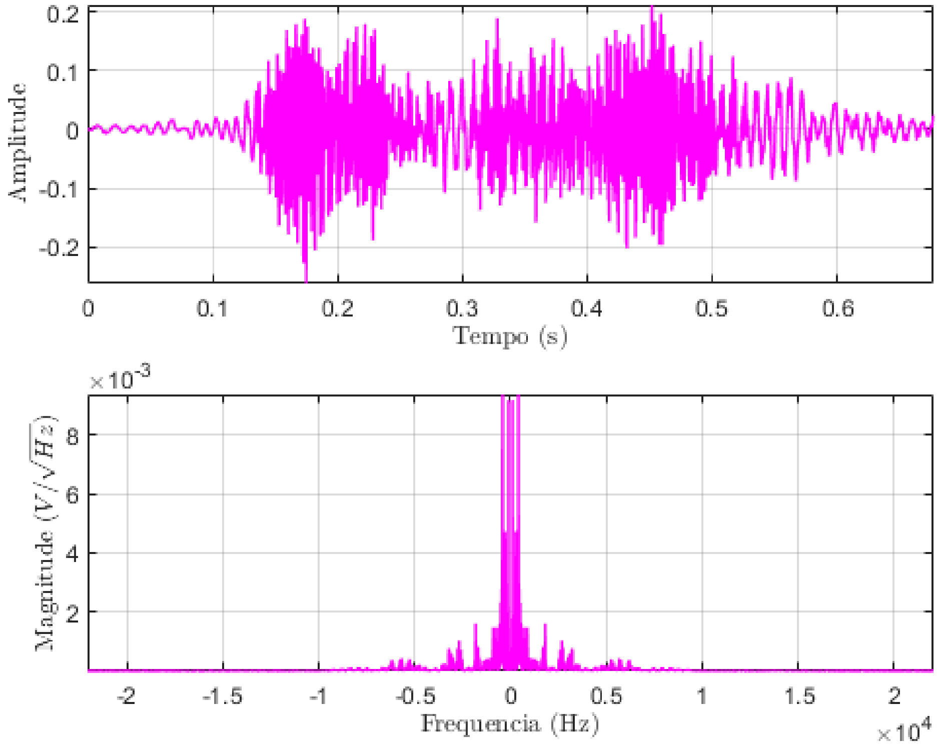


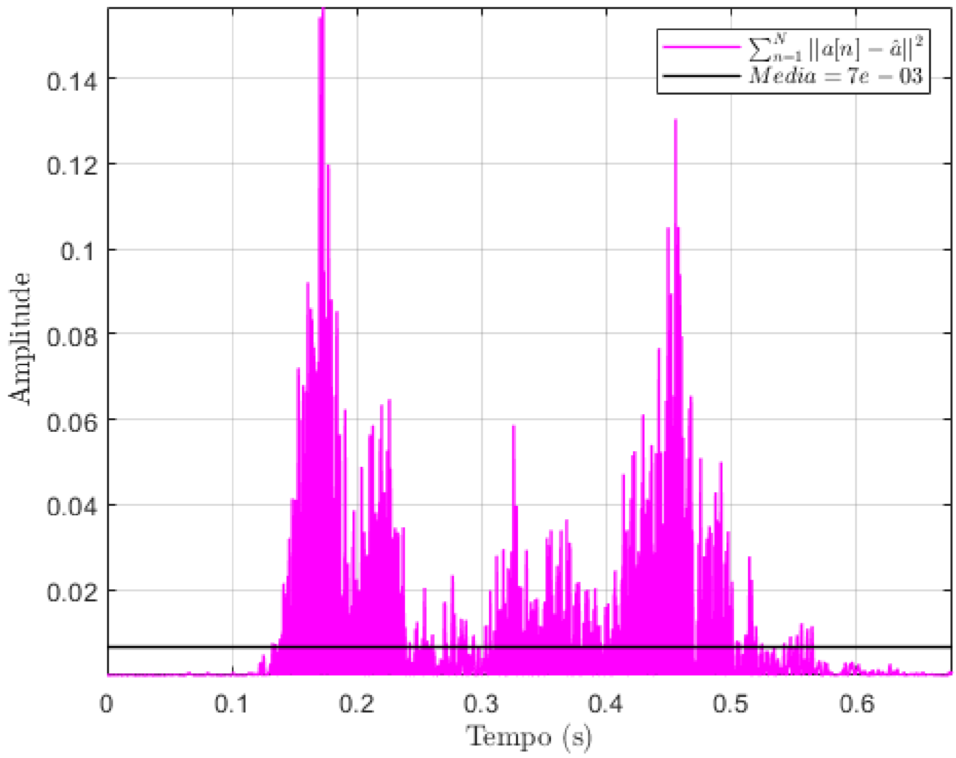
Problem 5

Equalizer

```
fprintf('Problema 5\n\n');
xfilt = retrieve(y, reverb.h);
audiowrite('..\audio\bomdia_restored.wav', xfilt, Fs)
hw2p5fig1 = plot_signal(xfilt, Fs, 'shifted', [fileName, '
 Equalized'], color_.Xfilt);

% Error
hw2p5fig2 = figure('name', 'Squared Error ');
sqerror = (x - xfilt).^2;
plot_time(sqerror, Fs, [], color_.Xfilt);
hold on
sqerror_mean = repelem(mean(sqerror), length(sqerror));
plot_time(sqerror_mean, Fs, [], []); % Error
hold off;
legend('$\sum_{n=1}^{N} || a[n] - \hat{a}||^2$', ...
sprintf('$Media = %1.0e$', mean(sqerror)), 'interpreter', ...
  'Latex', 'Location', 'northeast','Orientation', 'Vertical');
axis tight
%}
```

Problema 5


Amplitude
Tempo (s)
$\times 10^{-3}$
Magnitude $(V/\sqrt{Hz})$
Frequencia (Hz)
$\times 10^{4}$


$\sum_{n=1}^{N} ||a[n] - \hat{a}||^2$
$Media = 7e - 03$
Amplitude
Tempo (s)

Problem 6

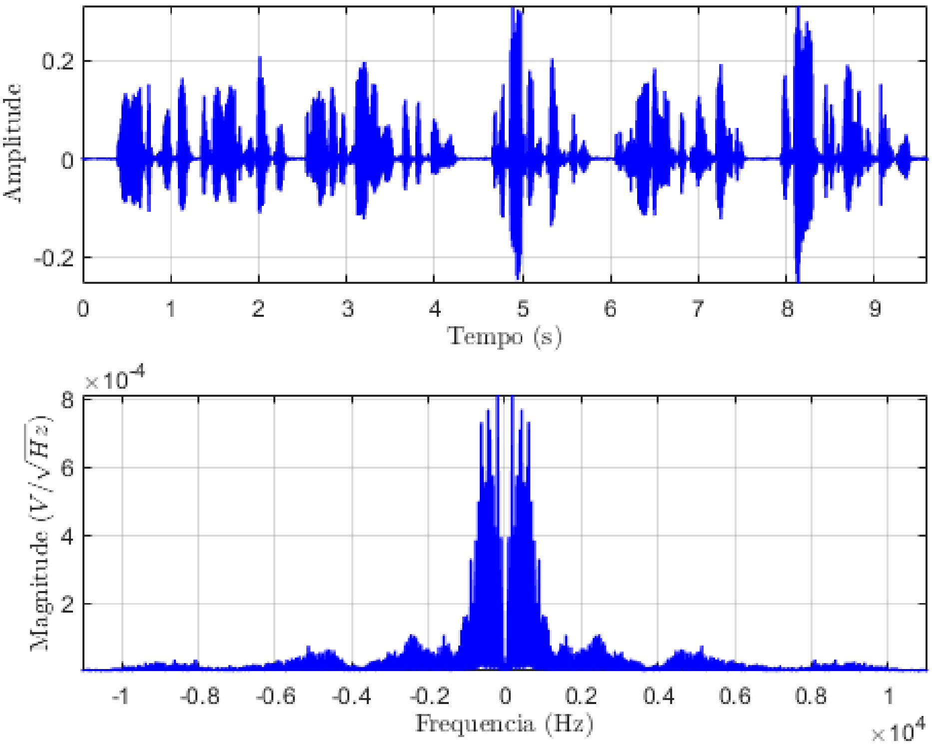
Input audio

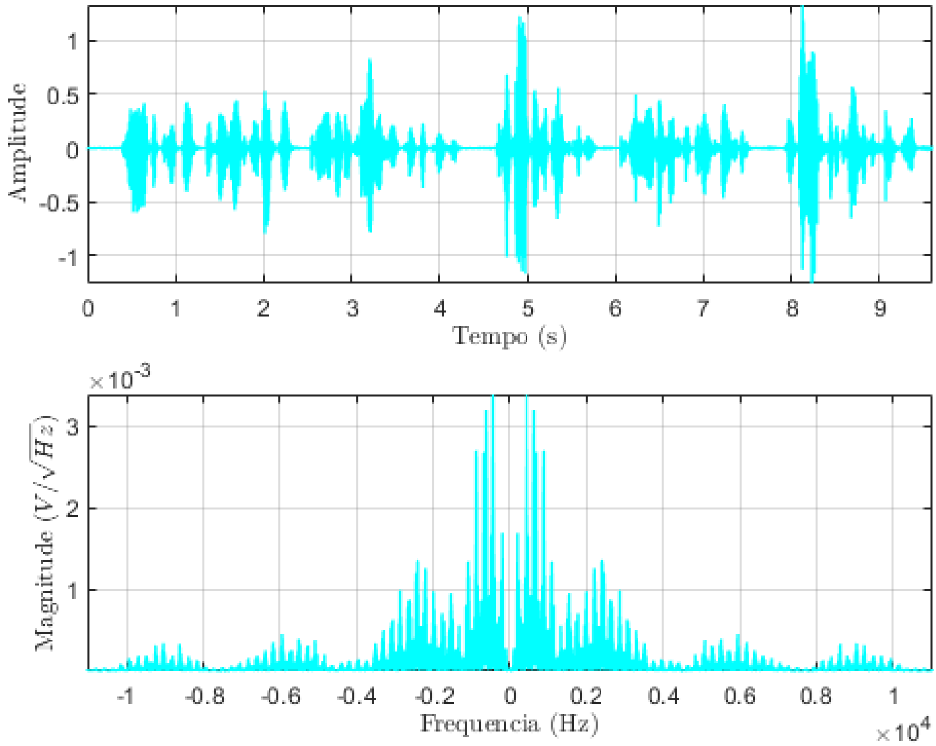
```
reverb = load('data/imp_resp.mat');
fileName = 'preamble';
fprintf('Problema 6 - %s \n\n', fileName);
[x, Fs] = audioread(['data\', fileName, '.wav']);
% Time and Frequencia Domain Analysis
hw2p6fig1 = plot_signal(x, Fs, 'shifted', fileName, color_.X);

% Convolve input x and system h, export y
y = conv(x, reverb.h, 'same');
audiowrite('..\audio\preamble_reverb.wav', y, Fs)
hw2p6fig2 = plot_signal(y, Fs, 'shifted', [fileName, ' Convolution'],
 color_.Y);

% Equalizer
fprintf('Equalizer\n\n');
xfilt = retrieve(y, reverb.h);
audiowrite('..\audio\preamble_restored.wav', xfilt, Fs)
hw2p6fig3 = plot_signal(xfilt, Fs, 'shifted', [fileName, '
 Equalized'], color_.Xfilt);

% Error
hw2p6fig4 = figure('name', 'Squared Error ');
sqerror = (x - xfilt).^2;
plot_time(sqerror, Fs, [], color_.Xfilt);
hold on
sqerror_mean = repelem(mean(sqerror), length(sqerror));
plot_time(sqerror_mean, Fs, [], []); % Error
hold off;
legend('$\sum_{n=1}^{N} || a[n] - \hat{a}||^2$', ...
  sprintf('$Media = %1.0e$', mean(sqerror)), 'interpreter', ...
  'Latex', 'Location', 'northwest','Orientation', 'Vertical');
axis tight


% Export figures as eps
%{
saveaseps(hw2p1fig1, 'hw2p1fig1', savefigPath)
saveaseps(hw2p2fig1, 'hw2p2fig1', savefigPath)
saveaseps(hw2p2fig2, 'hw2p2fig2', savefigPath)
saveaseps(hw2p2fig3, 'hw2p2fig3', savefigPath)
saveaseps(hw2p3fig1, 'hw2p3fig1', savefigPath)
saveaseps(hw2p3fig2, 'hw2p3fig2', savefigPath)
saveaseps(hw2p3fig3, 'hw2p3fig3', savefigPath)
saveaseps(hw2p3fig4, 'hw2p3fig4', savefigPath)
saveaseps(hw2p3fig5, 'hw2p3fig5', savefigPath)
saveaseps(hw2p4fig1, 'hw2p4fig1', savefigPath)
saveaseps(hw2p4fig2, 'hw2p4fig2', savefigPath)
saveaseps(hw2p4fig3, 'hw2p4fig3', savefigPath)
saveaseps(hw2p4fig4, 'hw2p4fig4', savefigPath)
```

```
saveaseps(hw2p5fig1, 'hw2p5fig1', savefigPath)
saveaseps(hw2p5fig2, 'hw2p5fig2', savefigPath)
saveaseps(hw2p6fig1, 'hw2p6fig1', savefigPath)
saveaseps(hw2p6fig2, 'hw2p6fig2', savefigPath)
saveaseps(hw2p6fig3, 'hw2p6fig3', savefigPath)
saveaseps(hw2p6fig4, 'hw2p6fig4', savefigPath)
%}
```

Problema 6 - preamble

Equalizer





Author Functions

```
function xfilt = retrieve(x, h)
  nPoints = 2e7;
  w = ifft(1./fft(h, nPoints), nPoints);
  xfilt = filter(w, 1, x);
end


function h = plot_signal(x, Fs, approach, figTitle, color)
  h = figure('name', sprintf('Signal Plot: Time and Frequencia Domain
 - %s', figTitle));
  subplot(2,1,1);
  title('Tempo')
  plot_time(x, Fs, figTitle, color);
  axis tight
  subplot(2,1,2);
  title('Frequencia')
  fft_dsp(x, Fs, approach, figTitle, color);
  axis tight
end


function [t] = plot_time(x, Fs, figTitle, color)
  if isempty(color); color = 'black'; end
```

```
  n = length(x);
  t = linspace(0, n/Fs, n);
  % h = figure('name', sprintf('Signal Plot: Time Domain - %s',
 figTitle));
  plot(t, x, 'Color', color, 'LineStyle', '-', 'LineWidth', 1.0);
  grid on;
  xlabel('Tempo (s)', 'interpreter', 'Latex');
  ylabel('Amplitude', 'interpreter', 'Latex');
  ylim([-1.1 1.1])
end


function [Xf, f] = fft_dsp(x, Fs, approach, figTitle, color)
  if isempty(color); color = 'black'; end

  if nargin < 3 || isempty(approach)
    approach = 'positive';
    figTitle = 'Signal';
  end
  if nargin < 4 || isempty(figTitle)
    figTitle = 'Signal';
  end

  n = length(x);
  xdft = fft(x);

  approachChoice = false; % Fix wrong choices for approach

  while ~approachChoice
    approachChoice = true;

    switch approach

      case 'positive'
          n_positive = fix(n/2)+1;
          f = linspace(0, 1, n_positive)*(Fs/2);  % Positive Frequency
          Xf = abs(xdft(1:length(f)))*2/n;      % Multiply By ‘2’ to
 scale magnitude since we use half Frequency
          % h = figure('name', sprintf('Positive FFT(x) - %s',
 figTitle));
          disp(figTitle)
          plot(f, Xf, 'Color', color, 'LineStyle', '-', 'LineWidth',
 1.0)
          xlabel('Frequencia (Hz)', 'interpreter', 'Latex')
          ylabel('Magnitude ($V/\sqrt{Hz}$)', 'interpreter', 'Latex')
          xlim([min(f) max(f)])
          ylim([0.9*min(Xf), 1.1*max(Xf)])
          grid on
        case 'shifted'
          xdftShift = fftshift(xdft);
          f = (-n/2+1:n/2)*(Fs/n);  % zero-centered Frequency range
          Xf = abs(xdftShift)/n;  % zero-centered magnitude
          % h = figure('name', sprintf('Reflected FFT(x) - %s',
 figTitle));
```

```
            plot(f,Xf, 'Color', color, 'LineStyle', '-', 'LineWidth',
 1.0)
            xlabel('Frequencia (Hz)', 'interpreter', 'Latex')
            ylabel('Magnitude ($V/\sqrt{Hz}$)', 'interpreter', 'Latex')
            xlim([min(f) max(f)])
            ylim([0.9*min(Xf), 1.1*max(Xf)])
            grid on
          case 'freq_norm'
            n_positive = fix(n/2)+1;
            f = linspace(0, 1, n_positive);     % Positive Frequency
            Xf = abs(xdft(1:length(f)))*2/n;  % Multiply By ‘2’ to scale
 magnitude since we use half Frequency
            % h = figure('name', sprintf('Normalized Frequency FFT(x) -
 %s', figTitle));
            plot(f, Xf, 'Color', color, 'LineStyle', '-', 'LineWidth',
 1.0)
            xlabel('Normalized Frequencia ($\times \pi radians/samples
$)', 'interpreter', 'Latex')
            ylabel('Magnitude', 'interpreter', 'Latex')
            xlim([min(f) max(f)])
            grid on
          case 'dB'
            xdft = xdft(1:n/2+1);
            psdx = (1/(Fs*n)) * abs(xdft).^2;
            psdx(2:end-1) = 2*psdx(2:end-1);
            f = 0:Fs/length(x):Fs/2;
            Xf = pow2db(psdx);
            % h = figure('name', sprintf('Positive FFT(x) in dB - %s',
 figTitle));
            plot(f, Xf, 'Color', color, 'LineStyle', '-', 'LineWidth',
 1.0)
            grid on
            xlabel('Frequencia (Hz)', 'interpreter', 'Latex')
            ylabel('Power/Frequencia (dB/Hz)', 'interpreter', 'Latex')
            ylim([-80 10])
          otherwise
            warning("Invalid approach. Replaced by 'positive'")
            approachChoice = false; % Return to switch beginning
            approach = 'positive';
      end
    end

end


function saveaseps(input, saveName, savefigPath)
  saveas(input, sprintf('%s%s', savefigPath, saveName),'epsc')
end


function h = plot_impz(b, a, figTitle, color)
  if isempty(color); color = 'black'; end
    [h_, t] = impz(b, a, 100);
```

```
    % h = figure('name', sprintf('Impulse Response Plot: Time Domain -
 %s', figTitle));
    plot(t, h_, 'Color', color, 'LineStyle', ':', 'LineWidth', 1.5);
    grid on;
    title(figTitle)
    xlabel('Amostras (n)', 'interpreter', 'Latex');
    ylabel('Amplitude', 'interpreter', 'Latex');
end


function h = plot_pspectrum(x, Fs, figTitle, color)
  if isempty(color); color = 'black'; end
  [p,f] = pspectrum(x, Fs);
  % h = figure('name', sprintf('Power spectrum: Frequency Domain -
 %s', figTitle));
  plot(f, mag2db(p), 'Color', color, 'LineStyle', '-', 'LineWidth',
 1.0);
  grid on;
  title(figTitle);
  xlabel('Frequencia (kHz)', 'interpreter', 'Latex');
  ylabel('Potencia (dB)', 'interpreter', 'Latex');
end
```

Published with MATLAB® R2021a